



TPC Benchmark™ C
Full Disclosure Report

NEC Express5800 HV8600 (Cluster)

with Oracle8i for Windows NT
and Microsoft Windows NT Server, Enterprise Edition 4.0

First Edition
Submitted for Review
June 29, 1999

NEC, the Sponsors of this benchmark test, believe 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 do 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 1999 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, 1998

NEC and Express5800 are registered trademarks of NEC Corporation.

ORACLE, SQL*Loader, Oracle8i, and Pro*C are registered trademarks of Oracle Corporation.

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

Microsoft, Windows NT are registered trademarks of Microsoft Corporation.

BEA and Tuxedo are registered trademarks of BEA Systems, Inc.

Intel, PentiumIII and Xeon are trademarks or registered trademarks of Intel Corporation.

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

Additional Copies

For additional copies, please contact:

Transaction Performance Processing Council
c/o Shanley Public Relations
777 North First Street, Suite 6000
San Jose, CA 95112-6311
408-295-8894

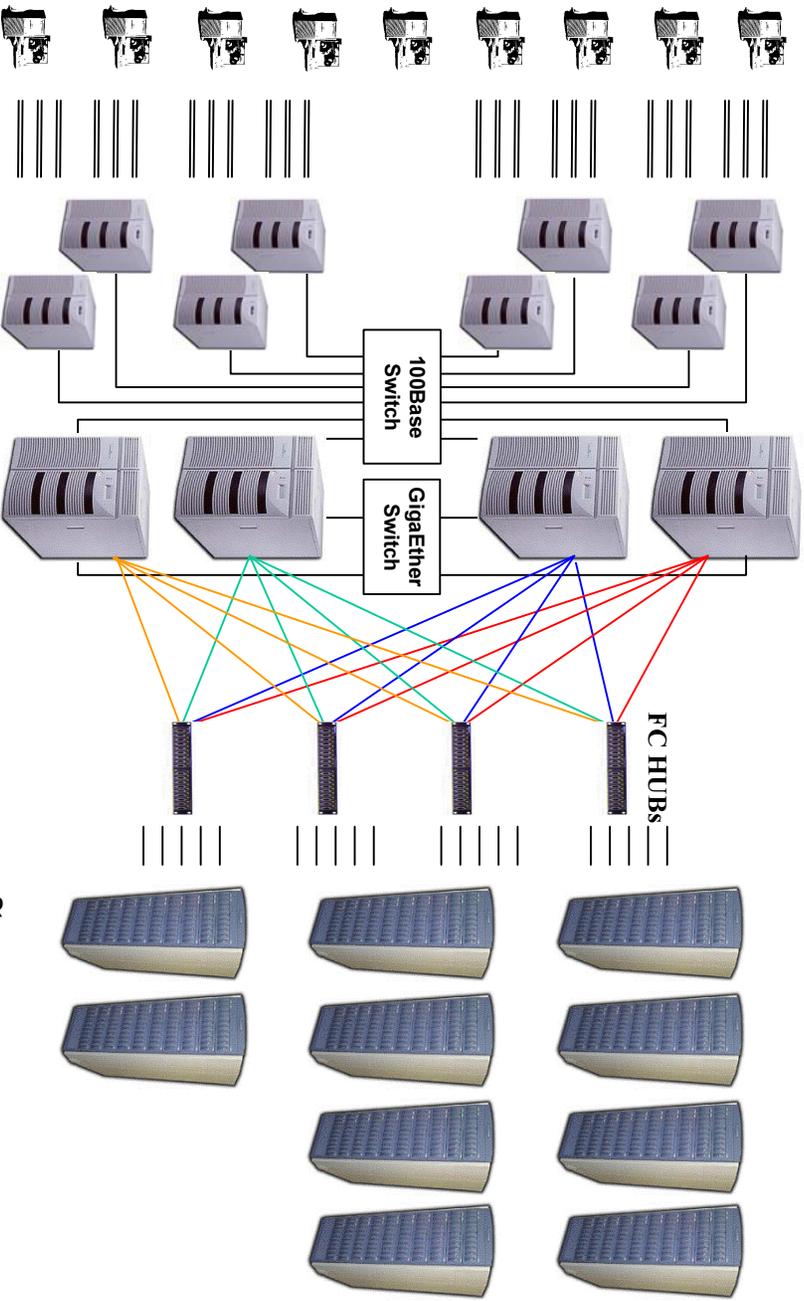


EXPRESS5800 HV8600 C/S

TPC-C Rev. 3.4
Reported Date
June 29, 1999

Total System Cost	TPC-C Throughput	Price/Performance	Availability
\$4,723,164	50,208.43 tpmC	\$ 94.05per tpmC	November 30, 1999
Processors	DBMS	Operating System	Other Software
32 x Pentium ® III Xeon 500MHz 2MB L2 cache	Oracle®8i Enterprise Edition v8.1.6.0	Microsoft Windows NT 4.0 Enterprise Edition	Microsoft IIS3.0 BEA Tuxedo 6.4 CFS VC++, MKS Tool Kit
			Number of Users
			48,000

48,000 user connections to 8 clients
(6 segments per client)



Clients
Express5800 HX4600

Servers
Express5800 HV8600

Storage
Data General FC5700
FibreChannel Disk Array System

System Component	Servers		Each Client
Processors	32	Pentium ® III Xeon 500MHz	2
Cache		2MB	2MB
Memory	4	4096MB	1
Disk Controllers	16	Qlogic q2100 Fibre HBA Integrated SCSI	1
Disk Drives	740	9GB disk (8.339GB usable) 18GB disk (16.936GB usable)	1
Total Storage		6916 GB	
Others	1	CD-ROM Drive DAT Drive	1



NEC Express 5800 HV8600

C/S

TPC-C REV 3.4

Report Date: June 29, 1999

Description	Part Number	Brand	Third Party		Unit	Qty	Extended Price	5-year Maint Price
			Pricing	Price				
Server Hardware								
Express5800 HV8600	850131020	NEC	1	29,629	4	118,516	28,444	
Integrated SCSI, Ether. 1 CD-ROM	062-01602-000	NEC	1	63,249	4	252,996	60,720	
8x Pentium III Xeon 500MHz/2MB Optior	062-01544-000	NEC	1	1,449	32	46,368	89,027	
512MB(4x 128MB) Memory Option	203497	NEC	1	799	4	3,196	768	
9GB 10Krpm HDD Option	203117	NEC	1	1,099	4	4,396	0	
12/24GB SCSI DDS-3 4mm DAT Drive	QLA2100	Qlogic	5	774	18	13,932	0	
Qlogic QLA2100(+2spares)	ACENic PCI	Alteon	6	995	6	5,970	0	
Alteon ACENic(+2spares)	Alteon 180	Alteon	6	14,995	3	44,985	0	
Alteon ACEswitch 180(+2spares)	JC-1576VMA	NEC	1	299	4	1,196	288	
NEC 15" Multisync Monitor								
Disk Subsystem								
39U rack, dual AC inlet	C7661G-F7B	DG	2	6,500	10	65,000	0	
HA Model C5701 10 drive DPE	C5701R-A	DG	2	60,500	10	605,000	25,200	
HA Model C5051 10 drive DAE	C5051D-A	DG	2	8,300	62	514,600	30,008	
Cabinet Mounting Kit for DAE	C7680G-A	DG	2	100	62	6,200	0	
8.8GB 10Krpm HDD Option(+10%spares)	C6810FLG-A	DG	2	1,850	740	1,369,000	0	
18GB 10Krpm HDD Option(+10%spares)	C1710FG-A	DG	2	2,850	44	125,400	0	
1024MB Cache Option(+2spares)	C77512G-A	DG	2	14,000	12	168,000	0	
Dual Rackmount SPS Option(+2spares)	C7714G-A	DG	2	4,200	12	50,400	0	
Single 9-port copper FC Hub	C5H01D-V1	DG	2	4,400	4	17,600	1,936	
10m Cable(+10%spares)	154950E033	DG	2	200	22	4,400	0	
(Storage Volume Discount)		DG	2			(1,170,240)		
Subtotal				2,246,915		236,391		
Server Software								
Microsoft Windows NT Server Enterprise Edition 4.0		Microsoft	1	0	4	0	0	
(included in Express5800 HV8600)								
Oracle 8i v8.1.6.0 Bronze Level support		Oracle	3	808,889	1	808,889	808,889	
MKS Toolkit 6.1 for Win32	20014552	MKS	7	252	8	2,016	0	
Subtotal				810,905		808,889		
Client Hardware								
Express5800 HX4600	850125020	NEC	1	11,699	8	93,592	22,463	
2x Pentium III Xeon 500MHz/2MB Optior	062-01539-000	NEC	1	14,929	8	119,432	28,664	
512MB(4x 128MB) Memory Option	062-01577-000	NEC	1	1,449	16	23,184	11,129	
9GB 10Krpm HDD Option	203497	NEC	1	799	8	6,392	1,535	
Adaptec ANA-6944A/TX(+2spares)	ANA-6944A/TX	Adaptec	10	517	18	9,306	0	
NEC 15" Multisync Monitor	JC-1576VMA	NEC	1	299	8	2,392	576	
Subtotal				254,298		64,367		
Client Software								
Microsoft Windows NT Server 4.0		Microsoft	1	0	8	0	0	
(included in Express5800 HX4600)								
Visual C++ Professional 6.0	20002203	Microsoft	7	364	1	364	0	
Tuxedo 6.4 Core functionality Services for NT		BEA	4	3,000	8	24,000	18,000	
Subtotal				24,364		18,000		
User Connectivity								
Cisco Catalyst/2912-XL-A (+2 spares)	WS-C2912-XL-A	Cisco	8	1,245	3	3,735	0	
Compex 16-pt 10Mbps HUB (+10% spares)	TP1016C	Compex	9	77	3300	254,100	0	
Subtotal				257,835		0		
TOTAL				3,594,317		1,127,647		

Notes:

- 1-Pricing: 1-NEC 2-DG 3-Oracle 4-BEA 5-Unique technologies 6-Alteon 7-BVSOFT.COM
- 8-Bottom Line Telecom 9-COI 10-NECX
- 2-offers volume discount 5,6,8,9,10 : 5-year warranty

Five-Year Cost of Ownership: **\$4,721,964**

tpmC Rating: **50208.43**

\$ / tpmC: 94.05

Audited by Francois Raab of InfoSizing, Inc

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

Numerical Quantities Summary

MQTh, Computed Maximum Qualified Throughput				50208.43tpmC
%throughput difference ,reported & reproducibility runs				0.13 %
Response Times(in seconds)	90%	Average	Maximum	
New-Order	2.36	0.99	93.86	
Payment	2.07	0.78	63.61	
Order-Status	1.88	0.71	16.83	
Delivery(interactive portion)	1.35	0.56	6.75	
Delivery(deferred portion)	1.50	0.66	6.13	
Stock-Level	3.15	1.88	9.61	
Menu	0.35	0.28	5.99	
Response time delay added for emulated components				0.1
Transaction Mix , in percent of total transaction				
New-Order				44.77%
Payment				43.09%
Order-Status				4.05%
Delivery				4.05%
Stock-Level				4.04%
Keying/Think Times (in seconds)	Min.	Average	Max	
New-Order	21.05	14.04	21.74	140.40
Payment	3.50	14.04	3.97	140.43
Order-status	2.35	11.71	2.71	116.94
Delivery	2.35	5.83	2.75	58.41
Stock-level	2.35	5.86	2.63	58.43
Test Duration				
Ramp-up time				21.6 minutes
Measurement interval				30 minutes
Number of checkpoints				1
Checkpoint interval				30 minutes
Number of transactions (all types) completed in measurement interval				3,362,812

ABSTRACT	1
TPC BENCHMARK™ C METRICS	1
STANDARD AND EXECUTIVE SUMMARY STATEMENTS.....	1
AUDITOR	1
PREFACE	2
TPC BENCHMARK™ C OVERVIEW	2
DOCUMENT STRUCTURE	2
GENERAL ITEMS	3
ORDER AND TITLES.....	3
SUMMARY STATEMENT	3
NUMERICAL QUANTITIES SUMMARY	3
APPLICATION PROGRAM	3
SPONSOR.....	4
PARAMETERS AND OPTIONS.....	4
CONFIGURATION DIAGRAMS	4
MEASURED CONFIGURATION	5
PRICED SYSTEM CONFIGURATION	6
CLAUSE 1 : LOGICAL DATABASE DESIGN AND RELATED ITEMS	7
TABLE DEFINITIONS.....	7
TABLE ORGANIZATION.....	7
INSERT AND DELETE OPERATIONS.....	7
DISCLOSURE OF PARTITIONING.....	7
REPLICATION OF TABLES.....	7
ADDITIONAL AND/OR DUPLICATED ATTRIBUTES IN ANY TABLE.....	7
CLAUSE 2 : TRANSACTION AND TERMINAL PROFILES RELATED ITEMS	8
RANDOM NUMBER GENERATION.....	8
TERMINAL INPUT/OUTPUT SCREEN LAYOUT	8
TERMINAL FEATURE VERIFICATION.....	8
PRESENTATION MANAGER OR INTELLIGENT TERMINAL	8
TRANSACTION PROFILES.....	8
TRANSACTION MIX.....	9
QUEUING MECHANISM.....	9
CLAUSE 3 : TRANSACTION AND SYSTEM PROPERTIES RELATED ITEMS	10
TRANSACTION SYSTEM PROPERTIES (ACID)	10
ATOMICITY TESTS.....	10
Completed Transactions	10
Aborted Transactions	10
CONSISTENCY TESTS	10
ISOLATION.....	10
DURABILITY.....	11
<i>Instantaneous interruption , Loss of Memory</i>	<i>11</i>
<i>Loss of log and data disk</i>	<i>11</i>
<i>Loss of cluster interconnect</i>	<i>12</i>
<i>Loss of mirrored write-back cache</i>	<i>12</i>
CLAUSE 4 : SCALING AND DATABASE POPULATION RELATED ITEMS	13
INITIAL CARDINALITY OF TABLES	13
DISTRIBUTION OF TABLES AND LOGS	14
TYPE OF DATABASE.....	15
DATABASE MAPPING.....	16
180-DAYS SPACE.....	16

CLAUSE 5 : PERFORMANCE METRICS AND RESPONSE TIME RELATED ITEMS	17
THROUGHPUT	17
RESPONSE TIMES	17
KEYING AND THINK TIMES	17
RESPONSE TIME FREQUENCY DISTRIBUTION CURVES AND OTHER GRAPHS	18
RESPONSE TIME VERSUS THROUGHPUT PERFORMANCE CURVE	20
NEW-ORDER THINK TIME	21
NEW-ORDER THROUGHPUT VS. ELAPSED TIME	21
STEADY STATE	22
WORK PERFORMED DURING STEADY STATE	22
REPRODUCIBILITY	22
MEASUREMENT PERIOD DURATION	22
REGULATION OF TRANSACTION MIX	22
TRANSACTION STATISTICS	22
CHECKPOINT COUNT AND LOCATION	22
CLAUSE 6 : SUT, DRIVER, AND COMMUNICATION DEFINITION RELATED ITEMS	23
DESCRIPTIONS OF RTE	23
EMULATED COMPONENTS	23
FUNCTIONAL DIAGRAMS AND DETAIL OF DRIVER SYSTEM	23
NETWORK CONFIGURATIONS AND DRIVER SYSTEM	23
NETWORK BANDWIDTH	23
OPERATOR INTERVENTION	23
CLAUSE 7 : PRICING RELATED ITEMS	24
HARDWARE AND SOFTWARE COMPONENTS	24
AVAILABILITY	24
THROUGHPUT, AND PRICE PERFORMANCE	24
COUNTRY SPECIFIC PRICING	24
USAGE PRICING	24
CLAUSE 9 : AUDIT RELATED ITEMS	25
AUDITOR'S REPORT	25
AVAILABILITY OF THE FULL DISCLOSURE REPORT	25
AUDITOR'S LETTER	26
APPENDIX A : APPLICATION SOURCE CODE	28
APPENDIX B : DATABASE DESIGN	152
APPENDIX C : TUNABLE PARAMETERS	269
APPENDIX D : SPACE CALCULATION	299
APPENDIX E : PRICE QUOTATION	300

Abstract

This report documents the compliance of NEC Corporations TPC Benchmark™ C tests on the NEC Express 5800 HV8600 model2 client/server system with version 3.4 of the TPC Benchmark C Standard Specification. 4 Clients (NEC Express5800 HX4600) were used as the front-end clients.

The operating system and the DBMS used on the server were Microsoft Windows NT, Enterprise Edition 4.0 and Oracle8i v8.1.6. The operating system on the clients was Microsoft Windows NT Server 4.0. Those clients ran Microsofts IIS server 3.0 and Tuxedo 6.4 CFS for Windows NT.

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 (five year capital cost per measured tpmC) are reported.

System	SW	Total System Cost	tpmC	\$ per tpmC	Availability Date
NEC Express5800 HV8600 model2	Oracle8i v8.1.6 Tuxedo 6.4 CFS Microsoft Windows NT, Enterprise Edition 4.0	\$4,723,164	50,208.43	\$94.05	November 30th , 1999

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 InfoSizing, 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 3.4.

TPC Benchmark™ C Overview

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

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

The performance metric reported by TPC-C is a business throughput¹ measuring the number of orders processed per minute. Multiple transactions are used to simulate the business activity of processing an order, and each transaction is subject to a response time constraint. The performance metric for this benchmark is expressed in transactions-per-minute-C (tpmC). To be compliant with the TPC-C standard, all references to tpmC results must include the tpmC rate, the associated price-per-tpmC, and the availability date of the priced configuration.

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 benchmark when critical capacity planning and/or product evaluation decisions are contemplated.

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 for TPC-C standard specification. 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.

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,*
- *computed maximum Qualified Throughput in tpmC,*
- *percentage difference between reported throughput and throughput obtained in reproducibility run,*
- *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, and 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 and Oracle Corporation . Packard Bell NEC has authorized NEC Corp. to publish TPC-C performance and price/performance results for the NEC Epress5800 HV8600. Price quotations contained in Appendix E correspond to the NEC Express5800 HV8600 server.

Parameters and Options

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

- *Database tuning options*
- *Recovery/locking options*
- *Operating system and application configuration parameters*

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

Configuration Diagrams

Provide diagrams of both the measured and priced configurations, accompanied by a description of the differences. This includes, but not limited to:

- *Number and type of processors*
- *Size of allocated memory, and any specific mapping/partitioning or memory unique to the test*
- *Number and type of disk drive 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.*
- *Type and the run-time execution location of software components(e.g., DBMS, client processes, transaction monitors, software drivers, etc.)*

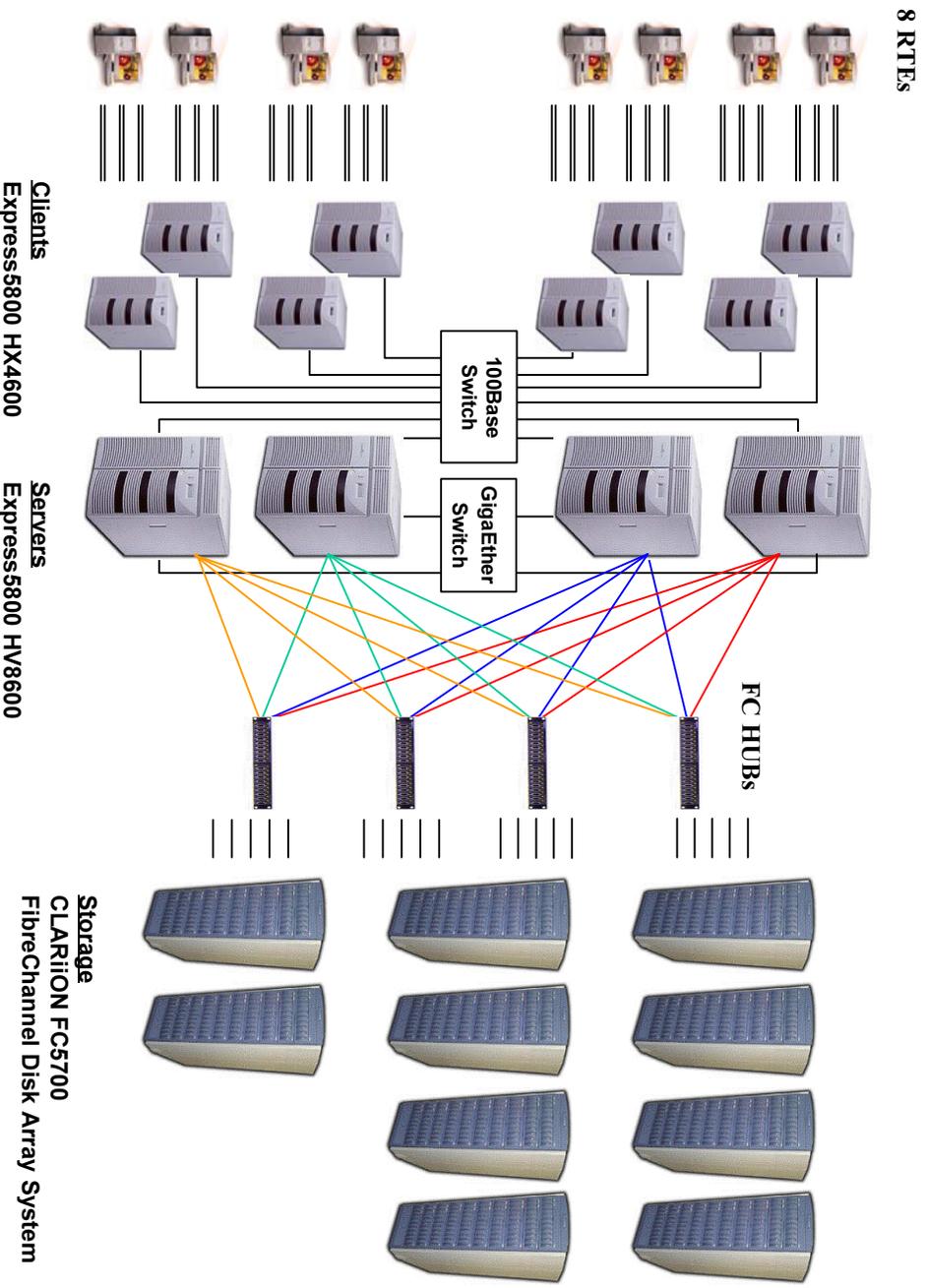
Figure 1.1 shows the measured configuration diagram.

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 Measured Configuration Diagram

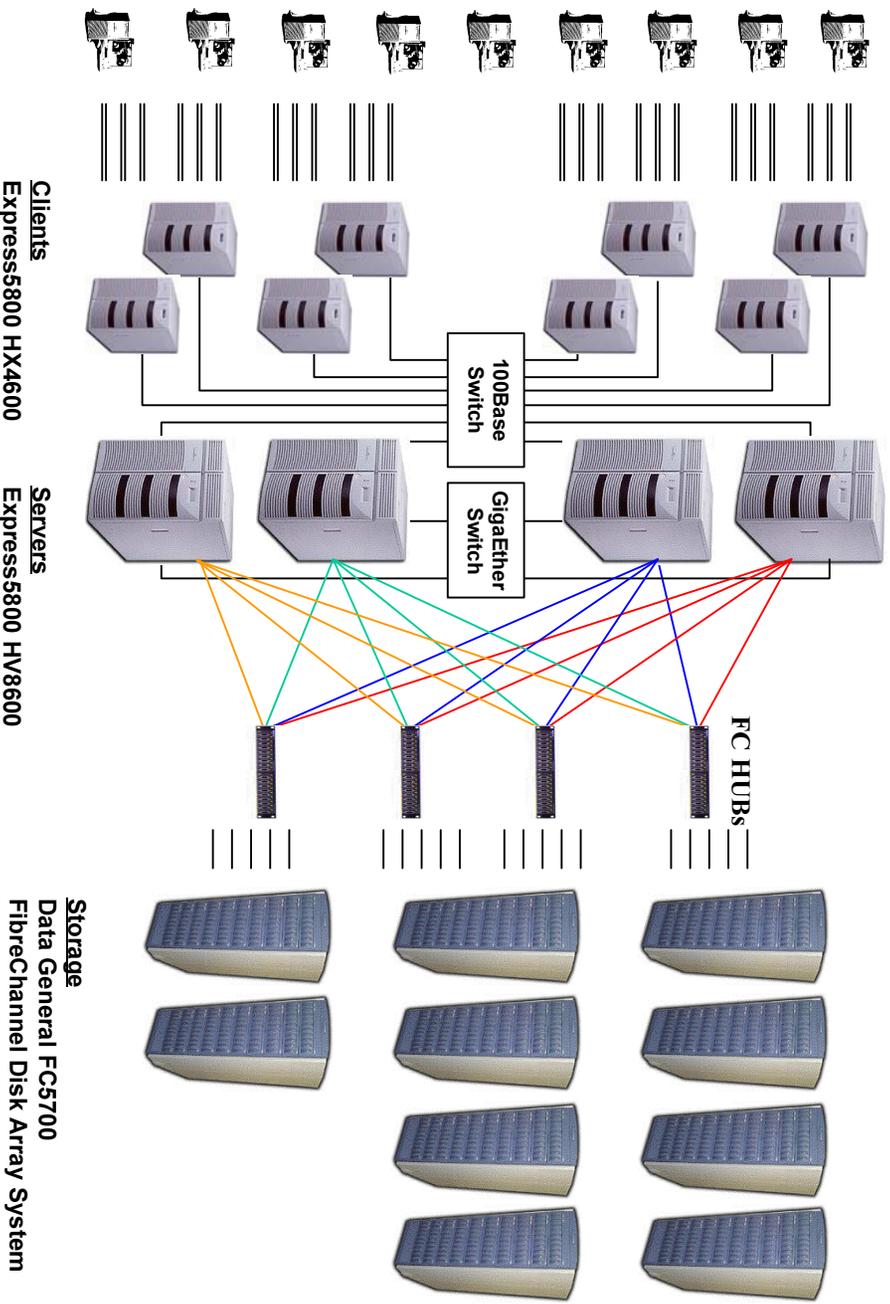


Priced System Configuration

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

Figure 1.2: Priced Configuration Diagram

48,000 user connections to 8 clients
(6 segments per client)



Clause 1 : Logical Database Design and Related Items

Table Definitions

Listing 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 restrictions 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 minimum 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.

Horizontal partitioning based on warehouse ID was used on the new_order, order_line, orders and history tables.

Replication of Tables

Replication of tables, if used, must be disclosed.

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.

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 layout 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 8.1.3.3.3 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.

Comment1: *The intent of this clause is to describe any special manipulations performed by a local terminal or workstation to off-load work from the SUT. This includes, but is not limited to : screen presentations, message bundling, and local storage of TPC-C rows.*

Comment2: *This disclosure also requires that all data manipulation functions also be described. Within this disclosure, the purpose of such additional function(s) 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 section 3.5 through 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	1.00%
	Average items per order	9.91
Payment	Home warehouse payments	85.04%
	Remote warehouse payments	14.96%
	Accessed by last name	59.96%
Order Status	Accessed by last name	59.94%
Delivery	Skipped deliveries	0
Transaction Mix	New Order	44.77%
	Payment	43.09%
	Order Status	4.05%
	Delivery	4.05%
	Stock Level	4.04%

Queuing Mechanism

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

The client application submitted to delivery transaction servers 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 the 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 for 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 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 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.

Consistency conditions one through four were tested using a script to issue queries to the database. The results of the queries verified that the database was consistent for all four tests. A run was executed over 10 minutes and included a checkpoint under full scaled users condition . The shell script was executed before and after the run. The result of the same queries verified that the database remained consistent after the run.

Isolation

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 between the nodes, 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

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

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

Instantaneous interruption , Loss of Memory

Because the loss of power erases the contents of memory, both of instantaneous interruption and loss of memory were combined into a single 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 connected to SUT and started submitting transactions.
3. A checkpoint was initiated after 5 minutes.
4. After the global checkpoint was finished, the running continued for 2 minutes.
5. One of four nodes was powered off.
6. Oracle OPS started cluster reconfiguration and one of survived three nodes recovered database automatically.
7. After the autorecovery finished, survived three nodes restored processing transactions.
8. RTE was stopped.
9. A new count of D_NEXT_O_ID was taken on a survived node.
10. This number was compared with the number of new orders reported by the RTE.

Loss of log and data disk

Loss of log and Loss of data disk were combined to a one test and were demonstrated on a 10 Warehouse database shared with 2 nodes for convenience. The standard driving mechanism was used to generate the transaction load of 100 users for the test. To demonstrate recovery from a permanent failure of durable media containing TPC-C tables and log, the following steps were performed.

A fully scaled database would also pass this test.

1. A 10 Warehouse database was built having similar characteristics to the large database.
2. The database was backed up .
3. A sum of D_NEXT_O_ID was taken.
4. RTE was started running with 100 users.
5. After 5 minutes running, a checkpoint was initiated.
6. One disk drive containing redo log was removed after 2minutes of checkpoint completion.
7. Since the drive was mirrored to another disk, the processing was not interrupted.
8. One disk drive containing data file was removed after 1minutes.
9. Oracle OPS reported IO error. The database and RTE was then shut down.
10. The 10 Warehouse database data was restored from backup.
11. The database was started and recovered by using recover command from svrmgl.
12. The database was opened and ORACLE8 performed instance recovery.
13. Consistency conditions were executed and verified.
14. A new count of D_NEXT_O_ID was taken.
15. This number was compared with the number of new orders reported by the RTE.

Loss of cluster interconnect

Loss of cluster interconnect was demonstrated on a 200 warehouse database shared with 2 nodes for convenience. The standard driving mechanism was used to generate the transaction load of 2000 users for the test. To demonstrate recovery from interconnection, the following steps were performed. 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. RTE was started running with 2000 users.
3. A checkpoint was initiated after 5 minutes.
4. After the global checkpoint was finished, the running continued for 2 minutes.
5. Power of switching HUB for interconnection was turned off.
6. One of two nodes automatically started database recovery.
7. After the autorecovery finished, RTE was stopped.
8. A new count of D_NEXT_O_ID was taken.
9. 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 SP modules, which are powered and protected from loss of power by independent UPSs. Because mirrored drives used to contain redo log files were configured to enable write-back caching, loss of write-back cache was demonstrated on 10 warehouse database shared with two nodes, by pulling of f one of SP 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. RTE was started running with 100 users.
3. A checkpoint was initiated after 5 minutes.
4. After the global checkpoint was finished, the running continued for 2 minutes.
5. A SP module, which manages write-back cache of mirrored drives containing redo log files, was pulled off.
6. Fibre system reported system error resulted in IO error for Oracle database.
7. Contents on mirrored cache on another SP was automatically saved to temporal disk area and the Fibre Array house was eventually stopped. System was shut down.
8. The Fibre array was re-powered.
9. Saved data was automatically restored to mirrored cache and flushed to drives containing redo log files.
10. A Oracle was started up and database was recovered automatically.
11. A new count of D_NEXT_O_ID was taken.
12. 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. number of rows) of each table, as it existed at the start of the benchmark run, must be disclosed. If the database was over-scaled and inactive rows of the WAREHOUSE table were deleted, 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 4800 warehouses.

Table 2 Number of Rows for Server

Table	Cardinality as benchmarked
Warehouse	4,800
Distinct	48,000
Customer	144,000,000
History	144,000,000
Orders	144,000,000
New Order	43,200,000
Order Line	1,440,025,184
Stock	480,000,000
Item	100,000

Distribution of Tables and Logs

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

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

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

Table 3 : Data Distribution

NT div. No.	Physical Layout	RAID Level	Disk Size (MB)	comment
1	SCSI2:Array5:LUN0	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/4 iware.
2	SCSI2:Array5:LUN1	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/4 idist. backup
3	SCSI2:Array5:LUN2	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/4 iitem. backup
4	SCSI2:Array5:LUN3	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/4 inord.
5	SCSI2:Array5:LUN4	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/4 icust.1
6	SCSI2:Array5:LUN5	5 : 10striped	75,055	Backup
7	SCSI2:Array6:LUN0	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/4 istok.
8	SCSI2:Array6:LUN1	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/12 iordl.
9	SCSI2:Array6:LUN2	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/12 iordl.
10	SCSI2:Array6:LUN3	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/12 iordl.
11	SCSI2:Array6:LUN4	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/2 control.
12	SCSI2:Array6:LUN5	1/0 : 10mirrored	84,680	Log1
13	SCSI2:Array10:LUN2	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. backup
14	SCSI2:Array10:LUN3	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp.
15	SCSI2:Array10:LUN4	5 : 10striped	108,413	Backup
16	SCSI3:Array7:LUN0	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/4 iware.
17	SCSI3:Array7:LUN1	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/4 idist.
18	SCSI3:Array7:LUN2	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/4 iitem. backup
19	SCSI3:Array7:LUN3	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/4 inord.
20	SCSI3:Array7:LUN4	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/4 icust.1
21	SCSI3:Array7:LUN5	5 : 10striped	75,055	Backup
22	SCSI3:Array8:LUN0	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/4 istok.
23	SCSI3:Array8:LUN1	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/12 iordl.
24	SCSI3:Array8:LUN2	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/12 iordl.
25	SCSI3:Array8:LUN3	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/12 iordl.
26	SCSI3:Array8:LUN4	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. system. backup
27	SCSI3:Array8:LUN5	1/0 : 10mirrored	84,680	Log2

28	SCSI3:Array10:LUN0	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp, backup
29	SCSI3:Array10:LUN1	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp.
30	SCSI4:Array3:LUN0	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/4 iware.
31	SCSI4:Array3:LUN1	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/4 idist, backup
32	SCSI4:Array3:LUN2	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/4 item, backup
33	SCSI4:Array3:LUN3	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/4 inord.
34	SCSI4:Array3:LUN4	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/4 icust.1
35	SCSI4:Array3:LUN5	5 : 10striped	75,055	Backup
36	SCSI4:Array4:LUN0	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/4 istok.
37	SCSI4:Array4:LUN1	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/12 iordl.
38	SCSI4:Array4:LUN2	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/12 iordl.
39	SCSI4:Array4:LUN3	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/12 iordl.
40	SCSI4:Array4:LUN4	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/2 control, backup
41	SCSI4:Array4:LUN5	1/0 : 10mirrored	84,680	Log3
42	SCSI4:Array9:LUN0	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp.
43	SCSI4:Array9:LUN1	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp.
44	SCSI5:Array1:LUN0	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/4 iware.
45	SCSI5:Array1:LUN1	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/4 idist, backup
46	SCSI5:Array1:LUN2	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/4 item, backup
47	SCSI5:Array1:LUN3	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/4 inord.
48	SCSI5:Array1:LUN4	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/4 icust.1
49	SCSI5:Array1:LUN5	5 : 10striped	75,055	Backup
50	SCSI5:Array2:LUN0	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/4 istok.
51	SCSI5:Array2:LUN1	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/12 iordl.
52	SCSI5:Array2:LUN2	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/12 iordl.
53	SCSI5:Array2:LUN3	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp. 1/12 iordl.
54	SCSI5:Array2:LUN4	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp.
55	SCSI5:Array2:LUN5	1/0 : 10mirrored	84,680	Log4
56	SCSI5:Array9:LUN2	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp, backup
57	SCSI5:Array9:LUN3	0 : 14striped	116,753	1/48 Stok, Cust, Icust2, Item, Ord, Ord, Iordr1, Iordr2, nord, hist, disk, ware, roll and temp.
58	SCSI5:Array9:LUN4	5 : 10striped	108,413	Backup

Type of Database

A statement must be provided that describes:

- 1) The data model implemented by DBMS used (e.g. relational, network, hierarchical).
- 2) The database interface (e.g. embedded, call level) and access language (e.g. SQL, PL/I, COBOL read/write used to implement the TPC-C transaction. 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.

Oracle8i, a relational database, was used in this benchmark. Anonymous block PL/SQL and stored procedures were accessed through the ORACLE Call Interface. Application code is included in Appendix A.

Database Mapping

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

Horizontal partitioning based on warehouse ID was used on the new_order, order_line, orders and history tables

180-Days Space

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

The detail of 180-day space calculation is shown in Appendix D.

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

1. Logfile usage was obtained from system view.
2. The space used was calculated as the difference between the first and second query.
3. The number of NEW-ORDERS was verified from an RTE report covering the entire run.
4. The space used was divided by the number of NEW-ORDERS giving a spaceused per NEW-ORDER transaction.
5. The space used per transaction was multiplied by the measured tpmC rate times 480 minutes.

The same methodology was used to compute growth requirements for dynamic tables Order, Order-Line and History.

Clause 5 : Performance Metrics and Response Time Related Items

Throughput

Measured tpmC must be reported

Table 4 : Measured tpmC

50208.43 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	0.99	93.86	2.36
Payment	0.78	63.61	2.07
Order Status	0.71	16.83	1.88
Interactive Delivery	0.56	6.75	1.35
Deferred Delivery	0.66	6.13	1.50
Stock Level	1.88	9.61	3.15
Menu	0.28	5.99	0.35

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	Minimum	Average	Maximum
New-Order	21.05	21.05	21.74
Payment	3.50	3.51	3.97
Order Status	2.35	2.35	2.71
Interactive Delivery	2.35	2.35	2.75
Stock Level	2.35	2.35	2.63

Table 7 : Think Times

Type	Minimum	Average	Maximum
New-Order	0.00	14.04	140.40
Payment	0.00	14.04	140.43
Order Status	0.00	11.71	116.94
Interactive Delivery	0.00	5.83	58.41
Stock Level	0.00	5.86	58.43

Response Time Frequency Distribution Curves and Other Graphs

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

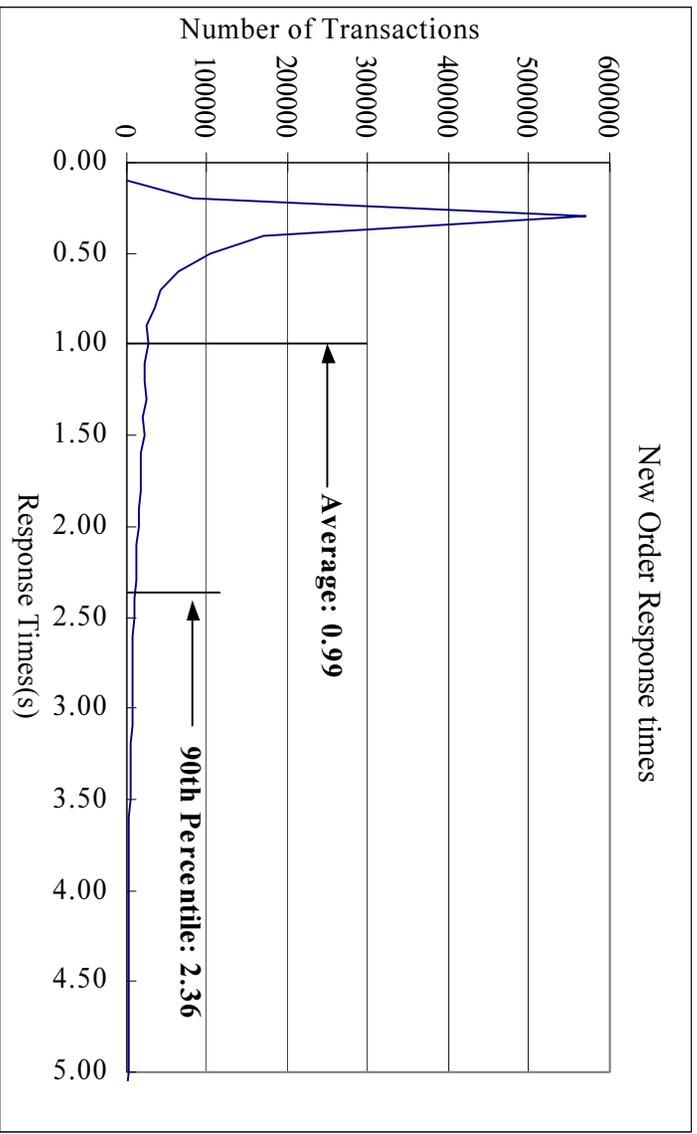
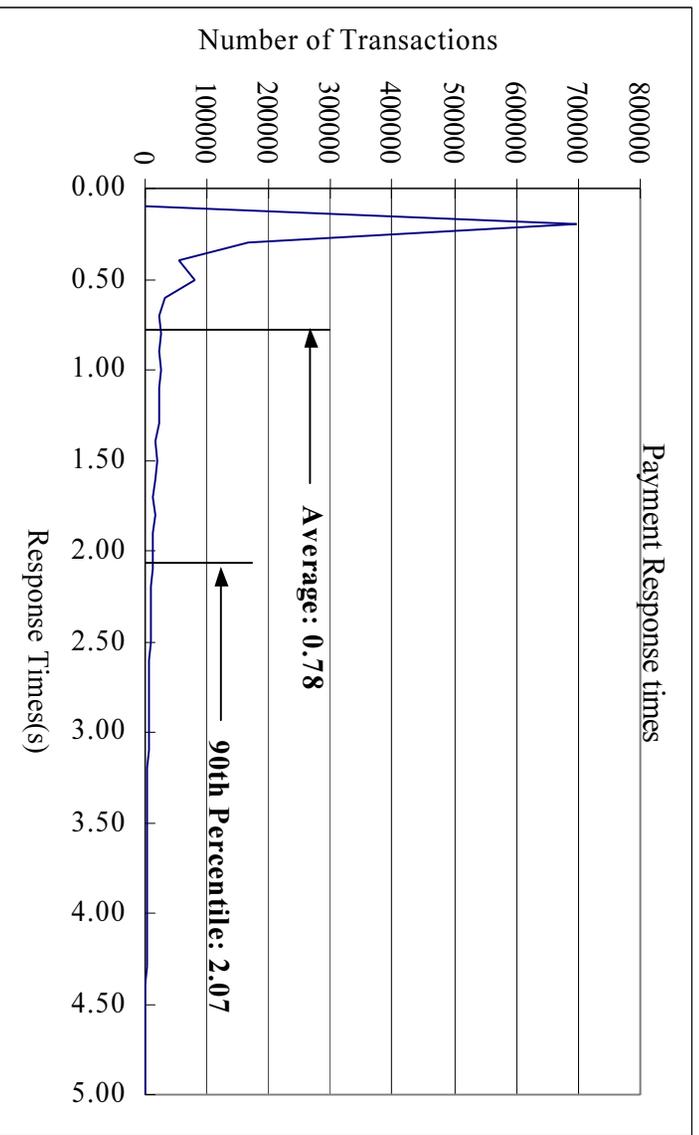


Figure 2.2 : Payment Response Time Distribution



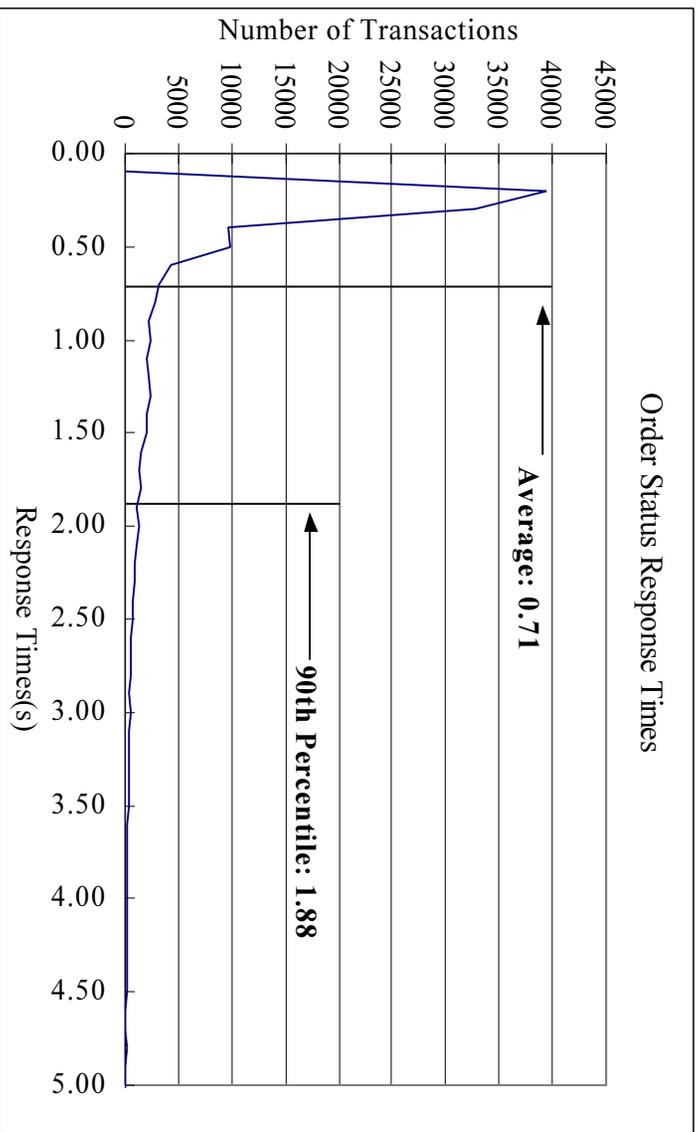


Figure 2.3 : Order Status Response Time Distribution

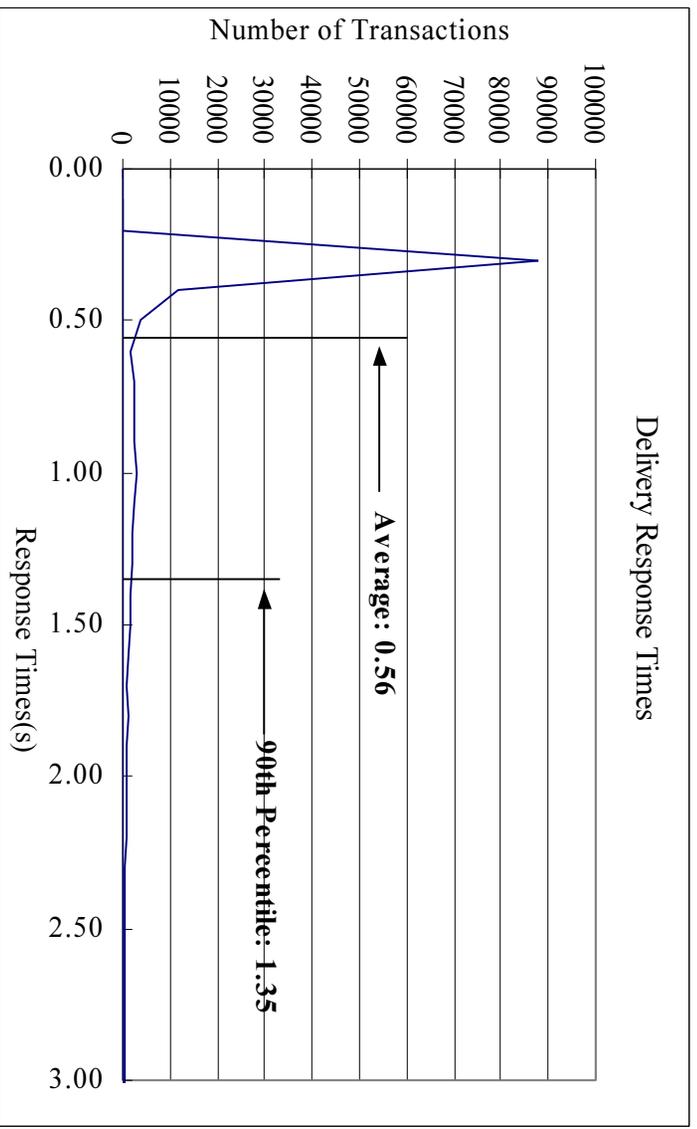
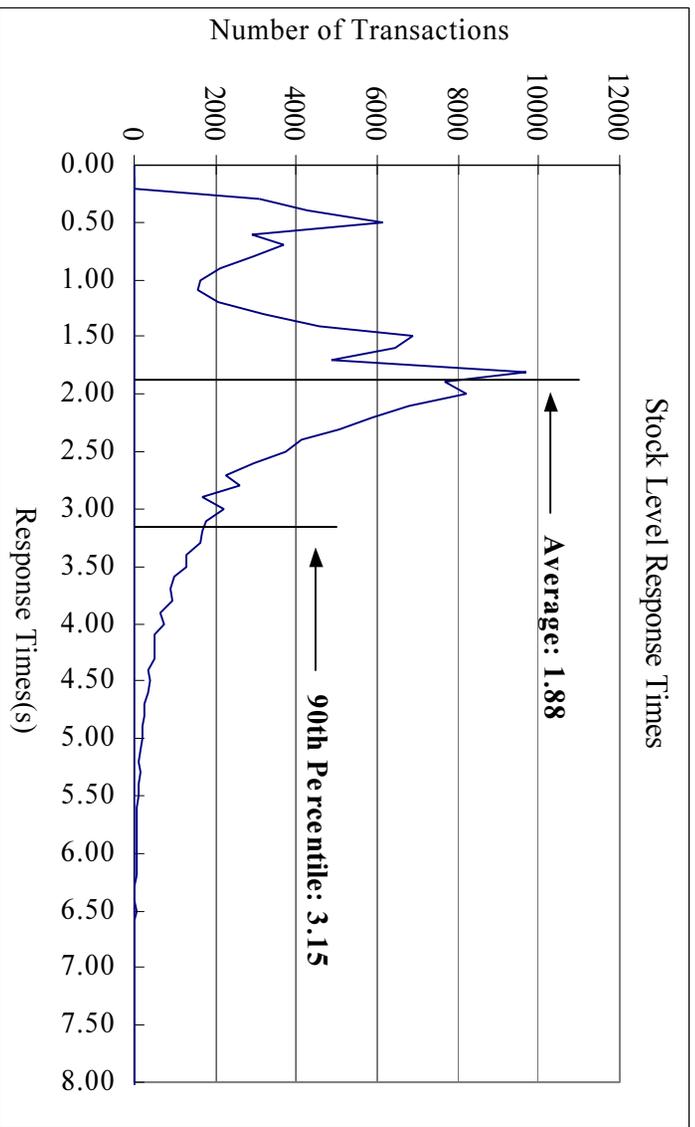


Figure 2.4 : Delivery Response Time Distribution

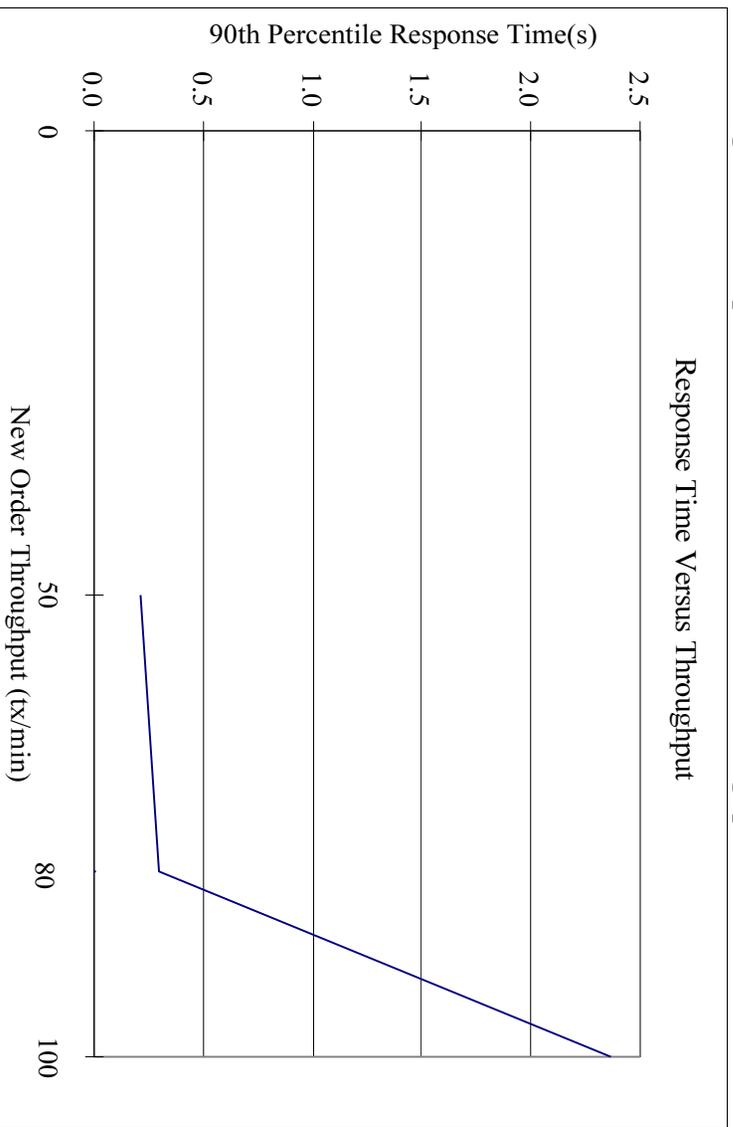
Figure 2.5 : Stock Level Response Time Distribution



Response time versus Throughput Performance Curve

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

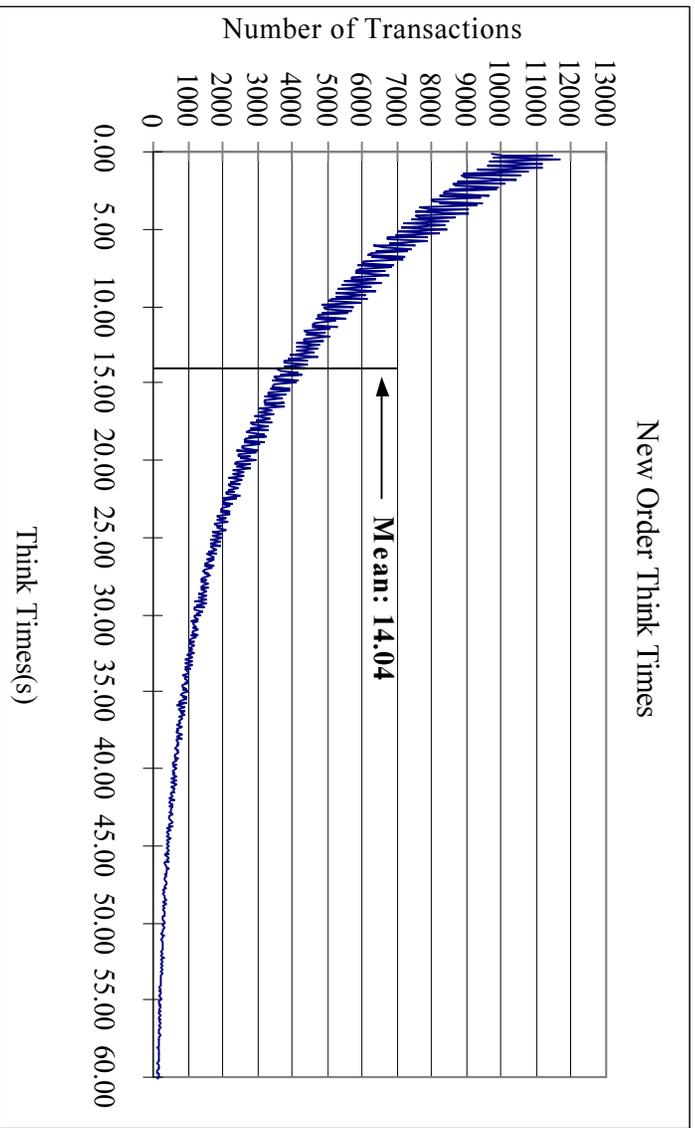
Figure 2.6 Response Time Performance vs. Throughput Curve



NEW-Order Think Time

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

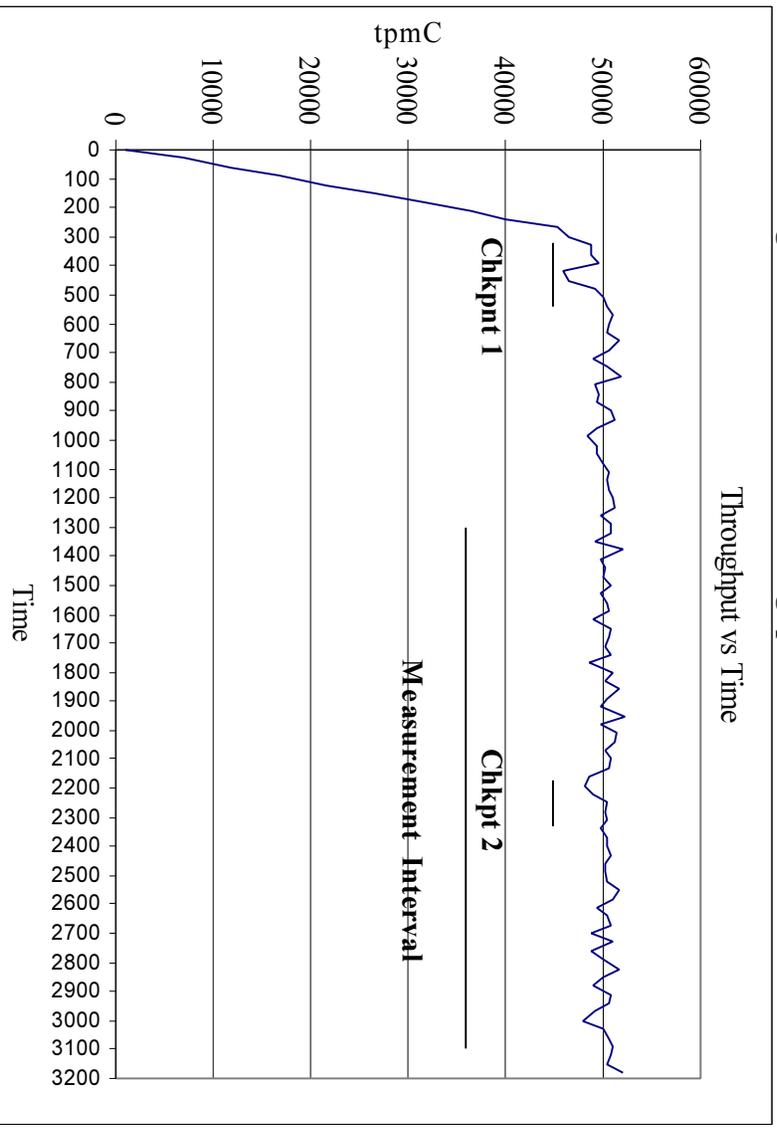
Figure 2.7 New-Order Think Time



New-Order Throughput vs. Elapsed Time

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

Figure 2.8 New Order Throughput vs. Time



Steady State

The method used to determine that the SUT had reached a steady state prior to commencing the measurement interval must be disclosed.

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 checkpoint in Oracle8i writes to disk all updated memory pages among nodes that have not been yet actually written to disk. Oracle8i recovery interval parameter was set to the maximum allowable value to perform checkpoint at specific intervals. A checkpoint script, which issues specified number of checkpoint at specified (30 minutes) intervals, was started after all users logged in and sending transactions.

Reproducibility

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

The reproducibility test result is taken from another, non-overlapping, measurement interval of the same duration as the reported interval. The throughput difference measured over that interval was within 0.13 % of reported interval result.

Measurement Period Duration

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

The reported measured interval was exactly 30 minutes long.

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 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.

Checkpoint Count and Location

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.

Initial checkpoint was started 6 minutes after the start of ramp-up. Second checkpoint was started 30 minutes after the 1st checkpoint. The time from the start of the Measurement interval was 14.5 minutes after. In accord with Clause 5.5.22, there is no checkpoint within the 'guard zones'¹1800/4=450 seconds from the beginning and end of the measurement interval.

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 proprietary. The RTE input parameters are listed in Appendix C.

Emulated Components

It must be demonstrated that the functionality and performance of the components being emulated in the Driver System are equivalent to 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 hardware and software functionality being performed on the Driver System and its interface to the SUT must be disclosed.

The diagrams in figure 1.1 and 1.2 show the tested and priced benchmark configurations

Network configurations and Driver system

The network configuration of both the tested services and 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.

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

The front-end clients were connected over one 100Mbps 100Base-T Ethernet segments to the back-end. Each front-end client were connected to the RTE over three 10Mbps 10Base-T Ethernet segments.

The priced PCs are also connected using 10Mbps Ethernet to the front-end clients.

Network Bandwidth

The bandwidth of the networks used in the tested/priced configuration must be disclosed.

The Ethernet used in the local area network (LAN) between the emulated terminals and the front-end system complies with the IEEE 802.3 standard and has a bandwidth of 10Mbps.

Operator Intervention

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

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 data. 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 and effective date(s) of price(s) must also be reported.

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

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 calculation must be reported. When the priced system included 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. The 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.

NEC Express5800 HV8600 is currently available.
Oracle®i v8.1.6.0 will be available by November 30th , 1999.

All other components used in the priced system are currently available.

Throughput, and Price Performance

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

The details of price calculation was shown in price shee in the top of this documentation.

- Maximum Qualified Throughput 50208.43 tpmC
- Price per tpmC : 94.05\$ per tpmC
- Total 5-year cost of ownership

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 being 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.*

None

Clause 9 : 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 InfoSizing 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 the charges for similar documents by the test sponsor. The report must be made available when results are made public. In order to use the phrase "TPC Benchmark™ 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
c/o Shanley Public Relations
777 North First Street, Suite 6000
San Jose, CA 95112-6311
or your local NEC / Packard Bell - NEC office.

Auditor's letter

INFO SIZING



Test Sponsors:

Frank Stephens
Manager: NT Benchmarks
Oracle Corporation
400 Oracle Parkway
Redwood Shores, CA 94065

Eiichi Kennai
NEC Corporation
3rd Development Dept
3rd Computers Software Dept
Fuchu City Tokyo 183, Japan

June 28, 1999

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

Platform: NEC EXPRESS5800 HV8600 c/s
Operating system: Microsoft Windows NT4.0 Enterprise Edition
Database Manager: Oracle8i Enterprise Edition Version 8.1.6.0
Transaction Manager: BEA Tuxedo Version 6.4 CFS

The results were:

CPU's	Memory	Disks	NewOrder 90% Response Time	tpmC
Server: Four nodes Express5800 HV8600				
8 x Intel Pentium III Xeon/node (500 MHz)	2 MB L2/cpu 4096 MB Main/node	672 x 9 GB 40 x 18 GB	2.36 Seconds	50,208.43
Eight Clients: Express5800 HX4600 (Specification for each)				
2 x Intel Pentium III Xeon (500 MHz)	2 MB L2/cpu 1024 MB Main	1 x 9 GB	n/a	n/a

In my opinion, these performance results were produced in compliance with the TPC requirements for Revision 3.4 of the benchmark. The following verification items were given special attention:

- The transactions were correctly implemented
- The database records were the proper size
- The database was properly scaled and populated
- The ACID properties were met
- Input data was generated according to the specified percentages

- The transaction cycle times included the required keying and think times
- The reported response times were correctly measured.
- At least 90% of all delivery transactions met the 80 Second completion time limit
- All 90% response times were under the specified maximums
- The measurement interval was representative of steady state conditions
- The reported measurement interval was 30 minutes (1800 seconds)
- One checkpoints per node was taken during the measurement interval
- Measurement repeatability was verified
- The 180 day storage requirement was correctly computed
- The system pricing was verified for major components and maintenance

Additional Audit Notes:

None.

Respectfully Yours,

A handwritten signature in black ink, appearing to read "François Raab". The signature is written in a cursive, flowing style with a long horizontal stroke at the end.

François Raab
President

Appendix A : Application Source Code

A1. Front-End Source Code, Scripts

TPCCAPI.H

```
# DESCRIPTION: Builds TPCC Application Code (Front and backend for
clients)

CPU                = i386

CLIB_DIR           = C:\program files\devstudio\vc\lib
#Oracle Installation Location is %ORACLE_HOME%
ORACLE_HOME       = c:\oracle\ora81

#TPCC Kit location is %TPCC_HOME%
#TPCC_HOME        = c:\usr\rkambo\oracletpcnt
TPCC_HOME         = ..\..\..\oracletpcnt

#TopEnd installation location is %TM_HOME% or
#TUXEDO installation

CFG="TUXEDO"

!IF "$(CFG)" != "TOPEND" && "$(CFG)" != "TUXEDO"
!MESSAGE Invalid configuration "$(CFG)" specified.
!MESSAGE You can specify a configuration when running NMAKE
!MESSAGE by defining the macro CFG on the command line. For example:
!MESSAGE
!MESSAGE NMAKE /f "makefile.oracle" CFG="TUXEDO"
!MESSAGE
!MESSAGE Possible choices for configuration are:
!MESSAGE
!MESSAGE TUXEDO - for tuxedo linking
!MESSAGE TOPEND - for topend linking
!MESSAGE
!ERROR An invalid configuration is specified.
!ENDIF

OCI_SUB           = oci
OCI_INCLUDE       = $(ORACLE_HOME)\$(OCI_SUB)\include
OCI_LIB_HOME     = $(ORACLE_HOME)\$(OCI_SUB)\lib\msvc
#OCI_LIB         = $(OCI_LIB_HOME)\ociw32.lib
OCI_LIB          = $(OCI_LIB_HOME)\oci.lib

DPB_LIB_DIR      = $(TPCC_HOME)\source\lib
DPB_LIB          = $(DPB_LIB_DIR)\dpcbnt.lib

CSRC_DIR        = $(TPCC_HOME)\source\client
SSRC_DIR        = $(TPCC_HOME)\source\server
```

```
!IF "$(CFG)" == "TUXEDO"
TM_HOME          = C:\TUXEDO
TM_LOC           = $(TM_HOME)\LIB
TM_LIB           = $(TM_LOC)\libtux.lib $(TM_LOC)\libgp.lib
TM_FW           =
TM_INC           = $(TM_HOME)\include
TM_INCLUDES     = /I$(TM_INC)

TM_CFLAGS       = /DTUX /D_CONSOLE /DWIN32 /D_TMSTHEADS
```

```
ORA_DEFINES     =
ORA_INCLUDES    =
APP_INCLUDES    = /I$(CSRC_DIR) /I$(SSRC_DIR)
```

```
#
# Destination for executables.
#
BIN_DIR         = c:\inetpub\wwwroot
!ENDIF
```

```
#Visual C++ libs
SOCKETS_LIB     = "$(CLIB_DIR)\wsock32.lib"
ADVAPI_LIB     = "$(CLIB_DIR)\advapi32.lib"
USER32_LIB     = "$(CLIB_DIR)\user32.lib"
MSVCRT_LIB     = "$(CLIB_DIR)\msvcrt.lib"
```

```
CL              = cl
CFLAGS          = -c -W3 -D X86 =1
COPT            = -Ox -Oy- -Gf -
#CDEBUG        = -Od -Z7 -Ge
```

```
CL_OPS          = $(CFLAGS) $(COPT) $(CDEBUG) $(ORA_DEFINES)
$(TM_CFLAGS) $(ORA_INCLUDES) $(TM_INCLUDES) $(APP_INCLUDES)
```

```
LINK           = link
#LFLAGS        = /debug:full /debugtype:cv /pdb:none
```

```
LIBRARIAN      = lib
#
# Private C compilation rule.
```

```
.c.obj:
    @$ (CL) $(CL_OPS) /Fo$@ $*.c
```

```
#
# Rule for building an "exe" in $(BIN_DIR) and using object files
in current
# directory.
#
}{.obj}{$(BIN_DIR)}.exe:
    @$ (LINK) $(LFLAGS) /OUT:$@ $(DPB_LIB) $(OCI_LIB) $**
```

```
!IF "$(CFG)" == "TUXEDO"
```

```
all: tpcc.dll
```

```
!ENDIF
```

```

tpcc.obj: $(CSRC_DIR)\tpcc.c $(CSRC_DIR)\tpccapi.h
$(CSRC_DIR)\tpccerr.h ; @$ (CL) /MT $(CL_OPS) $(CSRC_DIR)\tpcc.c

tpcc.dll: tpcc.obj $(TM_LIB) $(TM_FW) $(SOCKETS_LIB) $(USER32_LIB)
$(ADVAPI_LIB)
    @$ (LINK) /dll /def:"$(CSRC_DIR)\tpcc.def" $(LFLAGS)
/NODEFAULTLIB:"library" /OUT:$@ $**

clean:
    @for %f in ( *.obj *.lib $(BIN_DIR)\*.exe ) do if exist %f
del %f
    @for %f in ( *.obj *.lib ) do if exist $(DPB_LIB_DIR)\%f del
$(DPB_LIB_DIR)\%f

```

TPCCAPI.H

```

#ifndef TPCCAPI_H
#define TPCCAPI_H

/*=====+
|           Copyright (c) 1997 Oracle Corp, Redwood Shores, CA
|           OPEN SYSTEMS PERFORMANCE GROUP
|           All Rights Reserved
|=====+
| FILENAME
|   tpccapi.h
| DESCRIPTION
|   header file to 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

    //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)

    //error message structure used in ErrorMessage API
typedef struct _SERRORMSG
{
    int      iError;          //error id of message
    char     szMsg[80];      //message to sent to browser
} SERRORMSG;

    //This structure defines the data necessary to keep distinct
for each
    //terminal or client connection.
typedef struct _CLIENTDATA
{
    int      inUse;          //in use flag allows client entries to
be reused
    int      w_id;          //warehouse id assigned at welcome form
    int      d_id;          //district id assigned at welcome form
    int      iProcessRequestCounter; //request counter
    int      iSyncId;       //synchronization id
    int      iTickCount;    //time of last access;
    int      iTermId;       //terminal id of http stream connection
    char     szBuffer[4096]; //form buffer; HTML form built for
client here
    //for TOPEND
    char     type;
    int      retval;
    int      client;
    BOOL     bFailed;

#ifndef TOPEND
    long     *tp_structadr;
#else // assume tuxedo
    struct newstruct      *newOrderDataPtr;    //new order
Tuxedo buffer
    struct paystruct      *paymentDataPtr;     //payment
Tuxedo buffer
    struct ordstruct      *orderStatusDataPtr; //order status
Tuxedo buffer
    struct delstruct      *deliveryDataPtr;    //delivery
Tuxedo buffer
    struct stostruct      *stockLevelDataPtr;  //stock level
Tuxedo buffer
#endif
    union {
        struct newstruct      newOrderData;    //new order form data
        struct paystruct      paymentData;     //payment form data
        struct ordstruct      orderStatusData; //order status form
    } data
    struct delstruct      deliveryData;       //delivery form data
    struct stostruct      stocklevelData;     //stock level form data
} CLIENTDATA;

typedef CLIENTDATA *PCLIENTDATA; //pointer to client structure
    //This structure is used to define the operational interface
for
    //terminal id support
typedef struct _TERM
{
    int      iAvailable;    //total allocated terminal array entries
    int      iNext;         //next available terminal array element
    int      iMasterSyncId; //synchronization id
    BOOL     bInit;        //structure has been initialized flag
CLIENTDATA *pClientData; //pointer to allocated client data

```

```

void (*Init)(void); //API to initialize this structure
int (*Allocate)(void); //API to allocate new terminal entry;array
id returnd
void (*Restore)(void); //API to free terminal data
int (*Add)(EXTENSION_CONTROL_BLOCK *pECB,
char *pQueryString); //API to add a terminal id to array,
//this context will be passed
from the
//browser to the tpcc.dll in the
//TERMINID= key in the HTP string.
void (*Delete)(EXTENSION_CONTROL_BLOCK *pECB, int id); //API to free
//resources used
by a
//terminal array
//entry
} TERM;
typedef TERM *PTERM; //pointer to terminal structure type
//this structure allows the EXTENSION CONTROL BLOCK to be passed
to the
//msg and error handlers.
typedef struct _ECBINFO
{
int iTermId; //terminal id
int iSyncId; //browser sync id
BOOL bDeadlock; //deadlock condition flag
BOOL bFailed; //cleared before sql transaction, set
in err
EXTENSION_CONTROL_BLOCK *pECB; //inetsrv current connection
//information
} ECBINFO, *PECBINFO;
//function prototypes
#ifdef TOPEND // in tuxedo defined in atmi.h
extern int tpsvrinit(int argc, char *argv[]);
// argv arguments are int pid, char *uid, char *pwd, int
txntype
extern int NEWORDER(CLIENTDATA *jobData, NewOrderData *neword, int
deadlock);
extern int PAYMENT(CLIENTDATA *jobData, PaymentData *paydata, int
deadlock);
extern int ORDERSTATUS(CLIENTDATA *jobData, OrderStatusData *orddata,
int deadlock);
extern int STOCKLEVEL(CLIENTDATA *jobData, StockLevelData *stodata, int
deadlock);
extern int DELIVERY(CLIENTDATA *jobData, DeliveryData *deldata, int
deadlock);
#endif
#ifdef TUX
extern void NEWORDER(TPSVCINFO *msg);
extern void PAYMENT(TPSVCINFO *msg);
extern void ORDERSTATUS(TPSVCINFO *msg);
extern void STOCKLEVEL(TPSVCINFO *msg);
extern void DELIVERY(TPSVCINFO *msg);
#endif

```

```

BOOL APIENTRY DllMain(HANDLE hModule, DWORD ul_reason_for_call, LPVOID
lpReserved);
static BOOL IsValidTermId(int TermId);
BOOL ProcessQueryString(EXTENSION_CONTROL_BLOCK *pECB, int *pCmd, int
*pFormId, int *pTermId, int *pSyncId);
void NewOrderForm(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int
iTermId, int iSyncId);
void PaymentForm(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int
iTermId, int iSyncId);
void DeliveryForm(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int
iTermId, int iSyncId);
void OrderStatusForm(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int
iTermId, int iSyncId);
void StockLevelForm(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int
iTermId, int iSyncId);
void Exitcmd(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int iTermId,
int iSyncId);
void SubmitCmd(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int iTermId,
int iSyncId);
void BeginCmd(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int iTermId,
int iSyncId);
void ProcessCmd(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int iTermId,
int iSyncId);
void ClearCmd(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int iTermId,
int iSyncId);
void MenuCmd(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int iTermId,
int iSyncId);
static void WriteZString(EXTENSION_CONTROL_BLOCK *pECB, char *szStr);
static void h_printf(EXTENSION_CONTROL_BLOCK *pECB, char *format, ...);
void ErrorMessage(EXTENSION_CONTROL_BLOCK *pECB, int iError, int
iErrorType, char *szMsg, int iTermId, int iSyncId);
static BOOL GetKeyValue(char *pQueryString, char *pKey, char *pValue,
int iMax);
static void TermInit(void);
static void TermRestore(void);
static int TermAllocate(void);
static int TermAdd(EXTENSION_CONTROL_BLOCK *pECB, char *pQueryString);
static void TermDelete(EXTENSION_CONTROL_BLOCK *pECB, int id);
BOOL Init(EXTENSION_CONTROL_BLOCK *pECB, int iTermId, int iSyncId, char
*szServer, char *szUser, char *szPassword, char *szDatabase);
static BOOL Close(EXTENSION_CONTROL_BLOCK *pECB, int iTermId, int
iSyncId);
static void FormatString(char *szDest, char *szPic, char *szSrc);
static char *MakeStockLevelForm(int iTermId, int iSyncId, BOOL bInput);
static char *MakeMainMenuForm(int iTermId, int iSyncId);
static char *MakeWelcomeForm(void);
static char *MakeNewOrderForm(int iTermId, int iSyncId, BOOL bInput,
BOOL bValid, char *execution status);
static char *MakePaymentForm(int iTermId, int iSyncId, BOOL bInput);
static char *MakeOrderStatusForm(int iTermId, int iSyncId, BOOL
bInput);
static char *MakeDeliveryForm(int iTermId, int iSyncId, BOOL bInput,
char *execution status);
void UtilStrCpy(char *pDest, char *pSrc, int n);
static void ProcessNewOrderForm(EXTENSION_CONTROL_BLOCK *pECB, int
iTermId, int iSyncId);
static void ProcessPaymentForm(EXTENSION_CONTROL_BLOCK *pECB, int
iTermId, int iSyncId);
static void ProcessOrderStatusForm(EXTENSION_CONTROL_BLOCK *pECB, int
iTermId, int iSyncId);
static void ProcessDeliveryForm(EXTENSION_CONTROL_BLOCK *pECB, int
iTermId, int iSyncId);
static void ProcessStockLevelForm(EXTENSION_CONTROL_BLOCK *pECB, int
iTermId, int iSyncId);

```

```

static int GetNewOrderData(LPSTR lpszQueryString, NewOrderData
*pNewOrderData);
static int GetPaymentData(LPSTR lpszQueryString, PaymentData
*pPaymentData);
static int GetOrderStatusData(LPSTR lpszQueryString, OrderStatusData
*pOrderStatusData);
static BOOL ReadRegistrySettings(void);
static BOOL IsNumeric(char *ptr);
static void FormatHTMLString(char *szBuff, char *szStr, int iLen);
static void Log(char *szType, char *szStr);
int do_neworder(NewOrderData *pNOData);
int do_payment(PaymentData *pPAYData);
int do_orderstatus(OrderStatusData *pOSData);
int do_delivery(pDeliveryData pDelivery);
int do_stocklevel(StockLevelData *pSTOData);
int end_neworder(NewOrderData *pNOData);
int end_payment(PaymentData *pPAYData);
int end_orderstatus(OrderStatusData *pOSData);
int end_stocklevel(StockLevelData *pSTOData);
int do_all_txns(int txn);
int do_disconnect( );

```

```
#endif /* TPCAPI_H */
```

TPCCERR.H

```

#ifndef TPCERR_H
#define TPCERR_H

/*=====+
|          Copyright (c) 1997 Oracle Corp, Redwood Shores, CA
|          All Rights Reserved
|=====+
| FILENAME
|   tpccerr.h
| DESCRIPTION
|   Include file for TPC-C benchmark programs.
+=====*/

#define TP_MAX_RETRIES 1
#define ERR_TYPE_WEBDLL 1
#define ERR_TYPE_SQL 2
#define ERR_TYPE_OCI 3
#define ERR_TYPE_TUXEDO 4
#define ERR_DB_SUCCESS 0
/* Database transaction succeeded */

#define ERR_DB_ERROR 1 /* A database error has occurred
*/
#define ERR_TRANSPORT_ERROR 2 /* A transport error has occurred
*/
#define ERR_DB_INTERFACE 3 /* An error occurred setting up */
/* connection to DB */
#define ERR_DB_DEADLOCK_LIMIT 4 /* The DB deadlock retry limit
was */
/* reached */
#define ERR_DB_NOT_COMMITTED 5 /* Transaction not committed */
#define ERR_DB_DEAD 6 /* Database connection is invalid
*/
#define ERR_SUCCESS 1000 /*Success, no error.

```

```

#define ERR_COMMAND_UNDEFINED 1001 /*Command undefined.
#define ERR_NOT_IMPLEMENTED_YET 1002 /*Not Implemented Yet.
#define ERR_CANNOT_INIT_TERMINAL 1003 /*Cannot initialize
client connection.
#define ERR_OUT_OF_MEMORY 1004 /*insufficient memory.
#define ERR_NEW_ORDER_NOT_PROCESSED 1005 /*Cannot process new Order
form.
#define ERR_PAYMENT_NOT_PROCESSED 1006 /*Cannot process payment
form.
#define ERR_NO_SERVER_SPECIFIED 1007 /*No Server name
specified.
#define ERR_ORDER_STATUS_NOT_PROCESSED 1008 /*Cannot process order
status form.
#define ERR_W_ID_INVALID 1009 /*Invalid Warehouse ID.
#define ERR_CAN_NOT_SET_MAX_CONNECTIONS 1010 /*Insufficient
memory to
//allocate # connections.
#define ERR_NOSUCH_CUSTOMER 1011 /*No such customer.
#define ERR_D_ID_INVALID 1012 /*Invalid District ID Must be 1
to 10.
#define ERR_MAX_CONNECT_PARAM 1013 /*Max client connections
exceeded,
//run install to increase.
#define ERR_INVALID_SYNC_CONNECTION 1014 /*Invalid Terminal Sync ID.
#define ERR_INVALID_TERMID 1015 /*Invalid Terminal ID.
#define ERR_PAYMENT_INVALID_CUSTOMER 1016 /*Payment Form, No such
Customer.
#define ERR_SQL_OPEN_CONNECTION 1017 /*SQLOpenConnection API
Failed.
#define ERR_STOCKLEVEL_MISSING_THRESHOLD_KEY 1018 /*Stock Level
missing
//Threshold key "TT*".
#define ERR_STOCKLEVEL_THRESHOLD_INVALID 1019 /*Stock Level Threshold
invalid
//data type range = 1 - 99.
#define ERR_STOCKLEVEL_THRESHOLD_RANGE 1020 /*Stock Level Threshold
out of
//range, range must be 1 -
99.
#define ERR_STOCKLEVEL_NOT_PROCESSED 1021 /*Stock Level not processed.
#define ERR_NEWORDER_FORM_MISSING_DID 1022 /*NewOrder miss District
key "DID*".
#define ERR_NEWORDER_DISTRICT_INVALID 1023 /*NewOrder District ID
Invalid
//range 1 - 10.
#define ERR_NEWORDER_DISTRICT_RANGE 1024 /*New Order District ID out
of
//Range. Range = 1 - 10.
#define ERR_NEWORDER_CUSTOMER_KEY 1025 /*New Order missing Customer
key "CID*".
#define ERR_NEWORDER_CUSTOMER_INVALID 1026 /*NewOrder customer id
invalid data
//type, range = 1 to 3000.
#define ERR_NEWORDER_CUSTOMER_RANGE 1027 /*New Order customer id out
of
//range, range = 1 to 3000.
#define ERR_NEWORDER_MISSING_IID_KEY 1028 /*NewOrder missng ItemId
key "IID*".
#define ERR_NEWORDER_ITEM_BLANK_LINES 1029 /*NewOrder blank order
lines all
//orders must be continuous.
#define ERR_NEWORDER_ITEMID_INVALID 1030 /*New Order Item Id is
wrong data
//type, must be numeric.

```

```

#define ERR_NEWORDER_MISSING_SUPPW_KEY 1031 //NewOrder missing Supp_W
key
//SP###".
#define ERR_NEWORDER_SUPPW_INVALID 1032 //New Order Supp_W
invalid data
//type must be numeric.
#define ERR_NEWORDER_MISSING_QTY_KEY 1033 //NewOrder Missing Qty key
"Qty###".
#define ERR_NEWORDER_QTY_INVALID 1034 //NewOrder Qty invalid must
be
//numeric range 1 - 99.
#define ERR_NEWORDER_SUPPW_RANGE 1035 //New Order Supp_W
value out of
//range range = 1 - Max
Warehouses.
#define ERR_NEWORDER_ITEMID_RANGE 1036 //New Order Item Id is
out of range.
//Range = 1 to 999999.
#define ERR_NEWORDER_QTY_RANGE 1037 //New Order Qty is out of range.
//Range = 1 to 99.
#define ERR_PAYMENT_DISTRICT_INVALID 1038 //Payment District ID is
invalid
//must be 1 - 10.
#define ERR_NEWORDER_SUPPW_WITHOUT_ITEMID 1039 //NewOrder Supp_W field
entered
//without a corosponding
Item_Id.
#define ERR_NEWORDER_QTY_WITHOUT_ITEMID 1040 //NewOrder Qty entered
without a
//corosponding Item_Id.
#define ERR_NEWORDER_NOITEMS_ENTERED 1041 //NewOrder Blank Items
between
//items, items must be
continuous.
#define ERR_PAYMENT_MISSING_DID_KEY 1042 //Payment missing District
Key "DID*".
#define ERR_PAYMENT_DISTRICT_RANGE 1043 //Payment District Out
of range,
//range = 1 - 10.
#define ERR_PAYMENT_MISSING_CID_KEY 1044 //Payment missing Customer
Key "CID*".
#define ERR_PAYMENT_CUSTOMER_INVALID 1045 //PaymentCustomer data type
invalid,
//must be numeric.
#define ERR_PAYMENT_MISSING_CLT 1046 //Payment missing
Customer Last Name
//Key "CLT*".
#define ERR_PAYMENT_LAST_NAME_TO_LONG 1047 //Payment Customer last
name
// longer than 16 characters.
#define ERR_PAYMENT_CUSTOMER_RANGE 1048 //Payment Customer ID out
of range,
//must be 1 to 3000.
#define ERR_PAYMENT_CID_AND_CLT 1049 //Payment Customer ID and
Last Name
//entered must be one or other.
#define ERR_PAYMENT_MISSING_CDI_KEY 1050 //Payment missing Customer
district
//key "CDI*".
#define ERR_PAYMENT_CDI_INVALID 1051 //Payment Customer
district invalid
//must be numeric.
#define ERR_PAYMENT_CDI_RANGE 1052 //Payment Customer district out
of

```

```

//range must be 1 - 10.
#define ERR_PAYMENT_MISSING_CWI_KEY 1053 //Payment missing Customer
Warehouse
//key "CWI*".
#define ERR_PAYMENT_CWI_INVALID 1054 //Payment Customer
Warehouse invalid
//must be numeric.
#define ERR_PAYMENT_CWI_RANGE 1055 //Payment Customer Warehouse out
of
//range, 1 to Max Warehouses.
#define ERR_PAYMENT_MISSING_HAM_KEY 1056 //Payment missing Amount key
"HAM*".
#define ERR_PAYMENT_HAM_INVALID 1057 //Payment Amount invalid
data type
//must be numeric.
#define ERR_PAYMENT_HAM_RANGE 1058 //Payment Amount out of range,
//0 - 9999.99.
#define ERR_ORDERSTATUS_MISSING_DID_KEY 1059 //OrderStatus
missing District
//key "DID*".
#define ERR_ORDERSTATUS_DID_INVALID 1060 //Order Status District
invalid,
//value must be numeric 1 - 10.
#define ERR_ORDERSTATUS_DID_RANGE 1061 //Order Status District
out of range
//must be 1 - 10.
#define ERR_ORDERSTATUS_MISSING_CID_KEY 1062 //OrderStatus missing
Customer
//key "CID*".
#define ERR_ORDERSTATUS_MISSING_CLT_KEY 1063 //OrderStatus missing
Customer
//Last Name key "CLT*".
#define ERR_ORDERSTATUS_CLT_RANGE 1064 //Order Status Customer
last name
//longer than 16 characters.
#define ERR_ORDERSTATUS_CID_INVALID 1065 //Order Status Customer ID
invalid,
//range must be numeric 1 - 3000.
#define ERR_ORDERSTATUS_CID_RANGE 1066 //Order Status Customer
ID out of
//range must be 1 - 3000.
#define ERR_ORDERSTATUS_CID_AND_CLT 1067 //Order Status Customer ID
and
//LastName entered must be only
one."
#define ERR_DELIVERY_MISSING_OCD_KEY 1068 //Delivery missing Carrier
ID key
//\OCD\".
#define ERR_DELIVERY_CARRIER_INVALID 1069 //Delivery Carrier ID
invalid must
//be numeric 1 - 10.
#define ERR_DELIVERY_CARRIER_ID_RANGE 1070 //Delivery Carrier ID out
of range
//must be 1 - 10.
#define ERR_PAYMENT_MISSING_CLT_KEY 1071 //Payment missing Customer
Last
//Name key "CLT*".
#define TOPEND_SEND_ERROR 1072 //TOPEND client send error".
#define TOPEND_RECEIVE_ERROR 1073 //TOPEND client receive error".
#define ERR_TPINIT_BAD 5001 //Tuxedo tpinit Failed."
#define ERR_TPALLOC_BAD 5002 //Tuxedo tmalloc Failed."
#define ERR_TPCALL_BAD 5003 //Tuxedo tpcall Failed."
#endif /* TPCCERR_H */

```

HTTPEXT.H

```
/*  
* Copyright (c) 1995 Process Software Corporation  
* Copyright (c) 1995 Microsoft Corporation  
*  
* Module Name : HttpExt.h  
* Abstract :  
*  
* This module contains the structure definitions and prototypes  
for the  
* version 1.0 HTTP Server Extension interface.  
*  
*****/  
#ifndef _HTTPEXT_H  
#define _HTTPEXT_H  
#include <windows.h>  
#ifdef __cplusplus  
extern "C" {  
#endif  
#define HSE_VERSION_MAJOR 1 // major version of this  
spec  
#define HSE_VERSION_MINOR 0 // minor version of this  
spec  
#define HSE_LOG_BUFFER_LEN 80  
#define HSE_MAX_EXT_DLL_NAME_LEN 256  
typedef LPVOID HCONN;  
// the following are the status codes returned by the Extension DLL  
#define HSE_STATUS_SUCCESS 1  
#define HSE_STATUS_SUCCESS_AND_KEEP_CONN 2  
#define HSE_STATUS_PENDING 3  
#define HSE_STATUS_ERROR 4  
// The following are the values to request services with the  
ServerSupportFunction.  
// Values from 0 to 1000 are reserved for future versions of the  
interface  
#define HSE_REQ_BASE 0  
#define HSE_REQ_SEND_URL_REDIRECT_RESP ( HSE_REQ_BASE + 1 )  
#define HSE_REQ_SEND_URL ( HSE_REQ_BASE + 2 )  
#define HSE_REQ_SEND_RESPONSE_HEADER ( HSE_REQ_BASE + 3 )  
#define HSE_REQ_DONE_WITH_SESSION ( HSE_REQ_BASE + 4 )  
#define HSE_REQ_END_RESERVED 1000  
//  
// These are Microsoft specific extensions  
//  
#define HSE_REQ_MAP_URL_TO_PATH  
(HSE_REQ_END_RESERVED+1)  
#define HSE_REQ_GET_SSPI_INFO  
(HSE_REQ_END_RESERVED+2)  
//  
// passed to GetExtensionVersion  
//  
typedef struct _HSE_VERSION_INFO  
{  
    DWORD dwExtensionVersion;  
    CHAR lpszExtensionDesc[HSE_MAX_EXT_DLL_NAME_LEN];  
};
```

```
} HSE_VERSION_INFO, *LPHSE_VERSION_INFO;  
//  
// passed to extension procedure on a new request  
//  
typedef struct _EXTENSION_CONTROL_BLOCK  
{  
    DWORD cbSize; // size of this struct.  
    DWORD dwVersion; // version info of this spec  
    HCONN ConnID; // Context number not to be  
modified!  
    DWORD dwHttpStatusCode; // HTTP Status code  
    CHAR lpszLogData[HSE_LOG_BUFFER_LEN]; // null terminated log  
info specific to this Extension DLL  
    LPSTR lpszMethod; // REQUEST METHOD  
    LPSTR lpszQueryString; // QUERY_STRING  
    LPSTR lpszPathInfo; // PATH_INFO  
    LPSTR lpszPathTranslated; // PATH_TRANSLATED  
    DWORD cbTotalBytes; // Total bytes indicated from  
client  
    DWORD cbAvailable; // Available number of bytes  
    LPBYTE lpbData; // pointer to cbAvailable bytes  
    LPSTR lpszContentType; // Content type of client data  
    BOOL (WINAPI * GetServerVariable) ( HCONN hConn,  
LPSTR lpszVariableName,  
LPVOID lpvBuffer,  
LPDWORD lpdwSize );  
    BOOL (WINAPI * WriteClient) ( HCONN ConnID,  
LPVOID Buffer,  
LPDWORD lpdwBytes,  
DWORD dwReserved );  
    BOOL (WINAPI * ReadClient) ( HCONN ConnID,  
LPVOID lpvBuffer,  
LPDWORD lpdwSize );  
    BOOL (WINAPI * ServerSupportFunction) ( HCONN hConn,  
DWORD dwHSERequest,  
LPVOID lpvBuffer,  
LPDWORD lpdwSize,  
LPDWORD lpdwDataType );  
};  
EXTENSION_CONTROL_BLOCK, *LPEXTENSION_CONTROL_BLOCK;  
//  
// these are the prototypes that must be exported from the extension  
DLL  
//  
_declspec (dllexport)  
BOOL WINAPI GetExtensionVersion( HSE_VERSION_INFO *pVer );  
_declspec (dllexport)  
DWORD WINAPI HttpExtensionProc( EXTENSION_CONTROL_BLOCK *pECB );  
// the following type declarations are for the server side  
typedef BOOL (WINAPI * PFN_GETEXTENSIONVERSION) ( HSE_VERSION_INFO  
*pVer );  
typedef DWORD (WINAPI *  
PFN_HTTPEXTENSIONPROC ) ( EXTENSION_CONTROL_BLOCK *pECB );  
#ifdef __cplusplus  
}  
#endif  
#endif // end definition _HTTPEXT_H
```

TPCC.C

```
/*  
| Copyright (c) 1997 Oracle Corp, Redwood Shores, CA |
```

```

| All Rights Reserved |
+-----+
| FILE: TPCC.C |
| DESCRIPTION: Main module for TPCC.DLL |
| Master created: 15 Apr 97 |
*-----*/

#ifdef TOPEND
#define TP_MT_SOURCE
#define STRICT
#endif

#define LOCAL_ALLOC 1 /* force local alloc to be true */

#include <windows.h>
#include <process.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 "tpccerr.h"
#include "tpcc_info.h"
#include "httpext.h"

// #include "getval.h"

#ifdef TOPEND
#include <tp_csi.h>
BOOL TP_InitDone = FALSE;
#define DllExport _declspec(dllexport)
DllExport LONG tp_ChangeToGroup(LPTSTR, DWORD, DWORD);

tp_dif_structs_t *client_dif;

/*
 * Topend structure offset.
 */
#define TOPEND_STRUCT_OFFSET 4124

#endif

#ifdef TUX
#include <tmenv.h>
#include <xa.h>
#include <atmi.h>
#endif

#include "tpccapi.h"

static TPINIT *tpinf;
static DWORD TLSIsTpInitKey;
static DWORD TLSNewOrderKey;
static DWORD TLSPaymentKey;
static DWORD TLSOrderStatusKey;
static DWORD TLSDeliveryKey;
static DWORD TLSStockLevelKey;
static int ThrTpInit();

```

```

char szServer[32] = { 0 }; //global variables used with
this DLL
char szUser[32] = { 0 };
char szPassword[32] = { 0 };
char szDatabase[32] = "tpcc";
char szIID[16][8] =
{"IID00*", "IID01", "IID02", "IID03", "IID04", "IID05", "IID06", "IID07", "IID08", "IID09", "IID10", "IID11", "IID12", "IID13", "IID14", "IID15"};
char szSP[16][8] =
{"SP00*", "SP01", "SP02", "SP03", "SP04", "SP05", "SP06", "SP07", "SP08", "SP09", "SP10", "SP11", "SP12", "SP13", "SP14", "SP15"};
char szQty[16][8] =
{"Qty00*", "Qty01", "Qty02", "Qty03", "Qty04", "Qty05", "Qty06", "Qty07", "Qty08", "Qty09", "Qty10", "Qty11", "Qty12", "Qty13", "Qty14", "Qty15"};

BOOL bLog = FALSE;
BOOL dLog = FALSE;

#ifdef TOPEND
int wait_time;
int inactivity_time = 1500;
#endif

BOOL bGeneric = FALSE;
/* add structure to get delivery time stamp */
struct _timeb timebuffer;

int iThreads = 5;
int iMaxWareHouses = 625;
int iDelayMs = 100;
short iDeadlockRetry = (short)3;
short iMaxConnections = (short)625;
int iErrVal = 0;

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

//defined command string functions, called via CMD=command http string
from html client.

void (*DoCmd[])(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int iTermId,
int iSyncId) =
{
    NewOrderForm,
    PaymentForm,
    DeliveryForm,
    OrderStatusForm,
    StockLevelForm,
    Exitcmd,
    SubmitCmd,
    BeginCmd,
    ProcessCmd,
    MenuCmd,
    ClearCmd
};

//Terminal client id structure and interface definition

```

```

TERM Term = { 0, 0, 0, FALSE, NULL, TermInit, TermAllocate,
TermRestore, TermAdd, TermDelete };

//welcome to tpc-c html form buffer, this is first form client sees.
static char      *szWelcomeForm =      "<HTML>"

        "<HEAD><TITLE>Welcome To TPC-C</TITLE></HEAD><BODY>"
Identify your Warehouse and District for this session.<BR>"
ACTION="\tpcc.dll" METHOD="\GET\">"
TYPE="\hidden" NAME="\STATUSID" VALUE="\0\">"
TYPE="\hidden" NAME="\ERROR" VALUE="\0\">"
TYPE="\hidden" NAME="\FORMID" VALUE="\1\">"
TYPE="\hidden" NAME="\TERMIN" VALUE="\-2\">"
TYPE="\hidden" NAME="\SYNCID" VALUE="\0\">"
ID <INPUT NAME="\w_id" SIZE=4><BR>"
ID <INPUT NAME="\d_id" SIZE=2><BR>"
TYPE="\submit" NAME="\CMD" VALUE="\Submit\">"

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

static char szTpccLogPath[256]; //path to html log file if logging
turned on in registry.
static char szErrorLogPath[256]; //path to error log file.

static CRITICAL_SECTION      CriticalSection;
static CRITICAL_SECTION      ErrorLogCriticalSection;

static int                    bTpccExit;
//exit delivery disconnect loop as dll exiting.

#ifdef TOPEND
typedef struct DTEEntry {
HANDLE      hSync;
struct DTEEntry *next;
tp_dif_structs_t *dif_struct;
char        *message_buffer;
long        *usr_dlg_ptr;
long        usr_dlg_storage;
long        buffer_length;
int         rc;
} DialogTableEntry_t;

//a place to put free DTEEntry cells
static DialogTableEntry_t DTFreeList={NULL,NULL,NULL,NULL,NULL,0,0,0};

//a place to put active dialogs
static DialogTableEntry_t
DTActiveList={NULL,NULL,NULL,NULL,0,0,0};

//can use for tuning, keep high
#define RECEIVE THREAD_PRIORITY_THREAD_PRIORITY_TIME_CRITICAL
#define MT_LOG_ERROR {

```

```

char buf[160];\
sprintf(buf,"Problem at %d in %s\n",__LINE__,__FILE__);\
tp_system_log_text("TEthreads",buf);\
}

static void TEREceiveThread(void *ptr);
DialogTableEntry_t *GetDTE();
DialogTableEntry_t *FindOnList(long
tp_user_dialog_id,DialogTableEntry_t *head);
void RemoveFromList (DialogTableEntry_t *DTE, DialogTableEntry_t
*head);
void AddToList (DialogTableEntry_t *DTE, DialogTableEntry_t *head);
//prototypes for calls used in the application

int tp_mt_initialize(tp_application_info_t *application_info,
tp_func_struct_t *function_array[],
long member_functions);

int tp_mt_client_signon (tp_dialogue_info_t *info,
tp_dialogue_user_t *client,
long inactivity_time,
tp_service_name_t *service,
tp_input_format_t *input_format,
long message_length,
char *message_text);

int tp_mt_client_send (tp_dialogue_info_t *info,
tp_service_name_t *service,
tp_input_format_t *input_format,
long message_length,
char *message_text);

int tp_mt_client_receive (tp_dialogue_info_t *info,
long wait_time,
tp_output_format_t *output_format,
tp_service_name_t *service,
tp_location_t *location,
long *buffer_length,
char *message_buffer);

#endif

/* FUNCTION: BOOL WINAPIENTRY DllMain(HANDLE hModule, DWORD
ul_reason_for_call,
LPVOID lpReserved)
*
* 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.
* Connections are sent to this function as thread attachments.
*
* ARGUMENTS: HANDLE hModule module handle
* DWORD ul_reason_for_call reason for call
* LPVOID lpReserved reserved for future use
*
* RETURNS: BOOL FALSE errors ocurred in
initialization
* TRUE DLL successfully
initialized
*
* COMMENTS: None

```

```

*
*/
BOOL WINAPI DllMain(HANDLE hModule, DWORD ul_reason_for_call, LPVOID
lpReserved)
{
    int i;
    static SECURITY_ATTRIBUTES sa;
    static PSECURITY_DESCRIPTOR pSD;

#ifdef DEBUG_ENTRY
    DebugBreak();
#endif

    switch( ul_reason_for_call )
    {
        case DLL_PROCESS_ATTACH:
            if ( ReadRegistrySettings() )
            {
                MessageBox(NULL, "Cannot Find TPCC Key in registry
(run install.exe).",
                "Init", MB_OK | MB_ICONSTOP);
                return FALSE;
            }

            InitializeCriticalSection(&CriticalSection);
            InitializeCriticalSection(&ErrorLogCriticalSection);

            (*Term.Init)();
            if ( !(*Term.Allocate)() )
            {
                MessageBox(NULL, "Error Trm.Allocate().", "Init",
                MB_OK |
                MB_ICONSTOP);
                return FALSE;
            }

            for(i=Term.iNext; i<Term.iAvailable; i++)
            {
                Term.pClientData[i].inUse = 0;
                Term.pClientData[0].inUse = 1;

                later
                TLSIsTpInitedKey = TlsAlloc(); // check for failure
                TLSNewOrderKey = TlsAlloc();
                TLSPaymentKey = TlsAlloc();
                TLSOrderStatusKey = TlsAlloc();
                TLSDeliveryKey = TlsAlloc();
                TLSStockLevelKey = TlsAlloc();
                // assumption:value inited to 0
                break;
            }

        case DLL_THREAD_ATTACH:
            break;

        case DLL_THREAD_DETACH:
            break;

        case DLL_PROCESS_DETACH:
            if ( pSD )
                free( pSD );

            bTpccExit = TRUE;
    }
}

```

```

(*Term.Restore)();

DeleteCriticalSection(&CriticalSection);
DeleteCriticalSection(&ErrorLogCriticalSection);

TlsFree(TLSIsTpInitedKey);
TlsFree(TLSNewOrderKey);
TlsFree(TLSPaymentKey);
TlsFree(TLSOrderStatusKey);
TlsFree(TLSDeliveryKey);
TlsFree(TLSStockLevelKey);

        break;
    }
    return TRUE;
}

/* FUNCTION: BOOL WINAPI GetExtensionVersion(HSE_VERSION_INFO *pVer)
*
* 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.
* COMMENTS: None
*/
_declspec (dllexport)
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: DWORD WINAPI HttpExtensionProc(EXTENSION_CONTROL_BLOCK
*pECB)
*
* 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
*
*/
_declspec (dllexport)
DWORD WINAPI HttpExtensionProc(EXTENSION_CONTROL_BLOCK *pECB)
{
    int iCmd, FormId, TermId, iSyncId;
    FILE *fp;

    static BOOL bReadRegistry = FALSE;

    if ( iMaxConnections == -1 )
    {
        ErrorMessage(pECB, ERR_CAN_NOT_SET_MAX_CONNECTIONS,
                     ERR_TYPE_WEBDLL, NULL, -1, -1);
        return HSE_STATUS_SUCCESS;
    }

    //if registry setting is for html logging then show http string
    passed in.
    if ( bLog )
    {
        SYSTEMTIME systemTime;

        fp = fopen(szTpccLogPath, "ab");

        GetLocalTime(&systemTime);

        fprintf(fp, "* QUERY
* %2.2d/%2.2d/%2.2d %2.2d:%2.2d:%2.2d\r\n\r\n%s\r\n\r\n",
              systemTime.wYear, systemTime.wMonth,
systemTime.wDay,
              systemTime.wHour, systemTime.wMinute,
systemTime.wSecond,
              pECB->lpszQueryString);
        fclose(fp);
    }

    //process http query
    if ( !ProcessQueryString(pECB, &iCmd, &FormId, &TermId,
&iSyncId) )
    {
        if ( TermId < 0 ){
            FILE *fur;
            fur =
fopen("C:\\temp\\fur.txt", "ab");
            fprintf(fur,
"- Invalid termid = %d \n", TermId);
            fclose(fur);
            ErrorMessage(pECB, ERR_INVALID_TERMID, ERR_TYPE_WEBDLL,
NULL, TermId, iSyncId);
        }
        else
            ErrorMessage(pECB, ERR_COMMAND_UNDEFINED,
ERR_TYPE_WEBDLL, NULL, TermId, iSyncId);
        return HSE_STATUS_SUCCESS_AND_KEEP_CONN;
    }
}

```

```

if ( TermId != 0 )
{
    if ( !IsValidTermId(TermId) )
    {
        FILE *fur;
        fur =
fopen("C:\\temp\\fur.txt", "ab");
        fprintf(fur,
"non0 Invalid termid = %d \n", TermId);
        fclose(fur);
    }
    ErrorMessage(pECB, ERR_INVALID_TERMID, ERR_TYPE_WEBDLL,
NULL, TermId, iSyncId);
    return HSE_STATUS_SUCCESS_AND_KEEP_CONN;
}

//must have a valid syncid here since termid is valid
if ( iSyncId < 1 || iSyncId !=
Term.pClientData[TermId].iSyncId )
{
    ErrorMessage(pECB, ERR_INVALID_SYNC_CONNECTION,
ERR_TYPE_WEBDLL, NULL, TermId, iSyncId);
    return HSE_STATUS_SUCCESS_AND_KEEP_CONN;
}

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

//go execute http: command
(*DoCmd[iCmd])(pECB, FormId, TermId, iSyncId);

//finish up and keep connection
return HSE_STATUS_SUCCESS_AND_KEEP_CONN;
}

/* FUNCTION: static BOOL IsValidTermId(int TermId)
*
* PURPOSE: This function checks to see of the passed in terminal id
is valid.
*
* ARGUMENTS: int TermId client terminal id
*
* RETURNS: BOOL FALSE Terminal ID Invalid
TRUE Terminal ID valid
*
* COMMENTS: None
*/

static BOOL IsValidTermId(int TermId)
{
    return (BOOL) ( TermId > 0 && TermId <= Term.iAvailable &&
Term.pClientData[TermId].inUse );
}

```

```

/* FUNCTION: BOOL ProcessQueryString(EXTENSION_CONTROL_BLOCK *pECB, int
*pCmd,
*
*                               int *pFormId, int *pTermId, int
*pSyncId)
*
* PURPOSE: This function extracts the relevent information out of the
http
*
*           command passed in from the browser.
*
* ARGUMENTS:  EXTENSION_CONTROL_BLOCK *pECB   structure pointer to
passed in
*
*                               internet service
information.
*
*           int *pCmd   returned command id
*           int *pFormId returned active form
client
*
*                               browser is on
terminal id
*           int *pTermId returned client
*
* RETURNS:    BOOL   FALSE   success
*             TRUE    command passed in is invalid
*
* 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.
*/

```

```

BOOL ProcessQueryString(EXTENSION_CONTROL_BLOCK *pECB, int *pCmd, int
*pFormId,
*                               int *pTermId, int *pSyncId)
{
    char *ptr;
    char szBuffer[25];
    char szTmp[25];
    char *dest = szBuffer;
    int i;

    if ( (ptr = strstr(pECB->lpszQueryString, "FORMID=")) )
        *pFormId = *(ptr+7) & 0x0F;

    if ( (ptr = strstr(pECB->lpszQueryString, "TERMID=")) )
    {
        *pTermId = atoi((ptr+7));

        if ( *pTermId == 0 )           //terminal id 0 used internally
            *pTermId = -1;
        if ( *pTermId == -2 )         //login screen
            *pTermId = 0;
    }
    else
        *pTermId = 0;

    if ( (ptr = strstr(pECB->lpszQueryString, "SYNCID=")) )
        *pSyncId = atoi((ptr+7));
    else
        *pSyncId = 0;

    if ( !(ptr = strstr(pECB->lpszQueryString, "CMD=")) )
    {
        ptr = szBuffer;

```

```

    if ( !strcmp(szBuffer, "Default") )
        strcpy(szBuffer, "CMD=Begin");
    switch( *pFormId )
    {
        case WELCOME_FORM:
            strcpy(szBuffer, "CMD=Submit");
            break;
        case MAIN_MENU_FORM:
            strcpy(szBuffer, "CMD=NewOrder");
            break;
        case NEW_ORDER_FORM:
        case PAYMENT_FORM:
        case DELIVERY_FORM:
        case ORDER_STATUS_FORM:
        case STOCK_LEVEL_FORM:
            if ( !(*pTermId) )
                return FALSE;
            if ( GetKeyValue(pECB->lpszQueryString,"PI*",
szTmp, sizeof(szTmp)) )
                strcpy(szBuffer, "CMD=Process");
            else
            {
                strcpy(szBuffer, "CMD=");
                strcat(szBuffer, szCmds[*pFormId -
NEW_ORDER_FORM]);
            }
            break;
        default:
            return FALSE;
    }
}
ptr += 4;

while( *ptr && *ptr != '&' )
    *dest++ = *ptr++;
*dest = 0;

for(i=0; szCmds[i][0]; i++)
{
    if ( !strcmp(szCmds[i], szBuffer) )
    {
        *pCmd = i;
        return TRUE;
    }
}
return FALSE;
}

```

```

/* FUNCTION: void NewOrderForm(EXTENSION_CONTROL_BLOCK *pECB, int
iFormId,
*                               int iTermId, intiSyncId)
*
* PURPOSE: This function wraps the functionality needed for the TPC-C
New
*
*           Order Form.
*
* ARGUMENTS:  int   iFormId   unused
*             int   iTermId   id of calling browser, i.e.
TERMID=from http
*
*             command line
*             EXTENSION_CONTROL_BLOCK *pECB   structure
pointer to passed in

```

```

*
* internet
service information
*
* RETURNS:      None
*
* COMMENTS:    None
*
*/

void NewOrderForm(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int
iTermId, int iSyncId)
{
    WriteZString(pECB, MakeNewOrderForm(iTermId, iSyncId, TRUE,
FALSE, ""));
    UNUSEDPARAM(iFormId);
    return;
}

/* FUNCTION: void PaymentForm(EXTENSION_CONTROL_BLOCK *pECB, int
iFormId, int
*
* iTermId, int iSyncId)
*
* PURPOSE: This function wraps the functionality for the TPC-C Payment
Form.
*
* ARGUMENTS:  int      iFormId      unused
*             int      iTermId      id of calling browser, i.e.
TERMID=
*
*             int      iSyncId      from http command line
*                                     sync id of calling browser
*
*             EXTENSION_CONTROL_BLOCK *pECB  structure pointer to
passed in
*                                     internet service
information.
* RETURNS:    None
*
* COMMENTS:  None
*
*/

void PaymentForm(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int
iTermId, int iSyncId)
{
    WriteZString(pECB, MakePaymentForm(iTermId, iSyncId, TRUE) );
    UNUSEDPARAM(iFormId);
    return;
}

/* FUNCTION: void DeliveryForm(EXTENSION_CONTROL_BLOCK *pECB, int
iFormId, int
*
* iTermId, int iSyncId)
*
* PURPOSE: This function wraps the functionality for the TPC-C
Delivery Form.
*
* ARGUMENTS:  int      iFormId      unused
*             int      iTermId      id of calling browser, i.e.
TERMID=
*
*             int      iSyncId      from http command line
*                                     sync id of calling browser

```

```

*
* EXTENSION_CONTROL_BLOCK *pECB  structure pointer to
passed in
*
* internet service
information.
* RETURNS:    None
*
* COMMENTS:  None
*
*/

void DeliveryForm(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int
iTermId, int iSyncId)
{
    WriteZString(pECB, MakeDeliveryForm(iTermId, iSyncId, TRUE,
"" ) );
    UNUSEDPARAM(iFormId);

    return;
}

/* FUNCTION: void OrderStatusForm(EXTENSION_CONTROL_BLOCK *pECB, int
iFormId,
*
* iTermId, int iSyncId)
*
* PURPOSE: This function wraps the function of the TPC-C Order Status
Form.
*
* ARGUMENTS:  int      iFormId      unused
*             int      iTermId      id of calling browser, i.e.
TERMID=
*
*             int      iSyncId      from http command line
*                                     sync id of calling browser
*
*             EXTENSION_CONTROL_BLOCK *pECB  structure pointer to
passed in
*                                     internet service
information.
* RETURNS:    None
*
* COMMENTS:  None
*
*/

void OrderStatusForm(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int
iTermId, int iSyncId)
{
    WriteZString(pECB, MakeOrderStatusForm(iTermId, iSyncId, TRUE) );
    UNUSEDPARAM(iFormId);

    return;
}

/* FUNCTION: void StockLevelForm(EXTENSION_CONTROL_BLOCK *pECB, int
iFormId,
*
* iTermId, int iSyncId)
*
* PURPOSE: This function wraps the functions of the TPC-C Stock Level
Form.
*

```

```

* ARGUMENTS:  int    iFormId    unused
*             int    iTermId    id of calling browser, i.e.
TERMINID=
*             from http command line
*             int    iSyncId    sync id of calling browser
*             EXTENSION_CONTROL_BLOCK *pECB  structure pointer to
passed in
*             internet service
information.
* RETURNS:    None
*
* COMMENTS:    None
*
*/

void StockLevelForm(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int
iTermId, int iSyncId)
{
    WriteZString(pECB, MakeStockLevelForm(iTermId, iSyncId, TRUE) );
    return;
}

/* FUNCTION: void Exitcmd(EXTENSION_CONTROL_BLOCK *pECB, int iFormId,
int iTermId, int iSyncId)
*
* PURPOSE: This function removes a terminal id from use, the allocated
* structure however remains valid so the next request for a
new client
* will not require a new memory allocation.
*
* ARGUMENTS:  int    iFormId    unused
*             int    iTermId    id of calling browser, i.e.
TERMINID=
*             from http command line
*             int    iSyncId    sync id of calling browser
*             EXTENSION_CONTROL_BLOCK *pECB  structure pointer to
passed in
*             internet service
information.
* RETURNS:    None
*
* COMMENTS:    None
*
*/

void Exitcmd(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int iTermId,
int iSyncId)
{
    (*Term.Delete)(pECB, iTermId);
    WriteZString(pECB, MakeWelcomeForm() );
    UNUSEDPARAM(iFormId);
    UNUSEDPARAM(iSyncId);

    return;
}

/* FUNCTION: void SubmitCmd(EXTENSION_CONTROL_BLOCK *pECB, int iFormId,
int
* iTermId, int iSyncId)
*

```

```

* PURPOSE: This function allocates a new terminal id in the Term
struct array.
*
* ARGUMENTS:  int    iFormId    unused
*             int    iTermId    id of calling browser, i.e.
TERMINID=
*             from http command line
*             int    iSyncId    sync id of calling browser
*             EXTENSION_CONTROL_BLOCK *pECB  structure pointer to
passed in
*             internet service
information.
* RETURNS:    None
*
* COMMENTS:    A terminal id can be allocated but still be invalid if
the
* requested warehouse number is outside the range specified
in the
* registry. This then will force the client id to be
invalid and an
* error message sent to the users browser.
*/

void SubmitCmd(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int iTermId,
int iSyncId)
{
    int iCurrent;

    if ( (iCurrent = (*Term.Add)(pECB, pECB->lpszQueryString)) < 0 )
    {
        ErrorMessage(pECB, ERR_CANNOT_INIT_TERMINAL,
ERR_TYPE_WEBDLL, NULL, iCurrent, iSyncId);
        return;
    }

    if ( Term.pClientData[iCurrent].w_id > iMaxWareHouses ||
Term.pClientData[iCurrent].w_id < 1 )
    {
        ErrorMessage(pECB, ERR_W_ID_INVALID, ERR_TYPE_WEBDLL, NULL,
iCurrent, iSyncId);
        (*Term.Delete)(pECB, iCurrent);
        return;
    }

    if ( Term.pClientData[iCurrent].d_id < 1 ||
Term.pClientData[iCurrent].d_id > 10 )
    {
        ErrorMessage(pECB, ERR_D_ID_INVALID, ERR_TYPE_WEBDLL, NULL,
iCurrent, iSyncId);
        (*Term.Delete)(pECB, iCurrent);
        return;
    }

    WriteZString(pECB, MakeMainMenuForm(iCurrent,
Term.pClientData[iCurrent].iSyncId) );
    return;
}

/* FUNCTION: void BeginCmd(EXTENSION_CONTROL_BLOCK *pECB, int iFormId,
int iTermId, int iSyncId)
*
* PURPOSE: This function is the first command executed. It is
executed with

```

```

*           the command:  CMD=Begin?Server=xxx from the http command
line.
*
* ARGUMENTS:  int      iFormId      unused
*             int      iTermId      id of calling browser, i.e.
TERMID=
*
*             int      iSyncId      from http command line
*             int      iSyncId      sync id of calling browser
* EXTENSION_CONTROL_BLOCK *pECB structure pointer to
passed in
*
*             internet service
information.
*
* RETURNS:    None
*
* COMMENTS:   SQL server must be specified, however the user and
password
*
*             parameters are optional. The complete command line is
*             CMD=Begin&Server=server&User=sa&Psw=&. The & are used
*             to separate parameters which is internet browser
standard.
*/

void BeginCmd(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int iTermId,
int iSyncId)
{
    LPSTR pQueryString;
    pQueryString = pECB->lpszQueryString;

    WriteZString(pECB, MakeWelcomeForm() );

    UNUSEDPARAM(iFormId);
    return;
}

/* FUNCTION: void ProcessCmd(EXTENSION_CONTROL_BLOCK *pECB, int iFormId,
int
*
*             iTermId, int iSyncId)
*
* PURPOSE:    This function process the passed in http command
*
* ARGUMENTS:  int      iFormId      unused
*             int      iTermId      id of calling browser, i.e.
TERMID=
*
*             int      iSyncId      from http command line
*             int      iSyncId      sync id of calling browser
* EXTENSION_CONTROL_BLOCK *pECB structure pointer to
passed in
*
*             internet service
information.
* RETURNS:    None
*
* COMMENTS:   None
*/

void ProcessCmd(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int iTermId,
int iSyncId)
{
    switch( iFormId )
    {
        case WELCOME_FORM:
            return;
        case MAIN_MENU_FORM:

```

```

return;
case NEW_ORDER_FORM:
    ProcessNewOrderForm(pECB, iTermId, iSyncId);
    return;
case PAYMENT_FORM:
    ProcessPaymentForm(pECB, iTermId, iSyncId);
    return;
case DELIVERY_FORM:
    ProcessDeliveryForm(pECB, iTermId, iSyncId);
    return;
case ORDER_STATUS_FORM:
    ProcessOrderStatusForm(pECB, iTermId, iSyncId);
    return;
case STOCK_LEVEL_FORM:
    ProcessStockLevelForm(pECB, iTermId, iSyncId);
    return;
}

}

/* FUNCTION: void ClearCmd(EXTENSION_CONTROL_BLOCK *pECB, int iFormId,
int
*
*             iTermId, int iSyncId)
*
* PURPOSE:    This function frees all currently logged in terminal
ids.
*
* ARGUMENTS:  int      iFormId      unused
*             int      iTermId      id of calling browser, i.e.
TERMID=
*
*             int      iSyncId      from http command line
*             int      iSyncId      sync id of calling browser
* EXTENSION_CONTROL_BLOCK *pECB structure pointer to
passed in
*
*             internet service
information.
* RETURNS:    None
*
* COMMENTS:   Use this function with caution, it may cause
unpredictable
*
*             results if existing browsers attempt to use the web
client
*
*             without beginning at the login screen for each client.
*/

void ClearCmd(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int iTermId,
int iSyncId)
{
    int i;
    EnterCriticalSection(&CriticalSection);
    for(i=0; i<Term.iAvailable; i++)
    {
        if ( Term.pClientData[i].inUse )
            (*Term.Delete)(pECB, i);
    }
    Term.iNext          = 0;
    Term.iAvailable     = 0;
    Term.iMasterSyncId = 1;

    if ( Term.pClientData )
        free(Term.pClientData);

    Term.pClientData    = NULL;
    Term.bInit          = FALSE;

```

```

(*Term.Init)();

if ( !(*Term.Allocate)() )
{
    ErrorMessage(pECB, ERR_MAX_CONNECT_PARAM, ERR_TYPE_WEBDLL, NULL,
iTermId, iSyncId);
    return;
}

for(i=Term.iNext; i<Term.iAvailable; i++)
    Term.pClientData[i].inUse = 0;

Term.pClientData[0].inUse = 1;
LeaveCriticalSection(&CriticalSection);

WriteZString(pECB, MakeWelcomeForm() );
return;
}

/* FUNCTION: void MenuCmd(EXTENSION_CONTROL_BLOCK *pECB, int iFormId,
*
* int iTermId, int iSyncId)
*
* PURPOSE: This function causes an exit to the main menu
*
* ARGUMENTS: int iFormId unused
* int iTermId id of calling browser, i.e.
TERMID=
* from http command line
* int iSyncId sync id of calling browser
* EXTENSION_CONTROL_BLOCK *pECB structure pointer to
passed in
* internet service
information.
* RETURNS: None
*
* COMMENTS: None
*
*/

void MenuCmd(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int iTermId,
int iSyncId)
{
    WriteZString(pECB, MakeMainMenuForm(iTermId, iSyncId) );
    return;
}

/* FUNCTION: void WriteZString(EXTENSION_CONTROL_BLOCK *pECB, char
*szStr)
*
* PURPOSE: This function is the low level output function. It writes
a string
* of text back to the client browser.
*
* ARGUMENTS: EXTENSION_CONTROL_BLOCK *pECB passed in structure
pointer from
* inetsrv.
* char *szStr string to display in
the client
* browser.
*
* RETURNS: None

```

```

*
* COMMENTS: This function assumes that the string to written to the
client
* browser has been formatted in an HTML manner.
*/

static void WriteZString(EXTENSION_CONTROL_BLOCK *pECB, char *szStr)
{
    FILE *fp;
    int lpbSize;
    int iSize;
    char szHeader[128];
    char szHeader1[128];

    lpbSize = strlen(szStr)+1;

    if ( bLog )
    {
        SYSTEMTIME systemTime;

        fp = fopen(szTpccLogPath, "ab");

        GetLocalTime(&systemTime);

        fprintf(fp, " * HTML PAGE
* %2.2d/%2.2d/%2.2d %2.2d:%2.2d:%2.2d\r\n\r\n%s\r\n\r\n",
systemTime.wYear, systemTime.wMonth,
systemTime.wDay,
systemTime.wHour, systemTime.wMinute,
systemTime.wSecond,
szStr);

        fclose(fp);
    }

#ifdef PERFORMIX

    (*pECB->ServerSupportFunction)(pECB->ConnID,
HSE_REQ_DONE_WITH_SESSION, NULL, 0, 0);
    (*pECB->WriteClient)(pECB->ConnID, szStr, &lpbSize, 0);

#else

    iSize = sprintf(szHeader, "200 Ok");
    sprintf(szHeader1, "Connection: keep-alive\r\nContent-type:
text/html\r\nContent-length: %d\r\n\r\n", lpbSize);
    (*pECB->ServerSupportFunction)(pECB->ConnID,
HSE_REQ_SEND_RESPONSE_HEADER, szHeader, &iSize, (LPDWORD)szHeader1);
    (*pECB->WriteClient)(pECB->ConnID, szStr, &lpbSize, 0);

#endif

    return;
}

/* FUNCTION: void h_printf(EXTENSION_CONTROL_BLOCK *pECB, char
*format, ...)
*
* PURPOSE: This function forms a high level printf for an HTML
browser
*

```

```

* ARGUMENTS:  EXTENSION_CONTROL_BLOCK *pECB  passed in structure
pointer from
*
*             char             *format printf style format
string
*             ...             other arguments as required by
printf
*             style format string.
*
* RETURNS:    None
*
* COMMENTS:   This function is mainly used for developmental support.
*/

```

```

static void h_printf(EXTENSION_CONTROL_BLOCK *pECB, char *format, ...)
{
    int lpbSize;
    char szBuff[512];
    char szTmp[512];

    va_list marker;
    va_start( marker, format );
    vsprintf(szTmp, format, marker);
    va_end( marker );

    lpbSize = wsprintf(szBuff, "<HTML>%s</HTML>", szTmp) + 1;
    (*pECB->WriteClient)(pECB->ConnID, szBuff, &lpbSize, 0);

    return;
}

```

```

/* FUNCTION: void ErrorMessage(EXTENSION_CONTROL_BLOCK *pECB, int
iError, int
*
*             iErrorType, char *szMsg)
*
* PURPOSE:    This function displays an error message in the client
browser.
*
* ARGUMENTS:  EXTENSION_CONTROL_BLOCK *pECB  passed in structure
pointer from
*
*             int             iError  id of error message
*             int             iErrorType  error type,
ERR_TYPE_SQL,
*
*             ERR_TYPE_OCI, or
*             ERR_TYPE_WEBDLL
*             int             iTermId  terminal_id from
browser
*
*             int             iSyncid  sync id from browser
*             char            *szMsg   optional error message
string
*
*             used with ERR_TYPE_SQL and
ERR_TYPE_OCI
*
* RETURNS:    None
*
* COMMENTS:   If the error type is ERR_TYPE_WEBDLL the szmsg
parameter may be
*             NULL because it is ignored. If the error type is
ERR_TYPE_SQL or
*             ERR_TYPE_OCI then the szMsg parameter contains the text
of the
*             error message, so the szMsg parameter cannot be NULL.

```

```

*
*/

void ErrorMessage(EXTENSION_CONTROL_BLOCK *pECB, int iError, int
iErrorType, char *szMsg, int iTermId, int iSyncId)
{
    int i;
    static SERRORMSG errorMsgs[] =
    {
        { ERR_SUCCESS, "Success, no error." },
        { ERR_COMMAND_UNDEFINED, "Command undefined." },
        { ERR_NOT_IMPLEMENTED_YET, "Not Implemented Yet." },
        { ERR_CANNOT_INIT_TERMINAL, "Cannot initialize client
connection." },
        { ERR_OUT_OF_MEMORY, "insufficient memory." },
        { ERR_NEW_ORDER_NOT_PROCESSED, "Cannot process new Order
form." },
        { ERR_PAYMENT_NOT_PROCESSED, "Cannot process payment
form." },
        { ERR_NO_SERVER_SPECIFIED, "No Server name specified." },
        { ERR_ORDER_STATUS_NOT_PROCESSED, "Cannot process order
status form." },
        { ERR_W_ID_INVALID, "Invalid Warehouse ID." },
        { ERR_CAN_NOT_SET_MAX_CONNECTIONS, "Insufficient memory
to allocate # connections." },
        { ERR_NOSUCH_CUSTOMER, "No such customer." },
        { ERR_D_ID_INVALID, "Invalid District ID Must be 1 to
10." },
        { ERR_MAX_CONNECT_PARAM, "Max client connections exceeded,
run install to increase." },
        { ERR_INVALID_SYNC_CONNECTION, "Invalid Terminal Sync
ID." },
        { ERR_INVALID_TERMID, "Invalid Terminal ID." },
        { ERR_PAYMENT_INVALID_CUSTOMER, "Payment Form, No such
Customer." },
        { ERR_SQL_OPEN_CONNECTION, "SQLOpenConnection API
Failed." },
        { ERR_STOCKLEVEL_MISSING_THRESHOLD_KEY, "Stock Level
missing Threshold key \"TT*\"." },
        { ERR_STOCKLEVEL_THRESHOLD_INVALID, "Stock Level
Threshold invalid data type range = 1 - 99." },
        { ERR_STOCKLEVEL_THRESHOLD_RANGE, "Stock Level Threshold
out of range, range must be 1 - 99." },
        { ERR_STOCKLEVEL_NOT_PROCESSED, "Stock Level not
processed." },
        { ERR_NEWORDER_FORM_MISSING_DID, "New Order missing
District key \"DID*\"." },
        { ERR_NEWORDER_DISTRICT_INVALID, "New Order District ID
Invalid range 1 - 10." },
        { ERR_NEWORDER_DISTRICT_RANGE, "New Order District ID out
of Range. Range = 1 - 10." },
        { ERR_NEWORDER_CUSTOMER_KEY, "New Order missing Customer
key \"CID*\"." },
        { ERR_NEWORDER_CUSTOMER_INVALID, "New Order customer id
invalid data type, range = 1 to 3000." },
        { ERR_NEWORDER_CUSTOMER_RANGE, "New Order customer id out
of range, range = 1 to 3000." },
        { ERR_NEWORDER_MISSING_IID_KEY, "New Order missing Item
Id key \"IID*\"." },
        { ERR_NEWORDER_ITEM_BLANK_LINES, "New Order blank order
lines all orders must be continuous." },
        { ERR_NEWORDER_ITEMID_INVALID, "New Order Item Id is
wrong data type, must be numeric." },

```

```

    { ERR_NEWORDER_MISSING_SUPPW_KEY, "New Order missing
Supp_W key \"SP##*\".\" },
    { ERR_NEWORDER_SUPPW_INVALID, "New Order Supp_W invalid
data type must be numeric.\" },
    { ERR_NEWORDER_MISSING_QTY_KEY, "New Order Missing Qty
key \"Qty##*\".\" },
    { ERR_NEWORDER_QTY_INVALID, "New Order Qty invalid must
be numeric range 1 - 99.\" },
    { ERR_NEWORDER_SUPPW_RANGE, "New Order Supp_W value out
of range - 1 to MaxWarehouses.\" },
    { ERR_NEWORDER_ITEMID_RANGE, "New Order Item Id is out of
range. Range = 1 to 999999.\" },
    { ERR_NEWORDER_QTY_RANGE, "New Order Qty is out of range.
Range = 1 to 99.\" },
    { ERR_PAYMENT_DISTRICT_INVALID, "Payment District ID is
invalid must be 1 - 10.\" },
    { ERR_NEWORDER_SUPPW_WITHOUT_ITEMID, "New Order Supp_W
field entered without a Item Id.\" },
    { ERR_NEWORDER_QTY_WITHOUT_ITEMID, "New Order Qty entered
without a corresponding Item Id.\" },
    { ERR_NEWORDER_NOITEMS_ENTERED, "New Order Blank Items
between items, items must be continuous.\" },
    { ERR_PAYMENT_MISSING_DID_KEY, "Payment missing district
Key \"DID*\".\" },
    { ERR_PAYMENT_DISTRICT_RANGE, "Payment District Out of
range, range = 1 - 10.\" },
    { ERR_PAYMENT_MISSING_CID_KEY, "Payment missing Customer
Key \"CID*\".\" },
    { ERR_PAYMENT_CUSTOMER_INVALID, "Payment Customer data
type invalid, must be numeric.\" },
    { ERR_PAYMENT_MISSING_CLT, "Payment missing Customer Last
Name Key \"CLT*\".\" },
    { ERR_PAYMENT_LAST_NAME_TO_LONG, "Payment Customer last
name longer than 16 characters.\" },
    { ERR_PAYMENT_CUSTOMER_RANGE, "Payment Customer ID out of
range, must be 1 to 3000.\" },
    { ERR_PAYMENT_CID_AND_CLT, "Payment Customer ID and Last
Name entered must be one of other.\" },
    { ERR_PAYMENT_MISSING_CDI_KEY, "Payment missing Customer
district key \"CDI*\".\" },
    { ERR_PAYMENT_CDI_INVALID, "Payment Customer district
invalid must be numeric.\" },
    { ERR_PAYMENT_CDI_RANGE, "Payment Customer district out of
range must be 1 - 10.\" },
    { ERR_PAYMENT_MISSING_CWI_KEY, "Payment missing Customer
Warehouse key \"CWI*\".\" },
    { ERR_PAYMENT_CWI_INVALID, "Payment Customer Warehouse
invalid must be numeric.\" },
    { ERR_PAYMENT_CWI_RANGE, "Payment Customer Warehouse out
of range, 1 to Max Warehouses.\" },
    { ERR_PAYMENT_MISSING_HAM_KEY, "Payment missing Amount key
\"HAM*\".\" },
    { 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_ORDERSTATUS_MISSING_DID_KEY, "Order Status missing
District key \"DID*\".\" },
    { ERR_ORDERSTATUS_DID_INVALID, "Order Status District
invalid, value must be numeric 1 - 10.\" },
    { ERR_ORDERSTATUS_DID_RANGE, "Order Status District out of
range must be 1 - 10.\" },
    { 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_CLT_RANGE, "Order Status Customer last
name longer than 16 characters.\" },
    { ERR_ORDERSTATUS_CID_INVALID, "Order Status Customer ID
invalid, range must be numeric 1 - 3000.\" },
    { ERR_ORDERSTATUS_CID_RANGE, "Order Status Customer ID
out of range must be 1 - 3000.\" },
    { ERR_ORDERSTATUS_CID_AND_CLT, "Order Status Customer ID
and LastName entered must be only one.\" },
    { ERR_DELIVERY_MISSING_OCD_KEY, "Delivery missing Carrier
ID key \"OCD*\".\" },
    { ERR_DELIVERY_CARRIER_INVALID, "Delivery Carrier ID
invalid must be numeric 1 - 10.\" },
    { ERR_DELIVERY_CARRIER_ID_RANGE, "Delivery Carrier ID out
of range must be 1 - 10.\" },
    { ERR_PAYMENT_MISSING_CLT_KEY, "Payment missing Customer
Last Name key \"CLT*\".\" },
    { ERR_TPINIT_BAD,
"Tuxedo tpinit
Failed."
},
    { ERR_TPALLOC_BAD,
"Tuxedo tpalloc
Failed."
},
    { ERR_TPCALL_BAD,
"Tuxedo tpcall
Failed."
},
    {
        TOPEND_SEND_ERROR, "TOPEND client send error.",
        TOPEND_RECEIVE_ERROR, "TOPEND client receive error.",
        0, ""
    }
};

static char szNoMsg[] = "";
char *szForm;

if ( !szMsg )
    szMsg = szNoMsg;

if ( iTermId > 0 && IsValidTermId(iTermId) )
    szForm = Term.pClientData[iTermId].szBuffer; //if termid valid
use common terminal static buffer.
else
    szForm = Term.pClientData[0].szBuffer; //else term id invalid
so use common terminal static buffer.

switch(iErrorType)
{
case ERR_TYPE_WEBDLL:
for(i=0; errorMsgs[i].szMsg[0]; i++)
{
if ( iError == errorMsgs[i].iError )
break;
}

if ( !errorMsgs[i].szMsg[0] )
i = 1;

strcpy(szForm, "<HTML><HEAD><TITLE>Welcome To TPC-
C</TITLE></HEAD><BODY><FORM ACTION=\\\"tpcc.dll\\\" METHOD=\\\"GET\\\">");
wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\\\"hidden\\\"NAME=\\\"STATUSID\\\" VALUE=\\\"%d\\\">", iError);

```

```

        wsprintf(szForm+strlen(szForm), "<INPUT TYPE=\"hidden\"
NAME=\"ERROR\" VALUE=\"%d\">", iErrVal);
        wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\"NAME=\"TERMID\" VALUE=\"%d\">", iTermId);
        wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\"NAME=\"SYNCID\" VALUE=\"%d\">", iSyncId);
        wsprintf(szForm+strlen(szForm), "Error: TPCCWEB(%d): %s",
iError, errorMsgs[i].szMsg);
        strcat(szForm, "</FORM><BODY></HTML>");
        WriteZString(pECB, szForm);
        break;

    case ERR_TYPE_SQL:
        strcpy(szForm, "<HTML><HEAD><TITLE>Welcome To TPC-
C</TITLE></HEAD><BODY><FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">");
        wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\"NAME=\"STATUSID\" VALUE=\"%d\">", iError);
        wsprintf(szForm+strlen(szForm), "<INPUT TYPE=\"hidden\"
NAME=\"ERROR\" VALUE=\"%d\">", iErrVal);
        wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\"NAME=\"TERMID\" VALUE=\"%d\">", iTermId);
        wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\"NAME=\"SYNCID\" VALUE=\"%d\">", iSyncId);
        wsprintf(szForm+strlen(szForm), "Error: Oracle(%d): %s",
iError, szMsg);
        strcat(szForm, "</FORM><BODY></HTML>");
        WriteZString(pECB, szForm);
        break;

    case ERR_TYPE_OCI:
        strcpy(szForm, "<HTML><HEAD><TITLE>Welcome To TPC-
C</TITLE></HEAD><BODY><FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">");
        wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\"NAME=\"STATUSID\" VALUE=\"%d\">", iError);
        wsprintf(szForm+strlen(szForm), "<INPUT TYPE=\"hidden\"
NAME=\"ERROR\" VALUE=\"%d\">", iErrVal);
        wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\"NAME=\"TERMID\" VALUE=\"%d\">", iTermId);
        wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\"NAME=\"SYNCID\" VALUE=\"%d\">", iSyncId);
        wsprintf(szForm+strlen(szForm), "Error: OCI(%d): %s", iError,
szMsg);
        strcat(szForm, "</FORM><BODY></HTML>");
        WriteZString(pECB, szForm);
        break;

    case ERR_TYPE_TUXEDO:
        strcpy(szForm, "<HTML><HEAD><TITLE>Welcome To
TPC-C</TITLE></HEAD><BODY><FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">");
        wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"%d\">", iError);
        wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\" NAME=\"ERROR\" VALUE=\"%d\">", iErrVal);
        wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\" NAME=\"TERMID\" VALUE=\"%d\">", iTermId);
        wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"%d\">", iSyncId);
        wsprintf(szForm+strlen(szForm), "Error:
Tuxedo: %d %s", iError, szMsg);
        strcat(szForm, "</FORM><BODY></HTML>");
        WriteZString(pECB, szForm);
        break;
    }
    return;

```

```

}

/* FUNCTION: BOOL GetKeyValue(char *pQueryString, char *pKey, char
*pValue,
*
* int iMax, char **pBeyondKey)
*
* PURPOSE: This function parses http formatted string for specific key
values.
*
* ARGUMENTS: char *pQueryString http string from client browser
* char *pKey key value to look for
* char *pValue character array into which to place
key's value
* int iMax maximum length of key value
array.
* char **pBeyondKey Pointer to location beyond key
value
* RETURNS: BOOL FALSE key value not found
* TRUE key value found
*
* 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.
*/

static BOOL GetKeyValue(char *pQueryString, char *pKey, char *pValue,
int iMax)
{
    char *ptr;

    if ( !(ptr=strstr(pQueryString, pKey)) )
        return FALSE;
    if ( !(ptr=strchr(ptr, '=') )
        return FALSE;
    ptr++;
    iMax--;
    while( *ptr && *ptr != '&' && iMax)
    {
        *pValue++ = *ptr++;
        iMax--;
    }
    *pValue = 0;
    return TRUE;
}

/* FUNCTION: void TermInit(void)
*
* PURPOSE: This function initializes the client terminal structure it
is
* called when the TPCC.DLL is first loaded by the inet
service.
*
* ARGUMENTS: none
*
* RETURNS: None
*
* COMMENTS: None
*
*/

```

```

static void TermInit(void)
{
    if ( Term.bInit )
        return;
    Term.iNext = 0;
    Term.iMasterSyncId = 1;
    Term.iAvailable = 0;
    Term.pClientData = NULL;
    Term.bInit = TRUE;
    return;
}

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

static void TermRestore(void)
{
    Term.iNext = 0;
    Term.iAvailable = 0;
    Term.iMasterSyncId = 0;
    if ( Term.pClientData )
        free(Term.pClientData);
    Term.pClientData = NULL;
    Term.bInit = FALSE;
    return;
}

/* FUNCTION: int TermAllocate(void)
 *
 * PURPOSE: This funct allocates terminal array entries in the Term
structure.
 *
 * ARGUMENTS: None
 *
 * RETURNS: int TRUE or 1 if sucessfull
 *          int FALSE or 0 if terminal id cannot be allocated.
 *
 * COMMENTS: None
 */

static int TermAllocate(void)
{
    Term.iAvailable += 32;
    if ( !Term.pClientData )
        Term.pClientData = (PCLIENTDATA)malloc(Term.iAvailable *
sizeof(CLIENTDATA));
    else
        Term.pClientData = (PCLIENTDATA)realloc(Term.pClientData,
Term.iAvailable * sizeof(CLIENTDATA));
}

```

```

        return ( Term.pClientData ) ? 1 : 0;
    }

/* FUNCTION: int TermAdd(EXTENSION_CONTROL_BLOCK *pECB, char
*pQueryString, int iTermId)
 *
 * PURPOSE: This function assigns a terminal id which is used to
identify a
 *          client browser.
 *
 * ARGUMENTS: EXTENSION_CONTROL_BLOCK *pECB    passed instructure
pointer
 *
 *          char *pQueryString    from inetsrv.
 *          http query string
 *          passed to
 *
 *          this DLL.
 *
 *          int iTermId    terminal id from
browser
 * RETURNS: int assigned terminal id
 *          -1 cannot assign id error ocured.
 *
 * COMMENTS: if the terminal id cannot be assigned it is because of
 *          insufficient memory or the SQL connection cannot be
allocated.
 */

static int TermAdd(EXTENSION_CONTROL_BLOCK *pECB, char *pQueryString)
{
    char szTmp[32];

    int i, iCurrent, iTotConnections, iTickCount;

    EnterCriticalSection(&CriticalSection);
    for(i=0, iTotConnections = 0; i<Term.iAvailable; i++)
    {
        if ( Term.pClientData[i].inUse )
            iTotConnections++;
    }

    if ( iTotConnections >= iMaxConnections )
    {
        for(iCurrent = 1, i=1, iTickCount = 0x7FFFFFFF;
i<iMaxConnections; i++)
        {
            if ( iTickCount > Term.pClientData[i].iTickCount )
            {
                iTickCount = Term.pClientData[i].iTickCount;
                iCurrent = i;
            }
        }
    }
    else
    {
        for(i=0; i<Term.iAvailable; i++)
        {
            if ( !Term.pClientData[i].inUse )
                break;
        }
    }

    iCurrent = i;
}

```

```

}
if ( i == Term.iAvailable )
{
    Term.iNext = Term.iAvailable;
    if ( !(*Term.Allocate)() )
        goto TermAddErr1;

    for(i=Term.iNext; i<Term.iAvailable; i++)
        Term.pClientData[i].inUse = 0;

    iCurrent = Term.iNext;
}

Term.pClientData[iCurrent].inUse = 1;
if ( !GetKeyValue(pQueryString, "w_id", szTmp, sizeof(szTmp)) )
    goto TermAddErr1;

Term.pClientData[iCurrent].w_id = (short)atoi(szTmp);
if ( !GetKeyValue(pQueryString, "d_id", szTmp, sizeof(szTmp)) )
    goto TermAddErr1;

Term.pClientData[iCurrent].d_id = atoi(szTmp);
Term.pClientData[iCurrent].iTickCount = GetTickCount();
Term.pClientData[iCurrent].iSyncId = Term.iMasterSyncId++;
if ( Init(pECB, iCurrent, Term.pClientData[iCurrent].iSyncId,
szServer, szUser, szPassword, szDatabase) )
{
    (*Term.Delete)(pECB, iCurrent);
    goto TermAddErr1;
}

LeaveCriticalSection(&CriticalSection);
return iCurrent;

TermAddErr1:
LeaveCriticalSection(&CriticalSection);
return -1; //terminal unsuccessfully added
}

/* FUNCTION: void TermDelete(EXTENSION_CONTROL_BLOCK *pECB, int id)
*
* PURPOSE: This function makes a terminal entry in the Term array
available
*
*           for reuse.
*
* ARGUMENTS: EXTENSION_CONTROL_BLOCK *pECB passed in structure
pointer from
*
*           inetsrv.
*           int id Terminal id of client exiting
*
* RETURNS: None
*
* COMMENTS: None
*/

static void TermDelete(EXTENSION_CONTROL_BLOCK *pECB, int id)
{
    if ( id >= 0 && id < Term.iAvailable )
    {
        Close(pECB, id, -1);
        Term.pClientData[id].inUse = 0;
    }
}

```

```

#endif LOCAL_ALLOC
tpfree((char *)Term.pClientData[id].newOrderDataPtr);
tpfree((char *)Term.pClientData[id].paymentDataPtr);
tpfree((char *)Term.pClientData[id].orderStatusDataPtr);
tpfree((char *)Term.pClientData[id].deliveryDataPtr);
tpfree((char *)Term.pClientData[id].stockLevelDataPtr);
#endif

}

return;
}

/* FUNCTION: BOOL Init(EXTENSION_CONTROL_BLOCK *pECB, int iTermId, int
iSyncId,
*
*           char *szServer, char *szUser, char *szPassword,
char *szDatabase)
*
* PURPOSE: This function initializes the sql connection for use.
*
* ARGUMENTS: EXTENSION_CONTROL_BLOCK *pECB passed in structure
pointer
*
*           from inetsrv.
*           int iTermId id of browser client that this
connection is for.
*           int iSyncId sync id for this client session
*           char *szServer sql server name
*           char *szUser user name
*           char *szPassword user password
*           char *szDatabase database to use
*
* RETURNS: BOOL FALSE if successfull
*           TRUE if an error occurs and connection cannot be
established.
*
* COMMENTS: None
*/

BOOL Init(EXTENSION_CONTROL_BLOCK *pECB, int iTermId, int iSyncId, char
*szServer, char *szUser, char *szPassword, char *szDatabase)
{
    char szApp[32];
#ifdef LOCAL_ALLOC
    char buf[40];
    int iRc;
#endif

#ifdef TOPEND
    int i;
    void space_fill();
    tp_dif_structs T *tclient_dif;
    char ascii_pid[10];
    int msglen;
    FILE *fp1;
#else
    sprintf(szApp, "TPCC:%ld", (int)iTermId);
    /* Term.pClientData[iTermId].dbContext = NULL; - do not keep
context */
#endif
}

```

```

#ifdef TOPEND
EnterCriticalSection(&CriticalSection);
if (TP_InitDone == FALSE)
{
    TP_InitDone = TRUE;
    LeaveCriticalSection (&CriticalSection);

    if ((i=(int)tp_ChangeToGroup((LPSTR)getenv("TP_SYSTEM"),
        LOGON32_LOGON_INTERACTIVE,
        LOGON32_PROVIDER_DEFAULT))!=0)
    {
        if (bLog)
        {
            fpl = fopen(szTpccLogPath,"ab");
            fprintf(fpl,"Bad status from
tp_ChangeToGroup,status=%d*\n",i);
            fclose(fpl);
        }
        MessageBox(NULL, "Failed in tp_ChangeToGroup.",
            "Init", MB_OK | MB_ICONSTOP);
        return TRUE; //changed DMA
    }
    //initialize TOPEND

    if ((i=tp_mt_initialize(NULL,NULL,0L))!=TP_OK)
    {
        if (bLog)
        {
            fpl = fopen(szTpccLogPath,"ab");
            fprintf(fpl,"Bad status from
tp_mt_initialize,status=%d*\n",i);
            fclose(fpl);
        }

        return TRUE; //changed DMA
    }
} //if init
else
LeaveCriticalSection(&CriticalSection);

sprintf(szApp,"TPCC:%ld", (int) iTermId);
// Term.pClientData[iTermId].dbproc=NULL;

if((i=client_dif= tp_csi_alloc(TP_DIF_ALL))== NULL)
{
    if (bLog)
    {
        // SYSTEMTIME systemTime;
        fpl = fopen(szTpccLogPath,"ab");
        fprintf(fpl,"Bad status from tp_csi_alloc,status=%d*\n",i);
        fclose(fpl);
    }

    ErrorMessage(pECB, TOPEND_SEND_ERROR, ERR_TYPE_WEBDLL,
        NULL, iTermId, iSyncId);

    return TRUE;
}

(tp_dif_structs_t*)Term.pClientData[iTermId].tp_structadr =
client_dif;

```

```

    tclient_dif =
(tp_dif_structs_t*)Term.pClientData[iTermId].tp_structadr;

    sprintf(ascii_pid,"%d",iTermId);

    tclient_dif->info->tp_user_dialogue_id = iTermId;
    tclient_dif->info->tp_user_message_id = 0L;
    tclient_dif->info->tp_system_dialogue_id.tp_sys_dialogue = 0L;
    tclient_dif->info->tp_flags = TP_NOFLAGS;

    space_fill (tclient_dif->client->tp_userid,"tpcc
cli",TP_USERID_LEN);
    space_fill (tclient_dif->client-
>tp_endpoint,ascii_pid,TP_ENDPOINT_LEN);
    space_fill (tclient_dif->client->tp_password,"",TP_PASSWORD_LEN);
    space_fill
(tclient_dif->service-
>tp_product_name,"tpcc",TP_PROD_NAME_LEN);
    space_fill
(tclient_dif->output_format-
>tp_format_name,"",TP_FMT_NAME_LEN);
    space_fill
(tclient_dif->service-
>tp_function_name,"tpcc1",TP_FUNC_NAME_LEN);

    tclient_dif->service->tp_function_qualifier = 0;
    wait_time = TP_BLOCK;
    msglen = 0L;

    if ((i = tp_mt_client_signon(tclient_dif->info,tclient_dif-
>client,
        inactivity_time,NULL,NULL,0,NULL)) != TP_OK)
    {
        if (bLog)
        {
            fpl = fopen(szTpccLogPath,"ab");
            fprintf(fpl,"client signon
failed,status=%d,iTermId=%d,iSyncId=%d\n",
                i,iTermId,iSyncId);
            fclose(fpl);
        }

        ErrorMessage(pECB, TOPEND_SEND_ERROR, ERR_TYPE_WEBDLL,
            NULL, iTermId, iSyncId);

        return TRUE;
    }

    if ((i = tp_mt_client_receive(tclient_dif->info,wait_time,
        tclient_dif->output_format,tclient_dif->service,
        tclient_dif->location,&msglen,NULL)) != TP_OK)
    {
        if (bLog)
        {
            fpl = fopen(szTpccLogPath,"ab");
            fprintf(fpl,"client receive from signon
failed,status=%d\n",i);
            fclose(fpl);
        }

        ErrorMessage(pECB, TOPEND_SEND_ERROR, ERR_TYPE_WEBDLL,
            NULL, iTermId, iSyncId);

        return TRUE;
    }

```

```

    }

    if (bLog)
    {
        fp1 = fopen(szTpccLogPath,"ab");
        fprintf(fp1,"client %d signed on.\n",iTermId);
        fclose(fp1);
    }

#else
#ifdef TUX
#ifdef LOCAL_ALLOC /* only do if not doing local alloc of
tuxedo data structures */
// Add initialization of Tuxedo Structures

// NEWORDER

    if ((Term.pClientData[iTermId].newOrderDataPtr =
(NewOrderData *)tpalloc("CARRAY", NULL, sizeof(NewOrderData))) == NULL)
    {
        iRc = tperrno;
        sprintf(buf, "Tpccalloc %d", iRc);
        ErrorMessage(pECB, ERR_TPALLOC_BAD,
ERR_TYPE_TUXEDO, buf, 0, 0);
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, ">>>> Init Thread %d :
NewOrder tmalloc failed: iRc = %d \r\n",
                GetCurrentThreadId(), iRc);
            fclose(fp);
        }
        sprintf(buf, "Tpccalloc %d", iRc);
        ErrorMessage(pECB, ERR_TPALLOC_BAD,
ERR_TYPE_TUXEDO, buf, 0, 0);
        return TRUE;
    }

    if (dLog )
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "* Init Thread %d iTermId %d *
NewOrderDataPtr: %x \r\n",
            GetCurrentThreadId(), iTermId,
            &Term.pClientData[iTermId].newOrderDataPtr);
        fclose(fp);
    }

// PAYMENT

    if ((Term.pClientData[iTermId].paymentDataPtr =
(PaymentData *)tpalloc("CARRAY", NULL, sizeof(PaymentData))) == NULL)
    {
        iRc = tperrno;
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, ">>>> Init Thread %d :
Payment tmalloc failed: iRc = %d \r\n",
                GetCurrentThreadId(), iRc);

```

```

        fclose(fp);
    }
    sprintf(buf, "Tpccalloc %d", iRc);
    ErrorMessage(pECB, ERR_TPALLOC_BAD,
ERR_TYPE_TUXEDO, buf, 0, 0);
    return TRUE;
}

if (dLog )
{
    FILE *fp;

    fp = fopen(szTpccLogPath, "ab");
    fprintf(fp, "* Init Thread %d iTermId %d *
PaymentDataPtr: %x \r\n",
        GetCurrentThreadId(), iTermId,
        &Term.pClientData[iTermId].paymentDataPtr);
    fclose(fp);
}

// ORDERSTATUS

    if ((Term.pClientData[iTermId].orderStatusDataPtr =
(OrderStatusData *)tpalloc("CARRAY", NULL, sizeof(OrderStatusData))) ==
NULL)
    {
        iRc = tperrno;
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, ">>>> Init Thread %d :
OrderStatus tmalloc failed: iRc = %d \r\n",
                GetCurrentThreadId(), iRc);
            fclose(fp);
        }
        sprintf(buf, "Tpccalloc %d", iRc);
        ErrorMessage(pECB, ERR_TPALLOC_BAD,
ERR_TYPE_TUXEDO, buf, 0, 0);
        return TRUE;
    }

    if (dLog )
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "* Init Thread %d iTermId %d *
OrderStatusDataPtr: %x \r\n",
            GetCurrentThreadId(), iTermId,
            &Term.pClientData[iTermId].orderStatusDataPtr);
        fclose(fp);
    }

// DELIVERY

    if ((Term.pClientData[iTermId].deliveryDataPtr =
(DeliveryData *)tpalloc("CARRAY", NULL, sizeof(DeliveryData))) == NULL)
    {
        iRc = tperrno;
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");

```

```

        fprintf(fp, ">>>> Init Thread %d :
Delivery tmalloc failed: iRc = %d \r\n",
        GetCurrentThreadId(), iRc);
        fclose(fp);
    }
    sprintf(buf, "Tpcalloc %d", iRc);
    ErrorMessage(pECB, ERR_TPALLOC_BAD,
ERR_TYPE_TUXEDO, buf, 0, 0);
    return TRUE;
}

if ( dLog )
{
    FILE *fp;

    fp = fopen(szTpccLogPath, "ab");

    fprintf(fp, "* Init Thread %d iTermId %d *
DeliveryDataPtr: %x \r\n",
        GetCurrentThreadId(), iTermId,
&Term.pClientData[iTermId].deliveryDataPtr);
    fclose(fp);
}

//      STOCKLEVEL

if ((Term.pClientData[iTermId].stockLevelDataPtr =
(StockLevelData *)tmalloc("CARRAY", NULL, sizeof(StockLevelData))) ==
NULL)
{
    iRc = tperrno;
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, ">>>> Init Thread %d :
StockLevel tmalloc failed: iRc = %d \r\n",
        GetCurrentThreadId(), iRc);
        fclose(fp);
    }
    sprintf(buf, "Tpcalloc %d", iRc);
    ErrorMessage(pECB, ERR_TPALLOC_BAD,
ERR_TYPE_TUXEDO, buf, 0, 0);
    return TRUE;
}

if ( dLog )
{
    FILE *fp;

    fp = fopen(szTpccLogPath, "ab");

    fprintf(fp, "* Init Thread %d iTermId %d *
StockLevelDataPtr: %x \r\n",
        GetCurrentThreadId(), iTermId,
&Term.pClientData[iTermId].stockLevelDataPtr);
    fclose(fp);
}
#endif
#endif
return FALSE;
}

```

```

/* FUNCTION: BOOL Close(EXTENSION_CONTROL_BLOCK*pECB, int iTermId, int
iSyncId)
*
* PURPOSE:      This function closes the sql connection for use.
*
* ARGUMENTS:   EXTENSION_CONTROL_BLOCK *pECB   passed in structure
pointer from
*
*              int iTermId           id of browser client that this
cxn is for.
*              int iSyncId           sync id of client
browser
*
* RETURNS:     BOOL FALSE   if successfull
*              TRUE    if an error occurs and connection
cannot be
*
*              terminated.
*
* COMMENTS:    None
*/

static BOOL Close(EXTENSION_CONTROL_BLOCK *pECB, int iTermId, int
iSyncId)
{
#ifdef TOPEND

    tp_dif_structs_t *tclient_dif;
    tclient_dif = (tp_dif_structs_t
*)Term.pClientData[iTermId].tp_structadr;
    tclient_dif->info->tp_flags = TP_SIGNOFF_IMMED;

    tp_client_signoff(tclient_dif->info);
    tp_csi_free (tclient_dif);
    return TRUE;

#else /* do any database special closing, nothing at this time */
    return 0;
#endif
    UNUSEDPARAM(iSyncId);
}

/* FUNCTION: void FormatString(char *szDest, char *szPic, char *szSrc)
*
* PURPOSE:      This function formats a character string for inclusion
in the
*
*              HTML formatted page being constructed.
*
* ARGUMENTS:   char *szDest Destination buffer where formatted
string is to
*
*              be placed
*              char *szPic  picture string which describes how
character
*
*              value is to be formatted.
*              char *szSrc  character string value.
*
* RETURNS:     None
*
* COMMENTS:    This function is used to format TPC-C phone and zip value
strings.

```

```

*
*/
static void FormatString(char *szDest, char *szPic, char *szSrc)
{
    while( *szPic )
    {
        if ( *szPic == 'X' )
        {
            if ( *szSrc )
                *szDest++ = *szSrc++;
            else
                *szDest++ = ' ';
        }
        else
            *szDest++ = *szPic;

        szPic++;
    }

    *szDest = 0;
    return;
}

/* FUNCTION: char *MakeStockLevelForm(int iTermId, int iSyncId, BOOL
bInput)
*
* PURPOSE:      This function constructs the Stock Level HTML page.
*
* ARGUMENTS:   int      iTermId      client browser terminal id
               int      iSyncId     client browser sync id
               BOOL     bInput       TRUE if form is being
constructed for
*
* RETURNS:     char *                A pointer to buffer inside
client
*
*              structure where HTML form is
built.
*
* 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.
*/

```

```

static char *MakeStockLevelForm(int iTermId, int iSyncId, BOOL bInput)
{
    char      *szForm;
    szForm = (char *)Term.pClientData[iTermId].szBuffer;
    Term.pClientData[iTermId].trans.stocklevelData.stoin.w_id =
(short)Term.pClientData[iTermId].w_id;
    Term.pClientData[iTermId].trans.stocklevelData.stoin.d_id =
(short)Term.pClientData[iTermId].d_id;

    if ( bInput)
    {
        strcpy(szForm, "<HTML><HEAD><TITLE>TPC-C Stock
Level</TITLE></HEAD>\
<FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">\
<INPUT TYPE=\"hidden\" NAME=\"PI*\" VALUE=\"\">\
<INPUT TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"ERROR\" VALUE=\"0\">\

```

```

<INPUT TYPE=\"hidden\" NAME=\"FORMID\" VALUE=\"7\">); //
STOCK_LEVEL_FORM = 7

        wprintf(szForm+strlen(szForm), "<INPUT TYPE=\"hidden\"
NAME=\"TERMINID\" VALUE=\"%d\">\
<INPUT TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"%d\">", iTermId,
iSyncId);

        strcat(szForm, "<PRE>
Stock-Level<BR>");
        wprintf(szForm+strlen(szForm), "Warehouse: %4.4d
District: %2.2d<BR><BR>",

                Term.pClientData[iTermId].trans.stocklevelData.stoin.w_id,

                Term.pClientData[iTermId].trans.stocklevelData.stoin.d_id);

        strcat(szForm, "Stock Level Threshold: <INPUT
NAME=\"TT*\" SIZE=2><BR><BR>\
low stock: <BR><HR><INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"Process\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"Menu\"></FORM></HTML>");

    }
    else // Not bInput
    {

        strcpy(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=\"7\">); //
STOCK_LEVEL_FORM = 7

        wprintf(szForm+strlen(szForm), "<INPUT TYPE=\"hidden\"
NAME=\"TERMINID\" VALUE=\"%d\">\
<INPUT TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"%d\">", iTermId,
iSyncId);

        strcat(szForm, "<PRE>
Stock-Level<BR>");
        wprintf(szForm+strlen(szForm), "Warehouse: %4.4d
Stock Level Threshold: %2.2d<BR><BR>low stock: %3.3d</PRE><BR><HR>",

                Term.pClientData[iTermId].trans.stocklevelData.stoin.w_id,

                Term.pClientData[iTermId].trans.stocklevelData.stoin.d_id,

                Term.pClientData[iTermId].trans.stocklevelData.stoin.threshold,

                Term.pClientData[iTermId].trans.stocklevelData.stoout.low_stock

                );

        strcat(szForm, "<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>");
    }
}

```

```

    }

    return szForm;
}

/* FUNCTION: char *MakeMainMenuForm(int iTermId, int iSyncId)
 *
 * PURPOSE: This function makes the main menu HTML page
 *
 * ARGUMENTS:  int      iTermId      client browser terminal id
 *             int      iSyncId      client browser sync id
 *
 * RETURNS:    char *                A pointer to buffer inside
client
 *             structure where HTML form is
built.
 *
 * 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.
 */

static char *MakeMainMenuForm(int iTermId, int iSyncId)
{
    char      *szForm;
    szForm = (char *)Term.pClientData[iTermId].szBuffer;
    strcpy(szForm, "<HTML><HEAD><TITLE>TPC-C
MainMenu</TITLE></HEAD><BODY>"
           "Select Desired Transaction.<BR><HR>"
           "<FORM ACTION=\"tpcc.dll\"METHOD=\"GET\">");

    strcat(szForm, "<INPUT TYPE=\"hidden\" NAME=\"STATUSID\"
VALUE=\"0\">");
    strcat(szForm, "<INPUT TYPE=\"hidden\" NAME=\"ERROR\"
VALUE=\"0\">");
    sprintf(szForm+strlen(szForm), "<INPUT TYPE=\"hidden\"
NAME=\"TERMINID\" VALUE=\"%d\">", iTermId);
    sprintf(szForm+strlen(szForm), "<INPUT TYPE=\"hidden\"
NAME=\"SYNCID\" VALUE=\"%d\">", iSyncId);
    sprintf(szForm+strlen(szForm), "<INPUT TYPE=\"hidden\"
NAME=\"FORMID\" VALUE=\"%d\">", MAIN MENU FORM);
    strcat(szForm, "<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>");
    return szForm;
}

/* FUNCTION: char *MakeWelcomeForm(void)
 *
 * PURPOSE: This function makes the HTML based welcome form
 *
 * ARGUMENTS:  None
 *
 * RETURNS:    char *  A pointer to the static HTML welcome form.

```

```

 *
 * COMMENTS:   The welcome form is static.
 */

static char *MakeWelcomeForm(void)
{
    return szWelcomeForm;
}

/* FUNCTION: char *MakeNewOrderForm(int iTermId, BOOL bInput, BOOL
bValid, char *execution_status)
 *
 * PURPOSE: This function makes/presents the HTML based New Order Form
 *
 * ARGUMENTS:  int      iTermId      client browser terminal id
 *             int      iSyncId      client browser sync id
 *             BOOL     bInput        TRUE if form is being
constructed for
 *             BOOL     bValid        input else FALSE
ELSE FALSE    TRUE if NeworderData valid,
 *             effects output only
 *
 * RETURNS:    char *                A pointer to buffer inside
client
 *             structure where HTML form is
built.
 *
 * 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.
 */

static char *MakeNewOrderForm(int iTermId, int iSyncId, BOOL bInput,
BOOL bValid, char *execution_status)
{
    char      *szForm;
    // char      szName[146];
    // char      szCredit[14];
    int        i;

    szForm = (char *)Term.pClientData[iTermId].szBuffer;

    Term.pClientData[iTermId].trans.newOrderData.newin.w_id =
Term.pClientData[iTermId].w_id;

    if ( bInput)
    {
        strcpy(szForm, "<HTML><HEAD><TITLE>TPC-C New
Order</TITLE>"
           "</HEAD><BODY><FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">"
           "<INPUT TYPE=\"hidden\" NAME=\"PI\" VALUE=\"\">"
           "<INPUT TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"0\">"
           "<INPUT TYPE=\"hidden\" NAME=\"ERROR\" VALUE=\"0\">"
           "<INPUT TYPE=\"hidden\" NAME=\"FORMID\" VALUE=\"3\">"); //
NEW_ORDER_FORM = 3

        sprintf(szForm+strlen(szForm), "<INPUT TYPE=\"hidden\"
NAME=\"TERMINID\" VALUE=\"%d\">"
           "<INPUT TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"%d\">"
           "<PRE>
New Order<BR>"

```

```

Warehouse: %4.4d   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>","
                                iTermId,
                                iSyncId,

Term.pClientData[iTermId].trans.newOrderData.newin.w_id);

      strcat(szForm, " <INPUT NAME="\SP00*" SIZE=4> <INPUT
NAME="\IID00*" SIZE=6>
NAME="\Qty00*" SIZE=1><BR>\
  <INPUT NAME="\SP01*" SIZE=4> <INPUT NAME="\IID01*" SIZE=6>
<INPUT NAME="\Qty01*" SIZE=1><BR>\
  <INPUT NAME="\SP02*" SIZE=4> <INPUT NAME="\IID02*" SIZE=6>
<INPUT NAME="\Qty02*" SIZE=1><BR>\
  <INPUT NAME="\SP03*" SIZE=4> <INPUT NAME="\IID03*" SIZE=6>
<INPUT NAME="\Qty03*" SIZE=1><BR>\
  <INPUT NAME="\SP04*" SIZE=4> <INPUT NAME="\IID04*" SIZE=6>
<INPUT NAME="\Qty04*" SIZE=1><BR>\
  <INPUT NAME="\SP05*" SIZE=4> <INPUT NAME="\IID05*" SIZE=6>
<INPUT NAME="\Qty05*" SIZE=1><BR>\
  <INPUT NAME="\SP06*" SIZE=4> <INPUT NAME="\IID06*" SIZE=6>
<INPUT NAME="\Qty06*" SIZE=1><BR>\
  <INPUT NAME="\SP07*" SIZE=4> <INPUT NAME="\IID07*" SIZE=6>
<INPUT NAME="\Qty07*" SIZE=1><BR>\
  <INPUT NAME="\SP08*" SIZE=4> <INPUT NAME="\IID08*" SIZE=6>
<INPUT NAME="\Qty08*" SIZE=1><BR>\
  <INPUT NAME="\SP09*" SIZE=4> <INPUT NAME="\IID09*" SIZE=6>
<INPUT NAME="\Qty09*" SIZE=1><BR>\
  <INPUT NAME="\SP10*" SIZE=4> <INPUT NAME="\IID10*" SIZE=6>
<INPUT NAME="\Qty10*" SIZE=1><BR>\
  <INPUT NAME="\SP11*" SIZE=4> <INPUT NAME="\IID11*" SIZE=6>
<INPUT NAME="\Qty11*" SIZE=1><BR>\
  <INPUT NAME="\SP12*" SIZE=4> <INPUT NAME="\IID12*" SIZE=6>
<INPUT NAME="\Qty12*" SIZE=1><BR>\
  <INPUT NAME="\SP13*" SIZE=4> <INPUT NAME="\IID13*" SIZE=6>
<INPUT NAME="\Qty13*" SIZE=1><BR>\
  <INPUT NAME="\SP14*" SIZE=4> <INPUT NAME="\IID14*" SIZE=6>
<INPUT NAME="\Qty14*" SIZE=1><BR>\
Execution Status:
Total:<BR><HR>\
<INPUT TYPE="\submit\" NAME="\CMD\" VALUE="\Process\">\
<INPUT TYPE="\submit\" NAME="\CMD\" VALUE="\Menu\">\
</FORM></HTML>" );

}
else // not bInput
{
  if (bValid)
  {
    FormatHTMLString(szName,
Term.pClientData[iTermId].NewOrderData.c_last, 16),
    FormatHTMLString(szCredit,
Term.pClientData[iTermId].NewOrderData.c_credit, 2);

    strcpy(szForm, "<HTML><HEAD><TITLE>TPC-C New
Order</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="\3\">"); //
NEW_ORDER_FORM = 3

sprintf(szForm+strlen(szForm), "<INPUT TYPE="\hidden\" NAME="\TERMINID\"
VALUE="\d\">\
<INPUT TYPE="\hidden\" NAME="\SYNCID\" VALUE="\%d\">\
<PRE>
Warehouse: %4.4d   District: %2.2d                               Date: %s
<BR>\
Customer: %4.4d   Name: %s   Credit: %2.2s   %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>","
                                iTermId,
                                iSyncId,

Term.pClientData[iTermId].trans.newOrderData.newin.w_id,
Term.pClientData[iTermId].trans.newOrderData.newin.d_id,
Term.pClientData[iTermId].trans.newOrderData.newout.o_entry_d,
Term.pClientData[iTermId].trans.newOrderData.newin.c_id,
Term.pClientData[iTermId].trans.newOrderData.newout.c_last,
Term.pClientData[iTermId].trans.newOrderData.newout.c_credit,
Term.pClientData[iTermId].trans.newOrderData.newout.c_discount
* 100,
Term.pClientData[iTermId].trans.newOrderData.newout.o_id,
Term.pClientData[iTermId].trans.newOrderData.newout.o_ol_cnt,
Term.pClientData[iTermId].trans.newOrderData.newout.w_tax * 100,
Term.pClientData[iTermId].trans.newOrderData.newout.d_tax *
100);

// We could make 0-4 be hard coded in one sprintf, since we know that
there is at least 5
// order lines, or is this a benchmark special??
// etw
for(i=0;
i<Term.pClientData[iTermId].trans.newOrderData.newout.o_ol_cnt; i++)
{
  FormatHTMLString(szName,
Term.pClientData[iTermId].NewOrderData.OL[i].ol_i_name, 24);

  sprintf(szForm+strlen(szForm),
"%4.4d %6.6d %-24.24s %2.2d %3.3d %1c %$6.2f %$7.2f
<BR>",
Term.pClientData[iTermId].trans.newOrderData.newin.ol_supply_w
id[i],
Term.pClientData[iTermId].trans.newOrderData.newin.ol_i_id[i],
Term.pClientData[iTermId].trans.newOrderData.newout.i_name[i],

```



```

//      char      szTmpStr2[122];
//      char      szTmpStr3[122];
//      char      szTmpStr4[122];
int     i;
int     l;
char    *szZipPic = "XXXXX-XXXX";

szForm = (char *)Term.pClientData[iTermId].szBuffer;

Term.pClientData[iTermId].trans.paymentData.payin.w_id =
Term.pClientData[iTermId].w_id;

if ( bInput )
{
    strcpy(szForm, "<HTML><HEAD><TITLE>TPC-C
Payment</TITLE></HEAD><BODY>\
<FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">\
<INPUT TYPE=\"hidden\" NAME=\"PI*\" VALUE=\"\">\
<INPUT TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"ERROR\" VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"FORMID\" VALUE=\"4\">"); // PAYMENT_FORM
= 4

    sprintf(szForm+strlen(szForm), "<INPUT TYPE=\"hidden\"
NAME=\"TERMID\" VALUE=\"%d\">\
<INPUT TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"%d\">\
<PRE>
Date:<BR><BR>\
Warehouse: %4.4d",

                                iTermId,
                                iSyncId,

                                Term.pClientData[iTermId].trans.paymentData.payin.w_id);

    strcat(szForm,
"
District: <INPUT NAME=\"DID*\"
SIZE=1><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:
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></PRE><HR>\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"Process\"><INPUT
TYPE=\"submit\" NAME=\"CMD\" VALUE=\"Menu\">\
</BODY></FORM></HTML>");
}
else // Not bInput
{
//      FormatHTMLString(szTmpStr1,
Term.pClientData[iTermId].PaymentData.w_street_1, 20);
//      FormatHTMLString(szTmpStr2,
Term.pClientData[iTermId].PaymentData.d_street_1, 20);

//      FormatHTMLString(szTmpStr3,
Term.pClientData[iTermId].PaymentData.w_street_2, 20);

```

```

//      FormatHTMLString(szTmpStr4,
Term.pClientData[iTermId].PaymentData.d_street_2, 20);

//      FormatString(szW_Zip, szZipPic,
Term.pClientData[iTermId].trans.paymentData.payout.w_zip);
//      FormatString(szD_Zip, szZipPic,
Term.pClientData[iTermId].trans.paymentData.payout.d_zip);

//      FormatHTMLString(szTmpStr5,
Term.pClientData[iTermId].PaymentData.w_city, 20);
//      FormatHTMLString(szTmpStr6,
Term.pClientData[iTermId].PaymentData.w_state, 2);
//      FormatHTMLString(szTmpStr7,
Term.pClientData[iTermId].PaymentData.d_city, 20);
//      FormatHTMLString(szTmpStr8,
Term.pClientData[iTermId].PaymentData.d_state, 2);

//      FormatHTMLString(szTmpStr9,
Term.pClientData[iTermId].PaymentData.c_first, 16);
//      FormatHTMLString(szTmpStr10,
Term.pClientData[iTermId].PaymentData.c_middle, 2);
//      FormatHTMLString(szTmpStr11,
Term.pClientData[iTermId].PaymentData.c_last, 16);

//      FormatHTMLString(szTmpStr12,
Term.pClientData[iTermId].PaymentData.c_street_1, 20);
//      FormatHTMLString(szTmpStr13,
Term.pClientData[iTermId].PaymentData.c_credit, 2);
//
//      FormatHTMLString(szTmpStr14,
Term.pClientData[iTermId].PaymentData.d_street_2, 20);

//      FormatString(szC_Zip, szZipPic,
Term.pClientData[iTermId].trans.paymentData.payout.c_zip);
//      FormatString(szC_Phone, "XXXXXX-XXX-XXX-
XXXX",Term.pClientData[iTermId].trans.paymentData.payout.c_phone);

//      FormatHTMLString(szTmpStr15,
Term.pClientData[iTermId].PaymentData.c_city, 20);
//      FormatHTMLString(szTmpStr16,
Term.pClientData[iTermId].PaymentData.c_state, 2);

    strcpy(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=\"4\">"); // PAYMENT_FORM
= 4

    sprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\" NAME=\"TERMID\" VALUE=\"%d\">\
<INPUT TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"%d\">\
<PRE>
Date: %19s<BR><BR>\
Warehouse: %4.4d
District: %2.2d<BR>\
%-20s          %-20s<BR>\
%-20s          %-20s<BR>\
%-20s %-2s %10.10s      %-20s %-2s %10.10s<BR><BR>\
Customer: %4.4d Cust-Warehouse: %4.4d Cust-District: %2.2d<BR>\
Name:      %-16s %-2s %-16s      Since: %10s<BR>\
           %-20s          Credit: %s<BR>\
           %-20s          %%Disc: %5.2f<BR>\
           %-20s %-2s %10.10s      Phone: %-19.19s<BR><BR>\

```

```

Amount Paid:          $%7.2f      New Cust Balance: $%14.2f<BR>\
Credit Limit:      $%13.2f<BR><BR>",
                    iTermId,
                    iSyncId,
Term.pClientData[iTermId].trans.paymentData.payout.h_date,
Term.pClientData[iTermId].trans.paymentData.payin.w_id,
Term.pClientData[iTermId].trans.paymentData.payin.d_id,
Term.pClientData[iTermId].trans.paymentData.payout.w_street_1,
Term.pClientData[iTermId].trans.paymentData.payout.d_street_1,
Term.pClientData[iTermId].trans.paymentData.payout.w_street_2,
Term.pClientData[iTermId].trans.paymentData.payout.d_street_2,
Term.pClientData[iTermId].trans.paymentData.payout.w_city,
Term.pClientData[iTermId].trans.paymentData.payout.w_state,
szW_Zip,
Term.pClientData[iTermId].trans.paymentData.payout.d_city,
Term.pClientData[iTermId].trans.paymentData.payout.d_state,
szD_Zip,
Term.pClientData[iTermId].trans.paymentData.payout.c_id,
Term.pClientData[iTermId].trans.paymentData.payin.c_w_id,
Term.pClientData[iTermId].trans.paymentData.payin.c_d_id,
Term.pClientData[iTermId].trans.paymentData.payout.c_first,
Term.pClientData[iTermId].trans.paymentData.payout.c_middle,
Term.pClientData[iTermId].trans.paymentData.payout.c_last,
Term.pClientData[iTermId].trans.paymentData.payout.c_since,
Term.pClientData[iTermId].trans.paymentData.payout.c_street_1,
Term.pClientData[iTermId].trans.paymentData.payout.c_credit,
Term.pClientData[iTermId].trans.paymentData.payout.c_street_2,
Term.pClientData[iTermId].trans.paymentData.payout.c_discount*1
00,
Term.pClientData[iTermId].trans.paymentData.payout.c_city,
Term.pClientData[iTermId].trans.paymentData.payout.c_state,
szC_Zip, szC_Phone,
(float) (Term.pClientData[iTermId].trans.paymentData.payin.h_amo
unt)/100.0,
Term.pClientData[iTermId].trans.paymentData.payout.c_balance,

```

```

);
Term.pClientData[iTermId].trans.paymentData.payout.c_credit_lim
);
// This part can probably be made a little more efficient
//etw
ptr =
Term.pClientData[iTermId].trans.paymentData.payout.c_credit;
//for dedug katsuf
if (bLog){
FILE *fp1;
fp1 = fopen(szTpccLogPath, "ab");
fprintf(fp1, "c_cre= %s", ptr);
fprintf(fp1, "c_da
= %s",Term.pClientData[iTermId].trans.paymentData.payout.c_data);
fclose(fp1);
}
if ( *ptr == 'B' && *(ptr+1) == 'C' )
{
ptr =
Term.pClientData[iTermId].trans.paymentData.payout.c_data;
l = strlen( ptr ) / 50;
for(i=0; i<4; i++, ptr += 50)
{
if ( i <= 1 )
UtilStrCpy(szTmp, ptr, 50);
else
szTmp[0] = 0;
if ( !i )
{
FormatHTMLString(szTmpStr1, szTmp, 50);
wsprintf(szForm+strlen(szForm),
"Cust-Data: %-50s<BR>", szTmp);
}
else
{
FormatHTMLString(szTmpStr1, szTmp, 50);
wsprintf(szForm+strlen(szForm),
"
%-50s<BR>", szTmp);
}
}
else {
strcat(szForm, "Cust-Data: <BR><BR><BR><BR>");
}
strcat(szForm, "</PRE><HR><BR>\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..NewOrder..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Payment..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Delivery..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Order-Status..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Stock-Level..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Exit..\">\
</BODY></FORM></HTML>");
}
return szForm;
}
/* FUNCTION: char *MakeOrderStatusForm(int iTermId, int iSyncId, BOOL
bInput)
*

```

```

* PURPOSE: This function makes the HTML based Order Status Form
*
* ARGUMENTS:  int      iTermId  client browser terminal id
*             int      iSyncId  client browser sync id
*             BOOL     bInput   TRUE if form is being constructed for
input
*             else FALSE
*
* RETURNS:    char *          A pointer to buffer inside client
structure
*             where HTML form is built.
*
* 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.
*/
static char *MakeOrderStatusForm(int iTermId, int iSyncId, BOOL bInput)
{
    char      *szForm;
    //      char  c_first[98];
    //      char  c_middle[14];
    //      char  c_last[98];
    int       i;

    szForm = (char *)Term.pClientData[iTermId].szBuffer;

    Term.pClientData[iTermId].trans.orderStatusData.ordin.w_id =
Term.pClientData[iTermId].w_id;

    if ( bInput )
    {
        strcpy(szForm, "<HTML><HEAD><TITLE>TPC-C Order-
Status</TITLE></HEAD><BODY>\
<FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">\
<INPUT TYPE=\"hidden\" NAME=\"PI*\" VALUE=\"\">\
<INPUT TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"ERROR\" VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"FORMID\" VALUE=\"6\">"); //
ORDER_STATUS_FORM = 6

        wsprintf(szForm+strlen(szForm), "<INPUT TYPE=\"hidden\"
NAME=\"TERMINID\" VALUE=\"%d\">\
<INPUT TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"%d\">",
                iTermId,
                iSyncId);

        strcat(szForm, "<PRE>
Order-Status<BR>" );
        wsprintf(szForm+strlen(szForm), "Warehouse: %4.4d ",
Term.pClientData[iTermId].trans.orderStatusData.ordin.w_id);

        strcat(szForm, "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></PRE>\
<HR><INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"Process\"><INPUT
TYPE=\"submit\" NAME=\"CMD\" VALUE=\"Menu\">\
</BODY></FORM></HTML>" );
    }
}

```

```

}
else // Not bInput
{
    strcpy(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=\"6\">"); //
ORDER_STATUS_FORM = 6

    wsprintf(szForm+strlen(szForm), "<INPUT TYPE=\"hidden\"
NAME=\"TERMINID\" VALUE=\"%d\">\
<INPUT TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"%d\">",
            iTermId,
            iSyncId);

    strcat(szForm, "<PRE>
Status<BR>" );

    sprintf(szForm+strlen(szForm), "Warehouse: %4.4d
District: %2.2d<BR>\
Customer: %4.4d Name: %-16s %-2s %-16s<BR>\
Cust-Balance: $%9.2f<BR><BR>\
Order-Number: %8.8d Entry-Date: %19s
Carrier-Number: %2.2d<BR>",

            Term.pClientData[iTermId].trans.orderStatusData.ordin.w_id,
            Term.pClientData[iTermId].trans.orderStatusData.ordin.d_id,
            Term.pClientData[iTermId].trans.orderStatusData.ordout.c_id,
            Term.pClientData[iTermId].trans.orderStatusData.ordout.c_first,
            Term.pClientData[iTermId].trans.orderStatusData.ordout.c_middle,
            Term.pClientData[iTermId].trans.orderStatusData.ordout.c_last,
            Term.pClientData[iTermId].trans.orderStatusData.ordout.c_balanc
e,
            Term.pClientData[iTermId].trans.orderStatusData.ordout.o_id,
            Term.pClientData[iTermId].trans.orderStatusData.ordout.o_entry_
d,
            Term.pClientData[iTermId].trans.orderStatusData.ordout.o_carrie
r_id);

    strcat(szForm+strlen(szForm), "Supply-W      Item-Id
Qty      Amount      Delivery-Date<BR>");

    for(i=0;
i<Term.pClientData[iTermId].trans.orderStatusData.ordout.o_ol_cnt; i++)
    {
        sprintf(szForm+strlen(szForm),
                " %4.4d      %6.6d      %2.2d      $%8.2f      %10s      <BR>",

                Term.pClientData[iTermId].trans.orderStatusData.ordout.ol_suppl
y_w_id[i],
                Term.pClientData[iTermId].trans.orderStatusData.ordout.ol_i_id[
i],

```

```

        Term.pClientData[iTermId].trans.orderStatusData.ordout.ol_quant
ity[i],
        Term.pClientData[iTermId].trans.orderStatusData.ordout.ol_ammoun
t[i],
        Term.pClientData[iTermId].trans.orderStatusData.ordout.ol_deliv
ery_d[i]);
    }
    strcat(szForm, "<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..\">\
</BODY></FORM></HTML>" );
}
return szForm;
}

```

```

/* FUNCTION: char *MakeDeliveryForm(int iTermId, int iSyncId, BOOL
bInput, char *execution_status)
*
* PURPOSE:      This function puts out the HTML for the delivery form
*
* ARGUMENTS:   int      iTermId client browser terminal id
*              int      iSyncId client browser sync id
*              BOOL     bInput TRUE if form is being constructed for
input
*              else FALSE
*
* RETURNS:     char *   A pointer to buffer inside client structure
where HTML
*              form is built.
*
* 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.
*/
static char *MakeDeliveryForm(int iTermId, int iSyncId, BOOL bInput,
char *execution_status)
{
    char      *szForm;
    szForm = (char *)Term.pClientData[iTermId].szBuffer;
    Term.pClientData[iTermId].trans.deliveryData.delin.w_id =
Term.pClientData[iTermId].w_id;

```

```

    if ( bInput )
    {
        strcpy( szForm, "<HTML><HEAD><TITLE>TPC-C
Delivery</TITLE></HEAD><BODY>\
<FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">\
<INPUT TYPE=\"hidden\" NAME=\"PI*\" VALUE=\"\">\
<INPUT TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"ERROR\" VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"FORMID\" VALUE=\"5\">"; // DELIVERY_FORM
= 5

```

```

        sprintf(szForm+strlen(szForm), "<INPUT TYPE=\"hidden\"
NAME=\"TERMINID\" VALUE=\"%d\">\
<INPUT TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"%d\">",
                iTermId,
                iSyncId);
        strcat(szForm, "<PRE>
Delivery<BR>" );
        sprintf(szForm+strlen(szForm),
"Warehouse: %4.4d<BR><BR>",
Term.pClientData[iTermId].trans.deliveryData.delin.w_id);
        strcat( szForm, "Carrier Number: <INPUT NAME=\"OCD*\"
SIZE=1><BR><BR>\
Execution Status:<BR></PRE>\
<HR><INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"Process\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"Menu\">\
</BODY></FORM></HTML>" );
    }
    else // Not bInput
    {
        strcpy( szForm, "<HTML><HEAD><TITLE>TPC-C
Delivery</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=\"5\">"; // DELIVERY_FORM
= 5
        sprintf(szForm+strlen(szForm), "<INPUT TYPE=\"hidden\"
NAME=\"TERMINID\" VALUE=\"%d\">\
<INPUT TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"%d\">",
                iTermId,
                iSyncId);
        strcat(szForm, "<PRE>
Delivery<BR>" );
        sprintf(szForm+strlen(szForm), "Warehouse: %4.4d<BR><BR>",
Term.pClientData[iTermId].trans.deliveryData.delin.w_id);
        sprintf(szForm+strlen(szForm), "Carrier
Number: %2.2d<BR><BR>\
Execution Status: %25.25s<BR></PRE>",
                Term.pClientData[iTermId].trans.deliveryData.delin.o_carrier_id,
                execution_status);
        strcat(szForm, "<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>" );
    }
    return szForm;
}

```

```

}

/* FUNCTION: void UtilStrCpy(char * pDest, char * pSrc, int n)
 *
 * PURPOSE: This function copies n characters from string pSrc to pDst
 and
 *
 * places a null character at the end of the destination
 string.
 *
 * ARGUMENTS:  char   *pDest  destination string pointer
 *             char   *pSrc   source string pointer
 *             int    n       number of characters to copy
 *
 * RETURNS:    None
 *
 * COMMENTS:   Unlike strncpy this function ensures that the result
 string is
 *
 * always null terminated.
 */

static void UtilStrCpy(char * pDest, char * pSrc, int n)
{
    strncpy(pDest, pSrc, n);
    pDest[n] = '\0';

    return;
}

/* FUNCTION: void ProcessNewOrderForm(EXTENSION_CONTROL_BLOCK *pECB,
int
 *
 *                                     iTermId, int iSyncId)
 *
 * 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.
 *
 * ARGUMENTS:  EXTENSION_CONTROL_BLOCK *pECB  passed in structure
 pointer
 *
 *                                     from inetsrv.
 *
 *             int    iTermId client browser terminal id
 *             int    iSyncId client browser sync id
 *
 * RETURNS:    None
 *
 * COMMENTS:   None
 */

static void ProcessNewOrderForm(EXTENSION_CONTROL_BLOCK *pECB, int
iTermId, int iSyncId)
{
#ifdef TOPEND
    int          i, iRc;
    int msglen;
    tp_dif_structs t      *tclient_dif;
    void space_fill();
#else // assume tuxedo at this point

```

```

        int          iRc, iError;
        long ilen, *olen;
        char buf[40];
#endif

#ifdef LOCAL_ALLOC
    NewOrderData *newOrderDataPtr;
#endif

#ifdef TUX
    if((iRc = ThrTpInit(pECB)) <0)
    {
        // This is bad
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "* ProcessNewOrder Thread %d
iTermId %d Failed ThrTpInit \r\n",
                GetCurrentThreadId(), iTermId );
            fclose(fp);

        }
        sprintf(buf, "NewOrder ThrTpInit < 0 >> Thread %d",
            GetCurrentThreadId());
        ErrorMessage(pECB, ERR_TPINIT_BAD, ERR_TYPE_TUXEDO, buf, 0,
0);
    }
#endif

    memset(&Term.pClientData[iTermId].trans.newOrderData, 0,
sizeof(NewOrderData));
    Term.pClientData[iTermId].trans.newOrderData.newin.w_id =
Term.pClientData[iTermId].w_id;

    if ( (iError=GetNewOrderData(pECB->lpszQueryString,
&Term.pClientData[iTermId].trans.newOrderData)) != ERR_SUCCESS )
    {
        ErrorMessage(pECB, iError, ERR_TYPE_WEBDLL, NULL,
iTermId, iSyncId);
        return;
    }
#endif

#ifdef TOPEND
    tclient dif = (tp_dif_structs t
*)Term.pClientData[iTermId].tp_structadr;

    // Input top end neworder data
    /* Initialize Top End for tpcc transactions */
    tclient_dif->info->tp_user_dialogue_id = iTermId;
    if (bGeneric)
        space_fill(tclient_dif->service->tp_function_name,
"all_txn", TP_FUNC_NAME_LEN);
    else
        space_fill(tclient_dif->service->tp_function_name,
"neworder", TP_FUNC_NAME_LEN);

    space_fill(tclient_dif->service->tp_product_name, "tpcc",
TP_PROD_NAME_LEN);
    space_fill(tclient_dif->input_format->tp_format_name, "",
TP_FMT_NAME_LEN);

```

```

    space_fill(tclient_dif->input_format->tp_format_language, "",
TP_LANG_LEN);
    space_fill(tclient_dif->input_format->tp_format_type, "",
TP_FMTTYPE_LEN);
    tclient_dif->service->tp_function_qualifier = 0;
    msglen = sizeof(CLIENTDATA) - TOPEND_STRUCT_OFFSET;

    Term.pClientData[iTermId].type = 'N';
    if ((i = tp_mt_client_send(tclient_dif->info, tclient_dif-
>service, tclient_dif->input_format, msglen, (char
*)&Term.pClientData[iTermId].type)) != TP_OK)
    {

        if (bLog)
        {
            FILE *fp1;
            fp1 = fopen(szTpccLogPath, "ab");
            fprintf(fp1, "tp_mt_client_send returned error status
= %d\n, i");
            fclose(fp1);
        }

        ErrorMessage(pECB, TOPEND_SEND_ERROR, ERR_TYPE_WEBDLL, NULL,
iTermId, iSyncId);
        return;
    }

    msglen = sizeof(CLIENTDATA) - TOPEND_STRUCT_OFFSET;

    if ((i = tp_mt_client_receive(tclient_dif->info, wait_time,
tclient_dif->output_format, tclient_dif->service, tclient_dif->location,
&msglen, (char *)&Term.pClientData[iTermId].type)) != TP_OK)
    {
        if (bLog)
        {
            FILE *fp1;
            fp1 = fopen(szTpccLogPath, "ab");
            fprintf(fp1, "Received error status = %d from client
receive for neworder transaction\n", i);
            fclose(fp1);
        }

        ErrorMessage(pECB, TOPEND_RECEIVE_ERROR, ERR_TYPE_WEBDLL, NULL,
iTermId, iSyncId);
        return;
    }

    iRc = Term.pClientData[iTermId].retval

#else // assume we are running tuxedo at this point
#ifdef LOCAL_ALLOC // using local alloc method for tuxedo

    if ((newOrderDataPtr = (NewOrderData *) tmalloc("CARRAY", NULL,
sizeof(NewOrderData))) == NULL)
    {
        iRc = tperrno;
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "*** ProcessNewOrder Thread %d
iTermId %d tmalloc tperrno %d \r\n",

```

```

                                GetCurrentThreadId(), iTermId, iRc);
                                fclose(fp);
        }
        sprintf(buf, "NewOrder Tmalloc %d", iRc);
        ErrorMessage(pECB, ERR_TPALLOC_BAD, ERR_TYPE_TUXEDO, buf,
0, 0);
        tpfree((char *)newOrderDataPtr);
        return;
    }

    *newOrderDataPtr =
Term.pClientData[iTermId].trans.newOrderData;
    ilen = sizeof(NewOrderData);
    olen = &ilen;

    if ( dLog )
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "** ProcessNewOrder Thread %d iTermId %d
newOrderDataPtr: %x \r\n",
                                GetCurrentThreadId(), iTermId,
&newOrderDataPtr);
        fclose(fp);
    }

    if (tpcall("OPSTUXSERVER", (char *)newOrderDataPtr, ilen, (char
**)&newOrderDataPtr, (long *)olen, TPSIGRSTRT) == -1)
    {
        iRc = tperrno;
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "** ProcessNewOrder Thread %d
iTermId %d tpcall tperrno %d \r\n",
                                GetCurrentThreadId(), iTermId, iRc);
        }
        sprintf(buf, "Neworder tpcall %d", iRc);
        ErrorMessage(pECB, ERR_TPCALL_BAD, ERR_TYPE_TUXEDO,
buf, iTermId, iSyncId);
        tpfree((char *)newOrderDataPtr);
        return;
    }

    if ( dLog )
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "*** ProcessNewOrder Thread %d iTermId %d
newOrderDataPtr: %x \r\n",
                                GetCurrentThreadId(), iTermId,
&newOrderDataPtr);
        fclose(fp);
    }

    Term.pClientData[iTermId].trans.newOrderData = *newOrderDataPtr;
    iRc = newOrderDataPtr->retval;

```

```

    tpfree((char *)newOrderDataPtr);

#else // using global alloc method for tuxedo

*Term.pClientData[iTermId].newOrderDataPtr =
Term.pClientData[iTermId].trans.newOrderData;

ilen = sizeof(NewOrderData);
olen = &ilen;

if ( dLog )
{
    FILE *fp;

    fp = fopen(szTpccLogPath, "ab");
    fprintf(fp, "** ProcessNewOrder Thread %d iTermId %d
NewOrderDataPtr: %x \r\n",
            GetCurrentThreadId(), iTermId,
&Term.pClientData[iTermId].newOrderDataPtr);
    fclose(fp);
}

if (tpcall("NEWORDER", (char
*)Term.pClientData[iTermId].newOrderDataPtr, ilen,
(char **) &Term.pClientData[iTermId].newOrderDataPtr,
(long *)olen, TPSIGRSTRT) == -1)
{
    iRc = tperrno;
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "** ProcessNewOrder Thread %d
iTermId %d tpcall tperrno %d \r\n",
            GetCurrentThreadId(), iTermId, iRc);
        fclose(fp);
    }
    sprintf(buf, "Neworder tpcall %d", iRc);
    ErrorMessage(pECB, ERR_TPCALL_BAD, ERR_TYPE_TUXEDO, buf,
iTermId, iSyncId);
    // tpfree((char *)newOrderDataPtr);
    return;
}

if ( dLog )
{
    FILE *fp;

    fp = fopen(szTpccLogPath, "ab");
    fprintf(fp, "*** ProcessNewOrder Thread %d iTermId %d
NewOrderDataPtr: %x \r\n",
            GetCurrentThreadId(), iTermId,
&Term.pClientData[iTermId].newOrderDataPtr);
    fclose(fp);
}

Term.pClientData[iTermId].trans.newOrderData =
*Term.pClientData[iTermId].newOrderDataPtr;

iRc = Term.pClientData[iTermId].newOrderDataPtr->retval;

#endif
#endif

```

```

if(bLog){
    FILE *fp1;
    fp1 = fopen(szTpccLogPath, "ab");
    fprintf(fp1, ">> ProcessNewOrder iRc = %d \r\n", iRc);
    fclose(fp1);
}

if ( iRc < 0)
{
    ErrorMessage(pECB, ERR_NEW_ORDER_NOT_PROCESSED, ERR_TYPE_WEBDLL,
NULL, iTermId, iSyncId);
    return;
}

if (iRc == 0)
WriteZString(pECB, MakeNewOrderForm(iTermId, iSyncId, FALSE,
FALSE, "Item number is not valid"));
else
WriteZString(pECB, MakeNewOrderForm(iTermId, iSyncId, FALSE,
TRUE, "transaction committed"));
return;
}

/* FUNCTION: void ProcessPaymentForm(EXTENSION_CONTROL_BLOCK *pECB, int
iTermId,
*
* int iSyncId)
*
* 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
* int iSyncId client browser sync id
*
* RETURNS: None
*
* COMMENTS: None
*/

static void ProcessPaymentForm(EXTENSION_CONTROL_BLOCK *pECB, int
iTermId, int iSyncId)
{
#ifdef TOPEND
    int i, iRc;
    tp dif_structs_t *tclient_dif;
    int msglen;
#else // assume tuxedo at this point
    int iRc, iError;
    long ilen, *olen;
    // PECBINFO pEcbInfo;
    char buf[40];
#endif
}

```

```

#ifdef LOCAL_ALLOC
    PaymentData *paymentDataPtr;
#endif

#ifdef TUX
    if((iRc = ThrTpInit(pECB)) <0)
    {
        // This is bad
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "* ProcessPayment Thread %d
iTermId %d Failed ThrTpInit \r\n",
                GetCurrentThreadId(), iTermId );
            fclose(fp);
        }
        sprintf(buf, "Payment ThrTpInit < 0 >> Thread %d",
GetCurrentThreadId());
        ErrorMessage(pECB, ERR_TPINIT_BAD, ERR_TYPE_TUXEDO, buf, 0,
0);
    }
#endif

    memset(&Term.pClientData[iTermId].trans.paymentData, 0,
sizeof(PaymentData));
    Term.pClientData[iTermId].trans.paymentData.payin.w_id =
Term.pClientData[iTermId].w_id;

    if ( (iError=GetPaymentData(pECB->lpszQueryString,
&Term.pClientData[iTermId].trans.paymentData)) !=ERR_SUCCESS )
    {
        ErrorMessage(pECB, iError, ERR_TYPE_WEBDLL, NULL, iTermId,
iSyncId);
        return;
    }
#endif

#ifdef TOPEND

    tclient_dif = (tp_dif_structs_t
*)Term.pClientData[iTermId].tp_structadr;

    // Input topend information for call to process payment form

    tclient_dif->info->tp_user_dialogue_id = iTermId;

    if (bGeneric)
        space_fill(tclient_dif->service->tp_function_name,
"all_txn",TP_FUNC_NAME_LEN);
    else
        space_fill(tclient_dif->service->tp_function_name,
"payment",TP_FUNC_NAME_LEN);

    space_fill(tclient_dif->service->tp_product_name, "tpcc",
TP_PROD_NAME_LEN);
    space_fill(tclient_dif->input_format->tp_format_name, "",
TP_FMT_NAME_LEN);
    space_fill(tclient_dif->input_format->tp_format_language, "",
TP_LANG_LEN);
    space_fill(tclient_dif->input_format->tp_format_type, "",
TP_FMTTYPE_LEN);
    tclient_dif->service->tp_function_qualifier = 0;

```

```

msglen = sizeof(CLIENTDATA) - TOPEND_STRUCT_OFFSET;
    Term.pClientData[iTermId].type = 'P';

    // tp_system_log_text("ProcessPayment:", "before
tp_mt_client_send");

    if ((i = tp_mt_client_send(tclient_dif->info, tclient_dif-
>service,
        tclient_dif->input_format, msglen, (char
*)&Term.pClientData[iTermId].type)) != TP_OK)
    {
        ErrorMessage(pECB, TOPEND_SEND_ERROR, ERR_TYPE_WEBDLL, NULL,
iTermId, iSyncId);

        if (bLog)
        {
            FILE *fp1;
            fp1 = fopen(szTpccLogPath, "ab");
            fprintf(fp1, "Received error status = %d on payment send
fnt\n", i);
            fclose(fp1);
        }
        return;
    }

    msglen = sizeof(CLIENTDATA) - TOPEND_STRUCT_OFFSET;
    // tp_system_log_text("ProcessPayment:", "before
tp_mt_client_receive");
    if ((i = tp_mt_client_receive(tclient_dif->info, wait_time,
tclient_dif->output_format, tclient_dif->service,
tclient_dif->location, &msglen, (char
*)&Term.pClientData[iTermId].type)) != TP_OK)
    {
        //tp_system_log_text("ProcessPayment:", "Error on
tp_mt_client_receive");
        if (bLog)
        {
            FILE *fp1;
            fp1 = fopen(szTpccLogPath, "ab");
            fprintf(fp1, "Received error status = %d from client receive
for payment transaction\n", i);
            fclose(fp1);
        }

        ErrorMessage(pECB, TOPEND_RECEIVE_ERROR, ERR_TYPE_WEBDLL, NULL,
iTermId, iSyncId);
        return;
    }

    iRc = Term.pClientData[iTermId].retval;

#else // assume tuxedo running here

#ifdef LOCAL_ALLOC

    if ((paymentDataPtr = (PaymentData *) tmalloc("CARRAY", NULL,
sizeof(PaymentData))) == NULL)
    {
        iRc = tperrno;
    }

```

```

        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "** ProcessPayment Thread %d
iTermId %d tmalloc tperrno %d \r\n",
        GetCurrentThreadId(), iTermId, iRc);
        fclose(fp);
    }
    sprintf(buf, "Payment Tmalloc %d", iRc);
    ErrorMessage(pECB, ERR_TPALLOC_BAD, ERR_TYPE_TUXEDO, buf,
0, 0);

    tpfree((char *)paymentDataPtr);
    return;
}

*paymentDataPtr= Term.pClientData[iTermId].trans.paymentData;

ilen = sizeof(PaymentData);
olen = &ilen;

    if ( dLog )
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "** ProcessPayment Thread %d iTermId %d
paymentDataPtr: %x \r\n",
        GetCurrentThreadId(), iTermId, &paymentDataPtr);
        fclose(fp);
    }

    if (tpcall("OPSTUXSERVER", (char *)paymentDataPtr, ilen, (char
**) &paymentDataPtr, (long *)olen, TPSIGRSTRT) == -1)
    {
        iRc = tperrno;
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "** ProcessPayment Thread %d
iTermId %d tpcall tperrno %d \r\n",
            GetCurrentThreadId(), iTermId, iRc);
            fclose(fp);

            sprintf(buf, "Payment tpcall %d", iRc);
            ErrorMessage(pECB, ERR_TPCALL_BAD, ERR_TYPE_TUXEDO, buf,
iTermId, iSyncId);
            tpfree((char *)paymentDataPtr);
            return;
        }
    }

    if ( dLog )
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "** ProcessPayment Thread %d iTermId %d
paymentDataPtr: %x \r\n",
        GetCurrentThreadId(), iTermId, &paymentDataPtr);
        fclose(fp);
    }

    Term.pClientData[iTermId].trans.paymentData = *paymentDataPtr;

```

```

        iRc = paymentDataPtr->retval;

        tpfree((char *)paymentDataPtr);

    #else

        *Term.pClientData[iTermId].paymentDataPtr =
Term.pClientData[iTermId].trans.paymentData;

        ilen = sizeof(PaymentData);
        olen = &ilen;

        if ( dLog )
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "** ProcessPayment Thread %d iTermId %d
PaymentDataPtr: %x \r\n",
            GetCurrentThreadId(), iTermId,
&Term.pClientData[iTermId].paymentDataPtr);
            fclose(fp);
        }

        if (tpcall("PAYMENT", (char
*)Term.pClientData[iTermId].paymentDataPtr, ilen,
(char **) &Term.pClientData[iTermId].paymentDataPtr,
(long *)olen, TPSIGRSTRT) == -1)
        {
            iRc = tperrno;
            {
                FILE *fp;

                fp = fopen(szTpccLogPath, "ab");
                fprintf(fp, "** ProcessPayment Thread %d
iTermId %d tpcall tperrno %d \r\n",
                GetCurrentThreadId(), iTermId, iRc);
                fclose(fp);

                sprintf(buf, "Payment tpcall %d", iRc);
                ErrorMessage(pECB, ERR_TPCALL_BAD, ERR_TYPE_TUXEDO, buf,
iTermId, iSyncId);
                // tpfree((char *)PaymentDataPtr);
                return;
            }
        }

        if ( dLog )
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "** ProcessPayment Thread %d iTermId %d
PaymentDataPtr: %x \r\n",
            GetCurrentThreadId(), iTermId,
&Term.pClientData[iTermId].paymentDataPtr);
            fclose(fp);
        }

        Term.pClientData[iTermId].trans.paymentData =
*Term.pClientData[iTermId].paymentDataPtr;

        iRc = Term.pClientData[iTermId].paymentDataPtr->retval;

```

```

#endif
#endif

    if ( iRc == 0 )
        ErrorMessage(pECB, ERR_PAYMENT_INVALID_CUSTOMER,
ERR_TYPE_WEBDLL, NULL, iTermId, iSyncId);
    else if ( iRc < 0 )
        ErrorMessage(pECB, ERR_PAYMENT_NOT_PROCESSED, ERR_TYPE_WEBDLL,
NULL, iTermId, iSyncId);
    else
    {
        WriteZString(pECB, MakePaymentForm(iTermId, iSyncId, FALSE) );
    }

    return;
}

/* FUNCTION: void ProcessOrderStatusForm(EXTENSION_CONTROL_BLOCK *pECB,
int
*
*                               iTermId, int iSyncId)
*
* 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
*
*          int
*                               iSyncId client browser sync id
*
* RETURNS:    None
*
* COMMENTS:   None
*
*/
static void ProcessOrderStatusForm(EXTENSION_CONTROL_BLOCK *pECB, int
iTermId, int iSyncId)
{
#ifdef TOPEND
    int          i, iRc;
    int          iError;
    void space_fill();
    tp_dif_structs_t *tclient_dif;
    int msglen;
#else // assume tuxedo
    int          iRc, iError;
    long ilen, *olen;
    //          PECBINFO      pEcbInfo;
    char buf[40];
#endif

#ifdef LOCAL_ALLOC
    OrderStatusData *orderStatusDataPtr;
#endif

```

```

#ifdef TUX

    if((iRc = ThrTpInit(pECB)) <0)
    {
        // This is bad
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "* ProcessOrderStatus Thread %d
iTermId %d Failed ThrTpInit \r\n",
                GetCurrentThreadId(), iTermId );
            fclose(fp);

        }
        sprintf(buf, "OrderStatus ThrTpInit < 0 >> Thread %d",
GetCurrentThreadId());
        ErrorMessage(pECB, ERR_TPINIT_BAD, ERR_TYPE_TUXEDO, buf, 0,
0);
    }
}

#endif

    memset(&Term.pClientData[iTermId].trans.orderStatusData, 0,
sizeof(OrderStatusData));
    Term.pClientData[iTermId].trans.orderStatusData.ordin.w_id =
Term.pClientData[iTermId].w_id;
    if ( (iError=GetOrderStatusData(pECB->lpszQueryString,
&Term.pClientData[iTermId].trans.orderStatusData)) !=
ERR_SUCCESS )
    {
        ErrorMessage(pECB, iError, ERR_TYPE_WEBDLL, NULL, iTermId,
iSyncId);
        return;
    }

#ifdef TOPEND

    tclient_dif = (tp_dif_structs_t
*)Term.pClientData[iTermId].tp_structadr;

    /* Initialize Top End for tpcc transactions */
    tclient_dif->info->tp_user_dialogue_id = iTermId;

    if (bGeneric)
        space_fill(tclient_dif->service->tp_function_name,
"all_txn", TP_FUNC_NAME_LEN);
    else
        space_fill(tclient_dif->service->tp_function_name,
"ordstat", TP_FUNC_NAME_LEN);

    space_fill(tclient_dif->service->tp_product_name, "tpcc",
TP_PROD_NAME_LEN);
    space_fill(tclient_dif->input_format->tp_format_name, "",
TP_FMT_NAME_LEN);
    space_fill(tclient_dif->input_format->tp_format_language, "",
TP_LANG_LEN);
    space_fill(tclient_dif->input_format->tp_format_type, "",
TP_FMTTYPE_LEN);
    tclient_dif->service->tp function qualifier = 0;
    msglen = sizeof(CLIENTDATA) - TOPEND_STRUCT_OFFSET;

    Term.pClientData[iTermId].type = 'O';

```

```

    if ((i = tp_mt_client_send(tclient_dif->info, tclient_dif-
>service,
    tclient_dif->input_format, msglen, (char
*)&Term.pClientData[iTermId].type)) != TP_OK)
    {
        if (bLog)
        {
            FILE *fp1;
            fp1 = fopen(szTpccLogPath, "ab");
            fprintf(fp1, "Received error status = %d on client send for
order status transaction\n", i);
            fclose(fp1);
        }

        ErrorMessage(pECB, TOPEND_SEND_ERROR, ERR_TYPE_WEBDLL, NULL,
iTermId, iSyncId);
        return;
    }

    msglen = sizeof(CLIENTDATA) - TOPEND_STRUCT_OFFSET;

    if ((i = tp_mt_client_receive(tclient_dif->info, wait time,
tclient_dif->output_format, tclient_dif->service, tclient_dif->location,
&msglen, (char *)&Term.pClientData[iTermId].type)) != TP_OK)
    {
        if (bLog)
        {
            FILE *fp1;
            fp1 = fopen(szTpccLogPath, "ab");
            fprintf(fp1, "Received error status = %d from client receive
for order status transaction\n", i);
            fclose(fp1);
        }
        fprintf(stderr, "Received error status = %d from client receive
for order status transaction\n", i);
        ErrorMessage(pECB, TOPEND_RECEIVE_ERROR, ERR_TYPE_WEBDLL, NULL,
iTermId, iSyncId);
        return;
    }

    iRc = Term.pClientData[iTermId].retval;

#else

#ifdef LOCAL_ALLOC

    if ((orderStatusDataPtr = (OrderStatusData *) tmalloc("CARRAY",
NULL, sizeof(OrderStatusData))) == NULL)
    {
        iRc = tperrno;
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "** ProcessOrderStatus Thread %d
iTermId %d tmalloc tperrno %d \r\n",
GetCurrentThreadId(), iTermId, iRc);
            fclose(fp);
        }
        sprintf(buf, "OrderStatus Tmalloc %d", iRc);
        ErrorMessage(NULL, ERR_TALLOC_BAD, ERR_TYPE_TUXEDO, buf,
0, 0);
    }

```

```

        tpfree((char *)orderStatusDataPtr);
        return;
    }

    *orderStatusDataPtr =
Term.pClientData[iTermId].trans.orderStatusData;

    ilen = sizeof(OrderStatusData);
    olen = &ilen;

    if ( dLog )
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "** ProcessOrderStatus Thread %d iTermId %d
orderStatusDataPtr: %x \r\n",
GetCurrentThreadId(), iTermId,
&orderStatusDataPtr);
        fclose(fp);
    }

    if (tpcall("OPSTUXSERVER", (char *)orderStatusDataPtr, ilen, (char
**)&orderStatusDataPtr, (long *)olen, TPSIGRSTRT) == -1)
    {
        iRc = tperrno;
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "** ProcessOrderStatus Thread %d
iTermId %d tpcall tperrno %d \r\n",
GetCurrentThreadId(), iTermId, iRc);
            fclose(fp);
        }
        sprintf(buf, "OrderStatus tpcall %d", iRc);
        ErrorMessage(pECB, ERR_TPCALL_BAD, ERR_TYPE_TUXEDO, buf,
iTermId, iSyncId);
        tpfree((char *)orderStatusDataPtr);
        return;
    }

    if ( dLog )
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "** ProcessOrderStatus Thread %d iTermId %d
orderStatusDataPtr: %x \r\n",
GetCurrentThreadId(), iTermId,
&orderStatusDataPtr);
        fclose(fp);
    }

    Term.pClientData[iTermId].trans.orderStatusData =
*orderStatusDataPtr;

    iRc = orderStatusDataPtr->retval;

    tpfree((char *)orderStatusDataPtr);

#else

```

```

*Term.pClientData[iTermId].orderStatusDataPtr =
Term.pClientData[iTermId].trans.orderStatusData;

ilen = sizeof(OrderStatusData);
olen = &ilen;

if ( dLog )
{
    FILE *fp;

    fp = fopen(szTpccLogPath, "ab");
    fprintf(fp, "** ProcessOrderStatus Thread %d iTermId %d
OrderStatusDataPtr: %x \r\n",
            GetCurrentThreadId(), iTermId,
&Term.pClientData[iTermId].orderStatusDataPtr);
    fclose(fp);
}

if (tpcall("ORDERSTATUS", (char
*)Term.pClientData[iTermId].orderStatusDataPtr, ilen,
(char **) &Term.pClientData[iTermId].orderStatusDataPtr,
(long *)olen, TPSIGRSTRT) == -1)
{
    iRc = tperrno;
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "** ProcessOrderStatus Thread %d
iTermId %d tpcall tperrno %d \r\n",
                GetCurrentThreadId(), iTermId, iRc);
        fclose(fp);
    }
    sprintf(buf, "OrderStatus tpcall %d", iRc);
    ErrorMessage(pECB, ERR_TPCALL_BAD, ERR_TYPE_TUXEDO, buf,
iTermId, iSyncId);
    // tpfree((char
*)Term.pClientData[iTermId].orderStatusDataPtr);
    return;
}

if ( dLog )
{
    FILE *fp;

    fp = fopen(szTpccLogPath, "ab");
    fprintf(fp, "** ProcessOrderStatus Thread %d iTermId %d
OrderStatusDataPtr: %x \r\n",
            GetCurrentThreadId(), iTermId,
&Term.pClientData[iTermId].orderStatusDataPtr);
    fclose(fp);
}

Term.pClientData[iTermId].trans.orderStatusData =
*Term.pClientData[iTermId].orderStatusDataPtr;

iRc = Term.pClientData[iTermId].orderStatusDataPtr->retval;

#endif
#endif

if ( iRc == 0 )

```

```

    ErrorMessage(pECB, ERR_NOSUCH_CUSTOMER, ERR_TYPE_WEBDLL, NULL,
iTermId, iSyncId);
    else if ( iRc < 0 )
        ErrorMessage(pECB, ERR_ORDER_STATUS_NOT_PROCESSED,
ERR_TYPE_WEBDLL, NULL, iTermId, iSyncId);
    else
        WriteZString(pECB, MakeOrderStatusForm(iTermId, iSyncId,
FALSE) );
}

return;

}

/* FUNCTION: void ProcessDeliveryForm(EXTENSION_CONTROL_BLOCK *pECB,
*                                     int iTermId, int iSyncId)
*
* PURPOSE: This function gets and validates the input data from
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
*                                     int iSyncId clinet browser sync id
*
* RETURNS: None
*
* COMMENTS: None
*/

static void ProcessDeliveryForm(EXTENSION_CONTROL_BLOCK *pECB, int
iTermId, int iSyncId)
{
#ifdef TOPEND
    char szTmp[26];
    int i;
    void space fill();
    tp_dif_structs_t *tclient_dif;
    int msglen;
#else
    int iRc;
    // int iError;
    char szTmp[26];
    long ilen, *olen;
    // int tmp;
    char buf[40];
#endif

#ifdef LOCAL_ALLOC
    DeliveryData *deliveryDataPtr;
#endif

#ifdef TUX
    if((iRc = ThrTpInit(pECB)) <0)

```

```

    {
        // This is bad
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "** ProcessDelivery Thread %d
iTermId %d Failed ThrTpInit \r\n",
                GetCurrentThreadId(), iTermId );
            fclose(fp);
        }
        sprintf(buf, "Delivery ThrTpInit < 0 >> Thread %d",
GetCurrentThreadId());
        ErrorMessage(pECB, ERR_TPINIT_BAD, ERR_TYPE_TUXEDO, buf, 0,
0);
    }
#endif

    memset(&Term.pClientData[iTermId].trans.deliveryData, 0,
sizeof(DeliveryData));
    Term.pClientData[iTermId].trans.deliveryData.delin.w_id =
Term.pClientData[iTermId].w_id;

    if ( !GetKeyValue(pECB->lpszQueryString, "OCD*", szTmp,
sizeof(szTmp)) )
    {
        ErrorMessage(pECB, ERR_DELIVERY_MISSING_OCD_KEY,ERR_TYPE_WEBDLL,
NULL, iTermId, iSyncId);
        return;
    }

    if ( !IsNumeric(szTmp) )
    {
        ErrorMessage(pECB, ERR_DELIVERY_CARRIER_INVALID,ERR_TYPE_WEBDLL,
NULL, iTermId, iSyncId);
        return;
    }

    Term.pClientData[iTermId].trans.deliveryData.delin.o_carrier_id =
atoi(szTmp);
    if
( Term.pClientData[iTermId].trans.deliveryData.delin.o_carrier_id > 10
||
    Term.pClientData[iTermId].trans.deliveryData.delin.o_carrier_id
< 1 )
    {
        ErrorMessage(pECB, ERR_DELIVERY_CARRIER_ID_RANGE,ERR_TYPE_WEBDLL,
NULL, iTermId, iSyncId);
        return;
    }

    //post delivery info

#ifdef TOPEND

    tclient_dif = (tp_dif_structs_t
*)Term.pClientData[iTermId].tp_structadr;

    /* Initialize Top End for tpcc transactions */
    tclient_dif->info->tp_user_dialogue_id = iTermId;
    tclient_dif->info->tp_flags = TP_NO_RESPONSE;

```

```

        space_fill(tclient_dif->service->tp_function_name,
"del_bat",TP_FUNC_NAME_LEN);
        space_fill(tclient_dif->service->tp_product_name, "tpcc",
TP_PROD_NAME_LEN);
        space_fill(tclient_dif->input_format->tp_format_name, "",
TP_FMT_NAME_LEN);
        space_fill(tclient_dif->input_format->tp_format_language, "",
TP_LANG_LEN);
        space_fill(tclient_dif->input_format->tp_format_type, "",
TP_FMTTYPE_LEN);
        tclient_dif->service->tp_function_qualifier = 0;
        msglen = sizeof(CLIENTDATA) - TOPEND_STRUCT_OFFSET;

        Term.pClientData[iTermId].type = 'D';

        /* get the time the entry is put on the queue */
        ftime(&timebuffer);
        Term.pClientData[iTermId].trans.deliveryData.delin.qtime =
timebuffer.time + timebuffer.millitm/1000.0;

        if ((i = tp_mt_client_send(tclient_dif->info, tclient_dif-
>service,
            tclient_dif->input_format, msglen, (char
*)&Term.pClientData[iTermId].type)) != TP_OK)
        {
            if (bLog)
            {
                FILE *fp1;
                fp1 = fopen(szTpccLogPath, "ab");
                fprintf(fp1, "Received error status = %d on client send for
delivery transaction\n", i);
                fclose(fp1);
            }

            ErrorMessage(pECB, TOPEND_SEND_ERROR, ERR_TYPE_WEBDLL, NULL,
iTermId, iSyncId);
            return;
        }

        /* msglen might need to be 0, as not receiving anything put ack
from server */
        msglen = sizeof(CLIENTDATA) - TOPEND_STRUCT_OFFSET;

        if ((i = tp_mt_client_receive(tclient_dif->info, wait time,
tclient_dif->output_format, tclient_dif->service, tclient_dif->location,
&msglen, (char *)&Term.pClientData[iTermId].type)) != TP_OK)
        {
            if (bLog)
            {
                FILE *fp1;
                fp1 = fopen(szTpccLogPath, "ab");
                fprintf(fp1, "Received error status = %d from client receive
for order status transaction\n", i);
                fclose(fp1);
            }

            ErrorMessage(pECB, TOPEND_RECEIVE_ERROR, ERR_TYPE_WEBDLL, NULL,
iTermId, iSyncId);
            return;
        }

        /* reset flags for next txn */
        tclient_dif->info->tp_flags = TP_NOFLAGS;
    }
else

```

```

#ifdef LOCAL_ALLOC
    if ((deliveryDataPtr = (DeliveryData *) tmalloc("CARRAY", NULL,
    sizeof(DeliveryData))) == NULL)
    {
        iRc = tperrno;
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "* ProcessDelivery Thread %d
iTermId %d tmalloc tperrno %d \r\n",
            GetCurrentThreadId(), iTermId, iRc);
            fclose(fp);
        }
        sprintf(buf, "Delivery Tpcalloc %d", iRc);
        ErrorMessage(NULL, ERR_TPALLOC_BAD, ERR_TYPE_TUXEDO, buf,
0, 0);
        tpfree((char *)deliveryDataPtr);
        return;
    }

    *deliveryDataPtr =
Term.pClientData[iTermId].trans.deliveryData;

    ilen = sizeof(DeliveryData);
    olen = &ilen;

    if ( dLog )
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "* ProcessDelivery Thread %d iTermId %d
deliveryDataPtr: %x \r\n",
            GetCurrentThreadId(), iTermId,
&deliveryDataPtr);
        fclose(fp);
    }

    /* get the time the entry is put on the queue */
    _ftime(&timebuffer);
    deliveryDataPtr->delin.qtime = timebuffer.time +
timebuffer.millitm/1000.0;

    if (tpacall("OPSTUXSERVER", (char *)deliveryDataPtr, ilen,
TPNOREPLY|TPSIGRSTRT) != 0) {
        iRc = tperrno;
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "* ProcessDelivery Thread %d
iTermId %d tpcall tperrno %d \r\n",
            GetCurrentThreadId(), iTermId, iRc);
            fclose(fp);
        }
        sprintf(buf, "Delivery tpcall %d", iRc);
        ErrorMessage(pECB, ERR_TPCALL_BAD, ERR_TYPE_TUXEDO, buf,
iTermId, iSyncId);
        tpfree((char *)deliveryDataPtr);
        return;
    }
}

```

```

}

    if ( dLog )
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "** ProcessDelivery Thread %d iTermId %d
deliveryDataPtr: %x \r\n",
            GetCurrentThreadId(), iTermId,
&deliveryDataPtr);
        fclose(fp);
    }

    tpfree((char *)deliveryDataPtr);

    #else
    *Term.pClientData[iTermId].deliveryDataPtr =
Term.pClientData[iTermId].trans.deliveryData;

    ilen = sizeof(DeliveryData);
    olen = &ilen;

    if ( dLog )
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "* ProcessDelivery Thread %d iTermId %d
DeliveryDataPtr: %x \r\n",
            GetCurrentThreadId(), iTermId,
&Term.pClientData[iTermId].deliveryDataPtr);
        fclose(fp);
    }

    /* get the time the entry is put on the queue */
    _ftime(&timebuffer);
    Term.pClientData[iTermId].deliveryDataPtr->delin.qtime =
timebuffer.time + timebuffer.millitm/1000.0;

    if (tpcall("DELIVERY", (char
*)Term.pClientData[iTermId].deliveryDataPtr, ilen,
(char **)&Term.pClientData[iTermId].deliveryDataPtr,
(long *)olen, TPSIGRSTRT) == -1) {
        iRc = tperrno;
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "* ProcessDelivery Thread %d
iTermId %d tpcall tperrno %d \r\n",
            GetCurrentThreadId(), iTermId, iRc);
            fclose(fp);
        }
        sprintf(buf, "Delivery tpcall %d", iRc);
        ErrorMessage(pECB, ERR_TPCALL_BAD, ERR_TYPE_TUXEDO, buf,
iTermId, iSyncId);
        // tpfree((char *)Term.pClientData[iTermId].deliveryDataPtr);
        return;
    }

    if ( dLog )
    {
        FILE *fp;

```

```

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "*** ProcessDelivery Thread %d iTermId %d
DeliveryDataPtr: %x \r\n",
                GetCurrentThreadId(), iTermId,
                &Term.pClientData[iTermId].deliveryDataPtr);
        fclose(fp);
    }

    iRc = Term.pClientData[iTermId].deliveryDataPtr->retval;

#endif
#endif

    WriteZString(pECB, MakeDeliveryForm(iTermId, iSyncId, FALSE,
"Delivery has been queued.));
    return;
}

/* FUNCTION: void ProcessStockLevelForm(EXTENSION_CONTROL_BLOCK *pECB,
int
*
*           iTermId, int iSyncId)
*
* 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
*           int iSyncId client browser sync id
*
* RETURNS: None
*
* COMMENTS: None
*/

static void ProcessStockLevelForm(EXTENSION_CONTROL_BLOCK *pECB, int
iTermId, int iSyncId)
{
#ifdef TOPEND
    char szTmp[26];
    int i, iRc;
    int msglen;
    void space_fill();
    tp_dif_structs_t *tclient_dif;
#else
    int iRc;
    // int iError;
    char szTmp[26];
    // BOOL bRc;
    long ilen, *olen;
    // PECBINFO pEcbInfo;
    char buf[40];
#endif
#ifdef LOCAL_ALLOC

```

```

        StockLevelData *stockLevelDataPtr;
#endif

#ifdef TUX
    if((iRc = ThrTpInit(pECB)) <0)
    {
        // This is bad
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "* ProcessStockLevel Thread %d
iTermId %d Failed ThrTpInit \r\n",
                GetCurrentThreadId(), iTermId );
            fclose(fp);
        }
        sprintf(buf, "StockLevel ThrTpInit < 0 >> Thread %d",
GetCurrentThreadId());
        ErrorMessage(pECB, ERR_TPINIT_BAD, ERR_TYPE_TUXEDO, buf, 0,
0);
    }
#endif

    memset(&Term.pClientData[iTermId].trans.stocklevelData, 0,
sizeof(StockLevelData));
    Term.pClientData[iTermId].trans.stocklevelData.stoin.w_id =
Term.pClientData[iTermId].w_id;
    Term.pClientData[iTermId].trans.stocklevelData.stoin.d_id =
Term.pClientData[iTermId].d_id;
    if ( !GetKeyValue(pECB->lpszQueryString, "TT*", szTmp,
sizeof(szTmp)) )
    {
        ErrorMessage(pECB,
ERR_STOCKLEVEL_MISSING_THRESHOLD_KEY,ERR_TYPE_WEBDLL, NULL, iTermId,
iSyncId);
        return;
    }
    if ( !IsNumeric(szTmp) )
    {
        ErrorMessage(pECB,
ERR_STOCKLEVEL_THRESHOLD_INVALID,ERR_TYPE_WEBDLL, NULL, iTermId,
iSyncId);
        return;
    }
    Term.pClientData[iTermId].trans.stocklevelData.stoin.threshold =
atoi(szTmp);
    if
( Term.pClientData[iTermId].trans.stocklevelData.stoin.threshold >= 100
||
Term.pClientData[iTermId].trans.stocklevelData.stoin.threshold <= 0 )
    {
        ErrorMessage(pECB,
ERR_STOCKLEVEL_THRESHOLD_RANGE,ERR_TYPE_WEBDLL, NULL, iTermId,
iSyncId);
        return;
    }
#endif
#ifdef TOPEND
    tclient_dif = (tp_dif_structs_t
*)Term.pClientData[iTermId].tp_structadr;

```

```

/* Initialize Top End for tpc transactions */
tclient_dif->info->tp_user_dialogue_id = iTermId;

if (bGeneric)
    space_fill(tclient_dif->service->tp_function_name,
"all_txn", TP_FUNC_NAME_LEN);
else
    space_fill(tclient_dif->service->tp_function_name,
"stklevel", TP_FUNC_NAME_LEN);

    space_fill(tclient_dif->service->tp_product_name, "tpcc",
TP_PROD_NAME_LEN);
    space_fill(tclient_dif->input_format->tp_format_name, "",
TP_FMT_NAME_LEN);
    space_fill(tclient_dif->input_format->tp_format_language, "",
TP_LANG_LEN);
    space_fill(tclient_dif->input_format->tp_format_type, "",
TP_FMTTYPE_LEN);
    tclient_dif->service->tp_function_qualifier = 0;
    msglen = sizeof(CLIENTDATA) - TOPEND_STRUCT_OFFSET;

    Term.pClientData[iTermId].type = 'S';

    if ((i = tp_mt_client_send(tclient_dif->info, tclient_dif-
>service, tclient_dif->input_format, msglen, (char
*)&Term.pClientData[iTermId].type)) != TP_OK)
    {

        if (bLog)
        {
            FILE *fp1;
            fp1 = fopen(szTpccLogPath, "ab");
            fprintf(fp1, "Received error status = %d from client send
for stock level transaction\n", i);
            fclose(fp1);
        }

        ErrorMessage(pECB, TOPEND_SEND_ERROR, ERR_TYPE_WEBDLL, NULL,
iTermId, iSyncId);
        return;
    }

    msglen = sizeof(CLIENTDATA) - TOPEND_STRUCT_OFFSET;

    if ((i = tp_mt_client_receive(tclient_dif->info, wait time,
tclient_dif->output_format, tclient_dif->service, tclient_dif->location,
&msglen, (char *)&Term.pClientData[iTermId].type)) != TP_OK)
    {
        if (bLog)
        {
            FILE *fp1;
            fp1 = fopen(szTpccLogPath, "ab");
            fprintf(fp1, "Received error status = %d from client receive
for stock level transaction\n", i);
            fclose(fp1);
        }

        ErrorMessage(pECB, TOPEND_RECEIVE_ERROR, ERR_TYPE_WEBDLL, NULL,
iTermId, iSyncId);
        return;
    }

    iRc = Term.pClientData[iTermId].retval;
#else

```

```

#ifdef LOCAL_ALLOC
    if ((stockLevelDataPtr = (StockLevelData *) tmalloc("CARRAY",
NULL, sizeof(StockLevelData))) == NULL)
    {
        iRc = tperrno;
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "* ProcessStockLevel Thread %d
iTermId %d tmalloc tperrno %d \r\n",
GetCurrentThreadId(), iTermId, iRc);
            fclose(fp);
        }
        sprintf(buf, "Tpcalloc %d", iRc);
        ErrorMessage(NULL, ERR_TPALLOC_BAD, ERR_TYPE_TUXEDO, buf,
0, 0);
        tpfree((char *)stockLevelDataPtr);
        return;
    }

    *stockLevelDataPtr =
Term.pClientData[iTermId].trans.stocklevelData;

    ilen = sizeof(StockLevelData);
    olen = &ilen;

    if (dLog)
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "* ProcessStockLevel Thread %d iTermId %d
stockLevelDataPtr: %x \r\n",
GetCurrentThreadId(), iTermId,
&stockLevelDataPtr);
        fclose(fp);
    }

    if (tpcall("OPSTUXSERVER", (char *) stockLevelDataPtr, ilen, (char
**) &stockLevelDataPtr, (long *) olen, TPSIGRSTRT) == -1) {
        iRc = tperrno;
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "* ProcessStockLevel Thread %d
iTermId %d tpcall tperrno %d \r\n",
GetCurrentThreadId(), iTermId, iRc);
            fclose(fp);
        }
        sprintf(buf, "StockLevel tpcall %d", iRc);
        ErrorMessage(pECB, ERR_TPCALL_BAD, ERR_TYPE_TUXEDO, buf,
iTermId, iSyncId);
        tpfree((char *)stockLevelDataPtr);
        return;
    }

    if (dLog)
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");

```

```

        fprintf(fp, "*** ProcessStockLevel Thread %d iTermId %d
stockLevelDataPtr: %x \r\n",
                GetCurrentThreadId(), iTermId,
&stockLevelDataPtr);
        fclose(fp);
    }

    Term.pClientData[iTermId].trans.stocklevelData =
*stockLevelDataPtr;

    iRc = stockLevelDataPtr->retval;

    tpfree((char *)stockLevelDataPtr);

#else

*Term.pClientData[iTermId].stockLevelDataPtr =
Term.pClientData[iTermId].trans.stocklevelData;

    ilen = sizeof(StockLevelData);
    olen = &ilen;

    if ( dLog )
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "*** ProcessStockLevel Thread %d iTermId %d
StockLevelDataPtr: %x \r\n",
                GetCurrentThreadId(), iTermId,
&Term.pClientData[iTermId].stockLevelDataPtr);
        fclose(fp);
    }

    if (tpccall("STOCKLEVEL", (char
*)Term.pClientData[iTermId].stockLevelDataPtr, ilen,
(char **) &Term.pClientData[iTermId].stockLevelDataPtr,
(long *) olen, TPSIGRSTRT) == -1) {
        iRc = tperrno;
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "*** ProcessStockLevel Thread %d
iTermId %d tpccall tperrno %d \r\n",
                    GetCurrentThreadId(), iTermId, iRc);
            fclose(fp);

            sprintf(buf, "StockLevel tpccall %d", iRc);
            ErrorMessage(pECB, ERR_TPCALL_BAD, ERR_TYPE_TUXEDO, buf,
iTermId, iSyncId);
            tpfree((char *)StockLevelDataPtr);
        }
        return;
    }

    if ( dLog )
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "*** ProcessStockLevel Thread %d iTermId %d
StockLevelDataPtr: %x \r\n",
                GetCurrentThreadId(), iTermId,
&Term.pClientData[iTermId].stockLevelDataPtr);

```

```

        fclose(fp);
    }

    Term.pClientData[iTermId].trans.stocklevelData =
*Term.pClientData[iTermId].stockLevelDataPtr;

    iRc = Term.pClientData[iTermId].stockLevelDataPtr->retval;

    #endif
#endif

    if ( iRc < 0) {
        ErrorMessage(pECB, ERR_STOCKLEVEL_NOT_PROCESSED, ERR_TYPE_WEBDLL,
NULL, iTermId, iSyncId);
    }
    else {
        WriteZString(pECB, MakeStockLevelForm(iTermId, iSyncId, FALSE) );
    }

    return;
}

/* FUNCTION: int GetNewOrderData(LPSTR lpszQueryString, NewOrderData
*pNewOrderData)
*
* PURPOSE: This function extracts and validates the new order form
data from
*          an http command string.
*
* ARGUMENTS: LPSTR      lpszQueryString client browser http command
string
*              NewOrderData *pNewOrderData pointer to new order data
structure
*
* RETURNS:      int          error code indicating reason for
failure
*              ERR_SUCCESS   new order input data successfully
parsed
*
* COMMENTS:     None
*/
static int GetNewOrderData(LPSTR lpszQueryString, NewOrderData
*pNewOrderData)
{
    char    szTmp[26];
    int     i,j,skipped;
    short   items;

    //    BOOL    bCheck;
    //    char *   s;

    //    s=lpszQueryString;

    if ( !GetKeyValue(lpszQueryString, "DID*", szTmp,
sizeof(szTmp)) )
        return ERR_NEWORDER_FORM_MISSING_DID;

    if ( !IsNumeric(szTmp) )
        return ERR_NEWORDER_DISTRICT_INVALID;

    pNewOrderData->newin.d_id = atoi(szTmp);

```

```

        if ( !GetKeyValue(lpszQueryString, "CID*", szTmp,
sizeof(szTmp)) )
            return ERR_NEWORDER_CUSTOMER_KEY;

        if ( !IsNumeric(szTmp) )
            return ERR_NEWORDER_CUSTOMER_INVALID;

        pNewOrderData->newin.c_id = atoi(szTmp);
        /*pNewOrderData->o_all_local=1;*/

//        bCheck = FALSE;
        skipped = 0;
        for(i=0, items=0; i<15; i++)
        {
            if ( !GetKeyValue(lpszQueryString, szIID[i], szTmp,
sizeof(szTmp)) )
                return ERR_NEWORDER_MISSING_IID_KEY;

            if ( szTmp[0] )
            {
                //if blank lines between item ids
                if ( bCheck )
                    return ERR_NEWORDER_ITEM_BLANK_LINES;
                if ( !IsNumeric(szTmp) )
                    return ERR_NEWORDER_ITEMID_INVALID;
                pNewOrderData->newin.ol_i_id[i-skipped] =
atoi(szTmp);

                if ( !GetKeyValue(lpszQueryString, szSP[i],
szTmp, sizeof(szTmp)) )
                    return ERR_NEWORDER_MISSING_SUPPW_KEY;
                if ( !IsNumeric(szTmp) )
                    return ERR_NEWORDER_SUPPW_INVALID;
                pNewOrderData->newin.ol_supply_w_id[i-skipped] =
(short)atoi(szTmp);
                /*
                if ( pNewOrderData->o_all_local &&
pNewOrderData->o_ol[i].ol_supply_w_id != pNewOrderData->w_id )
                    pNewOrderData->o_all_local = 0; */

                if ( !GetKeyValue(lpszQueryString, szQty[i],
szTmp, sizeof(szTmp)) )
                    return ERR_NEWORDER_MISSING_QTY_KEY;

                if ( !IsNumeric(szTmp) )
                    return ERR_NEWORDER_QTY_INVALID;

                pNewOrderData->newin.ol_quantity[i-skipped] =
pNewOrderData->old_quantity[i-skipped] = atoi(szTmp);
                items++;

                if ( pNewOrderData->newin.ol_i_id[i-skipped] >=
1000000 || pNewOrderData->newin.ol_i_id[i-skipped] < 1 )
                    return ERR_NEWORDER_ITEMID_RANGE;
                if ( pNewOrderData->newin.ol_quantity[i-skipped]
>= 100 || pNewOrderData->newin.ol_quantity[i-skipped] < 1 )
                    return ERR_NEWORDER_QTY_RANGE;

                for ( j = i-1-skipped; j >= 0; j-- )
                    if ( pNewOrderData->newin.ol_i_id[i-skipped] ==
pNewOrderData->newin.ol_i_id[j]){

```

```

                pNewOrderData->newin.ol_quantity[i-
skipped] += pNewOrderData->newin.ol_quantity[j];
                j = -1;
            }
            skipped = 0;
        }
        else
        {
            if ( !GetKeyValue(lpszQueryString, szSP[i],
szTmp, sizeof(szTmp)) )
                return ERR_NEWORDER_MISSING_QTY_KEY;

            if ( szTmp[0] )
                return ERR_NEWORDER_SUPPW_WITHOUT_ITEMID;

            if ( !GetKeyValue(lpszQueryString, szQty[i],
szTmp, sizeof(szTmp)) )
                return ERR_NEWORDER_MISSING_QTY_KEY;

            if ( szTmp[0] )
                return ERR_NEWORDER_QTY_WITHOUT_ITEMID;

//            bCheck = TRUE;
            skipped ++;
        }
    }
    if ( items == 0 )
        return ERR_NEWORDER_NOITEMS_ENTERED;

    // pNewOrderData->newout.o_ol_cnt = items;

    return ERR_SUCCESS;
}

/* FUNCTION: int GetPaymentData(LPSTR lpszQueryString, PaymentData
*pPaymentData)
*
*
* PURPOSE: This function extracts and validates the payment form data
from an
*
*         http command string.
*
* ARGUMENTS: LPSTR     lpszQueryString client browser http command
string
*               PaymentData     *pPaymentData ptr to payment data
structure
*
* RETURNS:   int           error code indicating reason for
failure
*               ERR_SUCCESS      all input data successfully parsed
*
* COMMENTS:   None
*
*/
static int GetPaymentData(LPSTR lpszQueryString, PaymentData
*pPaymentData)
{
    char    szTmp[26];
    char    *ptr;
    char    *s;
    double  fTmp;

//    s=lpszQueryString;

```

```

if(!GetKeyValue(lpszQueryString, "DID*", szTmp, sizeof(szTmp)))
    return ERR_PAYMENT_MISSING_DID_KEY;

if ( !IsNumeric(szTmp) )
    return ERR_PAYMENT_DISTRICT_INVALID;
pPaymentData->payin.d_id = atoi(szTmp);

if(!GetKeyValue(lpszQueryString, "CID*", szTmp, sizeof(szTmp)))
    return ERR_PAYMENT_MISSING_CID_KEY;

if ( szTmp[0] && !IsNumeric(szTmp) )
    return ERR_PAYMENT_CUSTOMER_INVALID;

pPaymentData->payin.c_id = atoi(szTmp);
if (bLog){
    FILE *fp1;
    fp1 = fopen(szTpccLogPath, "ab");
    fprintf(fp1, "cid = %s\n",szTmp);
    fclose(fp1);
}

if ( szTmp[0] == 0 )
{
    if(!GetKeyValue(lpszQueryString, "CLT*", szTmp,
sizeof(szTmp)))
        return ERR_PAYMENT_MISSING_CLT;
        _strupr( szTmp );

    strcpy(pPaymentData->payin.c_last, szTmp);
    if ( strlen(pPaymentData->payin.c_last) > 16 )
        return ERR_PAYMENT_LAST_NAME_TO_LONG;
    pPaymentData->payin.bylastname=1;
}
else
{
    if(!GetKeyValue(lpszQueryString, "CLT*", szTmp,
sizeof(szTmp)))
        return ERR_PAYMENT_MISSING_CLT_KEY;
    if ( szTmp[0] )
        return ERR_PAYMENT_CID_AND_CLT;
}

// s=lpszQueryString;

if(!GetKeyValue(lpszQueryString, "CWI*", szTmp, sizeof(szTmp)))
    return ERR_PAYMENT_MISSING_CWI_KEY;

if ( !IsNumeric(szTmp) )
    return ERR_PAYMENT_CWI_INVALID;

pPaymentData->payin.c_w_id = atoi(szTmp);
if (bLog){
    FILE *fp1;
    fp1 = fopen(szTpccLogPath, "ab");
    fprintf(fp1, "cid = %s\n",szTmp);
    fclose(fp1);
}

if(!GetKeyValue(lpszQueryString, "CDI*", szTmp, sizeof(szTmp)))
    return ERR_PAYMENT_MISSING_CDI_KEY;
if ( !IsNumeric(szTmp) )
    return ERR_PAYMENT_CDI_INVALID;
pPaymentData->payin.c_d_id = atoi(szTmp);

```

```

if(!GetKeyValue(lpszQueryString, "HAM*", szTmp, sizeof(szTmp)))
    return ERR_PAYMENT_MISSING_HAM_KEY;

ptr = szTmp;

while( *ptr )
{
    if ( *ptr == '.' )
    {
        ptr++;
        if ( !*ptr )
            break;
        if ( *ptr < '0' || *ptr > '9' )
            return ERR_PAYMENT_HAM_INVALID;
        ptr++;
        if ( !*ptr )
            break;
        if ( *ptr < '0' || *ptr > '9' )
            return ERR_PAYMENT_HAM_INVALID;
        if ( !*ptr )
            return ERR_PAYMENT_HAM_INVALID;
    }
    else if ( *ptr < '0' || *ptr > '9' )
        return ERR_PAYMENT_HAM_INVALID;
    ptr++;

    fTmp = atof(szTmp);
    pPaymentData->payin.h_amount = (int) (fTmp * 100);
    if ( pPaymentData->payin.h_amount >= 1000000 || pPaymentData-
>payin.h_amount < 0 )
        return ERR_PAYMENT_HAM_RANGE;

    if (bLog){
        FILE *fp1;
        fp1 = fopen(szTpccLogPath, "ab");
        fprintf(fp1, "cid = %s\n",szTmp);
        fclose(fp1);
    }
    if (bLog){
        FILE *fp1;
        fp1 = fopen(szTpccLogPath, "ab");
        fprintf(fp1, "ouma
= %s\n",lpszQueryString);
        fclose(fp1);
    }

    return ERR_SUCCESS;
}

/* FUNCTION: int GetOrderStatusData(LPSTR lpszQueryString,
*                                     OrderStatusData *pOrderStatusData)
*
* PURPOSE: This function extracts and validates the payment form data
from an
*           http command string.
*
* ARGUMENTS:LPSTR           lpszQueryString      client browser http
command
*                                     string
*           OrderStatusData *pOrderStatusData  pointer to order
status
*
*                                     data structure

```

```

*
* RETURNS:      int          error code indicating reason for
failure
*              ERR_SUCCESS  successfully parsed all required input
data
*
* COMMENTS:     None
*
*/
static int GetOrderStatusData(LPSTR lpszQueryString, OrderStatusData
*pOrderStatusData)
{
    char    szTmp[26];
//    char    *s;
//    s=lpszQueryString;

    if(!GetKeyValue(lpszQueryString, "DID*", szTmp, sizeof(szTmp)))
        return ERR_ORDERSTATUS_MISSING_DID_KEY;
    if ( !IsNumeric(szTmp) )
        return ERR_ORDERSTATUS_DID_INVALID;
    pOrderStatusData->ordin.d_id = atoi(szTmp);

    if(!GetKeyValue(lpszQueryString, "CID*", szTmp, sizeof(szTmp)))
        return ERR_ORDERSTATUS_MISSING_CID_KEY;

    if ( szTmp[0] == 0 )
    {
        pOrderStatusData->ordin.c_id = 0;
        if(!GetKeyValue(lpszQueryString, "CLT*", szTmp,
sizeof(szTmp)))
            return ERR_ORDERSTATUS_MISSING_CLT_KEY;
        _strupr( szTmp );
        strcpy(pOrderStatusData->ordin.c_last, szTmp);
        if ( strlen(pOrderStatusData->ordin.c_last) > 16 )
            return ERR_ORDERSTATUS_CLT_RANGE;
        pOrderStatusData->ordin.bylastname = 1;
    }
    else
    {
        if ( !IsNumeric(szTmp) )
            return ERR_ORDERSTATUS_CID_INVALID;
        pOrderStatusData->ordin.c_id = atoi(szTmp);
        if(!GetKeyValue(lpszQueryString, "CLT*", szTmp,
sizeof(szTmp)))
            return ERR_ORDERSTATUS_MISSING_CLT_KEY;
        if ( szTmp[0] )
            return ERR_ORDERSTATUS_CID_AND_CLT;
    }

    return ERR_SUCCESS;
}

/* FUNCTION: BOOL ReadRegistrySettings(void)
*
* PURPOSE: This function reads the NT registry for startup parameters.
There
*          parameters are under the TPCC key.
*
* ARGUMENTS:  None
*
* RETURNS:    None
*

```

```

* COMMENTS:     This function also sets up required operation variables
to
*              their default value so if registry is not setup the
default
*              values will be used.
*
*/
static BOOL ReadRegistrySettings(void)
{
    HKEY    hKey;
    DWORD   size;
    DWORD   type;
    char    szTmp[256];

    bLog     = FALSE;
    dLog     = FALSE;
    bGeneric = FALSE;
    iMaxWarehouses = 625;
    iThreads  = 5;
    iDelayMs  = 100;
    iDeadlockRetry = (short)3;
    strcpy(szTpccLogPath, "tpcclog.");

    if ( RegOpenKeyEx(HKEY_LOCAL_MACHINE, "SOFTWARE\\ORACLE\\tpcc", 0,
KEY_READ, &hKey) != ERROR_SUCCESS )
        return TRUE;
    size = sizeof(szTmp);

    if ( RegQueryValueEx(hKey, "PATH", 0, &type, szTmp, &size) ==
ERROR_SUCCESS )
    {
        strcpy(szTpccLogPath, szTmp);
        strcat(szTpccLogPath, "tpcclog.");
        strcpy(szErrorLogPath, szTmp);
        strcat(szErrorLogPath, "tpccerr.");
    }

    size = sizeof(szTmp);
    if ( RegQueryValueEx(hKey, "LOG", 0, &type, szTmp, &size) ==
ERROR_SUCCESS )
    {
        if ( !strcmp(szTmp, "ON") )
            bLog = TRUE;
    }

    size = sizeof(szTmp);
    if ( RegQueryValueEx(hKey, "DEBUG", 0, &type, szTmp, &size) ==
ERROR_SUCCESS )
    {
        if ( !strcmp(szTmp, "ON") )
            dLog = TRUE;
    }

    size = sizeof(szTmp);
    if ( RegQueryValueEx(hKey, "MaximumWarehouses", 0, &type, szTmp,
&size) == ERROR_SUCCESS )
    {
        iMaxWarehouses = atoi(szTmp);
        if ( iMaxWarehouses == 0 )
            iMaxWarehouses = 500;
    }

    size = sizeof(szTmp);

```

```

    if ( RegQueryValueEx(hKey, "NumberOfDeliveryThreads", 0, &type,
szTmp, &size) == ERROR_SUCCESS )
        iThreads = atoi(szTmp);
    if ( !iThreads )
        iThreads = 5;

    size = sizeof(szTmp);
    if ( RegQueryValueEx(hKey, "BackoffDelay", 0, &type, szTmp,
&size) == ERROR_SUCCESS )
        iDelayMs = atoi(szTmp);
    if ( !iDelayMs )
        iDelayMs = 100;

    size = sizeof(szTmp);
    if ( RegQueryValueEx(hKey, "DeadlockRetry", 0, &type, szTmp,
&size) == ERROR_SUCCESS )
        iDeadlockRetry = (short)atoi(szTmp);
    if ( !iDeadlockRetry )
        iDeadlockRetry = (short)3;

    size = sizeof(szTmp);
    if ( RegQueryValueEx(hKey, "MaxConnections", 0, &type, szTmp,
&size) == ERROR_SUCCESS )
        iMaxConnections = (short)atoi(szTmp);
    if ( !iMaxConnections )
        iMaxConnections = (short)25;

    size = sizeof(szTmp);
    if ( RegQueryValueEx(hKey, "GenSrv", 0, &type, szTmp, &size) ==
ERROR_SUCCESS )
    {
        if ( !strcmp(szTmp, "ON") )
            bGeneric = TRUE;
    }

    RegCloseKey(hKey);
    return FALSE;
}

/* 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 all numeric characters i.e.
'0' - '9'
 *
 * COMMENTS:   None
 *
 */

static BOOL IsNumeric( char *ptr)
{
    if ( *ptr == 0 )
        return FALSE;
    while( *ptr && isdigit(*ptr) )
        ptr++;
    return ( !*ptr );
}

```

```

}

/* FUNCTION: void FormatHTMLString(char *szBuff, int iLen, char *szStr)
 *
 * PURPOSE: This function Handles translation of HTML specific
character field
 *          data when an HTML output form is generated.
 *
 * ARGUMENTS:  char    *szBuff    Returned string information
 *              char    *szStr    input string to be formatted.
 *              int     iLen      Length of returned string
 *
 * RETURNS:    none
 *
 * COMMENTS:   The length parameter is the absolute length of the
returned
 *              string in HTML characters. For example the input string
> would
 *              be returned as &gt; which would be counted as 1
character.If
 *              the number of input characters is less than the iLen
parameter
 *              spaces are appended to the end of the string to ensure
that at
 *              least iLen characters are returned in the szBuff
parameter.
 *
 */

static void FormatHTMLString(char *szBuff, char *szStr, int iLen)
{
    while( iLen && *szStr )
    {
        switch( *szStr )
        {
            case '>':
                *szBuff++ = '&';
                *szBuff++ = 'g';
                *szBuff++ = 't';
                *szBuff++ = ';';
                szStr++;
                break;
            case '<':
                *szBuff++ = '&';
                *szBuff++ = 'l';
                *szBuff++ = 't';
                *szBuff++ = ';';
                szStr++;
                break;
            case '&':
                *szBuff++ = '&';
                *szBuff++ = 'a';
                *szBuff++ = 'm';
                *szBuff++ = 'p';
                *szBuff++ = ';';
                szStr++;
                break;
            case '\\':
                *szBuff++ = '&';
                *szBuff++ = 'q';
                *szBuff++ = 'u';
                *szBuff++ = 'o';
                *szBuff++ = 't';
                *szBuff++ = ';';
        }
    }
}

```

```

        szStr++;
        break;
    default:
        *szBuff++ = *szStr++;
        break;
    }
    iLen--;
}
while( iLen-- )
    *szBuff++ = ' ';
*szBuff = 0;
return;
}

static int ThrTpInit(EXTENSION_CONTROL_BLOCK *pECB)
{
    // use CS for tptpalloc ?
    static int num_tpinit=0;
    static int x=1;
    static int once=0;
    static CRITICAL_SECTION TpCriticalSection;
    int iRc, TpRc, lasterr;
    char buf[40];
    int retry = 0;
    BOOL Success = FALSE;

    if(!TlsGetValue(TLSIsTpInitKey))
    {
        if (!once)
        {
            InitializeCriticalSection(&TpCriticalSection);
            once=1;
        }

        if ( dLog )
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "* In ThrTpInit Thread %d * \r\n",
GetCurrentThreadId());
            fclose(fp);
        }

        while ( retry < TP_MAX_RETRIES )
        {
            EnterCriticalSection(&TpCriticalSection);

            if(tpinf == NULL)
            {
                if ((tpinf = ( TPINIT *)tpalloc("TPINIT",
NULL, sizeof(TPINIT))) == NULL)
                {
                    TpRc = tperrno;
                    {
                        FILE *fp;

                        fp = fopen(szTpccLogPath, "ab");
                        fprintf(fp, ">>>> ThrTpInit %d :
tpalloc of tpinit failed: %d \r\n",
GetCurrentThreadId(),
TpRc);
                        fclose(fp);
                    }
                }
            }
        }
    }
}

```

```

        }
        tpinf = NULL;
        retry++;
        LeaveCriticalSection(&TpCriticalSection);
        Sleep(50); // Relinquish thread timeslice
        continue;
    }
    tpinf->flags |= TPMULTICONTEXTS;
}

// Do the TPINIT

itoa(++num_tpinit, tpinf->cltname, 10);
iRc = tpinit(tpinf);

// check tpinit() ?
if (iRc < 0 )
{
    lasterr = GetLastError();
    retry++ ;

    if ( dLog )
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "* ThrTpInit
Thread %d Retry : count %d * \r\n",
GetCurrentThreadId(),
retry);
        fclose(fp);
    }
    else
    {
        Success = TRUE;
        tpfree( ( char * ) tpinf);
        tpinf=NULL;
        break;
    }
    --num_tpinit;
    LeaveCriticalSection(&TpCriticalSection);
    Sleep(50); // Relinquish thread
timeslice

} // retry the tpinit if it failed the first time
LeaveCriticalSection(&TpCriticalSection);

if ( Success == FALSE )
{
    TpRc = tperrno;
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, ">>>> ThrTpInit Thread %d
Failed tperrno= %d Last Err= %d \r\n",
GetCurrentThreadId(), TpRc,
lasterr);
        fclose(fp);
    }
}
}

```

```

        sprintf(buf,"Thread %d: In ThrTpInit
iRc=%d Tperrno=%d",
                GetCurrentThreadId(),iRc, TpRc);
        ErrorMessage(pECB, ERR_TPINIT_BAD,
ERR_TYPE_TUXEDO, buf, 0, 0);
        return -1;
    }
    if ( Success == TRUE )
    {
        if ( dLog )
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "* ThrTpInit Thread %d
Success retry count %d * \r\n",
                GetCurrentThreadId(), retry);
            fclose(fp);
        }

        if ( ( iRc=TlsSetValue(TLSIsTpInitedKey,&x)) ==
0)
        {
            {
                FILE *fp;

                fp = fopen(szTpccLogPath, "ab");
                fprintf(fp, ">>>> ThrTpInit %d :
TlsSetValue Failed iRc: %d \r\n",
                    GetCurrentThreadId(), iRc);
                fclose(fp);
            }
            sprintf(buf,"TlsSetValue %d",iRc);
            ErrorMessage(pECB, ERR_TPINIT_BAD,
ERR_TYPE_TUXEDO, buf, 0, 0);
            return -1;
        }
    }
    else
    {
        if ( dLog )
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "* ThrTpInit Thread %d already
tpinited * \r\n", GetCurrentThreadId());
            fclose(fp);
        }
    }

    return 0;
}

#ifdef TOPEND
void space_fill(s1, s2, i)

char *s1, *s2;
int i;

```

```

{
    int ii;

    strncpy(s1,s2,i);

    for(ii=0; ii<i; ii++)
    {
        if (s1[ii] == '\0') s1[ii] = 0x20;
    }
    return;
}

/* This is the function for the new thread that is added to the process.
*/
/* The error handling needs to be improved. Almost every problem is
fatal. */
static void TereceiveThread(void *ptr)
{
    tp_dif_structs_t *dif_struct;
    long buffer_length;
    char *message_buffer;
    int rc;
    FILE *fp1;

    if ((rc = (int)tp_ChangeToGroup ((LPSTR)getenv("TP_SYSTEM"),
LOGON32_LOGON_INTERACTIVE,
LOGON32_PROVIDER_DEFAULT)) != 0)
    {
        if (bLog)
        {
            fp1 = fopen(szTpccLogPath, "ab");
            fprintf(fp1, "Received bad status from tp_ChangeToGroup,
status = %d ***\n", rc);
            fclose(fp1);
        }
        MT_LOG_ERROR;
        return;
    }

    dif_struct = tp_csi_alloc (TP_DIF_DIAL_INFO |
                                TP_DIF_SERVICE_NAME |
                                TP_DIF_OUTPUT_FORMAT |
                                TP_DIF_LOCATION);

    if (dif_struct == NULL)
    {
        MT_LOG_ERROR;
        return;
    }

    if ((message_buffer = (char *) malloc (TP_MAX_BUF_LEN)) == NULL)
    {
        MT_LOG_ERROR;
        return;
    }

    /* receive loop. Should there be a way to shut this down? */
    for (;;)
    {
        if (tp_WaitForSingleObject (NULL, INFINITE) == WAIT_TIMEOUT)
        {
            MT_LOG_ERROR;
            return;
        }
    }
}

```

```

}
dif_struct->info->tp_user_dialogue_id = TP_ANY_DIALOGUE;
dif_struct->info->tp_flags = TP_NOFLAGS;
buffer_length = TP_MAX_BUF_LEN;

rc = tp_client_receive (dif_struct->info, TP_NOBLOCK,
                        dif_struct->output_format,
                        dif_struct->service,
                        dif_struct->location,
                        &buffer_length,
                        message_buffer);

switch (rc)
{
  case TP_TIMEOUT:
  {
    break;
  }
  case TP_ADMIN:
  {
    /* This assumes that admin messages aren't used in the
application. */
    /* Receive the message and throw it away. */
    long flags, buf_len;
    char buffer[1];

    flags = TP_TRUNCATE;
    buf_len = 1;
    tp_system_admin (&flags, &buf_len, buffer);
    break;
  }
  case TP DISSOLVED:
  case TP_OK:
  case TP_RESET:
  case TP_SERVICE:
  case TP_SIGNON_INHIBITED:
  case TP_USER:
  {
    DialogTableEntry_t *DTE;
    void *ptr;

    DTE = FindOnList (dif_struct->info->tp_user_dialogue_id,
&DTActiveList);
    if (DTE == NULL)
    {
      /* We received in a dialog without outstanding work????
*/
      MT_LOG_ERROR;
      return;
    }

    /* trade data with DTE. */
    ptr = DTE->dif_struct->info;
    DTE->dif_struct->info = dif_struct->info;
    dif_struct->info = ptr;

    ptr = DTE->dif_struct->output_format;
    DTE->dif_struct->output_format = dif_struct-
>output_format;
    dif_struct->output_format = ptr;

    ptr = DTE->dif_struct->service;
    DTE->dif_struct->service = dif_struct->service;
    dif_struct->service = ptr;

```

```

ptr = DTE->dif_struct->location;
DTE->dif_struct->location = dif_struct->location;
dif_struct->location = ptr;

DTE->buffer_length = buffer_length;

if (DTE->message_buffer == NULL)
{
  if ((DTE->message_buffer = malloc (buffer_length)) ==
NULL)
  {
    MT_LOG_ERROR;
    return;
  }
}
else
{
  /* NONPORTABLE */
  if (_msize (DTE->message_buffer) < (size_t)
buffer_length)
  {
    if ((DTE->message_buffer = realloc (DTE-
>message_buffer,
buffer_length)) == NULL)
    {
      MT_LOG_ERROR;
      return;
    }
  }
  memcpy (DTE->message_buffer, message_buffer,
buffer_length);
  DTE->rc = rc;
  /* Signal that the data is ready. */
  SetEvent (DTE->hSync);
  break;
}
case TP_SHUTDOWN:
{
  DialogTableEntry_t *DTE;
  /* tell everyone */
  if (WaitForSingleObject (DTActiveList.hSync, INFINITE)
== WAIT_TIMEOUT)
  {
    MT_LOG_ERROR;
    return;
  }
  DTE = DTActiveList.next;
  while (DTE != NULL)
  {
    DTE->rc = TP_SHUTDOWN;
    SetEvent (DTE->hSync);
    DTE = DTE->next;
  }
  if (!ReleaseMutex (DTActiveList.hSync))
  {
    MT_LOG_ERROR;
    return;
  }
  break;
}
default:
{
  if (bLog)

```

```

        {
            fp1 = fopen(szTpccLogPath, "ab");
            fprintf(fp1, "TEReceiveThread received error status
= %d\n", rc);
            fclose(fp1);
        }
        MT_LOG_ERROR;
        return;
    }
}

int tp_mt_initialize(tp_application_info_t *application_info,
                    tp_funct_struct_t   function_array[],
                    long                  number_functions)
{
    static HANDLE hReceiveThread = NULL;
    int rc;

    rc = tp_initialize(application_info, function_array,
number_functions);

    if (rc == TP_OK)
    {
        /* Make sure that the global data is initialized. */
        if (DTFreeList.hSync == NULL)
        {
            if ((DTFreeList.hSync = CreateMutex(NULL, FALSE, NULL)) ==
NULL)
            {
                MT_LOG_ERROR;
                tp_terminate();
                return TP_DIFERR;
            }
        }
        if (DTActiveList.hSync == NULL)
        {
            if ((DTActiveList.hSync = CreateMutex(NULL, FALSE, NULL)) ==
NULL)
            {
                MT_LOG_ERROR;
                tp_terminate();
                return TP_DIFERR;
            }
        }
        /* Make sure that the receive thread is running */
        if (hReceiveThread == NULL)
        {
            if ((hReceiveThread = (HANDLE) _beginthread(TEReceiveThread,
0, NULL)) == (HANDLE) -1 )
            {
                MT_LOG_ERROR;
                tp_terminate();
                return TP_DIFERR;
            }
            if (!SetThreadPriority (hReceiveThread,
RECEIVE_THREAD_PRIORITY))
            {
                MT_LOG_ERROR;
            }
        }
    }
}

```

```

    }
    return rc;
}

int tp_mt_client_signon(tp_dialogue_info_t *info,
                       tp_dialogue_user_t *client,
                       long                inactivity_time,
                       tp_service_name_t  *service,
                       tp_input_format_t  *input_format,
                       long                message_length,
                       char                *message_text)
{
    DialogTableEntry_t *DTE = GetDTE();
    int rc;

    if (DTE == NULL)
    {
        MT_LOG_ERROR;
        return TP_DIFERR;
    }
    if (!ResetEvent (DTE->hSync))
    {
        MT_LOG_ERROR;
        return TP_DIFERR;
    }

    /* This is to handle TP UNIQUE DIALOGUE */
    DTE->usr_dlg_ptr = &(info->tp_user_dialogue_id);
    AddToList(DTE, &DTActiveList);

    rc = tp_client_signon(info, client, inactivity_time, service,
input_format, message_length, message_text);

    switch (rc)
    {
        case TP DISSOLVING:
        case TP_OK:
        {
            DTE->usr_dlg_storage = info->tp_user_dialogue_id;
            DTE->usr_dlg_ptr = &(DTE->usr_dlg_storage);
            break;
        }
        default:
        {
            /* there is nothing to receive */
            RemoveFromList (DTE, &DTActiveList);
            AddToList (DTE, &DTFreeList);
        }
    }
    return rc;
}

int tp_mt_client_send(tp_dialogue_info_t *info,
                      tp_service_name_t  *service,
                      tp_input_format_t  *input_format,
                      long                message_length,
                      char                *message_text)
{
    DialogTableEntry_t *DTE = GetDTE();
    int rc;

    if (DTE == NULL)

```

```

    {
        MT_LOG_ERROR;
        return TP_DIFERR;
    }
    if (!ResetEvent (DTE->hSync))
    {
        MT_LOG_ERROR;
        return TP_DIFERR;
    }
    DTE->usr_dlg_storage = info->tp_user_dialogue_id;
    DTE->usr_dlg_ptr = &(DTE->usr_dlg_storage);
    AddToList(DTE, &DTActiveList);

    rc = tp_client_send(info, service, input_format, message_length,
message_text);
    switch (rc)
    {
        case TP DISSOLVING:
        case TP_OK:
        case TP_RESET:
        {
            break;
        }
        default:
        {
            /* there is nothing to receive */
            RemoveFromList (DTE, &DTActiveList);
            AddToList (DTE, &DTFreeList);
        }
    }
    return rc;
}

```

```

int tp_mt_client_receive(tp_dialogue_info_t *info,
                        long wait_time,
                        tp_output_format_t *output_format,
                        tp_service_name_t *service,
                        tp_location_t *location,
                        long *buffer_length,
                        char *message_buffer)
{
    DialogTableEntry_t *DTE = FindOnList(info->tp_user_dialogue_id,
&DTActiveList);
    int rc;
    FILE *fp1;

    if (DTE == NULL)
    {
        MT_LOG_ERROR;
        return TP_DIFERR;
    }
    wait_time = TP_BLOCK;

    /* Wait for the receive to actually happen */
    if (WaitForSingleObject (DTE->hSync, INFINITE) == WAIT_TIMEOUT)
    {
        if (bLog)
        {
            fp1 = fopen (szTpccLogPath, "ab");
            fprintf (fp1, "tp_mt_client_receive got a timeout error\n");
            fclose (fp1);
        }
    }
}

```

```

    return TP_TIMEOUT;
}

if (DTE->buffer_length > *buffer_length)
{
    *buffer_length = DTE->buffer_length;
    return TP_BUFFSIZE;
}

RemoveFromList (DTE, &DTActiveList);

memcpy (info, DTE->dif_struct->info, sizeof
(tp_dialogue_info_t));
memcpy (output_format, DTE->dif_struct->output_format, sizeof
(tp_output_format_t));
memcpy (service, DTE->dif_struct->service, sizeof
(tp_service_name_t));
memcpy (location, DTE->dif_struct->location, sizeof
(tp_location_t));
*buffer_length = DTE->buffer_length;
memcpy (message_buffer, DTE->message_buffer, *buffer_length);
rc = DTE->rc;

AddToList (DTE, &DTFreeList);
return rc;
}

void AddToList (DialogTableEntry_t *DTE, DialogTableEntry_t *head)
{
    if (WaitForSingleObject (head->hSync, INFINITE) == WAIT_TIMEOUT)
    {
        MT_LOG_ERROR;
        return;
    }
    DTE->next = head->next;
    head->next = DTE;
    if (!ReleaseMutex (head->hSync))
    {
        MT_LOG_ERROR;
    }
    return;
}

void RemoveFromList (DialogTableEntry_t *DTE, DialogTableEntry_t *head)
{
    DialogTableEntry_t *ptr;

    if (WaitForSingleObject (head->hSync, INFINITE) == WAIT_TIMEOUT)
    {
        MT_LOG_ERROR;
        return;
    }
    ptr = head;

    while ((ptr != NULL) && (ptr->next != DTE))
    {
        ptr = ptr->next;
    }
    if (ptr == NULL)
    {
        MT_LOG_ERROR;
        ReleaseMutex (head->hSync);
    }
}

```

```

    return;
}
ptr->next = DTE->next;

if (!ReleaseMutex (head->hSync))
{
    MT_LOG_ERROR;
}
return;
}

```

```

DialogTableEntry_t *FindOnList (long tp_user_dialogue_id,
DialogTableEntry_t *head)
{
    DialogTableEntry_t *ptr;

    if (WaitForSingleObject (head->hSync, INFINITE) == WAIT_TIMEOUT)
    {
        MT_LOG_ERROR;
        return NULL;
    }
    ptr = head->next;

    while ((ptr != NULL) && (*(ptr->usr_dlg_ptr) !=
tp_user_dialogue_id))
    {
        ptr = ptr->next;
    }
    if (ptr == NULL)
    {
        MT_LOG_ERROR;
        ReleaseMutex (head->hSync);
        return NULL;
    }
    if (!ReleaseMutex (head->hSync))
    {
        MT_LOG_ERROR;
    }
    return ptr;
}

```

```

DialogTableEntry_t *GetDTE()
{
    DialogTableEntry_t *ptr = NULL;

    if (WaitForSingleObject (DTFreeList.hSync, INFINITE) ==
WAIT_TIMEOUT)
    {
        MT_LOG_ERROR;
        return NULL;
    }

    if (DTFreeList.next != NULL)
    {
        ptr = DTFreeList.next;
        DTFreeList.next = ptr->next;
    }

    if (!ReleaseMutex (DTFreeList.hSync))
    {
        MT_LOG_ERROR;
    }
}

```

```

if (ptr == NULL)
{
    if ((ptr = (DialogTableEntry_t *) malloc (sizeof
(DialogTableEntry_t))) ==NULL)
    {
        MT_LOG_ERROR;
        return NULL;
    }

    if ((ptr->dif_struct = tp_csi_alloc (TP_DIF_DIAL_INFO |
TP_DIF_SERVICE_NAME |
TP_DIF_OUTPUT_FORMAT |
TP_DIF_LOCATION)) == NULL)
    {
        MT_LOG_ERROR;
        free(ptr);
        return NULL;
    }
    ptr->buffer_length = 0;
    ptr->message_buffer = NULL;
    ptr->rc = 0;
    if ((ptr->hSync = CreateEvent (NULL, TRUE, FALSE, NULL)) ==
NULL)
    {
        MT_LOG_ERROR;
        tp_csi_free (ptr->dif_struct);
        free (ptr);
        return NULL;
    }
    ptr->next = NULL;
}
return ptr;
}
#endif

```

A2. Back-End Source Code, Scripts

GENE.MAK

```
# FILE: gene.mak
# DESCRIPTION: Builds TPCC tux server generic.exe

CPU = i386

# !IF "$(CFG)" != "TUXEDO" && "$(CFG)" != "TOPEND"
# !MESSAGE Invalid configuration "$(CFG)" specified.
# !MESSAGE You can specify a configuration when running NMAKE
# !MESSAGE by defining the macro CFG on the command line. For example:
# !MESSAGE
# !MESSAGE NMAKE /f "makefile.intel" CFG="TUXEDO"
# !MESSAGE
# !MESSAGE Possible choices for configuration are:
# !MESSAGE
# !MESSAGE TUXEDO - for tuxedo build
# !MESSAGE TOPEND - for topend build
# !MESSAGE
# !ERROR An invalid configuration is specified.
# !ENDIF

TPCCBENCH = ..\..\..\oracletpcnt
TPCCBIN = $(TPCCBENCH)\bin
ORACLE_HOME = C:\Oracle\Ora81
CLIB_DIR = C:\Program Files\DevStudio\VC\lib

SOURCE_DIR = $(TPCCBENCH)\source\server

OCI_SUB = oci
OCI_INCLUDE = $(ORACLE_HOME)\$(OCI_SUB)\include
OCI_LIB_HOME = $(ORACLE_HOME)\$(OCI_SUB)\lib\msvc
OCI_LIB = $(OCI_LIB_HOME)\Oci.lib

DPB_LIB_DIR = $(TPCCBENCH)\source\lib
#DPB_LIB = $(DPB_LIB_DIR)\dpbnt.lib

!IF "$(CFG)" == "TUXEDO"
COMMON_LIB_DIR = C:\inetpub\wwwroot
#COMMON_LIB = $(COMMON_LIB_DIR)\common_files.lib

!ELSE
COMMON_LIB_DIR = C:\inetpub\wwwroot
#rkambo
# COMMON_LIB =
COMMON_LIB = common_files.lib
#rkambo

!ENDIF

# Visual C++ libraries
SOCKETS_LIB = "$(CLIB_DIR)\wsock32.lib"
ADVAPI_LIB = "$(CLIB_DIR)\advapi32.lib"

CL = cl
CFLAGS = -c -W3 -D_X86_=1
COPT = -Ox -Oy -Gf
#CDEBUG = -Od -Z7 -Ge
```

```
!IF "$(CFG)" == "TUXEDO"
TM_CFLAGS= /DTUX /D_CONSOLE /DWIN32 /D_TMSTHEADS /IC:\TUXEDO\include
/I$(TPCCBENCH)\source\client
!ENDIF

!IF "$(CFG)" == "TOPEND"
TM_CFLAGS = /DTP
!ENDIF

ORA_DEFINES = /DORACLE /DOCI /DORA_NT
ORA_INCLUDES = /I$(OCI_INCLUDE) /I$(DPB_LIB_DIR)

CL_OPS = $(CFLAGS) $(COPT) $(CDEBUG) $(ORA_DEFINES)
$(TM_CFLAGS) $(ORA_INCLUDES)

LINK = link
#LFLAGS = /debug:full /debugtype:cv /pdb:none

LIBRARIAN = lib

#
# Private C compilation rule.
#
.c.obj:
    @$(CL) $(CL_OPS) /Fo$@ $*.c

#
# Destination for executables.
#
BIN_DIR = $(TPCCBIN)

#
# Rule for building an "exe" in $(BIN_DIR) and using object files
in current
# directory.
#
all: tpccsvr.obj tpccpl.obj plnew.obj plpay.obj pldel.obj plord.obj
plsto.obj \
    dpbtimef.obj generic.exe

plnew.obj: plnew.c tpcc.h tpccpl.h
    @$(CL) $(CL_OPS) /Fo$@ plnew.c

plpay.obj: plpay.c tpcc.h tpccpl.h
    @$(CL) $(CL_OPS) /Fo$@ plpay.c

plord.obj: plord.c tpcc.h tpccpl.h
    @$(CL) $(CL_OPS) /Fo$@ plord.c

pldel.obj: pldel.c tpcc.h tpccpl.h
    @$(CL) $(CL_OPS) /Fo$@ pldel.c

plsto.obj: plsto.c tpcc.h tpccpl.h
    @$(CL) $(CL_OPS) /Fo$@ plsto.c

tpccsvr.obj: tpccsvr.c tpcc.h tpcc_info.h
    @$(CL) $(CL_OPS) /Fo$@ tpccsvr.c

tpccpl.obj: tpccpl.c tpcc.h tpcc_info.h tpccpl.h
    @$(CL) $(CL_OPS) /Fo$@ tpccpl.c
```

```

dpbtimef.obj: $(DPB_LIB_DIR)\dpbtimef.c
                @$ (CL) /MD $(CL_OPS) /Fo$@
$(DPB_LIB_DIR)\dpbtimef.c

generic.exe: tpccsvr.obj tpccpl.obj plnew.obj plpay.obj pldel.obj
plord.obj plsto.obj dpbtimef.obj
                buildserver -v -o generic.exe -s OPSTUXSERVER -f tpccsvr.obj -f
tpccpl.obj -f plnew.obj \
                -f plpay.obj -f pldel.obj -f plord.obj -f plsto.obj -f
dpbtimef.obj -f $(OCI_LIB)

clean:
@for %f in ( *.obj *.lib $(BIN_DIR)\*.exe ) do if exist %f
del %f
@for %f in ( *.obj *.lib ) do if exist $(DPB_LIB_DIR)\%f del
$(DPB_LIB_DIR)\%f

```

TPCC.H

```

/*
 * $Header: tpcc.h 7030100.1 95/07/19 15:10:55 plai Generic<base>
 * Copyr (c) 1993 Oracle
 */
/*-----+
|          Copyright (c) 1995 Oracle Corp, Redwood Shores, CA
|          OPEN SYSTEMS PERFORMANCE GROUP
|          All Rights Reserved
|-----+
| FILENAME
|   tpcc.h
| DESCRIPTION
|   Include file for TPC-C benchmark programs.
|-----+*/

#ifndef TPCC_H
#define TPCC_H

#ifndef FALSE
# define FALSE 0
#endif

#ifndef TRUE
# define TRUE 1
#endif

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

#ifndef boolean
#define boolean int
#endif

#include <oratypes.h>
#include <oci.h>
#include <ocidfn.h>
/*
#ifdef __STDC__
#include "ociapr.h"
#else
#include "ocikpr.h"

```

```

#endif
*/

typedef struct cda_def csrdef;
typedef struct cda_def ldadef;

/* TPC-C transaction functions */

extern int TPCinit ();
extern int TPCnew ();
extern int TPCpay ();
extern int TPCord ();
extern int TPCdel ();
extern int TPCsto ();
extern void TPCexit ();
extern int TPCdumpinit ();
extern void TPCdumpnew ();
extern void TPCdumppay ();
extern void TPCdumpord ();
extern void TPCdumpdel ();
extern void TPCdumpsto ();
extern void TPCdumpexit ();
#ifndef TUX
extern void userlog ();
#endif

/* Error codes */

#define RECOVERR -10
#define IRRECERR -20
#define NOERR 111
#define DEL_ERROR -666
#define DEL_DATE_LEN 7
#define NDISTS 10
#define NITEMS 15
#define SQL_BUF_SIZE 8192

#define FULLDATE "dd-mon-yy.hh:mi:ss"
#define SHORTDATE "dd-mm-yyyy"

#define DELRT 80.0

extern int tkvcninit ();
extern int tkvcpinit ();
extern int tkvcoinit ();
extern int tkvcdinit ();
extern int tkvcsinit ();

extern int tkvcn ();
extern int tkvcp ();
extern int tkvco ();
extern int tkvcd ();
extern int tkvcs ();

extern void tkvcndone ();
extern void tkvcpdone ();
extern void tkvcodone ();
extern void tkvcddone ();
extern void tkvcsdone ();

```

```

extern int tkvcss (); /* for alter session to get memory size and trace
*/
extern boolean multitranx;
extern int ord_init;

extern void errrpt ();
extern int ocierror(char *fname, int lineno, OCIError *errhp, sword
status);
extern int sqlfile(char *fname, text *linebuf);

extern FILE *lfp;
extern FILE *fopen ();
extern int proc_no;
extern int doid[];

extern int execstatus;
extern int errcode;

extern OCIEnv *tpcenv;
extern OCIServer *tpcsrv;
extern OCIError *errhp;
extern OCISvcCtx *tpcsvc;
extern OCISession *tpcusr;
extern OCISmt *curntest;
/* The bind and define handles for each transaction are
included in their respective header files. */

/* for stock-level transaction */

extern int w_id;
extern int d_id;
extern int c_id;
extern int threshold;
extern int low_stock;

/* for delivery transaction */

extern int del_o_id[10];
extern int carrier_id;
extern int retries;

/* for order-status transaction */

extern int bylastname;
extern char c_last[17];
extern char c_first[17];
extern char c_middle[3];
extern double c_balance;
extern int o_id;
extern text o_entry_d[20];
extern int o_carrier_id;
extern int o_ol_cnt;
extern int ol_supply_w_id[15];
extern int ol_i_id[15];
extern int ol_quantity[15];
extern int ol_amount[15];
ub4 ol_del_len[15];
extern text ol_delivery_d[15][11];

/* for payment transaction */

```

```

extern int c_w_id;
extern int c_d_id;
extern int h_amount;
extern char w_street_1[21];
extern char w_street_2[21];
extern char w_city[21];
extern char w_state[3];
extern char w_zip[10];
extern char d_street_1[21];
extern char d_street_2[21];
extern char d_city[21];
extern char d_state[3];
extern char d_zip[10];
extern char c_street_1[21];
extern char c_street_2[21];
extern char c_city[21];
extern char c_state[3];
extern char c_zip[10];
extern char c_phone[17];
extern text c_since_d[11];
extern char c_credit[3];
extern int c_credit_lim;
extern float c_discount;
extern char c_data[201];
extern text h_date[20];

/* for new order transaction */

extern int nol_i_id[15];
extern int nol_supply_w_id[15];
extern int nol_quantity[15];
extern int nol_quant10[15];
extern int nol_quant19[15];
extern int nol_ytdqty[15];
extern int nol_amount[15];
extern int o_all_local;
extern float w_tax;
extern float d_tax;
extern float total_amount;
extern char i_name[15][25];
extern int i_name_strlen[15];
extern ub2 i_name_strlen_len[15];
extern ub2 i_name_strlen_rcode[15];
extern ub4 i_name_strlen_csize;
extern int s_quantity[15];
extern char brand_gen[15];
extern ub2 brand_gen_len[15];
extern ub2 brand_gen_rcode[15];
extern ub4 brand_gen_csize;
extern int i_price[15];
extern char brand_generic[15][1];
extern int status;
extern int tracelevel;

/* Miscellaneous */
extern OCIDate cr_date;
extern OCIDate c_since;
extern OCIDate o_entry_d_base;
extern OCIDate ol_d_base[15];

#ifdef DISCARD
# define DISCARD (void)
#endif

```

```

#ifndef sword
# define sword int
#endif

#define VER7          2

#define NA            -1      /* ANSI SQL NULL */
#define NLT           1      /* length for string null terminator */
#define DEADLOCK     60     /* ORA-00060: deadlock */
#define NO_DATA_FOUND 1403  /* ORA-01403: no data found */
#define NOT_SERIALIZABLE 8177 /* ORA-08177: transaction not
serializable */
#define SNAPSHOT_TOO_OLD 1555 /* ORA-01555: snapshot too old */

#ifndef NULLP
# define NULLP (void *)NULL
#endif /* NULLP */

#define ADR(object) ((ub1 *)&(object))
#define SIZ(object) ((sword)sizeof(object))

typedef char date[24+NLT];
typedef char varchar2;

#define min(x,y) ((x) < (y)) ? (x) : (y)

#define OCIERROR(errp,function)\
    ocierror(__FILE__, __LINE__, (errp), (function));

#define OCIBND(stmp, bndp, errp, sqlvar, progvl, ftype)\
    ocierror(__FILE__, __LINE__, (errp), \
    OCIHandleAlloc((stmp), (dvoid**) &(bndp), OCI_HTYPE_BIND, 0, (d
void**)0)); \
    ocierror(__FILE__, __LINE__, (errp), \
    OCIBindByName((stmp), &(bndp), (errp), \
    (text *) (sqlvar), strlen((sqlvar)), \
    (progvl), (progvl), (ftype), 0,0,0,0,0,OCI_DEFAULT));

#define
OCIBNDRA(stmp, bndp, errp, sqlvar, progvl, ftype, indp, alen, arcode) \
    ocierror(__FILE__, __LINE__, (errp), \
    OCIHandleAlloc((stmp), (dvoid**) &(bndp), OCI_HTYPE_BIND, 0, (d
oid**)0)); \
    ocierror(__FILE__, __LINE__, (errp), \
    OCIBindByName((stmp), &(bndp), (errp), (text
*) (sqlvar), strlen((sqlvar)), \
    (progvl), (progvl), (ftype), (indp), (alen), (arcod
e), 0,0,OCI_DEFAULT));

#define
OCIBNDRAD(stmp, bndp, errp, sqlvar, progvl, ftype, indp, ctxp, cbf_nodata, cbf_d
ata) \
    ocierror(__FILE__, __LINE__, (errp), \
    OCIHandleAlloc((stmp), (dvoid**) &(bndp), OCI_HTYPE_BIND, 0, (d
oid**)0)); \
    ocierror(__FILE__, __LINE__, (errp), \
    OCIBindByName((stmp), &(bndp), (errp), (text *) (sqlvar), \
    strlen((sqlvar)), 0, (progvl), (ftype), \
    indp, 0,0,0,0,OCI_DATA_AT_EXEC)); \
    ocierror(__FILE__, __LINE__, (errp), \
    OCIBindDynamic((bndp), (errp), (ctxp), (cbf_nodata), (ctxp), (cb
f_data));

```

```

#define
OCIBNDR(stmp, bndp, errp, sqlvar, progvl, ftype, indp, alen, arcode) \
    ocierror(__FILE__, __LINE__, (errp), \
    OCIHandleAlloc((stmp), (dvoid**) &(bndp), OCI_HTYPE_BIND, 0, (dvoi
d**)0)); \
    ocierror(__FILE__, __LINE__, (errp), \
    OCIBindByName((stmp), &(bndp), (errp), (text
*) (sqlvar), strlen((sqlvar)), \
    (progvl), (progvl), (ftype), (indp), (alen), (arcod
e), 0,0,OCI_DEFAULT));

#define
OCIBNDRAA(stmp, bndp, errp, sqlvar, progvl, ftype, indp, alen, arcode, ms,
cu) \
    ocierror(__FILE__, __LINE__, (errp), \
    OCIHandleAlloc((stmp), (dvoid**) &(bndp), OCI_HTYPE_BIND, 0, (dvoid
**)0)); \
    ocierror(__FILE__, __LINE__, (errp), \
    OCIBindByName((stmp), &(bndp), (errp), (text
*) (sqlvar), strlen((sqlvar)), \
    (progvl), (progvl), (ftype), (indp), (alen), (arcod
e), (ms), (cu), OCI_DE
FAULT));

#define OCIDEFINE(stmp, dfnp, errp, pos, progvl, ftype)\
    OCIDefineByPos((stmp), &(dfnp), (errp), (pos), (progvl), (progvl), (fty
pe), \
    0,0,0,OCI_DEFAULT);

#define OCIDEF(stmp, dfnp, errp, pos, progvl, ftype) \
    OCIHandleAlloc((stmp), (dvoid**) &(dfnp), OCI_HTYPE_DEFINE, 0, \
    (dvoid**)0); \
    OCIDefineByPos((stmp), &(dfnp), (errp), (pos), (progvl), (progvl), \
    (ftype), NULL, NULL, NULL, OCI_DEFAULT); \

#define
OCIDFNRA(stmp, dfnp, errp, pos, progvl, ftype, indp, alen, arcode) \
    OCIHandleAlloc((stmp), (dvoid**) &(dfnp), OCI_HTYPE_DEFINE, 0, \
    (dvoid**)0); \
    OCIDefineByPos((stmp), &(dfnp), (errp), (pos), (progvl), \
    (progvl), (ftype), (indp), (alen), \
    (arcod
e), OCI_DEFAULT);

#define
OCIDFNDRYN(stmp, dfnp, errp, pos, progvl, ftype, indp, ctxp, cbf_data) \
    ocierror(__FILE__, __LINE__, (errp), \
    OCIHandleAlloc((stmp), (dvoid**) &(dfnp), OCI_HTYPE_DEFINE, 0, \
    (dvoid**)0)); \
    ocierror(__FILE__, __LINE__, (errp), \
    OCIDefineByPos((stmp), &(dfnp), (errp), (pos), (progvl),
    (progvl), (ftype), \
    (indp), NULL, NULL,
    OCI_DYNAMIC_FETCH)); \
    ocierror(__FILE__, __LINE__, (errp), \
    OCIDefineDynamic((dfnp), (errp), (ctxp), (cbf_data));

#ifndef TUX
// New order

```

```

struct newinstruct {
    int w_id;
    int d_id;
    int c_id;
    int ol_i_id[15];
    int ol_supply_w_id[15];
    int ol_quantity[15];
};

struct newoutstruct {
    int terror;
    int o_id;
    int o_ol_cnt;
    char c_last[17];
    char c_credit[3];
    float c_discount;
    float w_tax;
    float d_tax;
    char o_entry_d[20];
    float total_amount;
    char i_name[15][25];
    int s_quantity[15];
    char brand_generic[15];
    float i_price[15];
    float ol_amount[15];
    char status[26];
    int retry;
};

struct newstruct {
    struct newinstruct newin;
    struct newoutstruct newout;
};

// Payment

struct payinstruct {
    int w_id;
    int d_id;
    int c_w_id;
    int c_d_id;
    int c_id;
    int bylastname;
    int h_amount;
    char c_last[17];
};

struct payoutstruct {
    int terror;
    char w_street_1[21];
    char w_street_2[21];
    char w_city[21];
    char w_state[3];
    char w_zip[10];
    char d_street_1[21];
    char d_street_2[21];
    char d_city[21];
    char d_state[3];
    char d_zip[10];
    int c_id;
    char c_first[17];
    char c_middle[3];
    char c_last[17];
};

```

```

    char c_street_1[21];
    char c_street_2[21];
    char c_city[21];
    char c_state[3];
    char c_zip[10];
    char c_phone[17];
    char c_since[11];
    char c_credit[3];
    double c_credit_lim;
    float c_discount;
    double c_balance;
    char c_data[201];
    char h_date[20];
    int retry;
};

struct paystruct {
    struct payinstruct payin;
    struct payoutstruct payout;
};

// Order status

struct ordinstruct {
    int w_id;
    int d_id;
    int c_id;
    int bylastname;
    char c_last[17];
};

struct ordoutstruct {
    int terror;
    int c_id;
    char c_last[17];
    char c_first[17];
    char c_middle[3];
    double c_balance;
    int o_id;
    char o_entry_d[20];
    int o_carrier_id;
    int o_ol_cnt;
    int ol_supply_w_id[15];
    int ol_i_id[15];
    int ol_quantity[15];
    float ol_amount[15];
    char ol_delivery_d[15][11];
    int retry;
};

struct ordstruct {
    struct ordinstruct ordin;
    struct ordoutstruct ordout;
};

// Delivery

struct delinstruct {
    int w_id;
    int o_carrier_id;
    double qtime;
    int in_timing_int;
};

```

```

};

struct deloutstruct {
    int terror;
    int retry;
};

struct delstruct {
    struct delinstruct delin;
    struct deloutstruct delout;
};

// Stock level /

struct stoinstruct {
    int w_id;
    int d_id;
    int threshold;
};

struct stooutstruct {
    int terror;
    int low_stock;
    int retry;
};

struct stostruct {
    struct stoinstruct stoin;
    struct stooutstruct stoout;
};

#endif
#endif

```

TPCC_INFO.H

```

/*
 * $Header: tpcc_info.h 7030100.1 95/07/19 15:11:37 plai Generic<base>
 * Copyr (c) 1995 Oracle
 */
/*=====+
|          Copyright (c) 1995 Oracle Corp, Redwood Shores, CA          |
|          OPEN SYSTEMS PERFORMANCE GROUP                             |
|          All Rights Reserved                                         |
+=====+
| FILENAME                                                             |
|   tpcc_info.h                                                       |
| DESCRIPTION                                                           |
|   Include file for TPC-C benchmark programs.                         |
+=====*/

#ifndef TPCC_INFO_H
#define TPCC_INFO_H

/* this set is duplicated in c_Defs.h, c_Defs.h is used for batch
driver */
#define MENTXN      0          /* menu txn */
#define NEWTXN      1          /* new order transaction */
#define PAYTXN      2          /* payment transaction */

```

```

#define ORDTXN      3          /* order status transaction */
#define DELTXN      4          /* delivery transaction */
#define STOTXN      5          /* stock level transaction */
#define ALLTXN      6          /* for processing all txns */
#define ALLTXNNODEL 7         /* for processing all txns except delivery
 */
/* New order */

struct newinstruct {
    int w_id;
    int d_id;
    int c_id;
    int ol_i_id[15];
    int ol_supply_w_id[15];
    int ol_quantity[15];
};

struct newoutstruct {
    int terror;
    int o_id;
    int o_ol_cnt;
    char c_last[17];
    char c_credit[3];
    float c_discount;
    float w_tax;
    float d_tax;
    char o_entry_d[20];
    float total_amount;
    char i_name[15][25];
    int s_quantity[15];
    char brand_generic[15];
    float i_price[15];
    float ol_amount[15];
    char status[26];
    int retry;
};

struct newstruct {
    int retval;
    int old_quantity[15];
    struct newinstruct newin;
    struct newoutstruct newout;
};

/* Payment */

struct payinstruct {
    int w_id;
    int d_id;
    int c_w_id;
    int c_d_id;
    int c_id;
    int bylastname;
    int h_amount;
    char c_last[17];
};

struct payoutstruct {
    int terror;
    char w_street_1[21];
    char w_street_2[21];
    char w_city[21];
    char w_state[3];

```

```

char w_zip[10];
char d_street_1[21];
char d_street_2[21];
char d_city[21];
char d_state[3];
char d_zip[10];
int c_id;
char c_first[17];
char c_middle[3];
char c_last[17];
char c_street_1[21];
char c_street_2[21];
char c_city[21];
char c_state[3];
char c_zip[10];
char c_phone[17];
char c_since[11];
char c_credit[3];
double c_credit_lim;
float c_discount;
double c_balance;
char c_data[201];
char h_date[20];
int retry;
};

struct paystruct {
    int retval;
    struct payinstruct payin;
    struct payoutstruct payout;
};

/* Order status */

struct ordinstruct {
    int w_id;
    int d_id;
    int c_id;
    int bylastname;
    char c_last[17];
};

struct ordoutstruct {
    int terror;
    int c_id;
    char c_last[17];
    char c_first[17];
    char c_middle[3];
    double c_balance;
    int o_id;
    char o_entry_d[20];
    int o_carrier_id;
    int o_ol_cnt;
    int ol_supply_w_id[15];
    int ol_i_id[15];
    int ol_quantity[15];
    float ol_amount[15];
    char ol_delivery_d[15][11];
    int retry;
};

struct ordstruct {
    int retval;

```

```

    struct ordinstruct ordin;
    struct ordoutstruct ordout;
};

/* Delivery */

struct delinstruct {
    int w_id;
    int o_carrier_id;
    double qtime;
    int in_timing_int;
};

struct deloutstruct {
    int terror;
    int retry;
};

struct delstruct {
    int retval;
    struct delinstruct delin;
    struct deloutstruct delout;
};

/* Stock level */

struct stoinstruct {
    int w_id;
    int d_id;
    int threshold;
};

struct stooutstruct {
    int terror;
    int low_stock;
    int retry;
};

struct stostruct {
    int retval;
    struct stoinstruct stoin;
    struct stooutstruct stoout;
};

/* used these definitions in client code only */
typedef struct delstruct DeliveryData, *pDeliveryData;
typedef struct newstruct NewOrderData, *pNewOrderData;
typedef struct paystruct PaymentData, *pPaymentData;
typedef struct ordstruct OrderStatusData, *pOrderStatusData;
typedef struct stostruct StockLevelData, *pStockLevelData;

#endif

```

TPCCFLAGS.H

```

/* #define DMLRETNO */
#define PLSQLNO
#define DMLRETDEL
/* #define PLSQLORD */

```

TPCCPL.H

```
/*
 * $Header: tpccpl.h 7030100.1 96/04/02 18:03:35 plai Generic<base>
 $ Copyr (c) 1994 Oracle
 */
/*=====+
|          Copyright (c) 1994 Oracle Corp, Redwood Shores, CA          |
|          OPEN SYSTEMS PERFORMANCE GROUP                             |
|          All Rights Reserved                                         |
+=====+
| FILENAME                                                             |
|   tpccpl.h                                                           |
| DESCRIPTION                                                           |
|   Header file for TPC-C transactions in PL/SQL.                     |
+=====+*/

#ifndef TPCCPL_H
#define TPCCPL_H

#include <stdio.h>
#include "tpcc.h"

#ifdef TUX
#define DELRT 5.0
#else
#define DELRT 80.0
#endif

extern int plnewinit ();
extern int plpayinit ();
extern int plordinit ();
extern int pldelinit ();
extern int plstoinit ();

extern int plnew ();
extern int plpay ();
extern int plord ();
extern int pldel ();
extern int plsto ();

extern void plnewdone ();
extern void plpaydone ();
extern void plorddone ();
extern void pldeldone ();
extern void plstodone ();

extern errrpt ();
extern void logerr();
extern int ocierror(char *fname, int lineno, OCLError *errhp, sword
status);

extern int sqlfile(char *fname, text *linebuf);

extern void cvtdmy ( unsigned char *orady, char *outdate);
extern void cvtdmyhms (unsigned char *orady, char *outdate);

extern FILE *lfp;
extern FILE *fopen ();
extern int proc no;
extern int doid[];
```

```
extern int execstatus;
extern int errcode;

extern OCIEnv *tpcenv;
extern OCIServer *tpcsrv;
extern OCLError *errhp;
extern OCISvcCtx *tpscvc;
extern OCISession *tpcusr;
extern OCISstmt *curn, *curn1, *curn2, *curn3[10], *curn4;
extern OCISstmt *curntest;
/* The bind and define handles for each transaction are
   included in their respective header files. */

extern ldadef tpclda;
extern csrdef curs;
extern csrdef curd;
extern csrdef curo0;
extern csrdef curo1;
extern csrdef curo2;
/*extern csrdef curp0;
extern csrdef curp1;
*/
/*extern csrdef curn, curn1, curn2, curn3[10], curn4;*/

extern unsigned long tpchda[];

/* for stock-level transaction */

extern int w_id;
extern int d_id;
extern int c_id;
extern int threshold;
extern int low_stock;

/* for delivery transaction */

extern int del_o_id[10];
extern int carrier_id;
extern int retries;

/* for order-status transaction */

extern int bylastname;
extern char c_last[17];
extern char c_first[17];
extern char c_middle[3];
extern double c_balance;
extern int o_id;
extern char o_entry_d[20];
extern int o_carrier_id;
extern int o_ol_cnt;
extern int ol_supply_w_id[15];
extern int ol_i_id[15];
extern int ol_quantity[15];
extern int ol_amount[15];
extern char ol_delivery_d[15][11];

/* for payment transaction */

extern int c_w_id;
extern int c_d_id;
extern int h_amount;
```

```

extern char w_street_1[21];
extern char w_street_2[21];
extern char w_city[21];
extern char w_state[3];
extern char w_zip[10];
extern char d_street_1[21];
extern char d_street_2[21];
extern char d_city[21];
extern char d_state[3];
extern char d_zip[10];
extern char c_street_1[21];
extern char c_street_2[21];
extern char c_city[21];
extern char c_state[3];
extern char c_zip[10];
extern char c_phone[17];
extern char c_since_d[11];
extern char c_credit[3];
extern int c_credit_lim;
extern int c_discount;
extern char c_data[201];
extern char h_date[20];
extern char bad_credit[33];

/* for new order transaction */

extern int nol_i_id[15];
extern int nol_supply_w_id[15];
extern int nol_quantity[15];
extern int nol_quant10[15];
extern int nol_quant91[15];
extern int nol_ytdqty[15];
extern int nol_amount[15];
extern int o_all_local;
extern int w_tax;
extern int d_tax;
extern float total_amount;
extern char i_name[15][25];
extern int i_name_strlen[15];
extern ub2 i_name_strlen_len[15];
extern ub2 i_name_strlen_rcode[15];
extern ub4 i_name_strlen_csize;
extern int s_quantity[15];
extern char brand_gen[15];
extern ub2 brand_gen_len[15];
extern ub2 brand_gen_rcode[15];
extern ub4 brand_gen_csize;
extern int i_price[15];
extern int status;

/* Miscellaneous */
extern unsigned char cr_date[7];
extern unsigned char c_since[7];
extern unsigned char o_entry_d_base[7];
extern unsigned char ol_d_base[15][7];

#ifdef DISCARD
# define DISCARD (void)
#endif

#ifdef sword
# define sword int
#endif

```

```

#define VER7                2

#define NA                   -1      /* ANSI SQL NULL */
#define NLT                   1      /* length for string null terminator */
#define DEADLOCK              60     /* ORA-00060: deadlock */
#define NO_DATA_FOUND         1403   /* ORA-01403: no data found */
#define NOT_SERIALIZABLE      8177   /* ORA-08177: transaction not
serializable */
#define SNAPSHOT_TOO_OLD     1555   /* ORA-01555: snapshot too old */

#ifdef NULLP
# define NULLP (void *)NULL
#endif /* NULLP */

#define ADR(object) ((ub1 *)&(object))
#define SIZ(object) ((sword)sizeof(object))

typedef char date[24+NLT];
typedef char varchar2;

#define OCIERROR(errp,function)\
    ocierror(__FILE__,__LINE__,(errp),(function));

#define OCIBND(stmp, bndp, errp, sqlvar, progv, progv1, ftype)\
    ocierror(__FILE__,__LINE__,(errp), \
OCIHandleAlloc((stmp), (dvoid*)&(bndp), OCI_HTYPE_BIND, 0, (dvoid**)0); \
    ocierror(__FILE__, __LINE__, (errp), \
        OCIBindByName((stmp), &(bndp), (errp), \
            (text *) (sqlvar), strlen((sqlvar)), \
            (progv), (progv1), (ftype), 0, 0, 0, 0, OCI_DEFAULT));

#define
OCIBNDRA(stmp, bndp, errp, sqlvar, progv, progv1, ftype, indp, alen, arcode) \
    ocierror(__FILE__, __LINE__, (errp), \
OCIHandleAlloc((stmp), (dvoid*)&(bndp), OCI_HTYPE_BIND, 0, (dvoid**)0); \
    ocierror(__FILE__, __LINE__, (errp), \
        OCIBindByName((stmp), &(bndp), (errp), (text
*) (sqlvar), strlen((sqlvar)), \
            (progv), (progv1), (ftype), (indp), (alen), (arcode), 0, 0, OCI_DEFAULT));
#define
OCIBNDRAD(stmp, bndp, errp, sqlvar, progv, progv1, ftype, indp, ctxp, cbf_nodata, cbf_d
ata) \
    ocierror(__FILE__, __LINE__, (errp), \
OCIHandleAlloc((stmp), (dvoid*)&(bndp), OCI_HTYPE_BIND, 0, (dvoid**)0); \
    ocierror(__FILE__, __LINE__, (errp), \
        OCIBindByName((stmp), &(bndp), (errp), (text *) (sqlvar), \
            strlen((sqlvar)), 0, (progv1), (ftype), \
            indp, 0, 0, 0, 0, OCI_DATA_AT_EXEC)); \
    ocierror(__FILE__, __LINE__, (errp), \
OCIBindDynamic((bndp), (errp), (ctxp), (cbf_nodata), (ctxp), (cbf_data)));

#define
OCIBNDR(stmp, bndp, errp, sqlvar, progv, progv1, ftype, indp, alen, arcode) \
    ocierror(__FILE__, __LINE__, (errp), \
OCIHandleAlloc((stmp), (dvoid*)&(bndp), OCI_HTYPE_BIND, 0, (dvoid**)0); \
    ocierror(__FILE__, __LINE__, (errp), \

```

```

        OCIBindByName((stmp), &(bndp), (errp), (text
*) (sqlvar), strlen((sqlvar)), \
(progvl), (progvl), (ftype), (indp), (alen), (arcod), 0, 0, OCI_DEFAULT));
#define
OCIBNDRAA(stmp, bndp, errp, sqlvar, progvl, progvl, ftype, indp, alen, arcod, ms,
cu) \
    ocierror(OCI_HANDLE, LINE, (errp), \
OCIHandleAlloc((stmp), &(bndp), OCI_HTYPE_BIND, 0, (dvoid**)0));
\
    ocierror(OCI_HANDLE, LINE, (errp), \
OCIBindByName((stmp), &(bndp), (errp), (text
*) (sqlvar), strlen((sqlvar)), \
(progvl), (progvl), (ftype), (indp), (alen), (arcod), (ms), (cu), OCI_DE
FAULT));
#define OCIDEFINE(stmp, dfnp, errp, pos, progvl, progvl, ftype) \
OCIDefineByPos((stmp), &(dfnp), (errp), (pos), (progvl), (progvl), (fty
pe), \
0, 0, 0, OCI_DEFAULT);
#define OCIDEF(stmp, dfnp, errp, pos, progvl, progvl, ftype) \
OCIHandleAlloc((stmp), (dvoid**) &(dfnp), OCI_HTYPE_DEFINE, 0, \
(dvoid**)0); \
OCIDefineByPos((stmp), &(dfnp), (errp), (pos), (progvl), (progvl), \
(ftype), NULL, NULL, NULL, OCI_DEFAULT); \
#define
OCIDFNRA(stmp, dfnp, errp, pos, progvl, progvl, ftype, indp, alen, arcod) \
OCIHandleAlloc((stmp), (dvoid**) &(dfnp), OCI_HTYPE_DEFINE, 0, \
(dvoid**)0); \
OCIDefineByPos((stmp), &(dfnp), (errp), (pos), (progvl), \
(progvl), (ftype), (indp), (alen), \
(arcod), OCI_DEFAULT); \
/*
OCIDefineArrayOfStruct((dfnp), (errp), (progvl), \
sizeof((indp)[0]), \
sizeof((alen)[0]), \
sizeof((arcod)[0]));
*/
/*
#define
OCIDFNRA(stmp, dfnp, errp, pos, progvl, progvl, ftype, indp, alen, arcod) \
ocierror(OCI_HANDLE, LINE, (errp), \
OCIHandleAlloc(Tpcenv, &(dfnp), OCI_HTYPE_DEFINE, 0, \
(dvoid**)0)); \
ocierror(OCI_HANDLE, LINE, (errp), \
OCIDefineByPos((stmp), &(dfnp), (errp), (pos), (progvl), \
(progvl), (ftype), (indp), (alen), \
(arcod), OCI_DEFAULT)); \
ocierror(OCI_HANDLE, LINE, (errp), \
OCIDefineArrayOfStruct((dfnp), (errp), (progvl), \
sizeof((indp)[0]), \
sizeof((alen)[0]), \
sizeof((arcod)[0]));
*/
#define OBNDRV(lda, cursor, sqlvar, progvl, progvl, ftype) \
if
(obndrv((cursor), (text*) (sqlvar), NA, (ub1*) (progvl), (progvl), (ftype), NA, \
(sb2 *)0, (text *)0, NA, NA)) \

```

```

{errrpt(lda, cursor); return(-1);} \
else \
DISCARD 0
#define OBNDRA(lda, cursor, sqlvar, progvl, progvl, ftype, indp, alen, arcod) \
if
(obndra((cursor), (text*) (sqlvar), NA, (ub1*) (progvl), (progvl), (ftype), NA, \
(indp), (alen), (arcod), (ub4)0, (ub4*)0, (text*)0, NA, NA)) \
{errrpt(lda, cursor); return(-1);} \
else \
DISCARD 0
#define
OBNDRAA(lda, cursor, sqlvar, progvl, progvl, ftype, indp, alen, arcod, ms, cs) \
if
(obndra((cursor), (text*) (sqlvar), NA, (ub1*) (progvl), (progvl), (ftype), NA, \
(indp), (alen), (arcod), (ub4) (ms), (ub4*) (cs), (text*)0, NA, NA)) \
{errrpt(lda, cursor); return(-1);} \
else \
DISCARD 0
#define
ODEFIN(lda, cursor, pos, buf, buf1, ftype, scale, indp, fmt, fmt1, fmtt, rlen, rcode) \
if
(odefin((cursor), (pos), (ub1*) (buf), (buf1), (ftype), (scale), (indp), \
(text*) (fmt), (fmt1), (fmtt), (rlen), (rcode))) \
{errrpt(lda, cursor); return(-1);} \
else \
DISCARD 0
#define OEXFET(lda, cursor, nrow, cancel, exact) \
if (oexfet((cursor), (nrow), (cancel), (exact))) \
{if ((cursor)->rc == 1403) \
{i=errrpt(lda, cursor); orol(lda); return(-1);} \
else if (errrpt(lda, cursor) == RECOVER) \
{orol(lda); return(RECOVER);} \
else {orol(lda); return(-1);}} \
else \
DISCARD 0
#define OOPEN(lda, cursor) \
if (oopen((cursor), (lda), (text*)0, NA, NA, (text*)0, NA)) \
{errrpt(lda, cursor); return(-1);} \
else \
DISCARD 0
#define OPARSE(lda, cursor, sqlstm, sql1, defflg, lngflg) \
if (oparse((cursor), (sqlstm), (sb4) (sql1), (defflg), (ub4) (lngflg))) \
{errrpt(lda, cursor); return(-1);} \
else \
DISCARD 0
#define OFEN(lda, cursor, nrow) \
if (ofen((cursor), (nrow))) \
{if (errrpt(lda, cursor) == RECOVER) \
{orol(lda); return(RECOVER);} \
else {orol(lda); return(-1);}} \
else \
DISCARD 0
#define OEXEC(lda, cursor) \
if (oexec((cursor))) \
{if (errrpt(lda, cursor) == RECOVER) \

```

```

        {orol(lda);return(RECOVERR);} \
    else{orol(lda);return(-1);}\
else\
    DISCARD 0

#define OCOM(lda,cursor)\
    if (ocom((lda)) \
        {errrpt(lda,cursor);orol(lda);return(-1);}\
    else\
        DISCARD 0

#define OEXN(lda,cursor,itors,rowoff)\
    if (oexn((cursor),(itors),(rowoff)) \
        {if (errrpt(lda,cursor)==RECOVERR) \
            {orol(lda);return(RECOVERR);} \
        else{orol(lda);return(-1);}\
    else\
        DISCARD 0

#endif

```

PLDEL.C

```

#ifndef RCSID
static char *RCSid =
"$Header: pldel.c 7030100.5 96/06/24 16:26:06 plai Generic<base>
$ Copyr (c) 1994 Oracle";
#endif /* RCSID */

/*=====+
|          Copyright (c) 1996 Oracle Corp, Redwood Shores, CA
|          OPEN SYSTEMS PERFORMANCE GROUP
|          All Rights Reserved
|=====+
| FILENAME
|   pldel.c
| DESCRIPTION
|   OCI version of DELIVERY transaction in TPC-C benchmark.
|=====+*/

#include "tpcc.h"
#ifdef TUX
#include <userlog.h>
#endif

/*
extern int userlog();
*/

#include "tpccflags.h"

#if defined(ISO) || defined(ISO5) || defined(ISO6) || defined(ISO8)
#define SQLTXT0 "SELECT substr(value,1,5) FROM v$parameter \
    WHERE name = 'instance_number'"
#endif

#ifdef PLSQLDEL
#define SQLTXT "BEGIN delivery.deliver (:w_id, :carrier_id, :order_id,\
    :retry); END;"
#else

```

```

# ifdef DMLRETDEL
#define SQLTXT1 "DELETE FROM nord WHERE no_d_id = :d_id \
    AND no_w_id = :w_id and rownum <= 1 \
    RETURNING no_o_id INTO :o_id "

# else
#define SQLTXT1A "\
SELECT /*+ USE_NL(nord ordr) ORDERED */ 1, no_o_id, nord.rowid, o_c_id,
    ordr.rowid \
FROM nord, ordr \
WHERE no_w_id = :w_id AND no_d_id = 1 AND o_w_id = :w_id AND o_d_id = 1
AND \
    o_id = no_o_id AND rownum <= 1 UNION ALL \
"

#define SQLTXT1B "\
SELECT /*+ USE_NL(nord ordr) ORDERED */ 2, no_o_id, nord.rowid, o_c_id,
    ordr.rowid \
FROM nord, ordr \
WHERE no_w_id = :w_id AND no_d_id = 2 AND o_w_id = :w_id AND o_d_id = 2
AND \
    o_id = no_o_id AND rownum <= 1 UNION ALL \
"

#define SQLTXT1C "\
SELECT /*+ USE_NL(nord ordr) ORDERED */ 3, no_o_id, nord.rowid, o_c_id,
    ordr.rowid \
FROM nord, ordr \
WHERE no_w_id = :w_id AND no_d_id = 3 AND o_w_id = :w_id AND o_d_id = 3
AND \
    o_id = no_o_id AND rownum <= 1 UNION ALL \
"

#define SQLTXT1D "\
SELECT /*+ USE_NL(nord ordr) ORDERED */ 4, no_o_id, nord.rowid, o_c_id,
    ordr.rowid \
FROM nord, ordr \
WHERE no_w_id = :w_id AND no_d_id = 4 AND o_w_id = :w_id AND o_d_id = 4
AND \
    o_id = no_o_id AND rownum <= 1 UNION ALL \
"

#define SQLTXT1E "\
SELECT /*+ USE_NL(nord ordr) ORDERED */ 5, no_o_id, nord.rowid, o_c_id,
    ordr.rowid \
FROM nord, ordr \
WHERE no_w_id = :w_id AND no_d_id = 5 AND o_w_id = :w_id AND o_d_id = 5
AND \
    o_id = no_o_id AND rownum <= 1 UNION ALL \
"

#define SQLTXT1F "\
SELECT /*+ USE_NL(nord ordr) ORDERED */ 6, no_o_id, nord.rowid, o_c_id,
    ordr.rowid \
FROM nord, ordr \
WHERE no_w_id = :w_id AND no_d_id = 6 AND o_w_id = :w_id AND o_d_id = 6
AND \
    o_id = no_o_id AND rownum <= 1 UNION ALL \
"

#define SQLTXT1G "\
SELECT /*+ USE_NL(nord ordr) ORDERED */ 7, no_o_id, nord.rowid, o_c_id,
    ordr.rowid \
FROM nord, ordr \

```

```

WHERE no_w_id = :w_id AND no_d_id = 7 AND o_w_id = :w_id AND o_d_id = 7
AND \
  o_id = no_o_id AND rownum <= 1 UNION ALL \
"

#define SQLTXT1H "\
SELECT /*+ USE_NL(nord ordr) ORDERED */ 8, no_o_id, nord.rowid, o_c_id,
ordr.rowid \
FROM nord, ordr \
WHERE no_w_id = :w_id AND no_d_id = 8 AND o_w_id = :w_id AND o_d_id = 8
AND \
  o_id = no_o_id AND rownum <= 1 UNION ALL \
"

#define SQLTXT1I "\
SELECT /*+ USE_NL(nord ordr) ORDERED */ 9, no_o_id, nord.rowid, o_c_id,
ordr.rowid \
FROM nord, ordr \
WHERE no_w_id = :w_id AND no_d_id = 9 AND o_w_id = :w_id AND o_d_id = 9
AND \
  o_id = no_o_id AND rownum <= 1 UNION ALL \
"

#define SQLTXT1J "\
SELECT /*+ USE_NL(nord ordr) ORDERED */ 10, no_o_id, nord.rowid, o_c_id,
ordr.rowid \
FROM nord, ordr \
WHERE no_w_id = :w_id AND no_d_id = 10 AND o_w_id = :w_id AND o_d_id =
10 AND \
  o_id = no_o_id AND rownum <= 1"

#define SQLTXT2 "DELETE FROM nord WHERE rowid = :no_rowid"
#endif

#ifdef DMLRETDEL
#define SQLTXT3 "UPDATE ordr SET o_carrier_id = :carrier_id \
WHERE o_id = :o_id and o_d_id = :d_id and o_w_id
= :w_id \
returning o_c_id into :o_c_id"
#else
#define SQLTXT3 "UPDATE ordr SET o_carrier_id = :carrier_id \
WHERE rowid = :o_rowid"
#endif

#ifdef DMLRETDEL
#define SQLTXT4 "UPDATE /*+ buffer */ ordl SET ol_delivery_d = :cr_date
\
WHERE ol_w_id = :w_id AND ol_d_id = :d_id AND ol_o_id = :o_id \
RETURNING ol_amount into :ol_amount "
#else
#define SQLTXT4 "UPDATE ordl SET ol_delivery_d = :cr_date \
WHERE ol_w_id = :w_id AND ol_d_id = :d_id AND ol_o_id = :o_id"

#define SQLTXT5A "\
SELECT :d_id1, SUM(ol_amount) FROM ordl WHERE ol_w_id = :w_id AND \
ol_d_id = :d_id1 AND ol_o_id = :o_id1 UNION ALL \
SELECT :d_id2, SUM(ol_amount) FROM ordl WHERE ol_w_id = :w_id AND \
ol_d_id = :d_id2 AND ol_o_id = :o_id2 UNION ALL \
"

#define SQLTXT5B "\
SELECT :d_id3, SUM(ol_amount) FROM ordl WHERE ol_w_id = :w_id AND \
ol_d_id = :d_id3 AND ol_o_id = :o_id3 UNION ALL \
SELECT :d_id4, SUM(ol_amount) FROM ordl WHERE ol_w_id = :w_id AND \

```

```

ol_d_id = :d_id4 AND ol_o_id = :o_id4 UNION ALL \
"

#define SQLTXT5C "\
SELECT :d_id5, SUM(ol_amount) FROM ordl WHERE ol_w_id = :w_id AND \
ol_d_id = :d_id5 AND ol_o_id = :o_id5 UNION ALL \
SELECT :d_id6, SUM(ol_amount) FROM ordl WHERE ol_w_id = :w_id AND \
ol_d_id = :d_id6 AND ol_o_id = :o_id6 UNION ALL \
"

#define SQLTXT5D "\
SELECT :d_id7, SUM(ol_amount) FROM ordl WHERE ol_w_id = :w_id AND \
ol_d_id = :d_id7 AND ol_o_id = :o_id7 UNION ALL \
SELECT :d_id8, SUM(ol_amount) FROM ordl WHERE ol_w_id = :w_id AND \
ol_d_id = :d_id8 AND ol_o_id = :o_id8 UNION ALL \
"

#define SQLTXT5E "\
SELECT :d_id9, SUM(ol_amount) FROM ordl WHERE ol_w_id = :w_id AND \
ol_d_id = :d_id9 AND ol_o_id = :o_id9 UNION ALL \
SELECT :d_id10, SUM(ol_amount) FROM ordl WHERE ol_w_id = :w_id AND \
ol_d_id = :d_id10 AND ol_o_id = :o_id10"

#endif
#endif /* PLSQLDEL */

#define SQLTXT6 "UPDATE cust SET c_balance = c_balance + :amt, \
c_delivery_cnt = c_delivery_cnt + 1 WHERE c_w_id = :w_id AND \
c_d_id = :d_id AND c_id = :c_id"

#define NDISTS 10
#define ROWIDLEN 20

struct delctx {
  sb2 del_o_id_ind[NDISTS];
  sb2 cons_ind[NDISTS];
  sb2 w_id_ind[NDISTS];
  sb2 d_id_ind[NDISTS];
  sb2 c_id_ind[NDISTS];
  sb2 del_date_ind[NDISTS];
  sb2 carrier_id_ind[NDISTS];
  sb2 amt_ind[NDISTS];
  sb2 no_rowid_ind[NDISTS];
  sb2 o_rowid_ind[NDISTS];
  #if defined(ISO) || defined(ISO5) || defined(ISO6) || defined(ISO8)
  sb2 inum_ind;
  #endif
#ifdef DMLRETDEL
  ub4 del_o_id_len[NDISTS];
  ub4 c_id_len[NDISTS];
  int oid_ctx;
  int cid_ctx;
  OCIBind *olamt_bp;
#else
  ub2 del_o_id_len[NDISTS];
  ub2 c_id_len[NDISTS];

  ub2 cons_len[NDISTS];
  ub2 w_id_len[NDISTS];
  ub2 d_id_len[NDISTS];
  ub2 del_date_len[NDISTS];

```

```

ub2 carrier_id_len[NDISTS];
ub2 amt_len[NDISTS];
ub2 no_rowid_len[NDISTS];
ub2 no_rowid_ptr_len[NDISTS];
ub2 o_rowid_len[NDISTS];
ub2 o_rowid_ptr_len[NDISTS];
#if defined(ISO) || defined(ISO5) || defined(ISO6) || defined(ISO8)
ub2 inum_len;
#endif

ub2 del_o_id_rcode[NDISTS];
ub2 cons_rcode[NDISTS];
ub2 w_id_rcode[NDISTS];
ub2 d_id_rcode[NDISTS];
ub2 c_id_rcode[NDISTS];
ub2 del_date_rcode[NDISTS];
ub2 carrier_id_rcode[NDISTS];
ub2 amt_rcode[NDISTS];
ub2 no_rowid_rcode[NDISTS];
ub2 o_rowid_rcode[NDISTS];
#if defined(ISO) || defined(ISO5) || defined(ISO6) || defined(ISO8)
ub2 inum_rcode;
#endif

int del_o_id[NDISTS];
int cons[NDISTS];
int w_id[NDISTS];
int d_id[NDISTS];
int c_id[NDISTS];
int carrier_id[NDISTS];
int amt[NDISTS];
ub4 del_o_id_rcnt;
int retry;
OCIRowid *no_rowid_ptr[NDISTS];
OCIRowid *o_rowid_ptr[NDISTS];
OCIDate del_date[NDISTS];
#if defined(ISO) || defined(ISO5) || defined(ISO6) || defined(ISO8)
char inum[10];
#endif
OCISmt *curd0;
OCISmt *curd1;
OCISmt *curd2;
OCISmt *curd3;
OCISmt *curd4;
OCISmt *curd5;
OCISmt *curd6;
OCISmt *curdtest;

OCIBind *w_id_bp;
OCIBind *w_id_bp3;
OCIBind *w_id_bp4;
OCIBind *w_id_bp5;
OCIBind *w_id_bp6;
OCIBind *d_id_bp;
OCIBind *d_id_bp3;
OCIBind *d_id_bp4;
OCIBind *d_id_bp6;
OCIBind *o_id_bp;
OCIBind *cr_date_bp;
OCIBind *c_id_bp;
OCIBind *c_id_bp3;
OCIBind *no_rowid_bp;
OCIBind *carrier_id_bp;
OCIBind *o_rowid_bp;

```

```

OCIBind *del_o_id_bp;
OCIBind *del_o_id_bp3;
OCIBind *amt_bp;
OCIBind *bstr1_bp[10];
OCIBind *bstr2_bp[10];
OCIBind *retry_bp;
OCIDefine *inum_dp;
OCIDefine *d_id_dp;
OCIDefine *del_o_id_dp;
OCIDefine *no_rowid_dp;
OCIDefine *c_id_dp;
OCIDefine *o_rowid_dp;
OCIDefine *cons_dp;
OCIDefine *amt_dp;

int norow;
};

typedef struct delctx delctx;

delctx *dctx;

#ifdef DMLRETDEL
struct amtctx {
int ol_amt[NDISTS][NITEMS];
sb2 ol_amt_ind[NDISTS][NITEMS];
ub4 ol_amt_len[NDISTS][NITEMS];
ub2 ol_amt_rcode[NDISTS][NITEMS];
int ol_cnt[NDISTS];
};
typedef struct amtctx amtctx;
amtctx *actx;
#endif

#ifdef DMLRETDEL
extern sb4 no_data();

sb4 TPC_oid_data(dvoid *ctxp, OCIBind *bp, ub4 iter, ub4 index,
dvoid **bufpp, ub4 **alenp, ub1 *piecep,
dvoid **indpp, ub2 **rcodepp)
{
*bufpp = &dctx->del_o_id[iter];
*indpp= &dctx->del_o_id_ind[iter];
dctx->del_o_id_len[iter]=sizeof(dctx->del_o_id[0]);
*alenp= &dctx->del_o_id_len[iter];
*rcodepp = &dctx->del_o_id_rcode[iter];
*piecep =OCI_ONE_PIECE;
return (OCI_CONTINUE);
}

sb4 cid_data(dvoid *ctxp, OCIBind *bp, ub4 iter, ub4 index,
dvoid **bufpp, ub4 **alenp, ub1 *piecep,
dvoid **indpp, ub2 **rcodepp)
{
*bufpp = &dctx->c_id[iter];
*indpp= &dctx->c_id_ind[iter];
dctx->c_id_len[iter]=sizeof(dctx->c_id[0]);
*alenp= &dctx->c_id_len[iter];
*rcodepp = &dctx->c_id_rcode[iter];
*piecep =OCI_ONE_PIECE;
return (OCI_CONTINUE);
}

```

```

sb4 amt_data(dvoid *ctxp, OCIBind *bp, ub4 iter, ub4 index,
             dvoid **bufpp, ub4 **alenp, ub1 *piecep,
             dvoid **indpp, ub2 **rcodepp)
{
    amtctx *actx;
    actx = (amtctx*)ctxp;
    actx->ol_cnt[iter]=actx->ol_cnt[iter]+1;
    *bufpp = &actx->ol_amt[iter][index];
    *indpp= &actx->ol_amt_ind[iter][index];
    actx->ol_amt_len[iter][index]=sizeof(actx->ol_amt[0][0]);
    *alenp= &actx->ol_amt_len[iter][index];
    *rcodepp = &actx->ol_amt_rcode[iter][index];
    *piecep =OCI_ONE_PIECE;
    return (OCI_CONTINUE);
}

#endif

tkvcddinit ()
{
    # ifndef DMLRETDEL
    int i,j;
    char bstr1[10];
    char bstr2[10];
    # endif /* !DMLRETDEL */
    text stmbuf[SQL_BUF_SIZE];

    dctx = (delctx *) malloc (sizeof(delctx));
    memset (dctx, (char)0, sizeof(delctx));
    dctx->norow = 0;
    #ifdef DMLRETDEL
    actx = (amtctx *) malloc (sizeof(amtctx));
    memset (actx, (char)0, sizeof(amtctx));
    #else
    for(i=0;i<NDISTS;i++) {
        OCIERROR(errhp, OCIDescriptorAlloc(tpcenv, (dvoid**)&dctx-
        >o_rowid_ptr[i],
            OCI_DTYPE_ROWID,0, (dvoid**)0);
        OCIERROR(errhp, OCIDescriptorAlloc(tpcenv, (dvoid**)&dctx-
        >no_rowid_ptr[i],
            OCI_DTYPE_ROWID,0, (dvoid**)0);
    }
    #endif

    #if defined(ISO) || defined(ISO5) || defined(ISO6) || defined(ISO8)
    OCIHandleAlloc(tpcenv, (dvoid **)&dctx->curd0, OCI_HTYPE_STMT, 0,
    (dvoid**)0);
    sprintf ((char *) stmbuf, SQLTXT0);
    OCIStmtPrepare(dctx->curd0, errhp, stmbuf, strlen((char
    *)stmbuf),OCI_NTV_SYNTAX, OCI_DEFAULT);

    OCIDFNRA(dctx->curd0, dctx->inum_dp, errhp, 1, dctx->inum, SIZ(dctx-
    >inum), SQLT_STR,
        &(dctx->inum_ind), &(dctx->inum_len), &(dctx->inum_rcode));
    #endif

    /* If PLSQDEL and ISO? are both defined, then they both try to use
    curd0! This could cause a problem. Will try to fix later - VMM
    12/30/97 */

```

```

#endif PLSQDEL
OCIHandleAlloc(tpcenv, (dvoid **)&dctx->curd0, OCI_HTYPE_STMT,
0, (dvoid**)0);
sprintf ((char *) stmbuf, SQLTXT);
OCIStmtPrepare(dctx->curd0, errhp, stmbuf, strlen((char *)stmbuf),
OCI_NTV_SYNTAX, OCI_DEFAULT);
OCIWND(dctx->curd0, dctx->w_id_bp, errhp, ":w_id",ADR(w_id),SIZ(int),
SQLT_INT);
OCIWND(dctx->curd0, dctx->carrier_id_bp, errhp, ":carrier_id",
ADR(dctx->carrier_id), SIZ(int), SQLT_INT);

OCIWNDRAA(dctx->curd0, dctx->o_id_bp, errhp, ":order_id",
dctx->del_o_id, SIZ(int), SQLT_INT, dctx->del_o_id_ind,
dctx->del_o_id_len, dctx->del_o_id_rcode, NDISTS,
&dctx->del_o_id_rcnt);
OCIWND(dctx->curd0, dctx->retry_bp, errhp, ":retry",ADR(dctx->retry),
SIZ(int), SQLT_INT);

#else
# ifdef DMLRETDEL
OCIHandleAlloc(tpcenv, (dvoid **)&dctx->curd1, OCI_HTYPE_STMT, 0,
(dvoid**)0);
sprintf ((char *) stmbuf, "%s", SQLTXT1);
OCIStmtPrepare(dctx->curd1, errhp, stmbuf, strlen((char
*)stmbuf),OCI_NTV_SYNTAX, OCI_DEFAULT);

OCIWND(dctx->curd1, dctx->w_id_bp, errhp, ":w_id",dctx->w_id, SIZ(int),
SQLT_INT);
OCIWNDRA(dctx->curd1, dctx->d_id_bp, errhp, ":d_id",dctx-
>d_id, SIZ(int),
SQLT_INT, NULL, NULL, NULL);

OCIWNDRAD(dctx->curd1, dctx->del_o_id_bp, errhp, ":o_id",
SIZ(int), SQLT_INT, NULL,
&dctx->oid_ctx, no_data, TPC_oid_data);

# else

OCIHandleAlloc(tpcenv, (dvoid **)&dctx->curd1, OCI_HTYPE_STMT, 0,
(dvoid**)0);
sprintf ((char *) stmbuf, "%s%s%s%s%s%s%s%s%s", SQLTXT1A,
SQLTXT1B,
SQLTXT1C,
SQLTXT1D,
SQLTXT1E,
SQLTXT1F,
SQLTXT1G,
SQLTXT1H,
SQLTXT1I,
SQLTXT1J

);
OCIStmtPrepare(dctx->curd1, errhp, stmbuf, strlen((char
*)stmbuf),OCI_NTV_SYNTAX, OCI_DEFAULT);

OCIERROR(errhp,
OCIAttrSet(dctx->curd1,OCI_HTYPE_STMT, (dvoid*)&dctx->norow,0,
OCI_ATTR_PREFETCH_ROWS, errhp));

/* bind variables */

OCIWND(dctx->curd1, dctx-
>w_id_bp, errhp, ":w_id",ADR(w_id),SIZ(int), SQLT_INT);

OCIDFNRA(dctx->curd1, dctx->d_id_dp, errhp, 1, dctx->d_id, SIZ(int),

```

```

        SQLT_INT, dctx->d_id_ind, dctx->d_id_len, dctx->d_id_rcode);
OCIDFNRA(dctx->curd1, dctx->del_o_id_dp, errhp, 2, dctx->del_o_id,
        SIZ(int), SQLT_INT, dctx->del_o_id_ind,
        dctx->del_o_id_len, dctx->del_o_id_rcode);
OCIDFNRA(dctx->curd1, dctx->no_rowid_dp, errhp, 3, dctx->no_rowid_ptr,
        SIZ(OCIRowid *), SQLT_RDD, dctx->no_rowid_ind,
        dctx->no_rowid_len, dctx->no_rowid_rcode);
OCIDFNRA(dctx->curd1, dctx->c_id_dp, errhp, 4, dctx->c_id, SIZ(dctx-
>c_id[0]),
        SQLT_INT, dctx->c_id_ind, dctx->c_id_len, dctx->c_id_rcode);
OCIDFNRA(dctx->curd1, dctx->o_rowid_dp, errhp, 5, dctx->o_rowid_ptr,
        SIZ(OCIRowid *), SQLT_RDD, dctx->o_rowid_ind,
        dctx->o_rowid_len, dctx->o_rowid_rcode);

/* open second cursor */

OCIHandleAlloc(tpcenv, (dvoid **)&dctx->curd2, OCI_HTYPE_STMT, 0,
(dvoid**)0);
sprintf((char *) stmbuf, SQLTXT2);
OCIStmtPrepare(dctx->curd2, errhp, stmbuf, strlen((char *)stmbuf),
        OCI_NT_V_SYNTAX, OCI_DEFAULT);

/* bind variables */
OCIBNDRA(dctx->curd2, dctx->no_rowid_bp, errhp, ":no_rowid", &(dctx-
>no_rowid_ptr[0]),
        SIZ(dctx->no_rowid_ptr[0]), SQLT_RDD, dctx->no_rowid_ind,
        dctx->no_rowid_len, dctx->no_rowid_rcode);

# endif /*DMLRETDEL*/

/* open third cursor */

OCIHandleAlloc(tpcenv, (dvoid **)&dctx->curd3, OCI_HTYPE_STMT, 0,
(dvoid**)0);
sprintf((char *) stmbuf, SQLTXT3);
OCIStmtPrepare(dctx->curd3, errhp, stmbuf, strlen((char *)stmbuf),
        OCI_NT_V_SYNTAX, OCI_DEFAULT);

/* bind variables */

OCIBNDRA(dctx->curd3, dctx->carrier_id_bp, errhp, ":carrier_id", dctx-
>carrier_id,
        SIZ(dctx->carrier_id[0]), SQLT_INT, dctx->carrier_id_ind,
        dctx->carrier_id_len, dctx->carrier_id_rcode);

# ifdef DMLRETDEL
OCIBNDRA(dctx->curd3, dctx->w_id_bp3, errhp, ":w_id", dctx-
>w_id, SIZ(int),
        SQLT_INT, NULL, NULL, NULL);
OCIBNDRA(dctx->curd3, dctx->d_id_bp3, errhp, ":d_id", dctx-
>d_id, SIZ(int),
        SQLT_INT, NULL, NULL, NULL);
OCIBNDRA(dctx->curd3, dctx->del_o_id_bp3, errhp, ":o_id", dctx-
>del_o_id,
        SIZ(int), SQLT_INT, NULL, NULL, NULL);
OCIBNDRAD(dctx->curd3, dctx->c_id_bp3, errhp, ":o_c_id", SIZ(int),
        SQLT_INT, NULL, &dctx->cid_ctx, no_data, cid_data);
# else
OCIBNDRA(dctx->curd3, dctx->o_rowid_bp, errhp, ":o_rowid", &(dctx-
>o_rowid_ptr[0]),
        SIZ(dctx->o_rowid_ptr[0]), SQLT_RDD, dctx->o_rowid_ind,
        dctx->o_rowid_ptr_len, dctx->o_rowid_rcode);

```

```

#endif

/* open fourth cursor */

OCIHandleAlloc(tpcenv, (dvoid **)&dctx->curd4, OCI_HTYPE_STMT, 0,
(dvoid**)0);
sprintf((char *) stmbuf, SQLTXT4);
OCIStmtPrepare(dctx->curd4, errhp, stmbuf, strlen((char *)stmbuf),
        OCI_NT_V_SYNTAX, OCI_DEFAULT);

/* bind variables */

OCIBND(dctx->curd4, dctx->w_id_bp4, errhp, ":w_id", dctx->w_id,
        SIZ(int), SQLT_INT);
OCIBND(dctx->curd4, dctx->d_id_bp4, errhp, ":d_id", dctx->d_id,
        SIZ(int), SQLT_INT);
OCIBND(dctx->curd4, dctx->o_id_bp, errhp, ":o_id", dctx->del_o_id,
        SIZ(int), SQLT_INT);
OCIBND(dctx->curd4, dctx->cr_date_bp, errhp, ":cr_date", dctx-
>del_date,
        SIZ(OCIDate), SQLT_ODT);
# ifdef DMLRETDEL
OCIBNDRAD(dctx->curd4, dctx->olamt_bp, errhp, ":ol amount",
        SIZ(int), SQLT_INT, NULL, actx, no_data, amt_data);
# else

/* open fifth cursor */

OCIHandleAlloc(tpcenv, (dvoid **)&dctx->curd5, OCI_HTYPE_STMT, 0,
(dvoid**)0);
sprintf((char *) stmbuf, "%s%s%s%s", SQLTXT5A,
        SQLTXT5B,
        SQLTXT5C,
        SQLTXT5D,
        SQLTXT5E
);
OCIStmtPrepare(dctx->curd5, errhp, stmbuf, strlen((char *)stmbuf),
        OCI_NT_V_SYNTAX, OCI_DEFAULT);

OCIERROR(errhp,
        OCIAttrSet(dctx->curd5, OCI_HTYPE_STMT, (dvoid*)&dctx->norow, 0,
        OCI_ATTR_PREFETCH_ROWS, errhp));

/* bind variables */

OCIBND(dctx->curd5, dctx-
>w_id_bp, errhp, ":w_id", ADR(w_id), SIZ(w_id), SQLT_INT);
for (i = 0; i < NDISTS; i++) {
    sprintf(bstr1, ":d_id%d", i + 1);
    sprintf(bstr2, ":o_id%d", i + 1);
    OCIBNDRA(dctx->curd5, dctx->bstr1_bp[i], errhp, bstr1, ADR(dctx-
>d_id[i]),
            SIZ(dctx->d_id[0]), SQLT_INT, &(dctx->d_id_ind[i]),
            &(dctx->d_id_len[i]), &(dctx->d_id_rcode[i]));
    OCIBNDRA(dctx->curd5, dctx->bstr2_bp[i], errhp, bstr2, ADR(dctx-
>del_o_id[i]),
            SIZ(dctx->del_o_id[0]), SQLT_INT, &(dctx-
>del_o_id_ind[i]),
            &(dctx->del_o_id_len[i]), &(dctx->del_o_id_rcode[i]));
}

```

```

    OCIDFNRA(dctx->curd5,dctx->cons_dp,errhp,1,dctx->cons,SIZ(dctx-
>cons[0]),SQLT_INT,
    dctx->cons_ind,dctx->cons_len,dctx->cons_rcode);
    OCIDFNRA(dctx->curd5,dctx->amt_dp,errhp,2,dctx->amt,SIZ(dctx-
>amt[0]),SQLT_INT,
    dctx->amt_ind,dctx->amt_len,dctx->amt_rcode);
#endif
/* open sixth cursor */

OCIHandleAlloc(tpcenv, (dvoid **>(&dctx->curd6), OCI_HTYPE_STMT, 0,
(dvoid**)0);
sprintf((char *) stmbuf, SQLT6);
OCIStmtPrepare(dctx->curd6, errhp, stmbuf, strlen((char *)stmbuf),
OCI_NT_V_SYNTAX, OCI_DEFAULT);

/* bind variables */

OCIBND(dctx->curd6,dctx->amt_bp,errhp,":amt",dctx->amt,SIZ(int),
SQLT_INT);
OCIBND(dctx->curd6,dctx->w_id_bp6,errhp,":w_id",dctx->w_id,SIZ(int),
SQLT_INT);
OCIBND(dctx->curd6,dctx->d_id_bp6,errhp,":d_id",dctx->d_id,SIZ(int),
SQLT_INT);
OCIBND(dctx->curd6,dctx->c_id_bp,errhp,":c_id",dctx->c_id,SIZ(int),
SQLT_INT);
#endif
return (0);
}

void shiftdata(from)
int from ;
{
    int i;
    for (i=from;i<NDISTS-1; i++)
    {
        dctx->del_o_id_ind[i] = dctx->del_o_id_ind[i+1];
        dctx->del_o_id[i] = dctx->del_o_id[i+1];
        dctx->w_id[i] = dctx->w_id[i+1];
        dctx->d_id[i] = dctx->d_id[i+1];
        dctx->carrier_id[i] = dctx->carrier_id[i+1];
    }
}

tkvcd ()
{
    int i, j;
    int rpc,rcount,count;
    int invalid;
#ifdef DMLRETDEL
    int tmp_id,v;
    int tmp_amt;
#endif /* !DMLRETDEL */

#ifdef ISO || defined(ISO5) || defined(ISO6) || defined(ISO8)
    int hasno;
    int reread;
    char sdate[30];

```

```

OCIStmtExecute(tpcsvc,dctx->curd0,errhp,1,0,0,OCI_DEFAULT);
sysdate (sdate);
printf ("Delivery started at %s on node %s\n", sdate, dctx->inum);
#endif
#ifdef PLSQLDEL
    for (i = 0; i < NDISTS; i++)
    {
        dctx->del_o_id_ind[i] = TRUE;
        dctx->del_o_id_len[i] = sizeof(int);
    }

    OCIERROR(errhp,
        OCIStmtExecute(tpcsvc,dctx->curd0,errhp,1,0,0,OCI_DEFAULT));

    for (i = 0; i < NDISTS; i++)
    {
        del_o_id[i] = 0;
        if (dctx->del_o_id_ind[i] == 0)
        {
            del_o_id[i] = dctx->del_o_id[i];
        }
    }
#else
retry:

#ifdef ISO || defined(ISO5) || defined(ISO6) || defined(ISO8)
    reread = 1;
#endif

#ifdef ISO || defined(ISO5) || defined(ISO6) || defined(ISO8)
iso:
#endif

        invalid = 0;

        /* initialization for array operations */

        for (i = 0; i < NDISTS; i++) {
            dctx->del_o_id_ind[i] = TRUE;
            dctx->cons_ind[i] = TRUE;
            dctx->w_id_ind[i] = TRUE;
            dctx->d_id_ind[i] = TRUE;
            dctx->c_id_ind[i] = TRUE;
            dctx->del_date_ind[i] = TRUE;
            dctx->carrier_id_ind[i] = TRUE;
            dctx->amt_ind[i] = TRUE;
            dctx->no_rowid_ind[i] = TRUE;
            dctx->o_rowid_ind[i] = TRUE;

            dctx->del_o_id_len[i] = SIZ(dctx->del_o_id[0]);
            dctx->cons_len[i] = SIZ(dctx->cons[0]);
            dctx->w_id_len[i] = SIZ(dctx->w_id[0]);
            dctx->d_id_len[i] = SIZ(dctx->d_id[0]);
            dctx->c_id_len[i] = SIZ(dctx->c_id[0]);
            dctx->del_date_len[i] = DEL_DATE_LEN;
            dctx->carrier_id_len[i] = SIZ(dctx->carrier_id[0]);
            dctx->amt_len[i] = SIZ(dctx->amt[0]);
            dctx->no_rowid_len[i] = ROWIDLEN;
            dctx->o_rowid_len[i] = ROWIDLEN;
            dctx->o_rowid_ptr_len[i] = SIZ(dctx->o_rowid_ptr[0]);
            dctx->no_rowid_ptr_len[i] = SIZ(dctx->no_rowid_ptr[0]);

            dctx->w_id[i] = w_id;

```

```

    dctx->d_id[i] = i+1;
    dctx->carrier_id[i] = o_carrier_id;
    memcpy(&dctx->del_date[i],&cr_date,sizeof(OCIDate));
}

#ifdef DMLRETDEL /* VMM 1/13/98 */
    memset(acts,(char)0,sizeof(amtctx));
#endif /* DMLRETDEL */
/* array select from new_order and orders tables */

    execstatus=OCISmtExecute(tpcsvc,dctx-
>curd1,errhp,NDISTS,0,0,OCI_DEFAULT);
    if((execstatus != OCI_SUCCESS) && (execstatus != OCI_NO_DATA)) {
        OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
        errcode = OCIERROR(errhp,execstatus);
        if(errcode == NOT_SERIALIZABLE) {
            retries++;
            goto retry;
        } else if (errcode == RECOVER) {
            retries++;
            goto retry;
        } else {
            return -1;
        }
    }
/* mark districts with no new order */
    OCIAttrGet(dctx-
>curd1,OCI_HTYPE_STMT,&rcount,0,OCI_ATTR_ROW_COUNT,errhp);
    rpc = rcount;
#ifdef DMLRETDEL /* we have to compress the array here */
    if (rcount != NDISTS )
    {
        int j = 0;
        for (i=0;i < NDISTS; i++)
        {
            if (dctx->del_o_id_ind[j] == 0) /* there is data here */
                j++;
            else
                shiftdata(j);
        }
    }
#else
    invalid = NDISTS - rcount;
    for (i = rpc; i < NDISTS; i++) {
        dctx->del_o_id_ind[i] = NA;
        dctx->w_id_ind[i] = NA;
        dctx->d_id_ind[i] = NA;
        dctx->c_id_ind[i] = NA;
        dctx->carrier_id_ind[i] = NA;
        dctx->no_rowid_ind[i] = NA;
        dctx->o_rowid_ind[i] = NA;
    }
#endif

#ifdef defined(ISO) || defined(ISO5) || defined(ISO6) || defined(ISO8)
    if (invalid) {
        sysdate (sdate);
        for (i = 1; i <= NDISTS; i++) {
            hasno = 0;
            for (j = 0; j < rpc; j++) {
                if (dctx->d_id[j] == i) {
                    hasno = 1;
                    break;
                }
            }
        }
    }
#endif

```

```

    }
    if (!hasno)
        printf ("Delivery [dist %d] found no new order at %s\n", i,
sdate);
}
if (reread) {
    sleep (60);
    sysdate (sdate);
    printf ("Delivery wake up at %s\n", sdate);
    reread = 0;
    goto iso;
}
}
#endif

#ifdef DMLRETDEL
/* array delete of new_order table */
    execstatus=OCISmtExecute(tpcsvc,dctx-
>curd2,errhp,rpc,0,0,OCI_DEFAULT);
    if(execstatus != OCI_SUCCESS) {
        OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
        errcode = OCIERROR(errhp,execstatus);
        if(errcode == NOT_SERIALIZABLE) {
            retries++;
            goto retry;
        } else if (errcode == RECOVER) {
            retries++;
            goto retry;
        } else {
            return -1;
        }
    }

/* mark districts with no new order */
    OCIAttrGet(dctx-
>curd2,OCI_HTYPE_STMT,&rcount,0,OCI_ATTR_ROW_COUNT,errhp);

    if (rcount != rpc) {
#ifdef TUX
        userlog ("Error in TPC-C server %d: %d rows selected, %d rows
deleted\n",
                proc_no, rpc, dctx->curd2.rpc);
#else
        fprintf (stderr,
                "Error in TPC-C server %d: %d rows selected, %d rows
deleted\n",
                proc_no, rpc, rcount);
#endif /* TUX */
        OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
        return (DEL_ERROR);
    }
#endif /* DMLRETDEL */

    execstatus=OCISmtExecute(tpcsvc,dctx-
>curd3,errhp,rpc,0,0,OCI_DEFAULT);
    if(execstatus != OCI_SUCCESS) {
        OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
        errcode = OCIERROR(errhp,execstatus);
        if(errcode == NOT_SERIALIZABLE) {
            retries++;
            goto retry;
        } else if (errcode == RECOVER) {
            retries++;
            goto retry;
        }
    }
}

```

```

    } else {
        return -1;
    }
}

OCIAttrGet(dctx-
>curd3,OCI_HTYPE_STMT,&rcount,0,OCI_ATTR_ROW_COUNT,errhp);

if (rcount != rpc) {
#ifdef TUX
    userlog ("Error in TPC-C server %d: %d rows selected, %d ords
updated\n",
        proc_no, rpc, rcount);
#else
    fprintf (stderr,
        "Error in TPC-C server %d: %d rows selected, %d ords
updated\n",
        proc_no, rpc, rcount);
#endif
    OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
    return (-1);
}

/* array update of order_line table */
execstatus=OCIStmtExecute(tpcsvc,dctx-
>curd4,errhp,rc,0,0,OCI_DEFAULT);
if(execstatus != OCI_SUCCESS) {
    OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
    errcode = OCIERROR(errhp,execstatus);
    if(errcode == NOT_SERIALIZABLE) {
        retries++;
        goto retry;
    } else if (errcode == RECOVER) {
        retries++;
        goto retry;
    } else {
        return -1;
    }
}
#ifdef DMLRETDL
OCIAttrGet(dctx-
>curd4,OCI_HTYPE_STMT,&rcount,NULL,OCI_ATTR_ROW_COUNT,errhp);
/* add up amounts */
count=0;
for (i=0;i<rpc;i++)
{
    dctx->amt[i]=0;
    for (j=0;j<actx->ol_cnt[i];j++)
        if ( actx->ol_amt_rcode[i][j] == 0)
        {
            dctx->amt[i] = dctx->amt[i] + actx->ol_amt[i][j];
            count = count+1;
        }
}
if (rcount > rpc*NITEMS) {
    userlog ("Error in TPC-C server %d: %d ordnrs updated, %d ordl
updated\n",
        proc_no, rpc, rcount);
}
#else
/* array select from order_line table */
execstatus=OCIStmtExecute(tpcsvc,dctx-
>curd5,errhp,rc,0,0,OCI_DEFAULT);
if((execstatus != OCI_SUCCESS) && (execstatus != OCI_NO_DATA)) {

```

```

OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
errcode = OCIERROR(errhp,execstatus);
if(errcode == NOT_SERIALIZABLE) {
    retries++;
    goto retry;
} else if (errcode == RECOVER) {
    retries++;
    goto retry;
} else {
    return -1;
}
}

OCIAttrGet(dctx-
>curd5,OCI_HTYPE_STMT,&rcount,0,OCI_ATTR_ROW_COUNT,errhp);
if (rcount != rpc) {
#ifdef TUX
    userlog ("Error in TPC-C server %d: %d rows selected, %d ordl
selected\n",
        proc_no, rpc, rcount);
#else
    fprintf (stderr,
        "Error in TPC-C server %d: %d rows selected, %d ordl
selected\n",
        proc_no, rpc, rcount);
#endif
    OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
    return (-1);
}

/* reorder amount selected if necessary */

for (i = 0; i < rpc; i++) {
    if (dctx->cons[i] != dctx->d_id[i]) {
#ifdef TUX
        userlog ("TPC-C server %d: reordering amount\n", proc_no);
#else
        fprintf (stderr, "TPC-C server %d: reordering amount\n",
proc_no);
#endif
        for (j = i + 1; j < rpc; j++) {
            if (dctx->cons[j] == dctx->d_id[i]) {
                tmp_id = dctx->cons[i];
                dctx->cons[i] = dctx->cons[j];
                dctx->cons[j] = tmp_id;
                tmp_amt = dctx->amt[i];
                dctx->amt[i] = dctx->amt[j];
                dctx->amt[j] = tmp_amt;
                break;
            }
        }
        if (j >= rpc) {
#ifdef TUX
            userlog ("Error in TPC-C server %d: missing ordl?\n",
proc_no);
#else
            fprintf (stderr,
                "Error in TPC-C server %d: missing ordl?\n",
proc no);
#endif
        }
        OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
        return (-1);
    }
}
}

```

```

    }
#endif
#if defined(ISO5) || defined(ISO6)
    printf ("d id:amount\n");
    for (i = 0; i < rpc; i++)
        printf ("%d:%.2f ", dctx->d_id[i], (float)dctx->amt[i]/100);
    printf ("\n");
#endif

    /* array update of customer table */
#if defined(ISO5) || defined(ISO6)
    execstatus=OCIStmtExecute(tpcsvc,dctx->curd6,errhp,rpc,0,0,0,
        OCI_DEFAULT);
#else
    execstatus=OCIStmtExecute(tpcsvc,dctx->curd6,errhp,rpc,0,0,0,
        OCI_COMMIT_ON_SUCCESS | OCI_DEFAULT);
#endif

    if(execstatus != OCI_SUCCESS) {
        OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
        errcode = OCIERROR(errhp,execstatus);
        if(errcode == NOT_SERIALIZABLE) {
            retries++;
            goto retry;
        } else if (errcode == RECOVER) {
            retries++;
            goto retry;
        } else {
            return -1;
        }
    }

    OCIAttrGet(dctx-
>curd6,OCI_HTYPE_STMT,&rcount,0,OCI_ATTR_ROW_COUNT,errhp);

    if (rcount != rpc) {
#ifdef TUX
        userlog ("Error in TPC-C server %d: %d rows selected, %d cust
updated\n",
            proc_no, rpc, rcount);
#else
        fprintf (stderr,
            "Error in TPC-C server %d: %d rows selected, %d cust
updated\n",
            proc_no, rpc, rcount);
#endif
        OCITransRollback(tpcsvc, errhp, OCI_DEFAULT);
        return (-1);
    }

#if defined(ISO5) || defined(ISO6)
    sysdate (sdate);
#endif
#ifdef ISO5
    printf ("Delivery sleep before commit at %s\n", sdate);
#else
    printf ("Delivery sleep before abort at %s\n", sdate);
#endif
    sleep (60);
    sysdate (sdate);
    printf ("Delivery wake up at %s\n", sdate);
#endif

#ifdef ISO6
    printf("Delivery ISO6 Rolling back.\n");

```

```

        OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
#endif

#ifdef ISO5
    OCITransCommit(tpcsvc, errhp, OCI_DEFAULT);
#endif

#if defined(ISO5) || defined(ISO6)
    sysdate (sdate);
    printf ("Delivery completed at: %s\n", sdate);
#endif

    /* return o_id's in district id order */

    for (i = 0; i < NDISTS; i++)
        del_o_id[i] = 0;
    for (i = 0; i < rpc; i++)
        del_o_id[dctx->d_id[i] - 1] = dctx->del_o_id[i];
#endif

    return (0);
}

void tkvcddone ()
{
    if (dctx)
    {
#ifdef ISO5 || defined(ISO6) || defined(ISO8)
        OCIHandleFree((dvoid *)dctx->curd0,OCI_HTYPE_STMT);
#endif
#ifdef PLSQLEL
        OCIHandleFree((dvoid *)dctx->curd0,OCI_HTYPE_STMT);
#else
        /* Again the above will cause a problem if both PLSQLEL and ISO
are
        defined - VMM 12/30/97 */
        OCIHandleFree((dvoid *)dctx->curd1,OCI_HTYPE_STMT);
        OCIHandleFree((dvoid *)dctx->curd2,OCI_HTYPE_STMT);
        OCIHandleFree((dvoid *)dctx->curd3,OCI_HTYPE_STMT);
        OCIHandleFree((dvoid *)dctx->curd4,OCI_HTYPE_STMT);
        OCIHandleFree((dvoid *)dctx->curd5,OCI_HTYPE_STMT);
        OCIHandleFree((dvoid *)dctx->curd6,OCI_HTYPE_STMT);
#endif
        free (dctx);
    }
}

```

PLDEL.C

```

#ifdef RCSID
static char *RCSid =
    "$Header: pldel.c 7030100.5 96/06/24 16:26:06 plai Generic<base>
$ Copyr (c) 1994 Oracle";
#endif /* RCSID */

/*=====+

```

```

|          Copyright (c) 1996 Oracle Corp, Redwood Shores, CA          |
|          OPEN SYSTEMS PERFORMANCE GROUP                              |
|          All Rights Reserved                                         |
+-----+
| FILENAME                                                              |
|   pldel.c                                                            |
| DESCRIPTION                                                            |
|   OCI version of DELIVERY transaction in TPC-C benchmark.            |
+-----+
#include "tpcc.h"
#ifndef TUX
#include <userlog.h>
#endif

/*
extern int userlog();
*/

#include "tpccflags.h"

#if defined(ISO) || defined(ISO5) || defined(ISO6) || defined(ISO8)
#define SQLTXT0 "SELECT substr(value,1,5) FROM v$parameter \
WHERE name = 'instance_number'"
#endif

#ifndef PLSQLDEL
#define SQLTXT "BEGIN delivery.deliver (:w_id, :carrier_id, :order_id, \
:retry); END;"
#else
# ifdef DMLRETDDEL
#define SQLTXT1 "DELETE FROM nord WHERE no_d_id = :d_id \
AND no_w_id = :w_id and rownum <= 1 \
RETURNING no_o_id into :o_id "
# else
#define SQLTXT1A "\
SELECT /*+ USE_NL(nord ord) ORDERED */ 1, no_o_id, nord.rowid, o_c_id, \
ordr.rowid \
FROM nord, ord \
WHERE no_w_id = :w_id AND no_d_id = 1 AND o_w_id = :w_id AND o_d_id = 1 \
AND \
o_id = no_o_id AND rownum <= 1 UNION ALL \
"
#define SQLTXT1B "\
SELECT /*+ USE_NL(nord ord) ORDERED */ 2, no_o_id, nord.rowid, o_c_id, \
ordr.rowid \
FROM nord, ord \
WHERE no_w_id = :w_id AND no_d_id = 2 AND o_w_id = :w_id AND o_d_id = 2 \
AND \
o_id = no_o_id AND rownum <= 1 UNION ALL \
"
#define SQLTXT1C "\
SELECT /*+ USE_NL(nord ord) ORDERED */ 3, no_o_id, nord.rowid, o_c_id, \
ordr.rowid \
FROM nord, ord \
WHERE no_w_id = :w_id AND no_d_id = 3 AND o_w_id = :w_id AND o_d_id = 3 \
AND \
o_id = no_o_id AND rownum <= 1 UNION ALL \
"
#define SQLTXT1D "\

```

```

SELECT /*+ USE_NL(nord ord) ORDERED */ 4, no_o_id, nord.rowid, o_c_id, \
ordr.rowid \
FROM nord, ord \
WHERE no_w_id = :w_id AND no_d_id = 4 AND o_w_id = :w_id AND o_d_id = 4 \
AND \
o_id = no_o_id AND rownum <= 1 UNION ALL \
"
#define SQLTXT1E "\
SELECT /*+ USE_NL(nord ord) ORDERED */ 5, no_o_id, nord.rowid, o_c_id, \
ordr.rowid \
FROM nord, ord \
WHERE no_w_id = :w_id AND no_d_id = 5 AND o_w_id = :w_id AND o_d_id = 5 \
AND \
o_id = no_o_id AND rownum <= 1 UNION ALL \
"
#define SQLTXT1F "\
SELECT /*+ USE_NL(nord ord) ORDERED */ 6, no_o_id, nord.rowid, o_c_id, \
ordr.rowid \
FROM nord, ord \
WHERE no_w_id = :w_id AND no_d_id = 6 AND o_w_id = :w_id AND o_d_id = 6 \
AND \
o_id = no_o_id AND rownum <= 1 UNION ALL \
"
#define SQLTXT1G "\
SELECT /*+ USE_NL(nord ord) ORDERED */ 7, no_o_id, nord.rowid, o_c_id, \
ordr.rowid \
FROM nord, ord \
WHERE no_w_id = :w_id AND no_d_id = 7 AND o_w_id = :w_id AND o_d_id = 7 \
AND \
o_id = no_o_id AND rownum <= 1 UNION ALL \
"
#define SQLTXT1H "\
SELECT /*+ USE_NL(nord ord) ORDERED */ 8, no_o_id, nord.rowid, o_c_id, \
ordr.rowid \
FROM nord, ord \
WHERE no_w_id = :w_id AND no_d_id = 8 AND o_w_id = :w_id AND o_d_id = 8 \
AND \
o_id = no_o_id AND rownum <= 1 UNION ALL \
"
#define SQLTXT1I "\
SELECT /*+ USE_NL(nord ord) ORDERED */ 9, no_o_id, nord.rowid, o_c_id, \
ordr.rowid \
FROM nord, ord \
WHERE no_w_id = :w_id AND no_d_id = 9 AND o_w_id = :w_id AND o_d_id = 9 \
AND \
o_id = no_o_id AND rownum <= 1 UNION ALL \
"
#define SQLTXT1J "\
SELECT /*+ USE_NL(nord ord) ORDERED */ 10, no_o_id, nord.rowid, o_c_id, \
ordr.rowid \
FROM nord, ord \
WHERE no_w_id = :w_id AND no_d_id = 10 AND o_w_id = :w_id AND o_d_id = \
10 AND \
o_id = no_o_id AND rownum <= 1"

#define SQLTXT2 "DELETE FROM nord WHERE rowid = :no_rowid"
#endif

```

```

#ifdef DMLRETDDEL
#define SQLTXT3 "UPDATE ordr SET o_carrier_id = :carrier_id \
                WHERE o_id = :o_id and o_d_id = :d_id and o_w_id
= :w_id \
                returning o_c_id into :o_c_id"
#else
#define SQLTXT3 "UPDATE ordr SET o_carrier_id = :carrier_id \
                WHERE rowid = :o_rowid"
#endif

#ifdef DMLRETDDEL
#define SQLTXT4 "UPDATE /*+ buffer */ ordl SET ol_delivery_d = :cr_date \
                WHERE ol_w_id = :w_id AND ol_d_id = :d_id AND ol_o_id = :o_id \
                RETURNING ol_amount into :ol_amount "
#else
#define SQLTXT4 "UPDATE ordl SET ol_delivery_d = :cr date \
                WHERE ol_w_id = :w_id AND ol_d_id = :d_id AND ol_o_id = :o_id"

#define SQLTXT5A "\
SELECT :d_id1, SUM(ol_amount) FROM ordl WHERE ol_w_id = :w_id AND \
        ol_d_id = :d_id1 AND ol_o_id = :o_id1 UNION ALL \
SELECT :d_id2, SUM(ol_amount) FROM ordl WHERE ol_w_id = :w_id AND \
        ol_d_id = :d_id2 AND ol_o_id = :o_id2 UNION ALL \
"

#define SQLTXT5B "\
SELECT :d_id3, SUM(ol_amount) FROM ordl WHERE ol_w_id = :w_id AND \
        ol_d_id = :d_id3 AND ol_o_id = :o_id3 UNION ALL \
SELECT :d_id4, SUM(ol_amount) FROM ordl WHERE ol_w_id = :w_id AND \
        ol_d_id = :d_id4 AND ol_o_id = :o_id4 UNION ALL \
"

#define SQLTXT5C "\
SELECT :d_id5, SUM(ol_amount) FROM ordl WHERE ol_w_id = :w_id AND \
        ol_d_id = :d_id5 AND ol_o_id = :o_id5 UNION ALL \
SELECT :d_id6, SUM(ol_amount) FROM ordl WHERE ol_w_id = :w_id AND \
        ol_d_id = :d_id6 AND ol_o_id = :o_id6 UNION ALL \
"

#define SQLTXT5D "\
SELECT :d_id7, SUM(ol_amount) FROM ordl WHERE ol_w_id = :w_id AND \
        ol_d_id = :d_id7 AND ol_o_id = :o_id7 UNION ALL \
SELECT :d_id8, SUM(ol_amount) FROM ordl WHERE ol_w_id = :w_id AND \
        ol_d_id = :d_id8 AND ol_o_id = :o_id8 UNION ALL \
"

#define SQLTXT5E "\
SELECT :d_id9, SUM(ol_amount) FROM ordl WHERE ol_w_id = :w_id AND \
        ol_d_id = :d_id9 AND ol_o_id = :o_id9 UNION ALL \
SELECT :d_id10, SUM(ol_amount) FROM ordl WHERE ol_w_id = :w_id AND \
        ol_d_id = :d_id10 AND ol_o_id = :o_id10"

#endif
#endif /* PLSQDEL */

#define SQLTXT6 "UPDATE cust SET c_balance = c_balance + :amt, \
                c_delivery_cnt = c_delivery_cnt + 1 WHERE c_w_id = :w_id AND \
                c_d_id = :d_id AND c_id = :c_id"

#define NDISTS 10
#define ROWIDLEN 20

```

```

struct delctx {
    sb2 del_o_id_ind[NDISTS];
    sb2 cons_ind[NDISTS];
    sb2 w_id_ind[NDISTS];
    sb2 d_id_ind[NDISTS];
    sb2 c_id_ind[NDISTS];
    sb2 del_date_ind[NDISTS];
    sb2 carrier_id_ind[NDISTS];
    sb2 amt_ind[NDISTS];
    sb2 no_rowid_ind[NDISTS];
    sb2 o_rowid_ind[NDISTS];
    #if defined(ISO) || defined(ISO5) || defined(ISO6) || defined(ISO8)
    sb2 inum_ind;
    #endif

#ifdef DMLRETDDEL
    ub4 del_o_id_len[NDISTS];
    ub4 c_id_len[NDISTS];
    int o_id_ctx;
    int cid_ctx;
    OCIBind *olamt_bp;
#else
    ub2 del_o_id_len[NDISTS];
    ub2 c_id_len[NDISTS];
    #endif

    ub2 cons_len[NDISTS];
    ub2 w_id_len[NDISTS];
    ub2 d_id_len[NDISTS];
    ub2 del_date_len[NDISTS];
    ub2 carrier_id_len[NDISTS];
    ub2 amt_len[NDISTS];
    ub2 no_rowid_len[NDISTS];
    ub2 no_rowid_ptr_len[NDISTS];
    ub2 o_rowid_len[NDISTS];
    ub2 o_rowid_ptr_len[NDISTS];
    #if defined(ISO) || defined(ISO5) || defined(ISO6) || defined(ISO8)
    ub2 inum_len;
    #endif

    ub2 del_o_id_rcode[NDISTS];
    ub2 cons_rcode[NDISTS];
    ub2 w_id_rcode[NDISTS];
    ub2 d_id_rcode[NDISTS];
    ub2 c_id_rcode[NDISTS];
    ub2 del_date_rcode[NDISTS];
    ub2 carrier_id_rcode[NDISTS];
    ub2 amt_rcode[NDISTS];
    ub2 no_rowid_rcode[NDISTS];
    ub2 o_rowid_rcode[NDISTS];
    #if defined(ISO) || defined(ISO5) || defined(ISO6) || defined(ISO8)
    ub2 inum_rcode;
    #endif

    int del_o_id[NDISTS];
    int cons[NDISTS];
    int w_id[NDISTS];
    int d_id[NDISTS];
    int c_id[NDISTS];
    int carrier_id[NDISTS];
    int amt[NDISTS];
    ub4 del_o_id_rcnt;
    int retry;
    OCIRowid *no_rowid_ptr[NDISTS];
}

```

```

    OCIRowid *o_rowid_ptr[NDISTS];
    OCIDate del_date[NDISTS];
    #if defined(ISO) || defined(ISO5) || defined(ISO6) || defined(ISO8)
    char inum[10];
    #endif
    OCISmt *curd0;
    OCISmt *curd1;
    OCISmt *curd2;
    OCISmt *curd3;
    OCISmt *curd4;
    OCISmt *curd5;
    OCISmt *curd6;
    OCISmt *curdtest;

    OCIBind *w_id_bp;
    OCIBind *w_id_bp3;
    OCIBind *w_id_bp4;
    OCIBind *w_id_bp5;
    OCIBind *w_id_bp6;
    OCIBind *d_id_bp;
    OCIBind *d_id_bp3;
    OCIBind *d_id_bp4;
    OCIBind *d_id_bp6;
    OCIBind *o_id_bp;
    OCIBind *cr_date_bp;
    OCIBind *c_id_bp;
    OCIBind *c_id_bp3;
    OCIBind *no_rowid_bp;
    OCIBind *carrier_id_bp;
    OCIBind *o_rowid_bp;
    OCIBind *del_o_id_bp;
    OCIBind *del_o_id_bp3;
    OCIBind *amt_bp;
    OCIBind *bstr1_bp[10];
    OCIBind *bstr2_bp[10];
    OCIBind *retry_bp;
    OCIDefine *inum_dp;
    OCIDefine *d_id_dp;
    OCIDefine *del_o_id_dp;
    OCIDefine *no_rowid_dp;
    OCIDefine *c_id_dp;
    OCIDefine *o_rowid_dp;
    OCIDefine *cons_dp;
    OCIDefine *amt_dp;

    int norow;
};

typedef struct delctx delctx;

delctx *dctx;

#ifdef DMLRETDEL
struct amtctx {
    int ol_amt[NDISTS][NITEMS];
    sb2 ol_amt_ind[NDISTS][NITEMS];
    ub4 ol_amt_len[NDISTS][NITEMS];
    ub2 ol_amt_rcode[NDISTS][NITEMS];
    int ol_cnt[NDISTS];
};
typedef struct amtctx amtctx;
amtctx *actx;
#endif

```

```

#ifdef DMLRETDEL
extern sb4 no_data();

sb4 TPC_oid_data(dvoid *ctxp, OCIBind *bp, ub4 iter, ub4 index,
                dvoid **bufpp, ub4 **alenp, ub1 *piecep,
                dvoid **indpp, ub2 **rcodepp)
{
    *bufpp = &dctx->del_o_id[iter];
    *indpp = &dctx->del_o_id_ind[iter];
    dctx->del_o_id_len[iter]=sizeof(dctx->del_o_id[0]);
    *alenp = &dctx->del_o_id_len[iter];
    *rcodepp = &dctx->del_o_id_rcode[iter];
    *piecep = OCI_ONE_PIECE;
    return (OCI_CONTINUE);
}

sb4 cid_data(dvoid *ctxp, OCIBind *bp, ub4 iter, ub4 index,
            dvoid **bufpp, ub4 **alenp, ub1 *piecep,
            dvoid **indpp, ub2 **rcodepp)
{
    *bufpp = &dctx->c_id[iter];
    *indpp = &dctx->c_id_ind[iter];
    dctx->c_id_len[iter]=sizeof(dctx->c_id[0]);
    *alenp = &dctx->c_id_len[iter];
    *rcodepp = &dctx->c_id_rcode[iter];
    *piecep = OCI_ONE_PIECE;
    return (OCI_CONTINUE);
}

sb4 amt_data(dvoid *ctxp, OCIBind *bp, ub4 iter, ub4 index,
            dvoid **bufpp, ub4 **alenp, ub1 *piecep,
            dvoid **indpp, ub2 **rcodepp)
{
    amtctx *actx;
    actx = (amtctx*)ctxp;
    actx->ol_cnt[iter]=actx->ol_cnt[iter]+1;
    *bufpp = &actx->ol_amt[iter][index];
    *indpp = &actx->ol_amt_ind[iter][index];
    actx->ol_amt_len[iter][index]=sizeof(actx->ol_amt[0][0]);
    *alenp = &actx->ol_amt_len[iter][index];
    *rcodepp = &actx->ol_amt_rcode[iter][index];
    *piecep = OCI_ONE_PIECE;
    return (OCI_CONTINUE);
}

#endif

tkvcldinit ()
{
    #ifndef DMLRETDEL
    int i,j;
    char bstr1[10];
    char bstr2[10];
    #endif /* !DMLRETDEL */
    text stmbuf[SQL_BUF_SIZE];

    dctx = (delctx *) malloc (sizeof(delctx));
    memset (dctx, (char)0, sizeof(delctx));
    dctx->norow = 0;
}

```

```

#ifdef DMLRETDDEL
    actx = (amtctx *) malloc (sizeof(amtctx));
    memset (actx, (char)0, sizeof(amtctx));
#else
    for(i=0;i<NDISTS;i++) {
        OCIERROR (errhp, OCIDescriptorAlloc (tpcenv, (dvoid**) &dctx-
>o_rowid_ptr[i],
            OCI_DTYPE_ROWID, 0, (dvoid**) 0));
        OCIERROR (errhp, OCIDescriptorAlloc (tpcenv, (dvoid**) &dctx-
>no_rowid_ptr[i],
            OCI_DTYPE_ROWID, 0, (dvoid**) 0));
    }
#endif

#if defined(ISO) || defined(ISO5) || defined(ISO6) || defined(ISO8)
    OCIHandleAlloc (tpcenv, (dvoid **) (&dctx->curd0), OCI_HTYPE_STMT, 0,
(dvoid**) 0);
    sprintf ((char *) stmbuf, SQLTXT0);
    OCIStmtPrepare (dctx->curd0, errhp, stmbuf, strlen((char
*) stmbuf), OCI_NTV_SYNTAX, OCI_DEFAULT);

    OCIDFNRA (dctx->curd0, dctx->inum_dp, errhp, 1, dctx->inum, SIZ (dctx-
>inum), SQLT_STR,
        &(dctx->inum_ind), &(dctx->inum_len), &(dctx->inum_rcode));
#endif

/* If PLSQLDEL and ISO? are both defined, then they both try to use
curd0! This could cause a problem. Will try to fix later - VMM
12/30/97 */

#ifdef PLSQLDEL
    OCIHandleAlloc (tpcenv, (dvoid **) (&dctx->curd0), OCI_HTYPE_STMT,
0, (dvoid**) 0);
    sprintf ((char *) stmbuf, SQLTXT);
    OCIStmtPrepare (dctx->curd0, errhp, stmbuf, strlen((char *) stmbuf),
OCI_NTV_SYNTAX, OCI_DEFAULT);
    OCIBND (dctx->curd0, dctx->w_id_bp, errhp, "w_id", ADR (w_id), SIZ (int),
SQLT_INT);
    OCIBND (dctx->curd0, dctx->carrier_id_bp, errhp, "carrier_id",
ADR (dctx->carrier_id), SIZ (int), SQLT_INT);

    OCIBNDRAA (dctx->curd0, dctx->o_id_bp, errhp, "order_id",
dctx->del_o_id, SIZ (int), SQLT_INT, dctx->del_o_id_ind,
dctx->del_o_id_len, dctx->del_o_id_rcode, NDISTS,
&dctx->del_o_id_rcnt);
    OCIBND (dctx->curd0, dctx->retry_bp, errhp, "retry", ADR (dctx->retry),
SIZ (int), SQLT_INT);
#else
#ifdef DMLRETDDEL
    OCIHandleAlloc (tpcenv, (dvoid **) (&dctx->curd1), OCI_HTYPE_STMT, 0,
(dvoid**) 0);
    sprintf ((char *) stmbuf, "%s", SQLTXT1);
    OCIStmtPrepare (dctx->curd1, errhp, stmbuf, strlen((char
*) stmbuf), OCI_NTV_SYNTAX, OCI_DEFAULT);

    OCIBND (dctx->curd1, dctx->w_id_bp, errhp, "w_id", dctx->w_id, SIZ (int),
SQLT_INT);
    OCIBNDRA (dctx->curd1, dctx->d_id_bp, errhp, "d_id", dctx-
>d_id, SIZ (int),
SQLT_INT, NULL, NULL, NULL);

    OCIBNDRAD (dctx->curd1, dctx->del_o_id_bp, errhp, "o_id",
SIZ (int), SQLT_INT, NULL,
&dctx->oid_ctx, no_data, TPC_oid_data);

```

```

# else

    OCIHandleAlloc (tpcenv, (dvoid **) (&dctx->curd1), OCI_HTYPE_STMT, 0,
(dvoid**) 0);
    sprintf ((char *) stmbuf, "%s%s%s%s%s%s%s%s%s", SQLTXT1A,
SQLTXT1B,
SQLTXT1C,
SQLTXT1D,
SQLTXT1E,
SQLTXT1F,
SQLTXT1G,
SQLTXT1H,
SQLTXT1I,
SQLTXT1J

    );
    OCIStmtPrepare (dctx->curd1, errhp, stmbuf, strlen((char
*) stmbuf), OCI_NTV_SYNTAX, OCI_DEFAULT);

    OCIERROR (errhp,
OCIAttrSet (dctx->curd1, OCI_HTYPE_STMT, (dvoid *) &dctx->norow, 0,
OCI_ATTR_PREFETCH_ROWS, errhp));

    /* bind variables */

    OCIBND (dctx->curd1, dctx-
>w_id_bp, errhp, "w_id", ADR (w_id), SIZ (int), SQLT_INT);

    OCIDFNRA (dctx->curd1, dctx->d_id_dp, errhp, 1, dctx->d_id, SIZ (int),
SQLT_INT, dctx->d_id_ind, dctx->d_id_len, dctx->d_id_rcode);
    OCIDFNRA (dctx->curd1, dctx->del_o_id_dp, errhp, 2, dctx->del_o_id,
SIZ (int), SQLT_INT, dctx->del_o_id_ind,
dctx->del_o_id_len, dctx->del_o_id_rcode);
    OCIDFNRA (dctx->curd1, dctx->no_rowid_dp, errhp, 3, dctx->no_rowid_ptr,
SIZ (OCIRowid *), SQLT_RDD, dctx->no_rowid_ind,
dctx->no_rowid_len, dctx->no_rowid_rcode);
    OCIDFNRA (dctx->curd1, dctx->c_id_dp, errhp, 4, dctx->c_id, SIZ (dctx-
>c_id[0]),
SQLT_INT, dctx->c_id_ind, dctx->c_id_len, dctx->c_id_rcode);
    OCIDFNRA (dctx->curd1, dctx->o_rowid_dp, errhp, 5, dctx->o_rowid_ptr,
SIZ (OCIRowid *), SQLT_RDD, dctx->o_rowid_ind,
dctx->o_rowid_len, dctx->o_rowid_rcode);

    /* open second cursor */

    OCIHandleAlloc (tpcenv, (dvoid **) (&dctx->curd2), OCI_HTYPE_STMT, 0,
(dvoid**) 0);
    sprintf ((char *) stmbuf, SQLTXT2);
    OCIStmtPrepare (dctx->curd2, errhp, stmbuf, strlen((char *) stmbuf),
OCI_NTV_SYNTAX, OCI_DEFAULT);

    /* bind variables */
    OCIBNDRA (dctx->curd2, dctx->no_rowid_bp, errhp, "no_rowid", &(dctx-
>no_rowid_ptr[0]),
SIZ (dctx->no_rowid_ptr[0]), SQLT_RDD, dctx->no_rowid_ind,
dctx->no_rowid_len, dctx->no_rowid_rcode);

# endif /*DMLRETDDEL*/

    /* open third cursor */

    OCIHandleAlloc (tpcenv, (dvoid **) (&dctx->curd3), OCI_HTYPE_STMT, 0,
(dvoid**) 0);

```

```

sprintf ((char *) stmbuf, SQLTXT3);
OCIStmtPrepare(dctx->curd3, errhp, stmbuf, strlen((char *)stmbuf),
              OCI_NT_V_SYNTAX, OCI_DEFAULT);

/* bind variables */

OCIBNDRA(dctx->curd3, dctx->carrier_id_bp, errhp, ":carrier_id", dctx->
carrier_id,
          SIZ(dctx->carrier_id[0]), SQLT_INT, dctx->carrier_id_ind,
          dctx->carrier_id_len, dctx->carrier_id_rcode);

# ifdef DMLRETDDEL
OCIBNDRA(dctx->curd3, dctx->w_id_bp3, errhp, ":w_id", dctx->
w_id, SIZ(int),
          SQLT_INT, NULL, NULL, NULL);
OCIBNDRA(dctx->curd3, dctx->d_id_bp3, errhp, ":d_id", dctx->
d_id, SIZ(int),
          SQLT_INT, NULL, NULL, NULL);
OCIBNDRA(dctx->curd3, dctx->del_o_id_bp3, errhp, ":o_id", dctx->
del_o_id,
          SIZ(int), SQLT_INT, NULL, NULL, NULL);
OCIBNDRAD(dctx->curd3, dctx->c_id_bp3, errhp, ":o_c_id", SIZ(int),
          SQLT_INT, NULL, &dctx->cid_ctx, no_data, cid_data);
# else

OCIBNDRA(dctx->curd3, dctx->o_rowid_bp, errhp, ":o_rowid", &(dctx->
o_rowid_ptr[0]),
          SIZ(dctx->o_rowid_ptr[0]), SQLT_RDD, dctx->o_rowid_ind,
          dctx->o_rowid_ptr_len, dctx->o_rowid_rcode);
#endif

/* open fourth cursor */

OCIHandleAlloc(tpcenv, (dvoid **)&(dctx->curd4), OCI_HTYPE_STMT, 0,
(dvoid**)0);
sprintf ((char *) stmbuf, SQLTXT4);
OCIStmtPrepare(dctx->curd4, errhp, stmbuf, strlen((char *)stmbuf),
              OCI_NT_V_SYNTAX, OCI_DEFAULT);

/* bind variables */

OCIBND(dctx->curd4, dctx->w_id_bp4, errhp, ":w_id", dctx->w_id,
          SIZ(int), SQLT_INT);
OCIBND(dctx->curd4, dctx->d_id_bp4, errhp, ":d_id", dctx->d_id,
          SIZ(int), SQLT_INT);
OCIBND(dctx->curd4, dctx->o_id_bp, errhp, ":o_id", dctx->del_o_id,
          SIZ(int), SQLT_INT);
OCIBND(dctx->curd4, dctx->cr_date_bp, errhp, ":cr_date", dctx->
del_date,
          SIZ(OCIDate), SQLT_ODT);
# ifdef DMLRETDDEL
OCIBNDRAD(dctx->curd4, dctx->olamt_bp, errhp, ":ol amount",
          SIZ(int), SQLT_INT, NULL, actx, no_data, amt_data);
# else

/* open fifth cursor */

OCIHandleAlloc(tpcenv, (dvoid **)&(dctx->curd5), OCI_HTYPE_STMT, 0,
(dvoid**)0);
sprintf ((char *) stmbuf, "%s%s%s%s", SQLTXT5A,
          SQLTXT5B,

```

```

          SQLTXT5C,
          SQLTXT5D,
          SQLTXT5E
);
OCIStmtPrepare(dctx->curd5, errhp, stmbuf, strlen((char *)stmbuf),
              OCI_NT_V_SYNTAX, OCI_DEFAULT);

OCIERROR(errhp,
          OCIAttrSet(dctx->curd5, OCI_HTYPE_STMT, (dvoid*)&dctx->norow, 0,
          OCI_ATTR_PREFETCH_ROWS, errhp));

/* bind variables */

OCIBND(dctx->curd5, dctx->
w_id_bp, errhp, ":w_id", ADR(w_id), SIZ(w_id), SQLT_INT);
for (i = 0; i < NDISTS; i++) {
    sprintf (bstr1, ":d_id%d", i + 1);
    sprintf (bstr2, ":o_id%d", i + 1);
    OCIBNDRA(dctx->curd5, dctx->bstr1_bp[i], errhp, bstr1, ADR(dctx->
d_id[i]),
              SIZ(dctx->d_id[0]), SQLT_INT, &(dctx->d_id_ind[i]),
              &(dctx->d_id_len[i]), &(dctx->d_id_rcode[i]));
    OCIBNDRA(dctx->curd5, dctx->bstr2_bp[i], errhp, bstr2, ADR(dctx->
del_o_id[i]),
              SIZ(dctx->del_o_id[0]), SQLT_INT, &(dctx->
del_o_id_ind[i]),
              &(dctx->del_o_id_len[i]), &(dctx->del_o_id_rcode[i]));
}

OCIDFNRA(dctx->curd5, dctx->cons_dp, errhp, 1, dctx->cons, SIZ(dctx->
cons[0]), SQLT_INT,
          dctx->cons_ind, dctx->cons_len, dctx->cons_rcode);
OCIDFNRA(dctx->curd5, dctx->amt_dp, errhp, 2, dctx->amt, SIZ(dctx->
amt[0]), SQLT_INT,
          dctx->amt_ind, dctx->amt_len, dctx->amt_rcode);
#endif
/* open sixth cursor */

OCIHandleAlloc(tpcenv, (dvoid **)&(dctx->curd6), OCI_HTYPE_STMT, 0,
(dvoid**)0);
sprintf ((char *) stmbuf, SQLTXT6);
OCIStmtPrepare(dctx->curd6, errhp, stmbuf, strlen((char *)stmbuf),
              OCI_NT_V_SYNTAX, OCI_DEFAULT);

/* bind variables */

OCIBND(dctx->curd6, dctx->amt_bp, errhp, ":amt", dctx->amt, SIZ(int),
          SQLT_INT);
OCIBND(dctx->curd6, dctx->w_id_bp6, errhp, ":w_id", dctx->w_id, SIZ(int),
          SQLT_INT);
OCIBND(dctx->curd6, dctx->d_id_bp6, errhp, ":d_id", dctx->d_id, SIZ(int),
          SQLT_INT);
OCIBND(dctx->curd6, dctx->c_id_bp, errhp, ":c_id", dctx->c_id, SIZ(int),
          SQLT_INT);
#endif
return (0);
}

void shiftdata(from)
int from;
{

```

```

int i;
for (i=from;i<NDISTS-1; i++)
{
    dctx->del_o_id_ind[i] = dctx->del_o_id_ind[i+1];
    dctx->del_o_id[i] = dctx->del_o_id[i+1];
    dctx->w_id[i] = dctx->w_id[i+1];
    dctx->d_id[i] = dctx->d_id[i+1];
    dctx->carrier_id[i] = dctx->carrier_id[i+1];
}
}

tkvcd ()
{
    int i, j;
    int rpc,rcount,count;
    int invalid;
# ifndef DMLRETDEL
    int tmp_id,v;
    int tmp_amt;
# endif /* !DMLRETDEL */

# if defined(ISO) || defined(ISO5) || defined(ISO6) || defined(ISO8)
    int hasno;
    int reread;
    char sdate[30];

    OCISmtExecute(tpcsvc,dctx->curd0,errhp,1,0,0,OCI_DEFAULT);
    sysdate (sdate);
    printf ("Delivery started at %s on node %s\n", sdate, dctx->inum);
# endif
# ifdef PLSQLDEL
    for (i = 0; i < NDISTS; i++)
    {
        dctx->del_o_id_ind[i] = TRUE;
        dctx->del_o_id_len[i] = sizeof(int);
    }

    OCIERROR(errhp,
        OCISmtExecute(tpcsvc,dctx->curd0,errhp,1,0,0,OCI_DEFAULT));

    for (i = 0; i < NDISTS; i++)
    {
        del_o_id[i] = 0;
        if (dctx->del_o_id_ind[i] == 0)
        {
            del_o_id[i] = dctx->del_o_id[i];
        }
    }
# else

retry:

# if defined(ISO) || defined(ISO5) || defined(ISO6) || defined(ISO8)
    reread = 1;
# endif

# if defined(ISO) || defined(ISO5) || defined(ISO6) || defined(ISO8)
iso:
# endif

    invalid = 0;

```

```

/* initialization for array operations */

for (i = 0; i < NDISTS; i++) {
    dctx->del_o_id_ind[i] = TRUE;
    dctx->cons_ind[i] = TRUE;
    dctx->w_id_ind[i] = TRUE;
    dctx->d_id_ind[i] = TRUE;
    dctx->c_id_ind[i] = TRUE;
    dctx->del_date_ind[i] = TRUE;
    dctx->carrier_id_ind[i] = TRUE;
    dctx->amt_ind[i] = TRUE;
    dctx->no_rowid_ind[i] = TRUE;
    dctx->o_rowid_ind[i] = TRUE;

    dctx->del_o_id_len[i] = SIZ(dctx->del_o_id[0]);
    dctx->cons_len[i] = SIZ(dctx->cons[0]);
    dctx->w_id_len[i] = SIZ(dctx->w_id[0]);
    dctx->d_id_len[i] = SIZ(dctx->d_id[0]);
    dctx->c_id_len[i] = SIZ(dctx->c_id[0]);
    dctx->del_date_len[i] = DEL_DATE_LEN;
    dctx->carrier_id_len[i] = SIZ(dctx->carrier_id[0]);
    dctx->amt_len[i] = SIZ(dctx->amt[0]);
    dctx->no_rowid_len[i] = ROWIDLEN;
    dctx->o_rowid_len[i] = ROWIDLEN;
    dctx->o_rowid_ptr_len[i] = SIZ(dctx->o_rowid_ptr[0]);
    dctx->no_rowid_ptr_len[i] = SIZ(dctx->no_rowid_ptr[0]);

    dctx->w_id[i] = w_id;
    dctx->d_id[i] = i+1;
    dctx->carrier_id[i] = o_carrier_id;
    memcpy(&dctx->del_date[i],&cr_date,sizeof(OCIDate));
}

# ifdef DMLRETDEL /* VMM 1/13/98 */
    memset(amtctx, (char)0, sizeof(amtctx));
# endif /* DMLRETDEL */
/* array select from new_order and orders tables */

execstatus=OCISmtExecute(tpcsvc,dctx-
>curd1,errhp,NDISTS,0,0,OCI_DEFAULT);
if((execstatus != OCI_SUCCESS) && (execstatus != OCI_NO_DATA)) {
    OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
    errcode = OCIERROR(errhp,execstatus);
    if(errcode == NOT_SERIALIZABLE) {
        retries++;
        goto retry;
    } else if (errcode == RECOVER) {
        retries++;
        goto retry;
    } else {
        return -1;
    }
}
/* mark districts with no new order */
OCIAttrGet(dctx-
>curd1,OCI_HTYPE_STMT,&rcount,0,OCI_ATTR_ROW_COUNT,errhp);
rpc = rcount;
# ifdef DMLRETDEL /* we have to compress the array here */
    if (rcount != NDISTS )
    {
        int j = 0;
        for (i=0;i < NDISTS; i++)
        {

```

```

        if (dctx->del_o_id_ind[j] == 0) /* there is data here */
            j++;
        else
            shiftdata(j);
    }
}
#else
invalid = NDISTS - rcount;
for (i = rpc; i < NDISTS; i++) {
    dctx->del_o_id_ind[i] = NA;
    dctx->w_id_ind[i] = NA;
    dctx->d_id_ind[i] = NA;
    dctx->c_id_ind[i] = NA;
    dctx->carrier_id_ind[i] = NA;
    dctx->no_rowid_ind[i] = NA;
    dctx->o_rowid_ind[i] = NA;
}
#endif

#if defined(ISO) || defined(ISO5) || defined(ISO6) || defined(ISO8)
if (invalid) {
    sysdate (sdate);
    for (i = 1; i <= NDISTS; i++) {
        hasno = 0;
        for (j = 0; j < rpc; j++) {
            if (dctx->d_id_ind[j] == i) {
                hasno = 1;
                break;
            }
        }
        if (!hasno)
            printf ("Delivery [dist %d] found no new order at %s\n", i,
sdate);
    }
    if (reread) {
        sleep (60);
        sysdate (sdate);
        printf ("Delivery wake up at %s\n", sdate);
        reread = 0;
        goto iso;
    }
}
#endif

#ifdef DMLRETDL
/* array delete of new_order table */
execstatus=OCISstmtExecute(tpcsvc,dctx-
>curd2,errhp,rcp,0,0,0,OCI_DEFAULT);
if(execstatus != OCI_SUCCESS) {
    OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
    errcode = OCIERROR(errhp,execstatus);
    if(errcode == NOT_SERIALIZABLE) {
        retries++;
        goto retry;
    } else if (errcode == RECOVER) {
        retries++;
        goto retry;
    } else {
        return -1;
    }
}

/* mark districts with no new order */

```

```

OCIAttrGet(dctx-
>curd2,OCI_HTYPE_STMT,&rcount,0,OCI_ATTR_ROW_COUNT,errhp);

if (rcount != rpc) {
#ifdef TUX
    userlog ("Error in TPC-C server %d: %d rows selected, %d rows
deleted\n",
        proc_no, rpc, dctx->curd2.rpc);
#else
    fprintf (stderr,
        "Error in TPC-C server %d: %d rows selected, %d rows
deleted\n",
        proc_no, rpc, rcount);
#endif /* TUX */
    OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
    return (DEL_ERROR);
}
#endif /* DMLRETDL */

execstatus=OCISstmtExecute(tpcsvc,dctx-
>curd3,errhp,rcp,0,0,0,OCI_DEFAULT);
if(execstatus != OCI_SUCCESS) {
    OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
    errcode = OCIERROR(errhp,execstatus);
    if(errcode == NOT_SERIALIZABLE) {
        retries++;
        goto retry;
    } else if (errcode == RECOVER) {
        retries++;
        goto retry;
    } else {
        return -1;
    }
}

OCIAttrGet(dctx-
>curd3,OCI_HTYPE_STMT,&rcount,0,OCI_ATTR_ROW_COUNT,errhp);

if (rcount != rpc) {
#ifdef TUX
    userlog ("Error in TPC-C server %d: %d rows selected, %d ords
updated\n",
        proc_no, rpc, rcount);
#else
    fprintf (stderr,
        "Error in TPC-C server %d: %d rows selected, %d ords
updated\n",
        proc_no, rpc, rcount);
#endif
    OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
    return (-1);
}

/* array update of order line table */
execstatus=OCISstmtExecute(tpcsvc,dctx-
>curd4,errhp,rcp,0,0,0,OCI_DEFAULT);
if(execstatus != OCI_SUCCESS) {
    OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
    errcode = OCIERROR(errhp,execstatus);
    if(errcode == NOT_SERIALIZABLE) {
        retries++;
        goto retry;
    } else if (errcode == RECOVER) {
        retries++;

```

```

        goto retry;
    } else {
        return -1;
    }
}
#ifdef DMLRETDEL
OCIAttrGet(dctx-
>curd4,OCI_HTYPE_STMT,&rcount,NULL,OCI_ATTR_ROW_COUNT,errhp);
/* add up amounts */
count=0;
for (i=0;i<rpc;i++)
{
    dctx->amt[i]=0;
    for (j=0;j<actx->ol_cnt[i];j++)
        if ( actx->ol_amt_rcode[i][j] == 0)
        {
            dctx->amt[i] = dctx->amt[i] + actx->ol_amt[i][j];
            count = count+1;
        }
}
if (rcount > rpc*NITEMS) {
    userlog ("Error in TPC-C server %d: %d ordnrs updated, %d ordl
updated\n",
            proc_no, rpc, rcount);
}
#else
/* array select from order_line table */
execstatus=OCIStmtExecute(tpcsvc,dctx-
>curd5,errhp,rc,0,0,OCI_DEFAULT);
if((execstatus != OCI_SUCCESS) && (execstatus != OCI_NO_DATA)) {
    OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
    errcode = OCIERROR(errhp,execstatus);
    if(errcode == NOT_SERIALIZABLE) {
        retries++;
        goto retry;
    } else if (errcode == RECOVER) {
        retries++;
        goto retry;
    } else {
        return -1;
    }
}
OCIAttrGet(dctx-
>curd5,OCI_HTYPE_STMT,&rcount,0,OCI_ATTR_ROW_COUNT,errhp);
if (rcount != rpc) {
#ifdef TUX
    userlog ("Error in TPC-C server %d: %d rows selected, %d ordl
selected\n",
            proc_no, rpc, rcount);
#else
    fprintf (stderr,
            "Error in TPC-C server %d: %d rows selected, %d ordl
selected\n",
            proc_no, rpc, rcount);
#endif
    OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
    return (-1);
}

/* reorder amount selected if necessary */

for (i = 0; i < rpc; i++) {
    if (dctx->cons[i] != dctx->d_id[i]) {

```

```

#ifdef TUX
    userlog ("TPC-C server %d: reordering amount\n", proc_no);
#else
    fprintf (stderr, "TPC-C server %d: reordering amount\n",
proc_no);
#endif
    for (j = i + 1; j < rpc; j++) {
        if (dctx->cons[j] == dctx->d_id[i]) {
            tmp_id = dctx->cons[i];
            dctx->cons[i] = dctx->cons[j];
            dctx->cons[j] = tmp_id;
            tmp_amt = dctx->amt[i];
            dctx->amt[i] = dctx->amt[j];
            dctx->amt[j] = tmp_amt;
            break;
        }
    }
    if (j >= rpc) {
#ifdef TUX
        userlog ("Error in TPC-C server %d: missing ordl?\n",
proc_no);
#else
        fprintf (stderr,
            "Error in TPC-C server %d: missing ordl?\n",
proc_no);
#endif
        OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
        return (-1);
    }
}
}
#endif
#ifdef defined(ISO5) || defined(ISO6)
    printf ("d_id:amount\n");
    for (i = 0; i < rpc; i++)
        printf ("%d:%.2f ", dctx->d_id[i], (float)dctx->amt[i]/100);
    printf ("\n");
#endif

/* array update of customer table */
#ifdef defined(ISO5) || defined (ISO6)
    execstatus=OCIStmtExecute(tpcsvc,dctx->curd6,errhp,rc,0,0,0,
OCI_DEFAULT);
#else
    execstatus=OCIStmtExecute(tpcsvc,dctx->curd6,errhp,rc,0,0,0,
OCI_COMMIT_ON_SUCCESS | OCI_DEFAULT);
#endif

if(execstatus != OCI_SUCCESS) {
    OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
    errcode = OCIERROR(errhp,execstatus);
    if(errcode == NOT_SERIALIZABLE) {
        retries++;
        goto retry;
    } else if (errcode == RECOVER) {
        retries++;
        goto retry;
    } else {
        return -1;
    }
}

OCIAttrGet(dctx-
>curd6,OCI_HTYPE_STMT,&rcount,0,OCI_ATTR_ROW_COUNT,errhp);

```

```

    if (rcount != rpc) {
#ifdef TUX
        userlog ("Error in TPC-C server %d: %d rows selected, %d cust
updated\n",
            proc_no, rpc, rcount);
#else
        fprintf (stderr,
            "Error in TPC-C server %d: %d rows selected, %d cust
updated\n",
            proc_no, rpc, rcount);
#endif
        OCITransRollback(tpcsvc, errhp, OCI_DEFAULT);
        return (-1);
    }

#ifdef IS05 || defined(IS06)
        sysdate (sdate);
#ifdef IS05
            printf ("Delivery sleep before commit at %s\n", sdate);
#else
            printf ("Delivery sleep before abort at %s\n", sdate);
#endif
        sleep (60);
        sysdate (sdate);
        printf ("Delivery wake up at %s\n", sdate);
#endif

#ifdef IS06
        printf("Delivery ISO6 Rolling back.\n");
        OCITransRollback(tpcsvc, errhp, OCI_DEFAULT);
#endif

#ifdef IS05
        OCITransCommit(tpcsvc, errhp, OCI_DEFAULT);
#endif

#ifdef IS05 || defined(IS06)
        sysdate (sdate);
        printf ("Delivery completed at: %s\n", sdate);
#endif

        /* return o_id's in district id order */

        for (i = 0; i < NDISTS; i++)
            del_o_id[i] = 0;
        for (i = 0; i < rpc; i++)
            del_o_id[dctx->d_id[i] - 1] = dctx->del_o_id[i];
#endif

        return (0);
    }

void tkvcdone ()
{
    if (dctx)
    {
#ifdef IS0 || defined(IS05) || defined(IS06) || defined(IS08)
        OCIHandleFree((dvoid *)dctx->curd0, OCI_HTYPE_STMT);
#endif
    }
}

```

```

#endif
#ifdef PLSQLDEL
    OCIHandleFree((dvoid *)dctx->curd0, OCI_HTYPE_STMT);
#else
    /* Again the above will cause a problem if both PSQLDEL and ISO
are
    defined - VMM 12/30/97 */
    OCIHandleFree((dvoid *)dctx->curd1, OCI_HTYPE_STMT);
    OCIHandleFree((dvoid *)dctx->curd2, OCI_HTYPE_STMT);
    OCIHandleFree((dvoid *)dctx->curd3, OCI_HTYPE_STMT);
    OCIHandleFree((dvoid *)dctx->curd4, OCI_HTYPE_STMT);
    OCIHandleFree((dvoid *)dctx->curd5, OCI_HTYPE_STMT);
    OCIHandleFree((dvoid *)dctx->curd6, OCI_HTYPE_STMT);
#endif
    free (dctx);
}
}

```

PLNEW.C

```

#ifdef RCSID
static char *RCSid =
    "$Header: tkvnew.c 21-apr-98.18:32:59 rdecker Exp $ Copyr (c) 1994
Oracle";
#endif /* RCSID */

/*=====+
|          Copyright (c) 1996 , 1997, 1998 Oracle Corp, Redwood Shores,
CA          |
|          OPEN SYSTEMS PERFORMANCE GROUP          |
|          All Rights Reserved          |
+=====+
| FILENAME
| plnew.c
| DESCRIPTION
| OCI version (using PL/SQL stored procedure) of
| NEW ORDER transaction in TPC-C benchmark.
+=====*/

#include "tpcc.h"
#ifdef TUX
#include <userlog.h>
#endif
#include "tpccflags.h"

#ifdef TUX
extern void userlog();
#endif

#ifdef PLSQLNO
#define SQLTXT2 "BEGIN initnew.new_init(:idxlarr); END;"
#else
#define SQLTXT2 "UPDATE stok SET s_order_cnt = s_order_cnt + 1, \
s_ytd = s_ytd + :ol_quantity, s_remote_cnt = s_remote_cnt
+ :s_remote, \
s_quantity = :s_quantity \
WHERE rowid = :s_rowid"
#endif

#define SQLTXT3 "\

```

```

SELECT 0, stok.rowid, i_price, i_name, i_data, s_dist_%02d, s_data, s_quantity
\
FROM item, stok WHERE i_id = :10 AND s_w_id = :30 AND s_i_id = i_id
UNION ALL \
SELECT 1, stok.rowid, i_price, i_name, i_data, s_dist_%02d, s_data, s_quantity
\
FROM item, stok WHERE i_id = :11 AND s_w_id = :31 AND s_i_id = i_id
UNION ALL \
SELECT 2, stok.rowid, i_price, i_name, i_data, s_dist_%02d, s_data, s_quantity
\
FROM item, stok WHERE i_id = :12 AND s_w_id = :32 AND s_i_id = i_id
UNION ALL \
SELECT 3, stok.rowid, i_price, i_name, i_data, s_dist_%02d, s_data, s_quantity
\
FROM item, stok WHERE i_id = :13 AND s_w_id = :33 AND s_i_id = i_id
UNION ALL \
SELECT 4, stok.rowid, i_price, i_name, i_data, s_dist_%02d, s_data, s_quantity
\
FROM item, stok WHERE i_id = :14 AND s_w_id = :34 AND s_i_id = i_id
UNION ALL \
SELECT 5, stok.rowid, i_price, i_name, i_data, s_dist_%02d, s_data, s_quantity
\
FROM item, stok WHERE i_id = :15 AND s_w_id = :35 AND s_i_id = i_id
UNION ALL \
SELECT 6, stok.rowid, i_price, i_name, i_data, s_dist_%02d, s_data, s_quantity
\
FROM item, stok WHERE i_id = :16 AND s_w_id = :36 AND s_i_id = i_id
UNION ALL \
SELECT 7, stok.rowid, i_price, i_name, i_data, s_dist_%02d, s_data, s_quantity
\
FROM item, stok WHERE i_id = :17 AND s_w_id = :37 AND s_i_id = i_id
UNION ALL \
SELECT 8, stok.rowid, i_price, i_name, i_data, s_dist_%02d, s_data, s_quantity
\
FROM item, stok WHERE i_id = :18 AND s_w_id = :38 AND s_i_id = i_id
UNION ALL \
SELECT 9, stok.rowid, i_price, i_name, i_data, s_dist_%02d, s_data, s_quantity
\
FROM item, stok WHERE i_id = :19 AND s_w_id = :39 AND s_i_id = i_id
UNION ALL \
SELECT
10, stok.rowid, i_price, i_name, i_data, s_dist_%02d, s_data, s_quantity \
FROM item, stok WHERE i_id = :20 AND s_w_id = :40 AND s_i_id = i_id
UNION ALL \
SELECT
11, stok.rowid, i_price, i_name, i_data, s_dist_%02d, s_data, s_quantity \
FROM item, stok WHERE i_id = :21 AND s_w_id = :41 AND s_i_id = i_id
UNION ALL \
SELECT
12, stok.rowid, i_price, i_name, i_data, s_dist_%02d, s_data, s_quantity \
FROM item, stok WHERE i_id = :22 AND s_w_id = :42 AND s_i_id = i_id
UNION ALL \
SELECT
13, stok.rowid, i_price, i_name, i_data, s_dist_%02d, s_data, s_quantity \
FROM item, stok WHERE i_id = :23 AND s_w_id = :43 AND s_i_id = i_id
UNION ALL \
SELECT
14, stok.rowid, i_price, i_name, i_data, s_dist_%02d, s_data, s_quantity \
FROM item, stok WHERE i_id = :24 AND s_w_id = :44 AND s_i_id = i_id"

#define SQLTXT4 "INSERT INTO ordl \
(ol_o_id, ol_d_id, ol_w_id, ol_number, ol_delivery_d, ol_i_id, \
ol_supply_w_id, ol_quantity, ol_amount, ol_dist_info)\

```

```

VALUES (:ol_o_id, :ol_d_id, \
:ol_w_id, :ol_number, :null_date, :ol_i_id, :ol_supply_w_id, :ol_qua
ntity, \
:ol_amount, :ol_dist_info)"
#endif /* PLSQLNO */

#define NITEMS 15
#define ROWIDLEN 20
#define OCIROWLEN 20

sb4 no_data(dvoid *ctxp, OCIBind *bp, ub4 iter, ub4 index,
dvoid **bufpp, ub4 *alenp, ub1 *piecep,
dvoid **indpp)
{
*bufpp = (dvoid*)0;
*alenp = 0;
*indpp = (dvoid*)0;
*piecep = OCI_ONE_PIECE;
return (OCI_CONTINUE);
}

struct newctx {
sb2 nol_i_id_ind[NITEMS];
sb2 nol_supply_w_id_ind[NITEMS];
sb2 nol_quantity_ind[NITEMS];
sb2 nol_amount_ind[NITEMS];
sb2 i_name_ind[NITEMS];
sb2 s_quantity_ind[NITEMS];
sb2 i_price_ind[NITEMS];
sb2 ol_w_id_ind[NITEMS];
sb2 ol_d_id_ind[NITEMS];
sb2 ol_o_id_ind[NITEMS];
sb2 ol_number_ind[NITEMS];
sb2 cons_ind[NITEMS];
sb2 s_rowid_ind[NITEMS];
sb2 s_remote_ind[NITEMS];
sb2 s_quant_ind[NITEMS];
sb2 i_data_ind[NITEMS];
sb2 s_data_ind[NITEMS];
sb2 s_dist_info_ind[NITEMS];
sb2 ol_dist_info_ind[NITEMS];
sb2 null_date_ind[NITEMS];
#ifdef PLSQLNO
sb2 s_bg_ind[NITEMS];
#endif

ub2 nol_i_id_len[NITEMS];
ub2 nol_supply_w_id_len[NITEMS];
ub2 nol_quantity_len[NITEMS];
ub2 nol_amount_len[NITEMS];
ub2 s_quantity_len[NITEMS];
ub2 i_name_len[NITEMS];
ub2 i_price_len[NITEMS];
ub2 i_data_len[NITEMS];
ub2 s_dist_info_len[NITEMS];
ub2 s_data_len[NITEMS];
ub2 ol_w_id_len[NITEMS];
ub2 ol_d_id_len[NITEMS];
ub2 ol_o_id_len[NITEMS];
ub2 ol_number_len[NITEMS];
ub2 cons_len[NITEMS];
ub2 s_rowid_len[NITEMS];
ub2 s_remote_len[NITEMS];
ub2 s_quant_len[NITEMS];
ub2 ol_dist_info_len[NITEMS];

```

```

ub2 null_date_len[NITEMS];
#ifdef PLSQLNO
ub2 s_bg_len[NITEMS];
#endif

ub2 nol_i_id_rcode[NITEMS];
ub2 nol_supply_w_id_rcode[NITEMS];
ub2 nol_quantity_rcode[NITEMS];
ub2 nol_amount_rcode[NITEMS];
ub2 i_name_rcode[NITEMS];
ub2 s_quantity_rcode[NITEMS];
ub2 i_price_rcode[NITEMS];
ub2 ol_w_id_rcode[NITEMS];
ub2 ol_d_id_rcode[NITEMS];
ub2 ol_o_id_rcode[NITEMS];
ub2 ol_number_rcode[NITEMS];
ub2 cons_rcode[NITEMS];
ub2 s_rowid_rcode[NITEMS];
ub2 s_remote_rcode[NITEMS];
ub2 s_quant_rcode[NITEMS];
ub2 i_data_rcode[NITEMS];
ub2 s_data_rcode[NITEMS];
ub2 s_dist_info_rcode[NITEMS];
ub2 ol_dist_info_rcode[NITEMS];
ub2 null_date_rcode[NITEMS];
#ifdef PLSQLNO
ub2 s_bg_rcode[NITEMS];
#endif

int ol_w_id[NITEMS];
int ol_d_id[NITEMS];
int ol_o_id[NITEMS];
int ol_number[NITEMS];
int cons[NITEMS];

OCIRowid *s_rowid_ptr[NITEMS];

int s_remote[NITEMS];
char i_data[NITEMS][51];
char s_data[NITEMS][51];
char s_dist_info[NITEMS][25];
OCIDate null_date[NITEMS]; /* base date for null date entry */
OCISmt *curn1;
#ifdef PLSQLNO
OCIBind *ol_i_id_bp;
OCIBind *ol_supply_w_id_bp;
OCIBind *i_price_bp;
OCIBind *i_name_bp;
OCIBind *s_bg_bp;
OCIBind *s_data_bp;
OCIBind *i_data_bp;
ub4 nol_i_count;
ub4 nol_s_count;
ub4 nol_q_count;
ub4 nol_item_count;
ub4 nol_name_count;
ub4 nol_qty_count;
ub4 nol_bg_count;
ub4 nol_am_count;
ub4 s_remote_count;
ub4 s_data_count;
ub4 i_data_count;
#endif
OCISmt *curn2;

```

```

OCISmt *curn3[10];
OCIBind *ol_i_id_bp4;
OCIBind *ol_supply_w_id_bp4;
OCIBind *ol_quantity_bp;
OCIBind *ol_quantity_bp4;
OCIBind *s_remote_bp;
OCIBind *s_quantity_bp;
OCISmt *curn4;
OCIBind *w_id_bp;
OCIBind *d_id_bp;
OCIBind *c_id_bp;
OCIBind *o_all_local_bp;
OCIBind *o_all_cnt_bp;
OCIBind *w_tax_bp;
OCIBind *d_tax_bp;
OCIBind *o_id_bp;
OCIBind *c_discount_bp;
OCIBind *c_credit_bp;
OCIBind *c_last_bp;
OCIBind *retries_bp;
OCIBind *cr_date_bp;
OCIBind *s_rowid_bp;
OCIBind *id_bp[10][15];
OCIBind *sd_bp[10][15];
OCIDefine *Dcons[10];
OCIDefine *Ds_rowid[10];
OCIDefine *Di_price[10];
OCIDefine *Di_data[10];
OCIDefine *Ds_dist_info[10];
OCIDefine *Ds_data[10];
OCIDefine *Ds_quantity[10];
OCIDefine *Di_name[10];
OCIBind *ol_o_id_bp;
OCIBind *ol_d_id_bp;
OCIBind *ol_w_id_bp;
OCIBind *ol_number_bp;
OCIBind *ol_amount_bp;
OCIBind *ol_dist_info_bp;
OCIBind *null_date_bp;

sb2 w_id_ind;
ub2 w_id_len;
ub2 w_id_rc;

sb2 d_id_ind;
ub2 d_id_len;
ub2 d_id_rc;

sb2 c_id_ind;
ub2 c_id_len;
ub2 c_id_rc;

sb2 o_all_local_ind;
ub2 o_all_local_len;
ub2 o_all_local_rc;

sb2 o_ol_cnt_ind;
ub2 o_ol_cnt_len;
ub2 o_ol_cnt_rc;

sb2 w_tax_ind;
ub2 w_tax_len;
ub2 w_tax_rc;

```

```

sb2 d_tax_ind;
ub2 d_tax_len;
ub2 d_tax_rc;

sb2 o_id_ind;
ub2 o_id_len;
ub2 o_id_rc;

sb2 c_discount_ind;
ub2 c_discount_len;
ub2 c_discount_rc;

sb2 c_credit_ind;
ub2 c_credit_len;
ub2 c_credit_rc;

sb2 c_last_ind;
ub2 c_last_len;
ub2 c_last_rc;

sb2 retries_ind;
ub2 retries_len;
ub2 retries_rc;

sb2 cr_date_ind;
ub2 cr_date_len;
ub2 cr_date_rc;

int cs;
int norow;

/* context holders */
int i_name_ctx;
int i_data_ctx;
int i_price_ctx;
int s_data_ctx;
int s_dist_info_ctx;
int s_quantity_ctx;
};

typedef struct newctx newctx;

newctx *nctx;

tkvcninit ()
{
    int i;
    text stmbuf[16*1024];
#ifdef PLSQLNO
    char sd[4];
    char id[4];
    int j;
#endif /* !PLSQLNO */

    nctx = (newctx *) malloc (sizeof(newctx));
    memset (nctx, (char)0, sizeof(newctx));
    nctx->cs = 1;
    nctx->norow=0;
    for(i=0; i<NITEMS; i++) {
        OCIERROR(errhp, OCIDescriptorAlloc(tpcenv, (dvoid**) &nctx-
>s_rowid_ptr[i],
        OCI_DTYPE_ROWID, 0, (dvoid**)0));
    }
}

```

```

nctx->w_id_ind = TRUE;
nctx->w_id_len = sizeof(w_id);
nctx->d_id_ind = TRUE;
nctx->d_id_len = sizeof(d_id);
nctx->c_id_ind = TRUE;
nctx->c_id_len = sizeof(c_id);
nctx->o_all_local_ind = TRUE;
nctx->o_all_local_len = sizeof(o_all_local);
nctx->o_ol_cnt_ind = TRUE;
nctx->o_ol_cnt_len = sizeof(o_ol_cnt);
nctx->w_tax_ind = TRUE;
nctx->w_tax_len = 0;
nctx->d_tax_ind = TRUE;
nctx->d_tax_len = 0;
nctx->o_id_ind = TRUE;
nctx->o_id_len = sizeof(o_id);
nctx->c_discount_ind = TRUE;
nctx->c_discount_len = 0;
nctx->c_credit_ind = TRUE;
nctx->c_credit_len = 0;
nctx->c_last_ind = TRUE;
nctx->c_last_len = 0;
nctx->retries_ind = TRUE;
nctx->retries_len = sizeof(retries);
nctx->cr_date_ind = TRUE;
nctx->cr_date_len = sizeof(cr_date);

/* open first cursor */
OCIERROR(errhp, OCIHandleAlloc(tpcenv, (dvoid **) (&nctx->curl),
    OCI_HTYPE_STMT, 0, (dvoid**)0));
#ifdef PLSQLNO
    sqlfile("../blocks/tkvcpnew.sql", stmbuf);
#else
    sqlfile("../blocks/tkvcbnew.sql", stmbuf);
#endif
OCIERROR(errhp, OCIStmtPrepare(nctx->curl, errhp, stmbuf,
    strlen((char *)stmbuf),
    OCI_NTV_SYNTAX, OCI_DEFAULT));

/* bind variables */

OCIBNDR(nctx->curl, nctx->w_id_bp, errhp,
":w_id", ADR(w_id), SIZ(w_id),
    SFLT_INT, &nctx->w_id_ind, &nctx->w_id_len, &nctx->w_id_rc);
OCIBNDR(nctx->curl, nctx->d_id_bp, errhp,
":d_id", ADR(d_id), SIZ(d_id),
    SFLT_INT, &nctx->d_id_ind, &nctx->d_id_len, &nctx->d_id_rc);
OCIBNDR(nctx->curl, nctx->c_id_bp, errhp,
":c_id", ADR(c_id), SIZ(c_id),
    SFLT_INT, &nctx->c_id_ind, &nctx->c_id_len, &nctx->c_id_rc);
OCIBNDR(nctx->curl, nctx->o_all_local_bp, errhp, ":o_all_local",
    ADR(o_all_local), SIZ(o_all_local), SFLT_INT, &nctx-
>o_all_local_ind,
    &nctx->o_all_local_len, &nctx->o_all_local_rc);
OCIBNDR(nctx->curl, nctx->o_all_cnt_bp, errhp,
":o_ol_cnt", ADR(o_ol_cnt),
    SIZ(o_ol_cnt), SFLT_INT, &nctx->o_ol_cnt_ind, &nctx-
>o_ol_cnt_len,
    &nctx->o_ol_cnt_rc);
OCIBNDR(nctx->curl, nctx->w_tax_bp, errhp,
":w_tax", ADR(w_tax), SIZ(w_tax),
    SFLT_FLT, &nctx->w_tax_ind, &nctx->w_tax_len, &nctx-
>w_tax_rc);

```

```

OCIBNDR(nctx->curn1, nctx->d_tax_bp, errhp,
":d_tax",ADR(d_tax),SIZ(d_tax),
SQLT_FLT, &nctx->d_tax_ind, &nctx->d_tax_len, &nctx-
>d_tax_rc);
OCIBNDR(nctx->curn1, nctx->o_id_bp, errhp,
":o_id",ADR(o_id),SIZ(o_id),
SQLT_INT, &nctx->o_id_ind, &nctx->o_id_len, &nctx->o_id_rc);
OCIBNDR(nctx->curn1, nctx->c_discount_bp, errhp, ":c_discount",
ADR(c_discount), SIZ(c_discount),SQLT_FLT,
&nctx->c_discount_ind, &nctx->c_discount_len, &nctx-
>c_discount_rc);
OCIBNDR(nctx->curn1, nctx->c_credit_bp, errhp, ":c_credit",c_credit,
SIZ(c_credit),SQLT_CHR,
&nctx->c_credit_ind, &nctx->c_credit_len, &nctx->c_credit_rc);
OCIBNDR(nctx->curn1, nctx->c_last_bp, errhp,
":c_last",c_last,SIZ(c_last),
SQLT_STR, &nctx->c_last_ind, &nctx->c_last_len, &nctx-
>c_last_rc);
OCIBNDR(nctx->curn1, nctx->retries_bp, errhp, ":retry",ADR(retries),
SIZ(retries),SQLT_INT,
&nctx->retries_ind, &nctx->retries_len, &nctx->retries_rc);
OCIBNDR(nctx->curn1, nctx->cr_date_bp, errhp,
":cr_date",&cr_date,SIZ(OCIDate),
SQLT_ODT, &nctx->cr_date_ind, &nctx->cr_date_len, &nctx->cr_date_rc);
#ifdef PLSQLNO
OCIBNDRAA(nctx->curn1, nctx->ol_i_id_bp, errhp, ":ol_i_id", nol_i_id,
SIZ(int), SQLT_INT, nctx->nol_i_id_ind, nctx->nol_i_id_len,
nctx->nol_i_id_rcode, NITEMS, &nctx->nol_i_count);
OCIBNDRAA(nctx->curn1, nctx->ol_supply_w_id_bp, errhp,
":ol_supply_w_id",
nol_supply_w_id, SIZ(int), SQLT_INT, nctx->nol_supply_w_id_ind,
nctx->nol_supply_w_id_len, nctx->nol_supply_w_id_rcode,
NITEMS, &nctx->nol_s_count);
OCIBNDRAA(nctx->curn1, nctx->
>ol_quantity_bp, errhp, ":ol_quantity", nol_quantity,
SIZ(int), SQLT_INT, nctx->nol_quantity_ind, nctx->
>nol_quantity_len,
nctx->nol_quantity_rcode, NITEMS, &nctx->nol_q_count);
OCIBNDRAA(nctx->curn1, nctx->
>i_price_bp, errhp, ":i_price", i_price, SIZ(int),
SQLT_INT, nctx->i_price_ind, nctx->i_price_len, nctx->
>i_price_rcode,
NITEMS, &nctx->nol_item_count);
OCIBNDRAA(nctx->curn1, nctx->i_name_bp, errhp, ":i_name", i_name,
SIZ(i_name[0]), SQLT_STR, nctx->i_name_ind, nctx->i_name_len,
nctx->i_name_rcode, NITEMS, &nctx->nol_name_count);
OCIBNDRAA(nctx->curn1, nctx->
>s_quantity_bp, errhp, ":s_quantity", s_quantity,
SIZ(int), SQLT_INT, nctx->s_quant_ind, nctx->s_quant_len,
nctx->s_quant_rcode, NITEMS, &nctx->nol_qty_count);
OCIBNDRAA(nctx->curn1, nctx->
>s_bg_bp, errhp, ":brand generic", brand_generic,
SIZ(char), SQLT_CHR, nctx->s_bg_ind, nctx->s_bg_len,
nctx->s_bg_rcode, NITEMS, &nctx->nol_bg_count);
OCIBNDRAA(nctx->curn1, nctx->
>ol_amount_bp, errhp, ":ol amount", nol_amount,
SIZ(int), SQLT_INT, nctx->nol_amount_ind, nctx->nol_amount_len,
nctx->nol_amount_rcode, NITEMS, &nctx->nol_am_count);
OCIBNDRAA(nctx->curn1, nctx->s_remote_bp, errhp, ":s_remote", nctx->
>s_remote,
SIZ(int), SQLT_INT, nctx->s_remote_ind, nctx->s_remote_len,
nctx->s_remote_rcode, NITEMS, &nctx->s_remote_count);

```

```

/* open second cursor */
OCIERROR(errhp,OCIHandleAlloc(tpcenv, (dvoid **)&nctx->curn2),
OCI_HTYPE_STMT,
0, (dvoid**)0));
sprintf((char *) stmbuf, SQLTXT2);
OCIERROR(errhp,OCIStmtPrepare(nctx->curn2, errhp, stmbuf,
strlen((char *)stmbuf), OCI_NT_V_SYNTAX, OCI_DEFAULT));

/* execute second cursor to init newinit package */
{
int idxlarr[NITEMS];
OCIBind *idxlarr_bp;
ub2 idxlarr_len[NITEMS];
ub2 idxlarr_rcode[NITEMS];
sb2 idxlarr_ind[NITEMS];
ub4 idxlarr_count;
ub2 idx;

for (idx = 0; idx < NITEMS; idx++) {
idxlarr[idx] = idx + 1;
idxlarr_ind[idx] = TRUE;
idxlarr_len[idx] = sizeof(int);
}
idxlarr_count = NITEMS;
o_ol_cnt = NITEMS;

/* Bind array */
OCIBNDRAA(nctx->curn2, idxlarr_bp, errhp, ":idxlarr", idxlarr,
SIZ(int), SQLT_INT, idxlarr_ind, idxlarr_len,
idxlarr_rcode, NITEMS, &idxlarr_count);

execstatus = OCIStmtExecute(tpcenv, nctx-
>curn2, errhp, 1, 0, 0, 0, OCI_DEFAULT);
if (execstatus != OCI_SUCCESS) {
OCITransRollback(tpcenv, errhp, OCI_DEFAULT);
errcode = OCIERROR(errhp, execstatus);
return -1;
}
}
#else
/* open second cursor */
OCIERROR(errhp,OCIHandleAlloc(tpcenv, (dvoid **)&nctx->curn2),
OCI_HTYPE_STMT,
0, (dvoid**)0));
sprintf((char *) stmbuf, SQLTXT2);
OCIERROR(errhp,OCIStmtPrepare(nctx->curn2, errhp, stmbuf,
strlen((char *)stmbuf), OCI_NT_V_SYNTAX, OCI_DEFAULT));

/* bind variables */
OCIBNDRA(nctx->curn2, nctx->
>s_quantity_bp, errhp, ":s_quantity", s_quantity,
SIZ(int), SQLT_INT, nctx->s_quant_ind, nctx->s_quant_len,
nctx->s_quant_rcode);
OCIBNDRA(nctx->curn2, nctx->s_rowid_bp, errhp, ":s_rowid", nctx->
>s_rowid_ptr,
sizeof(nctx->s_rowid_ptr[0]), SQLT_RDD, nctx->s_rowid_ind,
nctx->s_rowid_len, nctx->s_rowid_rcode);

```

```

OCIBNDRA(nctx->curn2, nctx-
>ol_quantity_bp,errhp,":ol_quantity",nol_quantity,
        SIZ(int),SQLT_INT,nctx->nol_quantity_ind,nctx-
>nol_quantity_len,
        nctx->nol_quantity_rcode);
OCIBNDRA(nctx->curn2, nctx->s_remote_bp, errhp, ":s_remote",nctx-
>s_remote,
        SIZ(int), SQLT_INT,nctx->s_remote_ind,nctx->s_remote_len,
        nctx->s_remote_rcode);

/* open third cursor and bind variables */

for (i = 0; i < 10; i++)
{
    j = i + 1;
    OCIERROR(errhp,OCIHandleAlloc(tpcenv, (dvoid **)&(nctx->curn3)[i]),
        OCI_HTYPE_STMT, 0, (dvoid**)0);

    sprintf((char *) stmbuf, SQLTXT3, j, j, j, j, j, j, j, j, j, j,
j,
        j, j, j);

    OCIERROR(errhp,OCIStmtPrepare((nctx->curn3)[i], errhp, stmbuf,
        strlen((char
*)stmbuf),OCI_NTV_SYNTAX,
        OCI_DEFAULT));

    OCIERROR(errhp,
        OCIAttrSet(nctx->curn3[i],OCI_HTYPE_STMT, (dvoid*)&nctx-
>norow,0,
        OCI_ATTR_PREFETCH_ROWS,errhp));
    for (j = 0; j < NITEMS; j++)
    {
        sprintf(id, ":%d", j + 10);
        sprintf(sd, ":%d", j + 30);
        OCIBNDRA((nctx->curn3)[i],(nctx-
>id_bp)[i][j],errhp,id,ADR(nol_i_id[j]),
        SIZ(int),SQLT_INT,
        &nctx->nol_i_id_ind[j],&nctx->nol_i_id_len[j],
        &nctx->nol_i_id_rcode[j]);
        OCIBNDRA((nctx->curn3)[i],(nctx->sd_bp)[i][j],errhp,sd,
        ADR(nol_supply_w_id[j]),SIZ(int),SQLT_INT,
        &nctx->nol_supply_w_id_ind[j],&nctx-
>nol_supply_w_id_len[j],
        &nctx->nol_supply_w_id_rcode[j]);
        nctx->nol_i_id_ind[j] = NA;
        nctx->nol_supply_w_id_ind[j] = NA;
        nctx->nol_i_id_len[j] = sizeof(int);
        nctx->nol_supply_w_id_len[j] = sizeof(int);
    }

    OCIDEF((nctx->curn3)[i],(nctx->Dcons)[i],errhp,1,&(nctx->cons[0]),
        SIZ(nctx->cons[0]),SQLT_INT);
    OCIDEF((nctx->curn3)[i],(nctx->Ds_rowid)[i],errhp,2,
        nctx->s_rowid_ptr, sizeof(nctx->s_rowid_ptr[0]),
SQLT_RDD);
    OCIDEF((nctx->curn3)[i],(nctx-
>Di_price)[i],errhp,3,i_price,SIZ(int),
        SQLT_INT);

    OCIDFNRA((nctx->curn3)[i],(nctx->Di_name)[i],errhp,4,i_name,
        SIZ(i_name[0]),SQLT_STR, nctx->i_name_ind,nctx->i_name_len,
        nctx->i_name_rcode);
    OCIDFNRA((nctx->curn3)[i],(nctx->Di_data)[i],errhp,5,nctx->i_data,

```

```

        SIZ(nctx->i_data[0]), SQLT_STR, NULL,nctx-
>i_data_len,NULL);
    OCIDFNRA((nctx->curn3)[i],(nctx->Ds_dist_info)[i],errhp,6,
        nctx->s_dist_info, SIZ(nctx->s_dist_info[0]),SQLT_STR,
        NULL,nctx->s_dist_info_len, NULL);
    OCIDFNRA((nctx->curn3)[i],(nctx->Ds_data)[i],errhp,7,nctx->s_data,
        SIZ(nctx->s_data[0]),SQLT_STR,NULL,nctx->s_data_len,NULL);
    OCIDEF((nctx->curn3)[i],(nctx->Ds_quantity)[i],errhp,8,s_quantity,
        SIZ(int),SQLT_INT);
}

/* open fourth cursor */
OCIHandleAlloc(tpcenv, (dvoid **)&(nctx->curn4), OCI_HTYPE_STMT, 0,
        (dvoid**)0);
sprintf((char *) stmbuf, SQLTXT4);
OCIStmtPrepare(nctx->curn4, errhp, stmbuf, strlen((char *)stmbuf),
        OCI_NTV_SYNTAX, OCI_DEFAULT);
/* bind variables */

OCIBNDRA(nctx->curn4, nctx->ol_o_id_bp,errhp,":ol_o_id",nctx->ol_o_id,
        SIZ(int),SQLT_INT, NULL,nctx->ol_o_id_len,
        NULL);
OCIBNDRA(nctx->curn4, nctx->ol_d_id_bp,errhp,":ol_d_id",nctx->ol_d_id,
        SIZ(int),SQLT_INT, NULL,nctx->ol_d_id_len,
        NULL);
OCIBNDRA(nctx->curn4, nctx->ol_w_id_bp,errhp,":ol_w_id",nctx->ol_w_id,
        SIZ(int),SQLT_INT, NULL,nctx->ol_w_id_len,
        NULL);
OCIBNDRA(nctx->curn4, nctx->ol_number_bp,errhp,":ol_number",nctx-
>ol_number,
        SIZ(int),SQLT_INT, NULL,nctx->ol_number_len,
        NULL);
OCIBNDRA(nctx->curn4, nctx-
>ol_i_id_bp4,errhp,":ol_i_id",nol_i_id,SIZ(int),
        SQLT_INT, NULL,nctx->nol_i_id_len, NULL);
OCIBNDRA(nctx->curn4, nctx-
>ol_supply_w_id_bp4,errhp,":ol_supply_w_id",
        nol_supply_w_id,SIZ(int),SQLT_INT, NULL,
        nctx->nol_supply_w_id_len, NULL);
OCIBNDRA(nctx->curn4, nctx-
>ol_quantity_bp4,errhp,":ol_quantity",nol_quantity,
        SIZ(int),SQLT_INT, NULL,nctx->nol_quantity_len,
        NULL);
OCIBNDRA(nctx->curn4, nctx-
>ol_amount_bp,errhp,":ol_amount",nol_amount,
        SIZ(int),SQLT_INT, NULL,nctx->nol_amount_len,
        NULL);
OCIBNDRA(nctx->curn4, nctx->ol_dist_info_bp,errhp,":ol_dist_info",
        nctx->s_dist_info, SIZ(nctx->s_dist_info[0]),SQLT_AFC,
        NULL, nctx->ol_dist_info_len,
        NULL);
OCIBNDRA(nctx->curn4, nctx->>null_date_bp,errhp,":null_date",nctx-
>>null_date,
        SIZ(OCIDate), SQLT_ODT,NULL,
        nctx->>null_date_len, NULL);

/* set up the null date Null date is 15-sep-11 */
for (i=0;i<NITEMS;i++)
{
    OCIDateSetDate(&nctx->>null_date[i],(sb2)1811,(ub1)9,(ub1)15);
}
#endif

```

```

return (0);
}

tkvcn ()
{
    int i;
    int rcount;
# ifndef PLSQLNO
    ub4 flags;
    int rowoff, rpc, rpc3, iters, j, k;
# endif /* !PLSQLNO */
    int failed = 0;

retry:
    status = 0;                /* number of invalid items */

    /* get number of order lines, and check if all are local */

    o_ol_cnt = NITEMS;
    o_all_local = 1;
    for (i = 0; i < NITEMS; i++) {
        if (nol_i_id[i] == 0) {
            o_ol_cnt = i;
            break;
        }
        if (nol_supply_w_id[i] != w_id) {
            nctx->s_remote[i] = 1;
            o_all_local = 0;
        }
        else
            nctx->s_remote[i] = 0;
    }

    nctx->w_id_ind = TRUE;
    nctx->w_id_len = sizeof(w_id);
    nctx->d_id_ind = TRUE;
    nctx->d_id_len = sizeof(d_id);
    nctx->c_id_ind = TRUE;
    nctx->c_id_len = sizeof(c_id);
    nctx->o_all_local_ind = TRUE;
    nctx->o_all_local_len = sizeof(o_all_local);
    nctx->o_ol_cnt_ind = TRUE;
    nctx->o_ol_cnt_len = sizeof(o_ol_cnt);
    nctx->w_tax_ind = TRUE;
    nctx->w_tax_len = 0;
    nctx->d_tax_ind = TRUE;
    nctx->d_tax_len = 0;
    nctx->o_id_ind = TRUE;
    nctx->o_id_len = sizeof(o_id);
    nctx->c_discount_ind = TRUE;
    nctx->c_discount_len = 0;
    nctx->c_credit_ind = TRUE;
    nctx->c_credit_len = 0;
    nctx->c_last_ind = TRUE;
    nctx->c_last_len = 0;
    nctx->retries_ind = TRUE;
    nctx->retries_len = sizeof(retries);
    nctx->cr_date_ind = TRUE;
    nctx->cr_date_len = sizeof(cr_date);

```

```

#ifdef PLSQLNO
/* this is the row count */
rcount = o_ol_cnt;
nctx->nol_i_count = o_ol_cnt;
nctx->nol_q_count = o_ol_cnt;
nctx->nol_s_count = o_ol_cnt;
nctx->s_remote_count = o_ol_cnt;

nctx->nol_qty_count = 0;
nctx->nol_bg_count = 0;
nctx->nol_item_count = 0;
nctx->nol_name_count = 0;
nctx->nol_am_count = 0;
/* following not relevant */
nctx->s_data_count = o_ol_cnt;
nctx->i_data_count = o_ol_cnt;

/* initialization for array operations */
for (i = 0; i < o_ol_cnt; i++) {
    nctx->ol_w_id[i] = w_id;
    nctx->ol_d_id[i] = d_id;
    nctx->ol_number[i] = i + 1;
    nctx->null_date_ind[i] = TRUE;
    nctx->nol_i_id_ind[i] = 0;
    nctx->nol_supply_w_id_ind[i] = TRUE;
    nctx->nol_quantity_ind[i] = TRUE;
    nctx->nol_amount_ind[i] = TRUE;
    nctx->ol_w_id_ind[i] = TRUE;
    nctx->ol_d_id_ind[i] = TRUE;
    nctx->ol_o_id_ind[i] = TRUE;
    nctx->ol_number_ind[i] = TRUE;
    nctx->ol_dist_info_ind[i] = TRUE;
    nctx->s_remote_ind[i] = TRUE;
    nctx->s_data_ind[i] = TRUE;
    nctx->i_data_ind[i] = TRUE;
    nctx->s_quant_ind[i] = TRUE;
    nctx->s_bg_ind[i] = TRUE;
    nctx->cons_ind[i] = TRUE;
    nctx->s_rowid_ind[i] = TRUE;
    nctx->nol_i_id_len[i] = sizeof(int);
    nctx->nol_supply_w_id_len[i] = sizeof(int);
    nctx->nol_quantity_len[i] = sizeof(int);
    nctx->nol_amount_len[i] = sizeof(int);
    nctx->ol_w_id_len[i] = sizeof(int);
    nctx->ol_d_id_len[i] = sizeof(int);
    nctx->ol_o_id_len[i] = sizeof(int);
    nctx->ol_number_len[i] = sizeof(int);
    nctx->ol_dist_info_len[i] = nctx->s_dist_info_len[i];
    nctx->null_date_len[i] = sizeof(OCIDate);
    nctx->s_remote_len[i] = sizeof(int);
    nctx->s_data_len[i] = sizeof(int);
    nctx->i_data_len[i] = sizeof(int);
    nctx->s_quant_len[i] = sizeof(int);
    nctx->s_rowid_len[i] = sizeof(nctx->s_rowid_ptr[0]);
    nctx->cons_len[i] = sizeof(int);
    nctx->i_name_len[i] = 0;
    nctx->s_bg_len[i] = 0;
}
for (i = o_ol_cnt; i < NITEMS; i++) {
    nctx->nol_i_id_ind[i] = NA;
    nctx->nol_supply_w_id_ind[i] = NA;
    nctx->nol_quantity_ind[i] = NA;
    nctx->nol_amount_ind[i] = NA;
    nctx->ol_w_id_ind[i] = NA;

```

```

nctx->ol_d_id_ind[i] = NA;
nctx->ol_o_id_ind[i] = NA;
nctx->ol_number_ind[i] = NA;
nctx->ol_dist_info_ind[i] = NA;
nctx->null_date_ind[i] = NA;
nctx->s_remote_ind[i] = NA;
nctx->s_data_ind[i] = NA;
nctx->i_data_ind[i] = NA;
nctx->s_quant_ind[i] = NA;
nctx->s_bg_ind[i] = NA;
nctx->cons_ind[i] = NA;
nctx->s_rowid_ind[i] = NA;

nctx->nol_i_id_len[i] = 0;
nctx->nol_supply_w_id_len[i] = 0;
nctx->nol_quantity_len[i] = 0;
nctx->nol_amount_len[i] = 0;
nctx->ol_w_id_len[i] = 0;
nctx->ol_d_id_len[i] = 0;
nctx->ol_o_id_len[i] = 0;
nctx->ol_number_len[i] = 0;
nctx->ol_dist_info_len[i] = 0;
nctx->null_date_len[i] = 0;
nctx->s_remote_len[i] = 0;
nctx->i_data_len[i] = 0;
nctx->s_data_len[i] = 0;
nctx->s_quant_len[i] = 0;
nctx->s_rowid_len[i] = 0;
nctx->cons_len[i] = 0;
nctx->i_name_len[i] = 0;
nctx->s_bg_len[i] = 0;
}

execstatus = OCIStmtExecute(tpcsvc, nctx->curl, errhp, 1, 0, 0, 0,
OCI_DEFAULT | OCI_COMMIT_ON_SUCCESS);

#else
execstatus = OCIStmtExecute(tpcsvc, nctx-
>curl, errhp, 1, 0, 0, 0, OCI_DEFAULT);
#endif

if(execstatus != OCI_SUCCESS) {
OCITransRollback(tpcsvc, errhp, OCI_DEFAULT);
errcode = OCIERROR(errhp, execstatus);
if(errcode == NOT_SERIALIZABLE) {
retries++;
goto retry;
} else if (errcode == RECOVER) {
retries++;
goto retry;
}
else {
return -1;
}
}

#ifdef PLSQLNO
/* did the txn succeed ? */
if (rcount != o_ol_cnt)
{
status = rcount - o_ol_cnt;
o_ol_cnt = rcount;
}
}

```

```

#endif

#ifdef DEBUG
printf("w_id = %d, d_id = %d, c_id = %d\n", w_id, d_id, c_id);
#endif

#ifdef PLSQLNO
/* initialization for array operations */

for (i = 0; i < o_ol_cnt; i++) {
nctx->ol_w_id[i] = w_id;
nctx->ol_d_id[i] = d_id;
nctx->ol_number[i] = i + 1;
nctx->null_date_ind[i] = TRUE;
nctx->nol_i_id_ind[i] = TRUE;
nctx->nol_supply_w_id_ind[i] = TRUE;
nctx->nol_quantity_ind[i] = TRUE;
nctx->nol_amount_ind[i] = TRUE;
nctx->ol_w_id_ind[i] = TRUE;
nctx->ol_d_id_ind[i] = TRUE;
nctx->ol_o_id_ind[i] = TRUE;
nctx->ol_number_ind[i] = TRUE;
nctx->ol_dist_info_ind[i] = TRUE;
nctx->s_remote_ind[i] = TRUE;
nctx->s_quant_ind[i] = TRUE;
nctx->cons_ind[i] = TRUE;
nctx->s_rowid_ind[i] = TRUE;

nctx->nol_i_id_len[i] = sizeof(int);
nctx->nol_supply_w_id_len[i] = sizeof(int);
nctx->nol_quantity_len[i] = sizeof(int);
nctx->nol_amount_len[i] = sizeof(int);
nctx->ol_w_id_len[i] = sizeof(int);
nctx->ol_d_id_len[i] = sizeof(int);
nctx->ol_o_id_len[i] = sizeof(int);
nctx->ol_number_len[i] = sizeof(int);
nctx->ol_dist_info_len[i] = nctx->s_dist_info_len[i];
nctx->null_date_len[i] = sizeof(OCIDate);
nctx->s_remote_len[i] = sizeof(int);
nctx->s_quant_len[i] = sizeof(int);
nctx->s_rowid_len[i] = sizeof(nctx->s_rowid_ptr[0]);
nctx->cons_len[i] = sizeof(int);
}

for (i = o_ol_cnt; i < NITEMS; i++) {
nctx->nol_i_id_ind[i] = NA;
nctx->nol_supply_w_id_ind[i] = NA;
nctx->nol_quantity_ind[i] = NA;
nctx->nol_amount_ind[i] = NA;
nctx->ol_w_id_ind[i] = NA;
nctx->ol_d_id_ind[i] = NA;
nctx->ol_o_id_ind[i] = NA;
nctx->ol_number_ind[i] = NA;
nctx->ol_dist_info_ind[i] = NA;
nctx->null_date_ind[i] = NA;
nctx->s_remote_ind[i] = NA;
nctx->s_quant_ind[i] = NA;
nctx->cons_ind[i] = NA;
nctx->s_rowid_ind[i] = NA;

nctx->nol_i_id_len[i] = 0;
nctx->nol_supply_w_id_len[i] = 0;
nctx->nol_quantity_len[i] = 0;
nctx->nol_amount_len[i] = 0;
nctx->ol_w_id_len[i] = 0;
}
}

```

```

nctx->ol_d_id_len[i] = 0;
nctx->ol_o_id_len[i] = 0;
nctx->ol_number_len[i] = 0;
nctx->ol_dist_info_len[i] = 0;
nctx->null_date_len[i] = 0;
nctx->s_remote_len[i] = 0;
nctx->s_quant_len[i] = 0;
nctx->s_rowid_len[i] = 0;
nctx->cons_len[i] = 0;
}

rpc3 = SelItemStk ();
if (rpc3 == -2)
    goto retry;
else if (rpc3 == -1)
    return (-1);

/* compute order line amounts, total amount and stock quantities */
total_amount = 0.0;
for (i = 0; i < o_ol_cnt; i++)
{
    nctx->ol_o_id[i] = o_id;
    if (nctx->nol_i_id_ind[i] != NA) {
        s_quantity[i] -= nol_quantity[i];
        if (s_quantity[i] < 10)
            s_quantity[i] += 91;
        nol_amount[i] = (nol_quantity[i] * i_price[i]);
        total_amount += nol_amount[i];
        if (strstr (nctx->i_data[i], "ORIGINAL") &&
            strstr (nctx->s_data[i], "ORIGINAL"))
            brand_gen[i] = 'B';
        else
            brand_gen[i] = 'G';
    }
}

total_amount *= ((float)(1 - c_discount)) * (1.0 + ((float)d_tax) +
((float)w_tax));
total_amount = total_amount/100;

rpc = UpdStk2 ();
if (rpc == -2)
    goto retry;
else if (rpc == -1)
    return (-1);

/* error processing - will keep it separated for readability */
/* number of items selected != number of stock updated */

if (rpc3 != rpc) {
    userlog ("Error in TPC-C server %d: %d rows of item read, ",
            proc_no, rpc3);
    userlog ("                but %d rows of stock
updated\n", rpc);
    /* rollback */
    OCITransRollback(tpcsvc, errhp, OCI_DEFAULT);
    return (-1);
}

/* common code for insert into order_line */
for (i=0; i < o_ol_cnt; i++) /* move district info in place */
{

```

```

nctx->ol_dist_info_len[i]=nctx->s_dist_info_len[i];
}

/* array insert into order line table */
flags= (status ? OCI_DEFAULT : (OCI_DEFAULT|OCI_COMMIT_ON_SUCCESS));
if ((o_ol_cnt - status) > 0)
{
    execstatus = OCISmtExecute (tpcsvc, nctx->curn4, errhp, o_ol_cnt -
status,
                                0, 0, 0, flags);
    if (execstatus != OCI_SUCCESS) {
        OCITransRollback(tpcsvc, errhp, OCI_DEFAULT);
        errcode = OCIERROR(errhp, execstatus);
        if (errcode == NOT_SERIALIZABLE) {
            retries++;
            goto retry;
        } else if (errcode == RECOVER) {
            retries++;
            goto retry;
        } else {
            return -1;
        }
    }
    OCIAttrGet (nctx->curn4, OCI_HTYPE_STMT, &rcount, NULL,
                OCI_ATTR_ROW_COUNT, errhp);
    if (rcount != (o_ol_cnt - status))
    {
        userlog ("Error in TPC-C server %d: array insert failed\n",
                proc_no);
        /* rollback */
        OCITransRollback (tpcsvc, errhp, OCI_DEFAULT);
        return (-1);
    }
}

/* commit if no invalid item */

if (status) {
    OCITransRollback (tpcsvc, errhp, OCI_DEFAULT);
    fflush(stdout);
}

#endif
total_amount = 0.0;
for (i = 0; i < o_ol_cnt; i++)
{
    if (nctx->nol_amount_ind[i] != NA) {
        total_amount += nol_amount[i];
    }
}

total_amount *= ((float)(1 - c_discount)) * (float)(1.0 +
((float)d_tax) + ((float)w_tax));
total_amount = total_amount/100;

return (0);
}

void tkvcndone ()
{

```

```

int i;

if (nctx)
{
    OCIHandleFree((dvoid *)nctx->curn1,OCI_HTYPE_STMT);
    OCIHandleFree((dvoid *)nctx->curn2,OCI_HTYPE_STMT);
    for (i = 0; i < 10; i++)
        OCIHandleFree((dvoid *) (nctx->curn3) [i],OCI_HTYPE_STMT);
    OCIHandleFree((dvoid *)nctx->curn4,OCI_HTYPE_STMT);
    free (nctx);
}

/* the arrays are initialized based on a successful select from */
/* stock/item. We need to shift the values in the orderline array */
/* one position up to compensate when we have an invalid item */

void shiftitemstock (i, j)

int i, j;

{
    /* shift up the values for the stock table */
    nctx->s_remote[i] = nctx->s_remote[j];

    /* shift up the order_line values */

    nctx->nol_i_id_ind[i]=nctx->nol_i_id_ind[j];
    nol_i_id[i] = nol_i_id[j];

    nctx->nol_quantity_ind[i] = nctx->nol_quantity_ind[j];
    nol_quantity[i] = nol_quantity[j];

    nctx->nol_supply_w_id_ind [i] = nctx->nol_supply_w_id_ind[j];
    nol_supply_w_id[i] = nol_supply_w_id[j];
}

void swapitemstock (i, j)

int i, j;

{

    int tempi;
    int tempf;
    char tempstr[52];
    ub2 tempub2;
    sb2 tempsb2;
    OCIRowid *tmprid;

    tempsb2 = nctx->cons_ind[i];
    nctx->cons_ind[i] = nctx->cons_ind[j];
    nctx->cons_ind[j] = tempsb2;
    tempub2 = nctx->cons_len[i];
    nctx->cons_len[i] = nctx->cons_len[j];
    nctx->cons_len[j] = tempub2;
    tempub2 = nctx->cons_rcode[i];
    nctx->cons_rcode[i] = nctx->cons_rcode[j];
    nctx->cons_rcode[j] = tempub2;
    tempi = nctx->cons[i];
    nctx->cons[i] = nctx->cons[j];
    nctx->cons[j] = tempi;
}

```

```

tempsb2 = nctx->s_rowid_ind[i];
nctx->s_rowid_ind[i] = nctx->s_rowid_ind[j];
nctx->s_rowid_ind[j] = tempsb2;
tempub2 = nctx->s_rowid_len[i];
nctx->s_rowid_len[i] = nctx->s_rowid_len[j];
nctx->s_rowid_len[j] = tempub2;
tempub2 = nctx->s_rowid_rcode[i];
nctx->s_rowid_rcode[i] = nctx->s_rowid_rcode[j];
nctx->s_rowid_rcode[j] = tempub2;
tmprid = nctx->s_rowid_ptr[i];
nctx->s_rowid_ptr[i] = nctx->s_rowid_ptr[j];
nctx->s_rowid_ptr[j]=tmprid;

```

```

tempsb2 = nctx->i_price_ind[i];
nctx->i_price_ind[i] = nctx->i_price_ind[j];
nctx->i_price_ind[j] = tempsb2;
tempub2 = nctx->i_price_len[i];
nctx->i_price_len[i] = nctx->i_price_len[j];
nctx->i_price_len[j] = tempub2;
tempub2 = nctx->i_price_rcode[i];
nctx->i_price_rcode[i] = nctx->i_price_rcode[j];
nctx->i_price_rcode[j] = tempub2;
tempf = i_price[i];
i_price[i] = i_price[j];
i_price[j] = tempf;

```

```

tempsb2 = nctx->i_name_ind[i];
nctx->i_name_ind[i] = nctx->i_name_ind[j];
nctx->i_name_ind[j] = tempsb2;
tempub2 = nctx->i_name_len[i];
nctx->i_name_len[i] = nctx->i_name_len[j];
nctx->i_name_len[j] = tempub2;
tempub2 = nctx->i_name_rcode[i];
nctx->i_name_rcode[i] = nctx->i_name_rcode[j];
nctx->i_name_rcode[j] = tempub2;
strncpy (tempstr, i_name[i], 25);
strncpy (i_name[i], i_name[j], 25);
strncpy (i_name[j], tempstr, 25);

```

```

tempsb2 = nctx->i_data_ind[i];
nctx->i_data_ind[i] = nctx->i_data_ind[j];
nctx->i_data_ind[j] = tempsb2;
tempub2 = nctx->i_data_len[i];
nctx->i_data_len[i] = nctx->i_data_len[j];
nctx->i_data_len[j] = tempub2;
tempub2 = nctx->i_data_rcode[i];
nctx->i_data_rcode[i] = nctx->i_data_rcode[j];
nctx->i_data_rcode[j] = tempub2;
strncpy (tempstr, nctx->i_data[i], 51);
strncpy (nctx->i_data[i], nctx->i_data[j], 51);
strncpy (nctx->i_data[j], tempstr, 51);

```

```

tempsb2 = nctx->s_quantity_ind[i];
nctx->s_quantity_ind[i] = nctx->s_quantity_ind[j];
nctx->s_quantity_ind[j] = tempsb2;
tempub2 = nctx->s_quantity_len[i];
nctx->s_quantity_len[i] = nctx->s_quantity_len[j];
nctx->s_quantity_len[j] = tempub2;
tempub2 = nctx->s_quantity_rcode[i];
nctx->s_quantity_rcode[i] = nctx->s_quantity_rcode[j];
nctx->s_quantity_rcode[j] = tempub2;
tempf = s_quantity[i];
s_quantity[i] = s_quantity[j];
s_quantity[j] = tempf;

```

```

tempub2 = nctx->s_dist_info_ind[i];
nctx->s_dist_info_ind[i] = nctx->s_dist_info_ind[j];
nctx->s_dist_info_ind[j] = tempub2;
tempub2 = nctx->s_dist_info_len[i];
nctx->s_dist_info_len[i] = nctx->s_dist_info_len[j];
nctx->s_dist_info_len[j] = tempub2;
tempub2 = nctx->s_dist_info_rcode[i];
nctx->s_dist_info_rcode[i] = nctx->s_dist_info_rcode[j];
nctx->s_dist_info_rcode[j] = tempub2;
strncpy (tempstr, nctx->s_dist_info[i], 25);
strncpy (nctx->s_dist_info[i], nctx->s_dist_info[j], 25);
strncpy (nctx->s_dist_info[j], tempstr, 25);

tempub2 = nctx->s_data_ind[i];
nctx->s_data_ind[i] = nctx->s_data_ind[j];
nctx->s_data_ind[j] = tempub2;
tempub2 = nctx->s_data_len[i];
nctx->s_data_len[i] = nctx->s_data_len[j];
nctx->s_data_len[j] = tempub2;
tempub2 = nctx->s_data_rcode[i];
nctx->s_data_rcode[i] = nctx->s_data_rcode[j];
nctx->s_data_rcode[j] = tempub2;
strncpy (tempstr, nctx->s_data[i], 51);
strncpy (nctx->s_data[i], nctx->s_data[j], 51);
strncpy (nctx->s_data[j], tempstr, 51);
}

SellItemStk ()
{
    int i, j, rpc3, rcount;

    /* array select from item and stock tables */
    execstatus=OCIStmtExecute(tpcsvc, (nctx->curn3)[d_id-
1], errhp, o_ol_cnt,
                                0, 0, 0, OCI_DEFAULT);
    if ((execstatus != OCI_SUCCESS) && (execstatus != OCI_NO_DATA)) {
        errcode = OCIERROR(errhp, execstatus);
        if (errcode == NOT_SERIALIZABLE) {
            retries++;
            OCITransRollback(tpcsvc, errhp, OCI_DEFAULT);
            return (-2);
        } else if (errcode == RECOVER) {
            /* In case of NO_DATA this should NOT return, but simply fall
through */
            OCITransRollback(tpcsvc, errhp, OCI_DEFAULT);
            retries++;
            return (-2);
        } else {
            OCITransRollback(tpcsvc, errhp, OCI_DEFAULT);
            return (-1);
        }
    }
    /* mark invalid items */
    OCIAttrGet((nctx->curn3)[d_id-1], OCI_HTYPE_STMT, &rcount, NULL,
                OCI_ATTR_ROW_COUNT, errhp);
    rpc3 = rcount;

    /* the result is in order, so we have to shift up to fill */
    /* the slot for the line with the invalid item. */

```

```

/* If more than one item is wrong, this is not an simulated */
/* error and we'll blow off */

    if ((status = o_ol_cnt - rcount) > 1)
    {
        userlog ("TPC-C server %d: more than 1 invalid item?\n",
proc_no);
        return (rpc3);
    }
    if (status == 0) return (rpc3);

    /* find the invalid item, transfer the rowid information */

    for (i = 0; i < o_ol_cnt; i++) {
        if (nctx->cons[i] != i) break; /* this item is invalid */
    }

        userlog ("TPC-C server %d: reordering items and stocks\n",
proc_no);

    /* not the last item - shift up */

    for (j = i; j < o_ol_cnt-1; j++)
    {
        shiftitemstock (j, j+1);
    }
    /* zero the last item */
    i = o_ol_cnt-1;
    nctx->nol_i_id_ind[i] = NA;
    nctx->nol_supply_w_id_ind[i] = NA;
    nctx->nol_quantity_ind[i] = NA;
    nctx->nol_amount_ind[i] = NA;
    nctx->ol_w_id_ind[i] = NA;
    nctx->ol_d_id_ind[i] = NA;
    nctx->ol_o_id_ind[i] = NA;
    nctx->>null_date_ind[i] = NA;
    nctx->ol_number_ind[i] = NA;
    nctx->ol_dist_info_ind[i] = NA;
    nctx->s_remote_ind[i] = NA;
    nctx->s_quant_ind[i] = NA;

    nctx->nol_i_id_len[i] = 0;
    nctx->nol_supply_w_id_len[i] = 0;
    nctx->nol_quantity_len[i] = 0;
    nctx->nol_amount_len[i] = 0;
    nctx->ol_w_id_len[i] = 0;
    nctx->ol_d_id_len[i] = 0;
    nctx->ol_o_id_len[i] = 0;
    nctx->ol_number_len[i] = 0;
    nctx->ol_dist_info_len[i] = 0;
    nctx->>null_date_ind[i] = 0;
    nctx->s_remote_len[i] = 0;
    nctx->s_quant_len[i] = 0;

    return (rpc3);
}

UpdStk2 ()
{

```

```

int rpc, rcount;

/* array update of stock table */

execstatus = OCIStmtExecute(tpcsvc,nctx->curn2,errhp,o_ol_cnt-
status,0,0,0,
OCI_DEFAULT);
if(execstatus != OCI_SUCCESS) {
OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
errcode = OCIERROR(errhp,execstatus);
if(errcode == NOT_SERIALIZABLE) {
retries++;
return (-2);
} else if (errcode == RECOVER) {
retries++;
return (-2);
} else {
return -1;
}
}
OCIAttrGet(nctx->curn2,OCI_HTYPE_STMT,&rcount,NULL,
OCI_ATTR_ROW_COUNT, errhp);
rpc = rcount;

if (rpc != (o_ol_cnt - status)) {
userlog ("Error in TPC-C server %d: array update failed\n",
proc_no);
OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
return (-1);
}

return (rpc);
}

```

PLORD.C

```

#ifdef RCSID
static char *RCSid =
"$Header: plord.c 7030100.1 95/07/19 14:46:13 plai Generic<base>
$ Copyr (c) 1994 Oracle";
#endif /* RCSID */

/*-----+
|          Copyright (c) 1995 Oracle Corp, Redwood Shores, CA          |
|          OPEN SYSTEMS PERFORMANCE GROUP                               |
|          All Rights Reserved                                          |
+-----+
| FILENAME                                                              |
| plord.c                                                               |
| DESCRIPTION                                                            |
| OCI version (using PL/SQL anonymous block) of                        |
| ORDER STATUS transaction in TPC-C benchmark.                          |
+-----*/

#include "tpcc.h"
#include "tpccflags.h"

```

```

#ifdef TUX
#include <userlog.h>
#endif

#ifdef PLSQLORD
#define SQLTXT "BEGIN orderstatus.getstatus (:w_id, :d_id, :c_id, :byln,
\
:c_last, :c_first, :c_middle, :c_balance, :o_id, :o_entry_d, :o_cr_i
d, \
:o_ol_cnt, :ol_s_w_id, :ol_i_id, :ol_quantity, :ol_amount, :ol_d_d);
END;"
#else
#define SQLCUR0 "SELECT rowid FROM cust \
WHERE c_d_id = :d_id AND c_w_id = :w_id AND c_last
= :c_last \
ORDER BY c_last, c_d_id, c_w_id, c_first"

#define SQLCUR1 "SELECT c_id, c_balance, c_first, c_middle, c_last, \
o_id, o_entry_d, o_carrier_id, o_ol_cnt \
FROM cust, ordr \
WHERE cust.rowid = :cust_rowid \
AND o_d_id = c_d_id AND o_w_id = c_w_id AND o_c_id =
c_id \
ORDER BY o_c_id, o_d_id, o_w_id, o_id DESC"

#define SQLCUR2 "SELECT c_balance, c_first, c_middle, c_last, \
o_id, o_entry_d, o_carrier_id, o_ol_cnt, c_id \
FROM cust, ordr \
WHERE c_id = :c_id AND c_d_id = :d_id AND c_w_id
= :w_id \
AND o_d_id = c_d_id AND o_w_id = c_w_id AND o_c_id =
c_id \
ORDER BY o_c_id, o_d_id, o_w_id, o_id DESC"

#define SQLCUR3 "SELECT ol_i_id, ol_supply_w_id, ol_quantity, ol_amount, \
ol_delivery_d \
FROM ordl \
WHERE ol_d_id = :d_id AND ol_w_id = :w_id AND ol_o_id
= :o_id"

#define SQLCUR4 "SELECT count(c_last) FROM cust \
WHERE c_d_id = :d_id AND c_w_id = :w_id AND c_last
= :c_last "
#endif

struct ordctx {
sb2 c_rowid_ind[100];
sb2 ol_supply_w_id_ind[NITEMS];
sb2 ol_i_id_ind[NITEMS];
sb2 ol_quantity_ind[NITEMS];
sb2 ol_amount_ind[NITEMS];
sb2 ol_delivery_d_ind[NITEMS];
sb2 ol_w_id_ind;
sb2 ol_d_id_ind;
sb2 ol_o_id_ind;
sb2 c_id_ind;
sb2 c_first_ind;
sb2 c_middle_ind;
sb2 c_balance_ind;

```

```

sb2 c_last_ind;
sb2 o_id_ind;
sb2 o_entry_d_ind;
sb2 o_carrier_id_ind;
sb2 o_ol_cnt_ind;

ub4 c_rowid_len[100];
ub2 ol_supply_w_id_len[NITEMS];
ub2 ol_i_id_len[NITEMS];
ub2 ol_quantity_len[NITEMS];
ub2 ol_amount_len[NITEMS];
ub2 ol_delivery_d_len[NITEMS];
ub2 ol_w_id_len;
ub2 ol_d_id_len;
ub2 ol_o_id_len;

ub2 c_rowid_rcode[100];
ub2 ol_supply_w_id_rcode[NITEMS];
ub2 ol_i_id_rcode[NITEMS];
ub2 ol_quantity_rcode[NITEMS];
ub2 ol_amount_rcode[NITEMS];
ub2 ol_delivery_d_rcode[NITEMS];
ub2 ol_w_id_rcode;
ub2 ol_d_id_rcode;
ub2 ol_o_id_rcode;

ub4 ol_supply_w_id_csize;
ub4 ol_i_id_csize;
ub4 ol_quantity_csize;
ub4 ol_amount_csize;
ub4 ol_delivery_d_csize;
ub4 ol_w_id_csize;
ub4 ol_d_id_csize;
ub4 ol_o_id_csize;

OCIStmt *curo0;
OCIBind *w_id_bp0;
OCIBind *d_id_bp0;
OCIBind *c_id_bp;
OCIBind *c_last_bp;
#ifdef PLSQLORD
OCIBind *byln_bp;
OCIBind *c_first_bp;
OCIBind *c_middle_bp;
OCIBind *c_balance_bp;
OCIBind *o_entry_d_bp;
OCIBind *o_cr_id_bp;
OCIBind *o_ol_cnt_bp;
OCIBind *ol_i_id_bp;
OCIBind *ol_supply_w_id_bp;
OCIBind *ol_quantity_bp;
OCIBind *ol_amount_bp;
OCIBind *ol_d_base_bp;
ub4 ol_i_id_cnt;
ub4 ol_sup_cnt;
ub4 ol_qty_cnt;
ub4 ol_amt_cnt;
ub4 ol_del_d_cnt;
#else
OCIStmt *curo1;
OCIStmt *curo2;
OCIStmt *curo3;
OCIStmt *curo4;
OCIBind *w_id_bp2;

```

```

OCIBind *w_id_bp3;
OCIBind *w_id_bp4;
OCIBind *d_id_bp2;
OCIBind *d_id_bp3;
OCIBind *d_id_bp4;
OCIBind *c_last_bp4;
OCIBind *o_id_bp;
OCIBind *c_rowid_bp;
OCIDefine *c_rowid_dp;
OCIDefine *c_last_dp;
OCIDefine *c_last_dp1;
OCIDefine *c_id_dp;
OCIDefine *c_id_dp1;
OCIDefine *c_first_dp1;
OCIDefine *c_first_dp2;
OCIDefine *c_middle_dp1;
OCIDefine *c_middle_dp2;
OCIDefine *c_balance_dp1;
OCIDefine *c_balance_dp2;
OCIDefine *o_id_dp1;
OCIDefine *o_id_dp2;
OCIDefine *o_entry_d_dp1;
OCIDefine *o_entry_d_dp2;
OCIDefine *o_cr_id_dp1;
OCIDefine *o_cr_id_dp2;
OCIDefine *o_ol_cnt_dp1;
OCIDefine *o_ol_cnt_dp2;
OCIDefine *ol_d_dp;
OCIDefine *ol_i_id_dp;
OCIDefine *ol_supply_w_id_dp;
OCIDefine *ol_quantity_dp;
OCIDefine *ol_amount_dp;
OCIDefine *ol_d_base_dp;
OCIDefine *c_count_dp;
OCIRowid *c_rowid_ptr[100];
OCIRowid *middle_cust;

int cs;
int cust_idx;
int norow;
int rcount;
int somerows;
#endif
};

typedef struct ordctx ordctx;

struct defctx
{
    boolean reexec;
    ub4 count;
};
typedef struct defctx defctx;

ordctx *octx;

defctx cbctx;

#ifdef PLSQLORD
sb4 rid_data(dvoid *ctxp, OCIDefine *dp, ub4 iter,
             dvoid **bufpp, ub4 **alenp, ub1 *piecep,
             dvoid **indpp, ub2 **rcodepp)
{
    ub4 i;

```

```

    if ((defctx*)ctxp)->reexec)/* if this is the second execute - use
entry 0 */
    {
        i = 0;
        ((defctx*)ctxp)->count--; /* count down */
    }
    else
        i = iter;
    *bufpp = octx->c_rowid_ptr[i];
    *indpp = &octx->c_rowid_ind[i];
    *alenp = &octx->c_rowid_len[i];
    *rcodepp = &octx->c_rowid_rcode[i];
    *piecep = OCI_ONE_PIECE;
    return (OCI_CONTINUE);
}
#endif

tkvcoinit ()
{
    int i;
    text stmbuf[SQL_BUF_SIZE];

    octx = (ordctx *) malloc (sizeof(ordctx));
    memset (octx, (char)0, sizeof(ordctx));
#ifdef PLSQLORD
    octx->cs = 1;
    octx->norow = 0;
    octx->somerows = 10;
/* get the rowid handles */
    for(i=0;i<100;i++) {
        OCIERROR(errhp, OCIDescriptorAlloc(tpcenv, (dvoid**) &octx-
>c_rowid_ptr[i],
        OCI_DTYPE_ROWID, 0, (dvoid**)0));
    }
#endif

    OCIERROR(errhp,
    OCIHandleAlloc(tpcenv, (dvoid**) &octx-
>curo0, OCI_HTYPE_STMT, 0, (dvoid**)0));
    OCIERROR(errhp,
    OCIHandleAlloc(tpcenv, (dvoid**) &octx-
>curo0, OCI_HTYPE_STMT, 0, (dvoid**)0));
#ifdef PLSQLORD
    OCIERROR(errhp,
    OCIHandleAlloc(tpcenv, (dvoid**) &octx-
>curo1, OCI_HTYPE_STMT, 0, (dvoid**)0));
    OCIERROR(errhp,
    OCIHandleAlloc(tpcenv, (dvoid**) &octx-
>curo2, OCI_HTYPE_STMT, 0, (dvoid**)0));
    OCIERROR(errhp,
    OCIHandleAlloc(tpcenv, (dvoid**) &octx-
>curo3, OCI_HTYPE_STMT, 0, (dvoid**)0));
    OCIERROR(errhp,
    OCIHandleAlloc(tpcenv, (dvoid**) &octx-
>curo4, OCI_HTYPE_STMT, 0, (dvoid**)0));
#endif
#ifdef PLSQLORD
    sprintf((char *) stmbuf, SQLTXT);
    OCIERROR(errhp,
    OCIStmtPrepare(octx->curo0, errhp, stmbuf, strlen((char *) stmbuf),

```

```

OCI_NTV_SYNTAX, OCI_DEFAULT));
#else
/* c_id = 0, use find customer by lastname. Get an array or rowid's
back*/
    sprintf((char *) stmbuf, SQLCUR0);
    OCIERROR(errhp,
    OCIStmtPrepare(octx->curo0, errhp, stmbuf, strlen((char *) stmbuf),
    OCI_NTV_SYNTAX, OCI_DEFAULT));
    OCIERROR(errhp,
    OCIAttrSet(octx->curo0, OCI_HTYPE_STMT, (dvoid*)&octx->norow, 0,
    OCI_ATTR_PREFETCH_ROWS, errhp));
/* get order/customer info back based on rowid */
    sprintf((char *) stmbuf, SQLCUR1);
    OCIERROR(errhp,
    OCIStmtPrepare(octx->curo1, errhp, stmbuf, strlen((char *) stmbuf),
    OCI_NTV_SYNTAX, OCI_DEFAULT));
    OCIERROR(errhp,
    OCIAttrSet(octx->curo1, OCI_HTYPE_STMT, (dvoid*)&octx->norow, 0,
    OCI_ATTR_PREFETCH_ROWS, errhp));

/* c_id == 0, use lastname to find customer */
    sprintf((char *) stmbuf, SQLCUR2);
    OCIERROR(errhp,
    OCIStmtPrepare(octx->curo2, errhp, stmbuf, strlen((char *) stmbuf),
    OCI_NTV_SYNTAX, OCI_DEFAULT));
    OCIERROR(errhp,
    OCIAttrSet(octx->curo2, OCI_HTYPE_STMT, (dvoid*)&octx->norow, 0,
    OCI_ATTR_PREFETCH_ROWS, errhp));

    sprintf((char *) stmbuf, SQLCUR3);
    OCIERROR(errhp,
    OCIStmtPrepare(octx->curo3, errhp, stmbuf, strlen((char *) stmbuf),
    OCI_NTV_SYNTAX, OCI_DEFAULT));
    OCIERROR(errhp,
    OCIAttrSet(octx->curo3, OCI_HTYPE_STMT, (dvoid*)&octx->norow, 0,
    OCI_ATTR_PREFETCH_ROWS, errhp));

    sprintf((char *) stmbuf, SQLCUR4);
    OCIERROR(errhp,
    OCIStmtPrepare(octx->curo4, errhp, stmbuf, strlen((char *) stmbuf),
    OCI_NTV_SYNTAX, OCI_DEFAULT));
    OCIERROR(errhp,
    OCIAttrSet(octx->curo4, OCI_HTYPE_STMT, (dvoid*)&octx->norow, 0,
    OCI_ATTR_PREFETCH_ROWS, errhp));
#endif

    for (i = 0; i < NITEMS; i++) {
        octx->ol_supply_w_id_ind[i] = TRUE;
        octx->ol_i_id_ind[i] = TRUE;
        octx->ol_quantity_ind[i] = TRUE;
        octx->ol_amount_ind[i] = TRUE;
        octx->ol_delivery_d_ind[i] = TRUE;

        octx->ol_supply_w_id_len[i] = sizeof(int);
        octx->ol_i_id_len[i] = sizeof(int);
        octx->ol_quantity_len[i] = sizeof(int);
        octx->ol_amount_len[i] = sizeof(int);
        octx->ol_delivery_d_len[i] = sizeof(ol_d_base[0]);
    }
    octx->ol_supply_w_id_csize = NITEMS;
    octx->ol_i_id_csize = NITEMS;
    octx->ol_quantity_csize = NITEMS;
    octx->ol_amount_csize = NITEMS;
    octx->ol_delivery_d_csize = NITEMS;

```

```

octx->ol_w_id_csize = NITEMS;
octx->ol_o_id_csize = NITEMS;
octx->ol_d_id_csize = NITEMS;
octx->ol_w_id_ind = TRUE;
octx->ol_d_id_ind = TRUE;
octx->ol_o_id_ind = TRUE;
octx->ol_w_id_len = sizeof(int);
octx->ol_d_id_len = sizeof(int);
octx->ol_o_id_len = sizeof(int);

/* bind variables */
#ifdef PLSQLORD

OCIBND(octx->curo0, octx->w_id_bp0, errhp, ":w_id",ADR(w_id),
SIZ(int),SQLT_INT);
OCIBND(octx->curo0, octx->d_id_bp0, errhp, ":d_id",ADR(d_id),
SIZ(int), SQLT_INT);
OCIBND(octx->curo0, octx->c_id_bp , errhp, ":c_id",ADR(c_id),
SIZ(c_id),SQLT_INT);
OCIBND(octx->curo0, octx->byln_bp , errhp, ":byln",ADR(bylastname),
SIZ(int),SQLT_INT);
OCIBND(octx->curo0, octx->c_last_bp , errhp, ":c_last",c_last,
SIZ(c_last),SQLT_STR);
OCIBND(octx->curo0, octx->c_first_bp , errhp, ":c_first",c_first,
SIZ(c_first),SQLT_STR);
OCIBND(octx->curo0, octx->c_middle_bp , errhp, ":c_middle",c_middle,
SIZ(c_middle),SQLT_STR);
OCIBND(octx->curo0, octx->c_balance_bp , errhp, ":c_balance",
ADR(c_balance),SIZ (float),SQLT FLT);
OCIBND(octx->curo0, octx->c_id_bp , errhp, ":o_id",ADR(o_id),
SIZ(int),SQLT INT);
OCIBND(octx->curo0, octx->o_entry_d_bp ,
errhp, ":o_entry_d",o_entry_d,
SIZ(o_entry_d),SQLT_STR);
OCIBND(octx->curo0, octx->o_cr_id_bp ,
errhp, ":o_cr_id",ADR(o_carrier_id),
SIZ(int), SQLT INT);
OCIBND(octx->curo0, octx->o_ol_cnt_bp ,
errhp, ":o_ol_cnt",ADR(o_ol_cnt),
SIZ(int),SQLT_INT);

OCIBNDRAA(octx->curo0, octx->ol_i_id_bp, errhp, ":ol_i_id",
ol_i_id,SIZ(int),SQLT_INT,
octx->ol_i_id_ind,octx->ol_i_id_len,
octx->ol_i_id_rcode,NITEMS,&octx->ol_i_id_cnt);
OCIBNDRAA(octx->curo0,octx->ol_supply_w_id_bp,errhp,":ol_s_w_id",
ol_supply_w_id,SIZ(int),SQLT_INT,
octx->ol_supply_w_id_ind,octx->ol_supply_w_id_len,
octx->ol_supply_w_id_rcode,NITEMS,&octx->ol_sup_cnt);
OCIBNDRAA(octx->curo0, octx->ol_quantity_bp,errhp,":ol_quantity",
ol_quantity,SIZ(int),SQLT_INT,
octx->ol_quantity_ind,octx->ol_quantity_len,
octx->ol_quantity_rcode,NITEMS,&octx->ol_qty_cnt);
OCIBNDRAA(octx->curo0,octx->
>ol_amount_bp,errhp,":ol_amount",ol_amount,
SIZ(float),SQLT FLT,octx->ol_amount_ind,
octx->ol_amount_len, octx->ol_amount_rcode,NITEMS,
&octx->ol_amt_cnt);
OCIBNDRAA(octx->curo0,octx->ol_d_base_bp,errhp,":ol_d_d",ol_d_base,
SIZ(OCIDate),SQLT_ODT,octx->ol_delivery_d_ind,
octx->ol_delivery_d_len, octx->ol_delivery_d_rcode,NITEMS,
&octx->ol_del_d_cnt);

#else

```

```

/* c_id (customer id) is not known */
OCIBND(octx->curo0,octx-
>w_id_bp0,errhp,":w_id",ADR(w_id),SIZ(int),SQLT_INT);
OCIBND(octx->curo0,octx-
>d_id_bp0,errhp,":d_id",ADR(d_id),SIZ(int),SQLT_INT);
OCIBND(octx->curo0,octx-
>c_last_bp,errhp,":c_last",c_last,SIZ(c_last),
SQLT_STR);
OCIDFNDYN(octx->curo0,octx->c_rowid_dp,errhp,1,octx->c_rowid_ptr,
SIZ(OCIRowid*), SQLT_RDD, octx->c_rowid_ind,
&cbctx,rid_data);

OCIBND(octx->curo1,octx->c_rowid_bp,errhp,":cust_rowid",
&octx->middle_cust, sizeof( octx->middle_cust),SQLT_RDD);

OCIDEF(octx->curo1,octx-
>c_id_dp,errhp,1,ADR(c_id),SIZ(int),SQLT_INT);
OCIDEF(octx->curo1,octx->c_balance_dp1,errhp,2,ADR(c_balance),
SIZ(double),SQLT FLT);
OCIDEF(octx->curo1,octx->c_first_dp1,errhp,3,c_first,SIZ(c_first)-1,
SQLT CHR);
OCIDEF(octx->curo1,octx->c_middle_dp1,errhp,4,c_middle,
SIZ(c_middle)-1,SQLT AFC);
OCIDEF(octx->curo1,octx->c_last_dp1,errhp,5,c_last,SIZ(c_last)-1,
SQLT CHR);
OCIDEF(octx->curo1,octx-
>o_id_dp1,errhp,6,ADR(o_id),SIZ(int),SQLT_INT);
OCIDEF(octx->curo1,octx->o_entry_d_dp1,errhp,7,
&o_entry_d_base,SIZ(OCIDate),SQLT_ODT);
OCIDEF(octx->curo1,octx->o_cr_id_dp1,errhp,8,ADR(o_carrier_id),
SIZ(int),SQLT_INT);
OCIDEF(octx->curo1,octx->o_ol_cnt_dp1,errhp,9,ADR(o_ol_cnt),
SIZ(int),SQLT_INT);

/* Bind for third cursor , no-zero customer id */
OCIBND(octx->curo2,octx-
>w_id_bp2,errhp,":w_id",ADR(w_id),SIZ(int),SQLT_INT);
OCIBND(octx->curo2,octx-
>d_id_bp2,errhp,":d_id",ADR(d_id),SIZ(int),SQLT_INT);
OCIBND(octx->curo2,octx-
>c_id_bp,errhp,":c_id",ADR(c_id),SIZ(int),SQLT_INT);
OCIDEF(octx->curo2,octx->c_balance_dp2,errhp,1,ADR(c_balance),
SIZ(double),SQLT FLT);
OCIDEF(octx->curo2,octx->c_first_dp2,errhp,2,c_first,SIZ(c_first)-1,
SQLT CHR);
OCIDEF(octx->curo2,octx->c_middle_dp2,errhp,3,c_middle,
SIZ(c_middle)-1,SQLT AFC);
OCIDEF(octx->curo2,octx->c_last_dp2,errhp,4,c_last,SIZ(c_last)-1,
SQLT CHR);
OCIDEF(octx->curo2,octx-
>o_id_dp2,errhp,5,ADR(o_id),SIZ(int),SQLT_INT);
OCIDEF(octx->curo2,octx->o_entry_d_dp2,errhp,6, &o_entry_d_base,
SIZ(OCIDate),SQLT_ODT);
OCIDEF(octx->curo2, octx->o_cr_id_dp2,errhp,7,ADR(o_carrier_id),
SIZ(int), SQLT_INT);
OCIDEF(octx->curo2,octx->o_ol_cnt_dp2,errhp,8,ADR(o_ol_cnt),
SIZ(int),SQLT_INT);
OCIDEF(octx->curo2,octx-
>c_id_dp1,errhp,9,ADR(c_id),SIZ(int),SQLT_INT);

/* Bind for last cursor */

```

```

    OCIBND(octx->curo3,octx-
>w_id_bp3,errhp,":w_id",ADR(w_id),SIZ(int),SQLT_INT);
    OCIBND(octx->curo3,octx-
>d_id_bp3,errhp,":d_id",ADR(d_id),SIZ(int),SQLT_INT);
    OCIBND(octx->curo3,octx-
>o_id_bp,errhp,":o_id",ADR(o_id),SIZ(int),SQLT_INT);

    OCIDFNRA(octx->curo3, octx->ol_i_id_dp, errhp, 1,
ol_i_id,SIZ(int),SQLT_INT,
    octx->ol_i_id_ind,octx->ol_i_id_len, octx->ol_i_id_rcode);
    OCIDFNRA(octx->curo3,octx->ol_supply_w_id_dp,errhp,2,ol_supply_w_id,
    SIZ(int),SQLT_INT, octx->ol_supply_w_id_ind,
    octx->ol_supply_w_id_len, octx->ol_supply_w_id_rcode);
    OCIDFNRA(octx->curo3, octx->ol_quantity_dp,errhp,3,
ol_quantity,SIZ(int),
    SQLT_INT, octx->ol_quantity_ind,octx->ol_quantity_len,
    octx->ol_quantity_rcode);
    OCIDFNRA(octx->curo3,octx->ol_amount_dp,errhp,4,ol_amount, SIZ(int),
    SQLT_INT,octx->ol_amount_ind, octx->ol_amount_len,
    octx->ol_amount_rcode);
    OCIDFNRA(octx->curo3,octx-
>ol_d_base_dp,errhp,5,ol_d_base,SIZ(OCIDate),
    SQLT_ODT, octx->ol_delivery_d_ind,octx->ol_delivery_d_len,
    octx->ol_delivery_d_rcode);

    OCIBND(octx->curo4,octx-
>w_id_bp4,errhp,":w_id",ADR(w_id),SIZ(int),SQLT_INT);
    OCIBND(octx->curo4,octx-
>d_id_bp4,errhp,":d_id",ADR(d_id),SIZ(int),SQLT_INT);
    OCIBND(octx->curo4,octx-
>c_last_bp4,errhp,":c_last",c_last,SIZ(c_last),
    SQLT_STR);
    OCIDF(octx->curo4,octx->c_count_dp,errhp,1,ADR(octx-
>rcount),SIZ(int),
    SQLT_INT);

#endif
    return (0);
}

tkvco ()
{
    int i;
    int rcount;

    for (i = 0; i < NITEMS; i++) {
        octx->ol_supply_w_id_ind[i] = TRUE;
        octx->ol_i_id_ind[i] = TRUE;
        octx->ol_quantity_ind[i] = TRUE;
        octx->ol_amount_ind[i] = TRUE;
        octx->ol_delivery_d_ind[i] = TRUE;
        octx->ol_supply_w_id_len[i] = sizeof(int);
        octx->ol_i_id_len[i] = sizeof(int);
        octx->ol_quantity_len[i] = sizeof(int);
        octx->ol_amount_len[i] = sizeof(int);
        octx->ol_delivery_d_len[i] = sizeof(OCIDate);
    }
    octx->ol_supply_w_id_csize = NITEMS;

```

```

    octx->ol_i_id_csize = NITEMS;
    octx->ol_quantity_csize = NITEMS;
    octx->ol_amount_csize = NITEMS;
    octx->ol_delivery_d_csize = NITEMS;
#ifdef PLSQLORD
    octx->ol_i_id_cnt = 0;
    octx->ol_sup_cnt = 0;
    octx->ol_qty_cnt = 0;
    octx->ol_amt_cnt = 0;
    octx->ol_del_d_cnt = 0;
    OCIERROR(errhp,
    OCISmtExecute(tpcsvc,octx->curo0,errhp,1,0,0,0,OCI_DEFAULT));
#else
retry:
    if (bylastname)
    {
        cbctx.reexec = FALSE;
        execstatus=OCISmtExecute(tpcsvc,octx-
>curo0,errhp,100,0,0,0,OCI_DEFAULT);
        /* will get OCI_NO_DATA if <100 found */
        if ((execstatus != OCI_NO_DATA) && (execstatus != OCI_SUCCESS))
        {
            errcode=OCIERROR(errhp, execstatus);
            if ((errcode == NOT_SERIALIZABLE) || (errcode == RECOVER))
            {
                OCITransCommit(tpcsvc,errhp,OCI_DEFAULT);
                retries++;
                goto retry;
            } else {
                return -1;
            }
        }
        if (execstatus == OCI_NO_DATA) /* there are no more rows */
        {
            /* get rowcount, find middle one */
            OCIAttrGet(octx-
>curo0,OCI_HTYPE_STMT,&rcount,NULL,OCI_ATTR_ROW_COUNT,errhp);
            if (rcount < 1)
            {
                userlog("No Data Found\n");
                return (-1);
            }
            octx->cust_idx=(rcount-1)/2 ;
        }
        else
        {
            /* count the number of rows */
            execstatus=OCISmtExecute(tpcsvc,octx-
>curo4,errhp,1,0,0,0,OCI_DEFAULT);
            if ((execstatus != OCI_NO_DATA) && (execstatus != OCI_SUCCESS))
            {
                if ((errcode == NOT_SERIALIZABLE) || (errcode == RECOVER))
                {
                    OCITransCommit(tpcsvc,errhp,OCI_DEFAULT);
                    retries++;
                    goto retry;
                } else {
                    return -1;
                }
            }
            if (octx->rcount+1 < 200 )
                octx->cust_idx=(octx->rcount-1)/2 ;
            else
                /* */
        {

```



```

    "$Header: plpay.c 7030100.1 95/07/19 14:44:59 plai Generic<base>
$ Copyr (c) 1994 Oracle";
#endif /* RCSID */

/*-----+
|          Copyright (c) 1995 Oracle Corp, Redwood Shores, CA          |
|          OPEN SYSTEMS PERFORMANCE GROUP                              |
|          All Rights Reserved                                          |
+-----+
| FILENAME                                                              |
|   plpay.c                                                            |
| DESCRIPTION                                                            |
|   OCI version (using PL/SQL stored procedure) of                    |
|   PAYMENT transaction in TPC-C benchmark.                            |
+-----+*/

#include "tpcc.h"
#include "tpccflags.h"
#include <userlog.h>

#define SQLTXT_INIT "BEGIN initpay.pay_init; END;"
#define SQLTXT_STP  "begin
payment.dopayment(:w_id,:d_id,:c_w_id,:c_d_id, \
                  :c_id,:by_lname,:h_amount,:c_last,:w_street_1,:w_st
reet_2, \
                  :w_city,:w_state,:w_zip,:d_street_1,:d_street_2,:d_
city, \
                  :d_state,:d_zip,:c_first,:c_middle,:c_street_1,
\
                  :c_street_2,:c_city,:c_state,:c_zip,:c_phone,:c_sin
ce, \
                  :c_credit,:c_credit_lim,:c_discount,:c_balance,:c_d
ata, \
                  :cr_date,:retry); end;"

struct payctx {
    OCISmt *curpi;
    OCISmt *curp0;
    OCISmt *curp1;
    OCIBind *w_id_bp;
    OCIBind *w_id_bpl;
    sb2 w_id_ind;
    ub2 w_id_len;
    ub2 w_id_rc;

    OCIBind *d_id_bp;
    OCIBind *d_id_bpl;
    sb2 d_id_ind;
    ub2 d_id_len;
    ub2 d_id_rc;

    OCIBind *c_w_id_bp;
    OCIBind *c_w_id_bpl;
    sb2 c_w_id_ind;
    ub2 c_w_id_len;
    ub2 c_w_id_rc;

    OCIBind *c_d_id_bp;
    OCIBind *c_d_id_bpl;
    sb2 c_d_id_ind;
    ub2 c_d_id_len;
    ub2 c_d_id_rc;

```

```

OCIBind *c_id_bp;
OCIBind *c_id_bpl;
sb2 c_id_ind;
ub2 c_id_len;
ub2 c_id_rc;

OCIBind *by_lname_bp;

OCIBind *h_amount_bp;
OCIBind *h_amount_bpl;
sb2 h_amount_ind;
ub2 h_amount_len;
ub2 h_amount_rc;

OCIBind *c_last_bp;
OCIBind *c_last_bpl;
sb2 c_last_ind;
ub2 c_last_len;
ub2 c_last_rc;

OCIBind *w_street_1_bp;
OCIBind *w_street_1_bpl;
sb2 w_street_1_ind;
ub2 w_street_1_len;
ub2 w_street_1_rc;

OCIBind *w_street_2_bp;
OCIBind *w_street_2_bpl;
sb2 w_street_2_ind;
ub2 w_street_2_len;
ub2 w_street_2_rc;

OCIBind *w_city_bp;
OCIBind *w_city_bpl;
sb2 w_city_ind;
ub2 w_city_len;
ub2 w_city_rc;

OCIBind *w_state_bp;
OCIBind *w_state_bpl;
sb2 w_state_ind;
ub2 w_state_len;
ub2 w_state_rc;

OCIBind *w_zip_bp;
OCIBind *w_zip_bpl;
sb2 w_zip_ind;
ub2 w_zip_len;
ub2 w_zip_rc;

OCIBind *d_street_1_bp;
OCIBind *d_street_1_bpl;
sb2 d_street_1_ind;
ub2 d_street_1_len;
ub2 d_street_1_rc;

OCIBind *d_street_2_bp;
OCIBind *d_street_2_bpl;
sb2 d_street_2_ind;
ub2 d_street_2_len;
ub2 d_street_2_rc;

OCIBind *d_city_bp;
OCIBind *d_city_bpl;

```

```

sb2 d_city_ind;
ub2 d_city_len;
ub2 d_city_rc;

OCIBind *d_state_bp;
OCIBind *d_state_bpl;
sb2 d_state_ind;
ub2 d_state_len;
ub2 d_state_rc;

OCIBind *d_zip_bp;
OCIBind *d_zip_bpl;
sb2 d_zip_ind;
ub2 d_zip_len;
ub2 d_zip_rc;

OCIBind *c_first_bp;
OCIBind *c_first_bpl;
sb2 c_first_ind;
ub2 c_first_len;
ub2 c_first_rc;

OCIBind *c_middle_bp;
OCIBind *c_middle_bpl;
sb2 c_middle_ind;
ub2 c_middle_len;
ub2 c_middle_rc;

OCIBind *c_street_1_bp;
OCIBind *c_street_1_bpl;
sb2 c_street_1_ind;
ub2 c_street_1_len;
ub2 c_street_1_rc;

OCIBind *c_street_2_bp;
OCIBind *c_street_2_bpl;
sb2 c_street_2_ind;
ub2 c_street_2_len;
ub2 c_street_2_rc;

OCIBind *c_city_bp;
OCIBind *c_city_bpl;
sb2 c_city_ind;
ub2 c_city_len;
ub2 c_city_rc;

OCIBind *c_state_bp;
OCIBind *c_state_bpl;
sb2 c_state_ind;
ub2 c_state_len;
ub2 c_state_rc;

OCIBind *c_zip_bp;
OCIBind *c_zip_bpl;
sb2 c_zip_ind;
ub2 c_zip_len;
ub2 c_zip_rc;

OCIBind *c_phone_bp;
OCIBind *c_phone_bpl;
sb2 c_phone_ind;
ub2 c_phone_len;
ub2 c_phone_rc;

```

```

OCIBind *c_since_bp;
OCIBind *c_since_bpl;
sb2 c_since_ind;
ub2 c_since_len;
ub2 c_since_rc;

OCIBind *c_credit_bp;
OCIBind *c_credit_bpl;
sb2 c_credit_ind;
ub2 c_credit_len;
ub2 c_credit_rc;

OCIBind *c_credit_lim_bp;
OCIBind *c_credit_lim_bpl;
sb2 c_credit_lim_ind;
ub2 c_credit_lim_len;
ub2 c_credit_lim_rc;

OCIBind *c_discount_bp;
OCIBind *c_discount_bpl;
sb2 c_discount_ind;
ub2 c_discount_len;
ub2 c_discount_rc;

OCIBind *c_balance_bp;
OCIBind *c_balance_bpl;
sb2 c_balance_ind;
ub2 c_balance_len;
ub2 c_balance_rc;

OCIBind *c_data_bp;
OCIBind *c_data_bpl;
sb2 c_data_ind;
ub2 c_data_len;
ub2 c_data_rc;

OCIBind *h_date_bp;
OCIBind *h_date_bpl;
sb2 h_date_ind;
ub2 h_date_len;
ub2 h_date_rc;

OCIBind *retries_bp;
OCIBind *retries_bpl;
sb2 retries_ind;
ub2 retries_len;
ub2 retries_rc;

OCIBind *cr_date_bp;
OCIBind *cr_date_bpl;
sb2 cr_date_ind;
ub2 cr_date_len;
ub2 cr_date_rc;

OCIBind *byln_bp;
sb2 byln_ind;
ub2 byln_len;
ub2 byln_rc;
};

typedef struct payctx payctx;

```

```

payctx *pctx;

int tkvcpinit (void)
{
    text stmbuf[SQL_BUF_SIZE];
    pctx = (payctx *)malloc(sizeof(payctx));
    memset(pctx, (char)0, sizeof(payctx));

/* cursor for init */
OCIERROR(errhp,OCIHandleAlloc(tpcenv, (dvoid **)&(pctx->curpi)),
    OCI_HTYPE_STMT,0,(dvoid**)0));

OCIERROR(errhp,OCIHandleAlloc(tpcenv, (dvoid **)&(pctx->curp0)),
    OCI_HTYPE_STMT,0,(dvoid**)0));
OCIERROR(errhp,OCIHandleAlloc(tpcenv, (dvoid **)&(pctx->curp1)),
    OCI_HTYPE_STMT,0,(dvoid**)0));

/* build the init statement and execute it */

sprintf((char*)stmbuf, SQLTXT_INIT);
OCIERROR(errhp,OCIStmtPrepare(pctx->curpi, errhp, stmbuf,
    strlen((char *)stmbuf), OCI_NT_V_SYNTAX, OCI_DEFAULT));
OCIERROR(errhp,
    OCIStmtExecute(tpcenv, pctx->curpi, errhp, 1, 0, 0, OCI_DEFAULT));
#ifdef PLSQLPAY
/* prepare the stub for calling plsql stored procedure */
sprintf((char*)stmbuf, SQLTXT_STP);
OCIERROR(errhp,OCIStmtPrepare(pctx->curp0, errhp, stmbuf,
    strlen((char *)stmbuf), OCI_NT_V_SYNTAX, OCI_DEFAULT));
#else
/* customer id != 0, go by last name */

sqlfile("../blocks/paynz.sql",stmbuf);
OCIERROR(errhp,OCIStmtPrepare(pctx->curp0, errhp, stmbuf,
    strlen((char *)stmbuf), OCI_NT_V_SYNTAX, OCI_DEFAULT));

/* customer id == 0, go by last name */

sqlfile("../blocks/payz.sql",stmbuf); /* sqlfile opens
$O/bench/.../blocks/... */
OCIERROR(errhp,OCIStmtPrepare(pctx->curp1, errhp, stmbuf,
    strlen((char *)stmbuf), OCI_NT_V_SYNTAX, OCI_DEFAULT));
#endif
pctx->w_id_ind = TRUE;
pctx->w_id_len = SIZ(w_id);
pctx->d_id_ind = TRUE;
pctx->d_id_len = SIZ(d_id);
pctx->c_w_id_ind = TRUE;
pctx->c_w_id_len = SIZ(c_w_id);
pctx->c_d_id_ind = TRUE;
pctx->c_d_id_len = SIZ(c_d_id);
pctx->c_id_ind = TRUE;
pctx->c_id_len = 0;
pctx->h_amount_len = SIZ(h_amount);
pctx->h_amount_ind = TRUE;
pctx->c_last_ind = TRUE;
pctx->c_last_len = 0;
pctx->w_street_1_ind = TRUE;
pctx->w_street_1_len = 0;

```

```

pctx->w_street_2_ind = TRUE;
pctx->w_street_2_len = 0;
pctx->w_city_ind = TRUE;
pctx->w_city_len = 0;
pctx->w_state_ind = TRUE;
pctx->w_state_len = 0;
pctx->w_zip_ind = TRUE;
pctx->w_zip_len = 0;
pctx->d_street_1_ind = TRUE;
pctx->d_street_1_len = 0;
pctx->d_street_2_ind = TRUE;
pctx->d_street_2_len = 0;
pctx->d_city_ind = TRUE;
pctx->d_city_len = 0;
pctx->d_state_ind = TRUE;
pctx->d_state_len = 0;
pctx->d_zip_ind = TRUE;
pctx->d_zip_len = 0;
pctx->c_first_ind = TRUE;
pctx->c_first_len = 0;
pctx->c_middle_ind = TRUE;
pctx->c_middle_len = 0;
pctx->c_street_1_ind = TRUE;
pctx->c_street_1_len = 0;
pctx->c_street_2_ind = TRUE;
pctx->c_street_2_len = 0;
pctx->c_city_ind = TRUE;
pctx->c_city_len = 0;
pctx->c_state_ind = TRUE;
pctx->c_state_len = 0;
pctx->c_zip_ind = TRUE;
pctx->c_zip_len = 0;
pctx->c_phone_ind = TRUE;
pctx->c_phone_len = 0;
pctx->c_since_ind = TRUE;
pctx->c_since_len = 0;
pctx->c_credit_ind = TRUE;
pctx->c_credit_len = 0;
pctx->c_credit_lim_ind = TRUE;
pctx->c_credit_lim_len = 0;
pctx->c_discount_ind = TRUE;
pctx->c_discount_len = 0;
pctx->c_balance_ind = TRUE;
pctx->c_balance_len = sizeof(double);
pctx->c_data_ind = TRUE;
pctx->c_data_len = 0;
pctx->h_date_ind = TRUE;
pctx->h_date_len = 0;
pctx->retries_ind = TRUE;
pctx->retries_len = 0;
pctx->cr_date_ind = TRUE;
pctx->cr_date_len = 7;

/* bind variables */

OCIBNDR(pctx->curp0, pctx->w_id_bp, errhp,":w_id",ADR(w_id),SIZ(int),
    SQLT_INT, &pctx->w_id_ind, NULL, NULL);
OCIBNDR(pctx->curp0, pctx->d_id_bp, errhp,":d_id",ADR(d_id),SIZ(int),
    SQLT_INT, &pctx->d_id_ind, NULL, NULL);
OCIBND(pctx->curp0, pctx->c_w_id_bp,
    errhp,":c_w_id",ADR(c_w_id),SIZ(int),

```

```

        SQLT_INT);
    OCIBND(pctx->curp0, pctx->c_d_id_bp,
errhp,":c_d_id",ADR(c_d_id),SIZ(int),
        SQLT_INT);
    OCIBND(pctx->curp0, pctx->c_id_bp, errhp,":c_id",ADR(c_id),SIZ(int),
        SQLT_INT);
#ifdef PLSQLPAY
    OCIBND(pctx->curp0, pctx->by_lname_bp,
errhp,":by_lname",ADR(bylastname),
        SIZ(int), SQLT_INT);
#endif
    OCIBNDR(pctx->curp0, pctx->h_amount_bp,
errhp,":h_amount",ADR(h_amount),
        SIZ(int),SQLT_INT, &pctx->h_amount_ind, &pctx->h_amount_len,
        &pctx->h_amount_rc);
    OCIBNDR(pctx->curp0, pctx->c_last_bp,
errhp,":c_last",c_last,SIZ(c_last),
        SQLT_STR, &pctx->c_last_ind, &pctx->c_last_len, &pctx-
>c_last_rc);
    OCIBNDR(pctx->curp0, pctx->w_street_1_bp,
errhp,":w_street_1",w_street_1,
        SIZ(w_street_1),SQLT_STR, &pctx->w_street_1_ind,
        &pctx->w_street_1_len, &pctx->w_street_1_rc);
    OCIBNDR(pctx->curp0, pctx->w_street_2_bp,
errhp,":w_street_2",w_street_2,
        SIZ(w_street_2),SQLT_STR, &pctx->w_street_2_ind,
        &pctx->w_street_2_len, &pctx->w_street_2_rc);
    OCIBNDR(pctx->curp0, pctx->w_city_bp,
errhp,":w_city",w_city,SIZ(w_city),
        SQLT_STR, &pctx->w_city_ind, &pctx->w_city_len, &pctx-
>w_city_rc);
    OCIBNDR(pctx->curp0, pctx->w_state_bp,
errhp,":w_state",w_state,SIZ(w_state),
        SQLT_STR, &pctx->w_state_ind, &pctx->w_state_len, &pctx-
>w_state_rc);
    OCIBNDR(pctx->curp0, pctx->w_zip_bp, errhp,":w_zip",w_zip,SIZ(w_zip),
        SQLT_STR, &pctx->w_zip_ind, &pctx->w_zip_len, &pctx-
>w_zip_rc);
    OCIBNDR(pctx->curp0, pctx->d_street_1_bp,
errhp,":d_street_1",d_street_1,
        SIZ(d_street_1),SQLT_STR, &pctx->d_street_1_ind,
        &pctx->d_street_1_len, &pctx->d_street_1_rc);
    OCIBNDR(pctx->curp0, pctx->d_street_2_bp,
errhp,":d_street_2",d_street_2,
        SIZ(d_street_2),SQLT_STR, &pctx->d_street_2_ind,
        &pctx->d_street_2_len, &pctx->d_street_2_rc);
    OCIBNDR(pctx->curp0, pctx->d_city_bp,
errhp,":d_city",d_city,SIZ(d_city),
        SQLT_STR, &pctx->d_city_ind, &pctx->d_city_len, &pctx-
>d_city_rc);
    OCIBNDR(pctx->curp0, pctx->d_state_bp,
errhp,":d_state",d_state,SIZ(d_state),
        SQLT_STR, &pctx->d_state_ind, &pctx->d_state_len, &pctx-
>d_state_rc);
    OCIBNDR(pctx->curp0, pctx->d_zip_bp, errhp,":d_zip",d_zip,SIZ(d_zip),
        SQLT_STR, &pctx->d_zip_ind, &pctx->d_zip_len, &pctx-
>d_zip_rc);
    OCIBNDR(pctx->curp0, pctx->c_first_bp,
errhp,":c_first",c_first,SIZ(c_first),
        SQLT_STR, &pctx->c_first_ind, &pctx->c_first_len, &pctx-
>c_first_rc);
    OCIBNDR(pctx->curp0, pctx->c_middle_bp, errhp,":c_middle",c_middle,2,
        SQLT_AFC, &pctx->c_middle_ind, &pctx->c_middle_len,
        &pctx->c_middle_rc);

```

```

    OCIBNDR(pctx->curp0, pctx->c_street_1_bp,
errhp,":c_street_1",c_street_1,
        SIZ(c_street_1),SQLT_STR, &pctx->c_street_1_ind,
        &pctx->c_street_1_len, &pctx->c_street_1_rc);
    OCIBNDR(pctx->curp0, pctx->c_street_2_bp,
errhp,":c_street_2",c_street_2,
        SIZ(c_street_2),SQLT_STR, &pctx->c_street_2_ind,
        &pctx->c_street_2_len, &pctx->c_street_2_rc);
    OCIBNDR(pctx->curp0, pctx->c_city_bp,
errhp,":c_city",c_city,SIZ(c_city),
        SQLT_STR, &pctx->c_city_ind, &pctx->c_city_len, &pctx-
>c_city_rc);
    OCIBNDR(pctx->curp0, pctx->c_state_bp,
errhp,":c_state",c_state,SIZ(c_state),
        SQLT_STR, &pctx->c_state_ind, &pctx->c_state_len, &pctx-
>c_state_rc);
    OCIBNDR(pctx->curp0, pctx->c_zip_bp, errhp,":c_zip",c_zip,SIZ(c_zip),
        SQLT_STR, &pctx->c_zip_ind, &pctx->c_zip_len, &pctx-
>c_zip_rc);
    OCIBNDR(pctx->curp0, pctx->c_phone_bp,
errhp,":c_phone",c_phone,SIZ(c_phone),
        SQLT_STR, &pctx-
>c_phone_ind, &pctx->c_phone_len, &pctx->c_phone_rc);
    OCIBNDR(pctx->curp0, pctx->c_since_bp, errhp,":c_since",&c_since,
        SIZ(OCIDate), SQLT_ODT, &pctx->c_since_ind, &pctx-
>c_since_len,
        &pctx->c_since_rc);
    OCIBNDR(pctx->curp0, pctx->c_credit_bp, errhp,":c_credit",c_credit,
        SIZ(c_credit),SQLT_CHR, &pctx->c_credit_ind, &pctx-
>c_credit_len,
        &pctx->c_credit_rc);
    OCIBNDR(pctx->curp0, pctx->c_credit_lim_bp, errhp,":c_credit_lim",
        ADR(c_credit_lim),SIZ(int), SQLT_INT, &pctx-
>c_credit_lim_ind,
        &pctx->c_credit_lim_len, &pctx->c_credit_lim_rc);
    OCIBNDR(pctx->curp0, pctx->c_discount_bp, errhp,":c_discount",
        ADR(c_discount),SIZ(c_discount), SQLT_FLT, &pctx-
>c_discount_ind,
        &pctx->c_discount_len, &pctx->c_discount_rc);
    OCIBNDR(pctx->curp0, pctx->c_balance_bp,
errhp,":c_balance",ADR(c_balance),
        SIZ(double),SQLT_FLT, &pctx->c_balance_ind, &pctx-
>c_balance_len,
        &pctx->c_balance_rc);
    OCIBNDR(pctx->curp0, pctx->c_data_bp,
errhp,":c_data",c_data,SIZ(c_data),
        SQLT_STR, &pctx->c_data_ind, &pctx->c_data_len, &pctx-
>c_data_rc);
    /*
    OCIBNDR(pctx->curp0, pctx->h_date_bp,
errhp,":h_date",h_date,SIZ(h_date),
        SQLT_STR, &pctx->h_date_ind, &pctx->h_date_len, &pctx-
>h_date_rc);
    */
    OCIBNDR(pctx->curp0, pctx->retries_bp,
errhp,":retry",ADR(retries),SIZ(int),
        SQLT_INT, &pctx->retries_ind, &pctx->retries_len, &pctx-
>retries_rc);
    OCIBNDR(pctx->curp0, pctx->cr_date_bp, errhp,":cr_date",ADR(cr_date),
        SIZ(OCIDate),SQLT_ODT, &pctx->cr_date_ind, &pctx-
>cr_date_len,
        &pctx->cr_date_rc);
#ifdef PLSQLPAY
    /* ---- Binds for the second cursor */

```

```

OCIBNDR(pctx->curpl, pctx->w_id_bpl,
errhp,":w_id",ADR(w_id),SIZ(int),
SQLT_INT, &pctx->w_id_ind, &pctx->w_id_len, &pctx->w_id_rc);
OCIBNDR(pctx->curpl, pctx->d_id_bpl,
errhp,":d_id",ADR(d_id),SIZ(int),
SQLT_INT, &pctx->d_id_ind, &pctx->d_id_len, &pctx->d_id_rc);
OCIBND(pctx->curpl, pctx->c_w_id_bpl,
errhp,":c_w_id",ADR(c_w_id),SIZ(int),
SQLT_INT);
OCIBND(pctx->curpl, pctx->c_d_id_bpl,
errhp,":c_d_id",ADR(c_d_id),SIZ(int),
SQLT_INT);
OCIBNDR(pctx->curpl, pctx->c_id_bpl,
errhp,":c_id",ADR(c_id),SIZ(int),
SQLT_INT, &pctx->c_id_ind, &pctx->c_id_len, &pctx->c_id_rc);
OCIBNDR(pctx->curpl, pctx->h_amount_bpl,
errhp,":h_amount",ADR(h_amount),
SIZ(int),SQLT_INT, &pctx->h_amount_ind, &pctx->h_amount_len,
&pctx->h_amount_rc);
OCIBND(pctx->curpl, pctx->c_last_bpl,
errhp,":c_last",c_last,SIZ(c_last),
SQLT_STR);
OCIBNDR(pctx->curpl, pctx->w_street_1_bpl,
errhp,":w_street_1",w_street_1,
SIZ(w_street_1),SQLT_STR, &pctx->w_street_1_ind,
&pctx->w_street_1_len, &pctx->w_street_1_rc);
OCIBNDR(pctx->curpl, pctx->w_street_2_bpl,
errhp,":w_street_2",w_street_2,
SIZ(w_street_2),SQLT_STR, &pctx->w_street_2_ind,
&pctx->w_street_2_len, &pctx->w_street_2_rc);
OCIBNDR(pctx->curpl, pctx->w_city_bpl,
errhp,":w_city",w_city,SIZ(w_city),
SQLT_STR, &pctx->w_city_ind, &pctx->w_city_len, &pctx-
>w_city_rc);
OCIBNDR(pctx->curpl, pctx->w_state_bpl,
errhp,":w_state",w_state,SIZ(w_state),
SQLT_STR, &pctx->w_state_ind, &pctx->w_state_len, &pctx-
>w_state_rc);
OCIBNDR(pctx->curpl, pctx->w_zip_bpl,
errhp,":w_zip",w_zip,SIZ(w_zip),
SQLT_STR, &pctx->w_zip_ind, &pctx->w_zip_len, &pctx-
>w_zip_rc);
OCIBNDR(pctx->curpl, pctx->d_street_1_bpl,
errhp,":d_street_1",d_street_1,
SIZ(d_street_1),SQLT_STR, &pctx->d_street_1_ind,
&pctx->d_street_1_len, &pctx->d_street_1_rc);
OCIBNDR(pctx->curpl, pctx->d_street_2_bpl,
errhp,":d_street_2",d_street_2,
SIZ(d_street_2),SQLT_STR, &pctx->d_street_2_ind,
&pctx->d_street_2_len, &pctx->d_street_2_rc);
OCIBNDR(pctx->curpl, pctx->d_city_bpl,
errhp,":d_city",d_city,SIZ(d_city),
SQLT_STR, &pctx->d_city_ind, &pctx->d_city_len, &pctx-
>d_city_rc);
OCIBNDR(pctx->curpl, pctx->d_state_bpl, errhp,":d_state",d_state,
SIZ(d_state), SQLT_STR, &pctx->d_state_ind, &pctx-
>d_state_len,
&pctx->d_state_rc);
OCIBNDR(pctx->curpl, pctx->d_zip_bpl,
errhp,":d_zip",d_zip,SIZ(d_zip),
SQLT_STR, &pctx->d_zip_ind, &pctx->d_zip_len, &pctx-
>d_zip_rc);

```

```

OCIBNDR(pctx->curpl, pctx->c_first_bpl, errhp,":c_first",c_first,
SIZ(c_first), SQLT_STR, &pctx->c_first_ind, &pctx-
>c_first_len,
&pctx->c_first_rc);
OCIBNDR(pctx->curpl, pctx->c_middle_bpl,
errhp,":c_middle",c_middle,2,
SQLT_AFC, &pctx->c_middle_ind, &pctx->c_middle_len,
&pctx->c_middle_rc);
OCIBNDR(pctx->curpl, pctx->c_street_1_bpl,
errhp,":c_street_1",c_street_1,
SIZ(c_street_1),SQLT_STR, &pctx->c_street_1_ind,
&pctx->c_street_1_len, &pctx->c_street_1_rc);
OCIBNDR(pctx->curpl, pctx->c_street_2_bpl,
errhp,":c_street_2",c_street_2,
SIZ(c_street_2),SQLT_STR, &pctx->c_street_2_ind,
&pctx->c_street_2_len, &pctx->c_street_2_rc);
OCIBNDR(pctx->curpl, pctx->c_city_bpl,
errhp,":c_city",c_city,SIZ(c_city),SQLT_STR,
&pctx->c_city_ind, &pctx->c_city_len, &pctx->c_city_rc);
OCIBNDR(pctx->curpl, pctx->c_state_bpl,
errhp,":c_state",c_state,SIZ(c_state), SQLT_STR, &pctx-
>c_state_ind, &pctx->c_state_len, &pctx->c_state_rc);
OCIBNDR(pctx->curpl, pctx->c_zip_bpl,
errhp,":c_zip",c_zip,SIZ(c_zip),
SQLT_STR, &pctx->c_zip_ind, &pctx->c_zip_len, &pctx-
>c_zip_rc);
OCIBNDR(pctx->curpl, pctx->c_phone_bpl,
errhp,":c_phone",c_phone,SIZ(c_phone),
SQLT_STR, &pctx->c_phone_ind, &pctx->c_phone_len, &pctx-
>c_phone_rc);
OCIBNDR(pctx->curpl, pctx->c_since_bpl, errhp,":c_since",&c_since,
SIZ(OCIDate), SQLT_ODT, &pctx->c_since_ind, &pctx-
>c_since_len,
&pctx->c_since_rc);
OCIBNDR(pctx->curpl, pctx->c_credit_bpl, errhp,":c_credit",c_credit,
SIZ(c_credit),SQLT_CHR, &pctx->c_credit_ind, &pctx-
>c_credit_len,
&pctx->c_credit_rc);
OCIBNDR(pctx->curpl, pctx->c_credit_lim_bpl, errhp,":c_credit_lim",
ADR(c_credit_lim),SIZ(int), SQLT_INT, &pctx-
>c_credit_lim_ind,
&pctx->c_credit_lim_len, &pctx->c_credit_lim_rc);
OCIBNDR(pctx->curpl, pctx->c_discount_bpl, errhp,":c_discount",
ADR(c_discount),SIZ(c_discount), SQLT_FLT, &pctx-
>c_discount_ind,
&pctx->c_discount_len, &pctx->c_discount_rc);
OCIBNDR(pctx->curpl, pctx->c_balance_bpl,
errhp,":c_balance",ADR(c_balance),
SIZ(double),SQLT_FLT, &pctx->c_balance_ind, &pctx-
>c_balance_len,
&pctx->c_balance_rc);
OCIBNDR(pctx->curpl, pctx->c_data_bpl,
errhp,":c_data",c_data,SIZ(c_data),
SQLT_STR, &pctx->c_data_ind, &pctx->c_data_len, &pctx-
>c_data_rc);
/*
OCIBNDR(pctx->curpl, pctx->h_date_bpl,
errhp,":h_date",h_date,SIZ(h_date),
SQLT_STR, &pctx->h_date_ind, &pctx->h_date_len, &pctx-
>h_date_rc);
*/
OCIBNDR(pctx->curpl, pctx->retries_bpl,
errhp,":retry",ADR(retries),SIZ(int),

```

```

        SQLT_INT, &pctx->retries_ind, &pctx->retries_len, &pctx-
>retries_rc);
    OCIBNDR(pctx->curpl, pctx->cr_date_bpl,
errhp, "cr_date",ADR(cr_date),
        SIZ(OCIDate),SQLT_ODT, &pctx->cr_date_ind, &pctx-
>cr_date_len,
        &pctx->cr_date_rc);
#endif

    return (0);
}

tkvcpl ()
{
retry:
    pctx->w_id_ind = TRUE;
    pctx->w_id_len = SIZ(w_id);
    pctx->d_id_ind = TRUE;
    pctx->d_id_len = SIZ(d_id);
    pctx->c_w_id_ind = TRUE;
    pctx->c_w_id_len = 0;
    pctx->c_d_id_ind = TRUE;
    pctx->c_d_id_len = 0;
    pctx->c_id_ind = TRUE;
    pctx->c_id_len = 0;
    pctx->h_amount_len = SIZ(h_amount);
    pctx->h_amount_ind = TRUE;
    pctx->c_last_ind = TRUE;
    pctx->c_last_len = SIZ(c_last);
    pctx->w_street_1_ind = TRUE;
    pctx->w_street_1_len = 0;
    pctx->w_street_2_ind = TRUE;
    pctx->w_street_2_len = 0;
    pctx->w_city_ind = TRUE;
    pctx->w_city_len = 0;
    pctx->w_state_ind = TRUE;
    pctx->w_state_len = 0;
    pctx->w_zip_ind = TRUE;
    pctx->w_zip_len = 0;
    pctx->d_street_1_ind = TRUE;
    pctx->d_street_1_len = 0;
    pctx->d_street_2_ind = TRUE;
    pctx->d_street_2_len = 0;
    pctx->d_city_ind = TRUE;
    pctx->d_city_len = 0;
    pctx->d_state_ind = TRUE;
    pctx->d_state_len = 0;
    pctx->d_zip_ind = TRUE;
    pctx->d_zip_len = 0;
    pctx->c_first_ind = TRUE;
    pctx->c_first_len = 0;
    pctx->c_middle_ind = TRUE;
    pctx->c_middle_len = 0;
    pctx->c_street_1_ind = TRUE;
    pctx->c_street_1_len = 0;
    pctx->c_street_2_ind = TRUE;
    pctx->c_street_2_len = 0;
    pctx->c_city_ind = TRUE;

```

```

    pctx->c_city_len = 0;
    pctx->c_state_ind = TRUE;
    pctx->c_state_len = 0;
    pctx->c_zip_ind = TRUE;
    pctx->c_zip_len = 0;
    pctx->c_phone_ind = TRUE;
    pctx->c_phone_len = 0;
    pctx->c_since_ind = TRUE;
    pctx->c_since_len = 0;
    pctx->c_credit_ind = TRUE;
    pctx->c_credit_len = 0;
    pctx->c_credit_lim_ind = TRUE;
    pctx->c_credit_lim_len = 0;
    pctx->c_discount_ind = TRUE;
    pctx->c_discount_len = 0;
    pctx->c_balance_ind = TRUE;
    pctx->c_balance_len = sizeof(double);
    pctx->c_data_ind = TRUE;
    pctx->c_data_len = 0;
    pctx->h_date_ind = TRUE;
    pctx->h_date_len = 0;
    pctx->retries_ind = TRUE;
    pctx->retries_len = 0;
    pctx->cr_date_ind = TRUE;
    pctx->cr_date_len = 7;

#ifdef PLSQLPAY
    execstatus=OCISmtExecute(tpcsvc,pctx-
>curp0,errhp,1,0,0,0,OCI_DEFAULT|OCI_COMMIT_ON_SUCCESS);
#else
    if (bylastname) {
        execstatus=OCISmtExecute(tpcsvc,pctx-
>curp1,errhp,1,0,0,0,OCI_DEFAULT|OCI_COMMIT_ON_SUCCESS);
    } else {
        execstatus=OCISmtExecute(tpcsvc,pctx-
>curp0,errhp,1,0,0,0,OCI_DEFAULT|OCI_COMMIT_ON_SUCCESS);
    }
#endif

    if (execstatus != OCI_SUCCESS) {
        OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
        errcode = OCIERROR(errhp,execstatus);

        if (errcode == NOT_SERIALIZABLE) {
            retries++;
            goto retry;
        } else if (errcode == RECOVERERR) {
            retries++;
            goto retry;
        } else {
            return -1;
        }
    }
    return 0;
}

void tkvcpldone ()
{
    if (pctx) {
        free(pctx);
    }
}

```

```
}
```

PLSTO.C

```
#ifndef RCSID
static char *RCSid =
"$Header: plsto.c 7010000.3 95/02/14 12:48:03 plai Generic<base>
$ Copyr (c) 1994 Oracle";
#endif /* RCSID */

/*-----+
|           Copyright (c) 1994 Oracle Corp, Redwood Shores, CA           |
|           OPEN SYSTEMS PERFORMANCE GROUP                               |
|           All Rights Reserved                                           |
+-----+
| FILENAME                                                                |
|   plsto.c                                                                |
| DESCRIPTION                                                              |
|   OCI version of STOCK LEVEL transaction in TPC-C benchmark.          |
+-----*/

#include "tpcc.h"
#include "tpccflags.h"

#ifdef PLSQLSTO
#define SQLTXT "BEGIN stocklevel.getstocklevel
(:w_id, :d_id, :threshold, \
:low_stock); END;"
#else
#define SQLTXT "SELECT count (DISTINCT s_i_id) \
FROM ordl, stok, dist \
WHERE d_id = :d_id AND d_w_id = :w_id AND \
d_id = ol_d_id AND d_w_id = ol_w_id AND \
ol_i_id = s_i_id AND ol_w_id = s_w_id AND \
s_quantity < :threshold AND \
ol_o_id BETWEEN (d_next_o_id - 20) AND (d_next_o_id -
1)"
/* query using functional index */
/*
#define SQLTXT "SELECT count (DISTINCT s_i_id) \
FROM ordl, stok, dist \
WHERE d_id = :d_id AND d_w_id = :w_id AND \
d_id = ol_d_id AND d_w_id = ol_w_id AND \
ol_o_id BETWEEN (d_next_o_id - 20) AND (d_next_o_id -
1) AND \
decode(SIGN(s_quantity -21) , -1, s_w_id*100000 +
s_i_id, NULL) \
= ol_w_id*100000 +
ol_i_id AND \
s_quantity < :threshold;"
*/
#endif

struct stoctx {
OCIStmt *curs;
OCIBind *w_id_bp;
OCIBind *d_id_bp;
OCIBind *threshold_bp;
}
```

```
#ifdef PLSQLSTO
OCIBind *low_stock_bp;
#else
OCIDefine *low_stock_bp;
#endif
int norow;
};

typedef struct stoctx stoctx;

stoctx *sctx;

tkvcsinit ()
{
text stmbuf[SQL_BUF_SIZE];
sctx = (stoctx *)malloc(sizeof(stoctx));
memset(sctx, (char)0, sizeof(stoctx));

sctx->norow=0;

OCIERROR(errhp,
OCIHandleAlloc(tpcenv, (dvoid**) &sctx-
>curs, OCI_HTYPE_STMT, 0, (dvoid**) 0));
sprintf((char *) stmbuf, SQLTXT);
OCIERROR(errhp, OCIStmtPrepare(sctx->curs, errhp, stmbuf, strlen((char
*) stmbuf), OCI_NTV_SYNTAX, OCI_DEFAULT));
#ifdef PLSQLSTO
OCIERROR(errhp,
OCIAttrSet(sctx->curs, OCI_HTYPE_STMT, (dvoid*) &sctx->norow, 0,
OCI_ATTR_PREFETCH_ROWS, errhp));
#endif

/* bind variables */

OCIBND(sctx->curs, sctx->w_id_bp, errhp, ":w_id",
ADR(w_id), sizeof(int),
SQLT_INT);
OCIBND(sctx->curs, sctx->d_id_bp, errhp, ":d_id",
ADR(d_id), sizeof(int),
SQLT_INT);
OCIBND(sctx->curs, sctx->threshold_bp, errhp, ":threshold",
ADR(threshold),
sizeof(int), SQLT_INT);
#ifdef PLSQLSTO
OCIBND(sctx->curs, sctx->low_stock_bp, errhp, ":low_stock" ,
ADR(low_stock),
sizeof(int), SQLT_INT);
#else
OCIDEFINE(sctx->curs, sctx->low_stock_bp, errhp, 1, ADR(low_stock),
sizeof(int), SQLT_INT);
#endif

return (0);
}

tkvcs ()
```

```

{
retry:
    execstatus= OCISstmtExecute(tpcsvc,sctx->curs,errhp,1,0,0,0,
                                OCI_COMMIT_ON_SUCCESS | OCI_DEFAULT);

    if (execstatus != OCI_SUCCESS)
    {
        errcode=OCIERROR(errhp,execstatus);
        OCITransCommit(tpcsvc,errhp,OCI_DEFAULT);
        errcode=OCIERROR(errhp,execstatus);
        if((errcode == NOT_SERIALIZABLE) || (errcode == RECOVER))
        {
            retries++;
            goto retry;
        } else {
            return -1;
        }
    }

    return (0);
}

void tkvcstdone ()
{
    if(sctx) free(sctx);
}

```

TPCCPL.C

```

#ifdef RCSID
static char *RCSid =
    "$Header: tpccpl.c 7030100.2 96/04/02 17:51:34 plai Generic<base>
$ Copyr (c) 1994 Oracle";
#endif /* RCSID */

/*-----+
|          Copyright (c) 1994  Oracle Corp, Redwood Shores, CA          |
|                               OPEN SYSTEMS PERFORMANCE GROUP          |
|                               All Rights Reserved                      |
+-----+
| FILENAME                                                                |
|   tpccpl.c                                                             |
| DESCRIPTION                                                             |
|   TPC-C transactions in PL/SQL.                                        |
+-----*/

#include <stdio.h>
#include <time.h>
#include "tpcc.h"
#ifdef TUX
#include "tpcc_info.h"
#include <userlog.h>
#else
#include <stdarg.h>
#endif

```

```

#define SQLTXT "alter session set isolation_level = serializable"
#define SQLTXTTRC "alter session set sql_trace = true"
#define SQLTXTTIM "alter session set timed_statistics = true"

FILE *lfp;
FILE *fopen ();
#ifdef ORA_NT
#undef boolean
#include "dpbcore.h"
#define gettime dpbtimef
#else
extern double gettime ();
#endif
int proc_no = 0;
int logon = 0;
int new_init = 0;
int pay_init = 0;
int ord_init = 0;
int del_init = 0;
int sto_init = 0;
int res_init = 0;

int execstatus;
int errcode;

OCIEnv *tpcenv;
OCIServer *tpcsrv;
OCIError *errhp;
OCISvcCtx *tpcsvc;
OCISession *tpcusr;
OCIStmt *curs;

/* for stock-level transaction */

int w_id;
int d_id;
int c_id;
int threshold;
int low_stock;

/* for delivery transaction */

int del_o_id[10];
int retries;

/* for order-status transaction */

int bylastname;
char c_last[17];
char c_first[17];
char c_middle[3];
double c_balance;
int o_id;
text o_entry_d[20];
ub4 datelen;
int o_carrier_id;
int o_ol_cnt;
int ol_supply_w_id[15];
int ol_i_id[15];
int ol_quantity[15];
int ol_amount[15];
ub4 ol_del_len[15];
text ol_delivery_d[15][11];

```

```

/* for payment transaction */

int c_w_id;
int c_d_id;
int h_amount;
char w_street_1[21];
char w_street_2[21];
char w_city[21];
char w_state[3];
char w_zip[10];
char d_street_1[21];
char d_street_2[21];
char d_city[21];
char d_state[3];
char d_zip[10];
char c_street_1[21];
char c_street_2[21];
char c_city[21];
char c_state[3];
char c_zip[10];
char c_phone[17];
ub4 s_inclen;
text c_since_d[11];
float c_discount;
char c_credit[3];
int c_credit_lim;
char c_data[201];
ub4 hlen;
text h_date[20];

/* for new order transaction */

int nol_i_id[15];
int nol_supply_w_id[15];
int nol_quantity[15];
int nol_quant10[15];
int nol_quant19[15];
int nol_ytdqty[15];
int nol_amount[15];
int o_all_local;
float w_tax;
float d_tax;
float total_amount;
char i_name[15][25];
int s_quantity[15];
char brand_gen[15];
int i_price[15];
char brand_generic[15][1];
int status;
int tracelevel = 0;

OCIDate cr_date;
OCIDate c_since;
OCIDate o_entry_d_base;
OCIDate ol_d_base[15];
dvoid *xmem;

/*
extern char oracle_home[256];
*/

/* NewOrder Binding stuff */

```

```

#ifndef TUX
void userlog (char* fmt, ...)
{
    va_list va;
    va_start (va, fmt);
    vfprintf(stderr, fmt, va);
    va_end(va);
}
#endif

/* vmm313 void ocierror(fname, lineno, errhp, status) */
int ocierror(fname, lineno, errhp, status)
char *fname;
int lineno;
OCIError *errhp;
sword status;
{
    text errbuf[512];
    sb4 errcode;
    sb4 lstat;
    ub4 recno=2;

    switch (status)
    case OCI_SUCCESS:
        break;
    case OCI_SUCCESS_WITH_INFO:
#ifdef TUX
        userlog ("Module %s Line %d\n", fname, lineno);
        userlog ("Error - OCI_SUCCESS_WITH_INFO\n");
#else
        fprintf(stderr, "Module %s Line %d\n", fname, lineno);
        fprintf(stderr, "Error - OCI_SUCCESS_WITH_INFO\n");
#endif
        lstat = OCIErrorGet (errhp, recno++, (text *) NULL, &errcode,
            errbuf,
                (ub4) sizeof(errbuf), OCI_HTYPE_ERROR);
#ifdef TUX
        userlog ("Error - %s\n", errbuf);
#else
        fprintf(stderr, "Error - %s\n", errbuf);
#endif
        break;
    case OCI_NEED_DATA:
#ifdef TUX
        userlog ("Module %s Line %d\n", fname, lineno);
        userlog ("Error - OCI_NEED_DATA\n");
#else
        fprintf(stderr, "Module %s Line %d\n", fname, lineno);
        fprintf(stderr, "Error - OCI_NEED_DATA\n");
#endif
        return (IRRECERR);
    case OCI_NO_DATA:
#ifdef TUX
        userlog ("Module %s Line %d\n", fname, lineno);
        userlog ("Error - OCI_NO_DATA\n");
#else
        fprintf(stderr, "Module %s Line %d\n", fname, lineno);
        fprintf(stderr, "Error - OCI_NO_DATA\n");
#endif
        return (IRRECERR);
    case OCI_ERROR:
        lstat = OCIErrorGet (errhp, (ub4) 1,

```

```

        (text *) NULL, &errcode, errbuf,
        (ub4) sizeof(errbuf), OCI_HTYPE_ERROR);
    if (errcode == NOT_SERIALIZABLE) return (errcode);
    while (lstat != OCI_NO_DATA)
    {
#ifdef TUX
        userlog ("Module %s Line %d\n", fname, lineno);
        userlog ("Error - %s\n", errbuf);
#else
        fprintf(stderr,"Module %s Line %d\n", fname, lineno);
        fprintf(stderr,"Error - %s\n", errbuf);
#endif
        lstat = OCIErrorGet (errhp, recno++, (text *) NULL, &errcode,
errbuf,
                        (ub4) sizeof(errbuf), OCI_HTYPE_ERROR);
    }
    return (errcode);
/* vmm313 TPCexit(1); */
/* vmm313 exit(1); */
    case OCI_INVALID_HANDLE:
#ifdef TUX
        userlog ("Module %s Line %d\n", fname, lineno);
        userlog ("Error - OCI_INVALID_HANDLE\n");
#else
        fprintf(stderr,"Module %s Line %d\n", fname, lineno);
        fprintf(stderr,"Error - OCI_INVALID_HANDLE\n");
#endif
        TPCexit(1);
        exit(-1);
    case OCI_STILL_EXECUTING:
#ifdef TUX
        userlog ("Module %s Line %d\n", fname, lineno);
        userlog ("Error - OCI_STILL_EXECUTE\n");
#else
        fprintf(stderr,"Module %s Line %d\n", fname, lineno);
        fprintf(stderr,"Error - OCI_STILL_EXECUTE\n");
#endif
        return (IRRECERR);
    case OCI_CONTINUE:
#ifdef TUX
        userlog ("Module %s Line %d\n", fname, lineno);
        userlog ("Error - OCI_CONTINUE\n");
#else
        fprintf(stderr,"Module %s Line %d\n", fname, lineno);
        fprintf(stderr,"Error - OCI_CONTINUE\n");
#endif
        return (IRRECERR);
    default:
#ifdef TUX
        userlog ("Module %s Line %d\n", fname, lineno);
        userlog ("Status - %s\n", status);
#else
        fprintf(stderr,"Module %s Line %d\n", fname, lineno);
        fprintf(stderr,"Status - %s\n", status);
#endif
        return (IRRECERR);
    }
    return (RECOVERERR);
}

FILE *vopen (fname,mode)
char *fname;
char *mode;

```

```

{
FILE *fd;

#ifdef DEBUG
    fprintf(stderr, "tkvuoopen() fname: %s, mode: %s\n", fname, mode);
#endif

    fd = fopen((char *) fname, (char *)mode);
    if (!fd){
#ifdef TUX
        userlog (" fopen on %s failed %d\n",fname,fd);
#else
        fprintf(stderr," fopen on %s failed %d\n",fname,fd);
#endif //TUX
        exit(-1);
    }
    return(fd);
}

int sqlfile(fname,linebuf)
char *fname;
text *linebuf;
{
FILE *fd;
int nulpt = 0;
char realfile[512];

#ifdef DEBUG
    fprintf(stderr, "sqlfile() fname: %s, linebuf: %x\n", fname,
linebuf);
#endif

/*
    sprintf(realfile,"%s/bench/tpc/tpcc/blocks/%s",oracle_home, fname);
*/
    sprintf(realfile,"%s",fname);
    fd = vopen(realfile,"r");
    while (fgets((char *)linebuf+nulpt, SQL_BUF_SIZE,fd))
    {
        nulpt = strlen((char *)linebuf);
    }
    return(nulpt);
}

#ifdef NOT
void vgetdate (unsigned char *oradt)
{
    struct tm *loctime;
    time_t int_time;

    struct ORADATE {
        unsigned char century;
        unsigned char year;
        unsigned char month;
        unsigned char day;
        unsigned char hour;
        unsigned char minute;
        unsigned char second;
    } Date;
    int century;
    int cnvrtOK;

/* assume convert is successful */
    cnvrtOK = 1;

```

```

/* get the current date and time as an integer */
time( &int_time);

/* Convert the current date and time into local time */
loctime = localtime( &int_time);

century = (1900+loctime->tm_year) / 100;

Date.century = (unsigned char)(century + 100);
if (Date.century < 119 || Date.century > 120) cnvrtOK = 0;
Date.year = (unsigned char)(loctime->tm_year+100);
if (Date.year < 100 || Date.year > 199) cnvrtOK = 0;
Date.month = (unsigned char)(loctime->tm_mon + 1);
if (Date.month < 1 || Date.month > 12) cnvrtOK = 0;
Date.day = (unsigned char)loctime->tm_mday;
if (Date.day < 1 || Date.day > 31) cnvrtOK = 0;
Date.hour = (unsigned char)(loctime->tm_hour + 1);
if (Date.hour < 1 || Date.hour > 24) cnvrtOK = 0;
Date.minute = (unsigned char)(loctime->tm_min + 1);
if (Date.minute < 1 || Date.minute > 60) cnvrtOK = 0;
Date.second = (unsigned char)(loctime->tm_sec + 1);
if (Date.second < 1 || Date.second > 60) cnvrtOK = 0;

if (cnvrtOK)
    memcpy(oracle, &Date, 7);
else
    *oracle = '\0';

return;
}

void cvtdmy (unsigned char *oracle, char *outdate)
{
    struct ORADATE {
        unsigned char    century;
        unsigned char    year;
        unsigned char    month;
        unsigned char    day;
        unsigned char    hour;
        unsigned char    minute;
        unsigned char    second;
    } Date;

    int day, month, year;

    memcpy(&Date, oracle, 7);

    year = (Date.century-100)*100 + Date.year-100;
    month = Date.month;
    day = Date.day;
    sprintf(outdate, "%02d-%02d-%4d", day, month, year);

    return;
}

void cvtdmyhms (unsigned char *oracle, char *outdate)
{
    struct ORADATE {
        unsigned char    century;

```

```

        unsigned char    year;
        unsigned char    month;
        unsigned char    day;
        unsigned char    hour;
        unsigned char    minute;
        unsigned char    second;
    } Date;

    int day, month, year;
    int hour, min, sec;

    memcpy(&Date, oracle, 7);

    year = (Date.century-100)*100 + Date.year-100;
    month = Date.month;
    day = Date.day;
    hour = Date.hour - 1;
    min = Date.minute - 1;
    sec = Date.second - 1;

    sprintf(outdate, "%02d-%02d-%4d %02d:%02d:%02d",
        day, month, year, hour, min, sec);

    return;
}
#endif

void TPCexit (void)
{
    if (new_init) {
        tkvcndone();
        new_init = 0;
    }
    if (pay_init) {
        tkvcpdone();
        pay_init = 0;
    }
    if (ord_init) {
        tkvcodone();
        ord_init = 0;
    }
    if (del_init) {
        tkvcddone();
        del_init = 0;
    }
    if (sto_init) {
        tkvcsdone();
        sto_init = 0;
    }
}

OCIHandleFree((dvoid *)tpcusr, OCI_HTYPE_SESSION);
OCIHandleFree((dvoid *)tpcsvc, OCI_HTYPE_SVCCTX);
OCIHandleFree((dvoid *)errhp, OCI_HTYPE_ERROR);
OCIHandleFree((dvoid *)tpcsrv, OCI_HTYPE_SERVER);
OCIHandleFree((dvoid *)tpcenv, OCI_HTYPE_ENV);

if (lfp) {
    fclose (lfp);
    lfp = NULL;
}

```

```

    }
}

TPCinit (id, uid, pwd)

int id;
char *uid;
char *pwd;

{
    char filename[40];
    text stmbuf[100];

    proc_no = id;
    sprintf (filename, "tpcc%d.del", proc_no);
    if ((lfp = fopen (filename, "w")) == NULL) {
#ifdef TUX
        userlog ("Error in TPC-C server %d: Failed to open %s\n",
            proc_no, filename);
#else
        fprintf (stderr, "Error in TPC-C server %d: Failed to open %s\n",
            proc_no, filename);
#endif
        return (-1);
    }

    OCIInitialize(OCI_DEFAULT|OCI_OBJECT, (dvoid *)0,0,0,0);
    OCIEnvInit(&tpcenv, OCI_DEFAULT, 0, (dvoid **)0);
    OCIHandleAlloc((dvoid *)tpcenv, (dvoid **)&tpcsrv, OCI_HTYPE_SERVER,
0, (dvoid **)0);
    OCIHandleAlloc((dvoid *)tpcenv, (dvoid **)&errhp, OCI_HTYPE_ERROR,
0, (dvoid **)0);
    OCIHandleAlloc((dvoid *)tpcenv, (dvoid **)&tpcsvc, OCI_HTYPE_SVCCTX,
0, (dvoid **)0);
    OCIServerAttach(tpcsrv, errhp, (text *)0,0,OCI_DEFAULT);
    OCIAttrSet((dvoid *)tpcsvc, OCI_HTYPE_SVCCTX, (dvoid *)tpcsrv,
(ub4)0,OCI_ATTR_SERVER, errhp);
    OCIHandleAlloc((dvoid *)tpcenv, (dvoid **)&tpcusr, OCI_HTYPE_SESSION,
0, (dvoid **)0);
    OCIAttrSet((dvoid *)tpcusr, OCI_HTYPE_SESSION, (dvoid *)uid,
(ub4)strlen(uid),OCI_ATTR_USERNAME, errhp);
    OCIAttrSet((dvoid *)tpcusr, OCI_HTYPE_SESSION, (dvoid *)pwd,
(ub4)strlen(pwd),
        OCI_ATTR_PASSWORD, errhp);
    OCIERROR(errhp, OCISessionBegin(tpcsvc, errhp, tpcusr,
OCI_CRED_RDBMS, OCI_DEFAULT));

    OCIAttrSet(tpcsvc, OCI_HTYPE_SVCCTX, tpcusr, 0, OCI_ATTR_SESSION,
errhp);

    /* run all transaction in serializable mode */

    OCIHandleAlloc(tpcenv, (dvoid **)&curi, OCI_HTYPE_STMT, 0,
(dvoid **)0);
    sprintf ((char *) stmbuf, SQLTXT);
    OCISstmtPrepare (curi, errhp, stmbuf, strlen((char *)stmbuf),
OCI_NTV_SYNTAX, OCI_DEFAULT);
    OCIERROR(errhp,OCISstmtExecute(tpcsvc, curi,
errhp,1,0,0,0,OCI_DEFAULT));
    OCIHandleFree (curi, OCI_HTYPE_STMT);

```

```

/*
This is done in cvdrv.c
if (tracelevel == 2) {
    OCIHandleAlloc(tpcenv, (dvoid **)&curi, OCI_HTYPE_STMT, 0,
(dvoid **)0);
    memset(stmbuf,0,100);
    sprintf ((char *) stmbuf, SQLTXTTRC);
    OCISstmtPrepare (curi, errhp, stmbuf, strlen((char *)stmbuf),
        OCI_NTV_SYNTAX, OCI_DEFAULT);
    OCIERROR(errhp, OCISstmtExecute(tpcsvc, curi,
errhp,1,0,0,0,OCI_DEFAULT));
    OCIHandleFree((dvoid *)curi, OCI_HTYPE_STMT);
}
*/
if (tracelevel == 3) {
    OCIHandleAlloc(tpcenv, (dvoid **)&curi, OCI_HTYPE_STMT, 0,
(dvoid **)0);
    memset(stmbuf,0,100);
    sprintf ((char *) stmbuf, SQLTXTTIM);
    OCISstmtPrepare (curi, errhp, stmbuf, strlen((char *)stmbuf),
        OCI_NTV_SYNTAX, OCI_DEFAULT);
    OCIERROR(errhp, OCISstmtExecute(tpcsvc, curi,
errhp,1,0,0,0,OCI_DEFAULT));
    OCIHandleFree((dvoid *)curi, OCI_HTYPE_STMT);
}

logon = 1;

OCIERROR(errhp,OCIDateSysDate (errhp,&cr_date));

if (tkvcninit ()) { /* new order */
    TPCexit ();
    return (-1);
}
else
    new_init = 1;

if (tkvcpinit ()) { /* payment */
    TPCexit ();
    return (-1);
}
else
    pay_init = 1;

if (tkvcoint ()) { /* order status */
    TPCexit ();
    return (-1);
}
else
    ord_init = 1;

if (tkvcdinit ()) { /* delivery */
    TPCexit ();
    return (-1);
}
else
    del_init = 1;

if (tkvcsinit ()) { /* stock level */
    TPCexit ();
    return (-1);
}

```

```

    }
    else
        sto_init = 1;

    return (0);
}

TPCnew (str)
struct newstruct *str;
{
    int i;

    w_id = str->newin.w_id;
    d_id = str->newin.d_id;
    c_id = str->newin.c_id;
    for (i = 0; i < 15; i++) {
        nol_i_id[i] = str->newin.ol_i_id[i];
        nol_supply_w_id[i] = str->newin.ol_supply_w_id[i];
        nol_quantity[i] = str->newin.ol_quantity[i];
    }
    retries = 0;
/*
    vgetdate(cr_date); */
    OCIERROR(errhp, OCIDateSysDate(errhp, &cr_date));

    if (str->newout.terror = tkvcn ()) {
        if (str->newout.terror != RECOVERR)
            str->newout.terror = IRRECERR;
        return (-1);
    }

/* fill in date for o_entry_d from time in beginning of txn*/
/*
    cvtdmyhms(cr_date, o_entry_d);
*/
    datelen = sizeof(o_entry_d);
    OCIERROR(errhp,
        OCIDateToText(errhp, &cr_date, (text*) FULLDATE, SIZ(FULLDATE), (text*)0, 0,
            &datelen, o_entry_d));

    str->newout.terror = NOERR;
    str->newout.o_id = o_id;
    str->newout.o_ol_cnt = o_ol_cnt;
    strncpy (str->newout.c_last, c_last, 17);
    strncpy (str->newout.c_credit, c_credit, 3);
    str->newout.c_discount = c_discount;
    str->newout.w_tax = (float)(w_tax);
    str->newout.d_tax = (float)(d_tax);
    strncpy (str->newout.o_entry_d, (char*)o_entry_d, 20);
    str->newout.total_amount = total_amount;
    for (i = 0; i < o_ol_cnt; i++) {
        strncpy (str->newout.i_name[i], i_name[i], 25);
        str->newout.s_quantity[i] = s_quantity[i];
        str->newout.brand_generic[i] = brand_generic[i][0];
        str->newout.i_price[i] = (float)(i_price[i])/100;
        str->newout.ol_amount[i] = (float)(nol_amount[i])/100;

```

```

    }
    if (status)
#ifdef TUX
        {
            strcpy (str->newout.status, "Item number is not valid");
            return (0);
        }
#else
        strcpy (str->newout.status, "Item number is not valid");
#endif
    else
        str->newout.status[0] = '\0';
    str->newout.retry = retries;
#ifdef TUX
    return(1);
#else
    return (0);
#endif
}

TPCpay (str)
struct paystruct *str;
{
    w_id = str->payin.w_id;
    d_id = str->payin.d_id;
    c_w_id = str->payin.c_w_id;
    c_d_id = str->payin.c_d_id;
    h_amount = str->payin.h_amount;
    bylastname = str->payin.bylastname;

/*
    vgetdate(cr_date); */
    OCIERROR(errhp, OCIDateSysDate(errhp, &cr_date));

    if (bylastname) {
        c_id = 0;
        strncpy (c_last, str->payin.c_last, 17);
    }
    else {
        c_id = str->payin.c_id;
        strncpy (c_last, " ");
    }
    retries = 0;

    if (str->payout.terror = tkvcn ()) {
        if (c_id == 0)
            return (0);
        //by katsuf, for no such cust
        else if (str->payout.terror != RECOVERR)
            str->payout.terror = IRRECERR;

        return (-1);
    }

/*
    cvtdmyhms(cr_date, h_date);
*/
    hlen=SIZ(h_date);
    OCIERROR(errhp, OCIDateToText(errhp, &cr_date,
        (text*) FULLDATE, strlen(FULLDATE), (text*)0, 0, &hlen, h_date
    e));

```

```

/*
cvtdmy(c_since,c_since_d);
*/
    sincelen=SZ(c_since_d);
    OCIERROR(errhp,OCIDateToText(errhp,&c_since,
    (text*)SHORTDATE,strlen(SHORTDATE),(text*)0,0,&sincelen,c_si
nce_d));

str->payout.terror = NOERR;
strncpy (str->payout.w_street_1, w_street_1, 21);
strncpy (str->payout.w_street_2, w_street_2, 21);
strncpy (str->payout.w_city, w_city, 21);
strncpy (str->payout.w_state, w_state, 3);
strncpy (str->payout.w_zip, w_zip, 10);
strncpy (str->payout.d_street_1, d_street_1, 21);
strncpy (str->payout.d_street_2, d_street_2, 21);
strncpy (str->payout.d_city, d_city, 21);
strncpy (str->payout.d_state, d_state, 3);
strncpy (str->payout.d_zip, d_zip, 10);
str->payout.c_id = c_id;
strncpy (str->payout.c_first, c_first, 17);
strncpy (str->payout.c_middle, c_middle, 3);
strncpy (str->payout.c_last, c_last, 17);
strncpy (str->payout.c_street_1, c_street_1, 21);
strncpy (str->payout.c_street_2, c_street_2, 21);
strncpy (str->payout.c_city, c_city, 21);
strncpy (str->payout.c_state, c_state, 3);
strncpy (str->payout.c_zip, c_zip, 10);
strncpy (str->payout.c_phone, c_phone, 17);
strncpy (str->payout.c_since, (char*)c_since_d, 11);
strncpy (str->payout.c_credit, c_credit, 3);
str->payout.c_credit_lim = (float)(c_credit_lim)/100;
str->payout.c_discount = c_discount;
str->payout.c_balance = (float)(c_balance)/100;
strncpy (str->payout.c_data, c_data, 201);
strncpy (str->payout.h_date, (Char*)h_date, 20);
str->payout.retry = retries;
#ifdef TUX
    return (1);
#else
    return (0);
#endif
}

TPCord (str)

struct ordstruct *str;

{
    int i;
    w_id = str->ordin.w_id;
    d_id = str->ordin.d_id;
    bylastname = str->ordin.bylastname;
    if (bylastname) {
        c_id = 0;
        strncpy (c_last, str->ordin.c_last, 17);
    }
    else {

```

```

        c_id = str->ordin.c_id;
        strncpy (c_last, " ");
    }
    retries = 0;

    if (str->ordout.terror = tkvco ()) {
        if (c_id == 0) return (0); //by katsuf, for no such
cust
        else if (str->ordout.terror != RECOVERR)
            str->ordout.terror = IRRECERR;
        return (-1);
    }

    datelen = sizeof(o_entry_d);
    OCIERROR(errhp,
        OCIDateToText(errhp,&o_entry_d_base,(text*)FULLDATE,SIZ(FULLDAT
E),(text*)0,0,
        &datelen,o_entry_d));

    str->ordout.terror = NOERR;
    str->ordout.c_id = c_id;
    strncpy (str->ordout.c_last, c_last, 17);
    strncpy (str->ordout.c_first, c_first, 17);
    strncpy (str->ordout.c_middle, c_middle, 3);
    str->ordout.c_balance = c_balance/100;
    str->ordout.o_id = o_id;
    strncpy (str->ordout.o_entry_d, (char*)o_entry_d, 20);
    if ( o_carrier_id == 11 )
        str->ordout.o_carrier_id = 0;
    else
        str->ordout.o_carrier_id = o_carrier_id;
    str->ordout.o_ol_cnt = o_ol_cnt;
    for (i = 0; i < o_ol_cnt; i++) {
        ol_delivery_d[i][10] = '\0';
        if ( !strcmp((char*)ol_delivery_d[i],"15-09-1911") )
            strncpy((char*)ol_delivery_d[i],"NOT DELIVR",10);
        str->ordout.ol_supply_w_id[i] = ol_supply_w_id[i];
        str->ordout.ol_i_id[i] = ol_i_id[i];
        str->ordout.ol_quantity[i] = ol_quantity[i];
        str->ordout.ol_amount[i] = (float)(ol_amount[i])/100;
        strncpy (str->ordout.ol_delivery_d[i], (char*)ol_delivery_d[i],
11);
    }
    str->ordout.retry = retries;
#ifdef TUX
    return (1);
#else
    return (0);
#endif
}

TPCdel (str)

struct delstruct *str;

{
    double tr_end;
    int i;

    w_id = str->delin.w_id;
    o_carrier_id = str->delin.o_carrier_id;

```

```

retries = 0;
/*
vgetdate(cr_date); */
OCIERROR(errhp,OCIDateSysDate(errhp,&cr_date));

if (str->delout.terror = tkvcd ()) {
    if(str->delout.terror == DEL_ERROR)
        return DEL_ERROR;
    if (str->delout.terror != RECOVERERR)
        str->delout.terror = IRRECERR;
    return (-1);
}

tr_end = gettime ();
// fprintf (lfp, "%d %d %f %f %d %d", str->delin.in_timing_int,
//          (tr_end - str->delin.qtime) <= DELRT ? 1 : 0,
//          str->delin.qtime, tr_end, w_id, o_carrier_id);
fprintf (lfp, "%f %f %d %d", str->delin.qtime, tr_end, w_id,
o_carrier_id);
for (i= 0; i < 10; i++) {
//    fprintf (lfp, " %d %d", i + 1, del_o_id[i]);    //katsuf
    fprintf (lfp, " %d", del_o_id[i]);
    if (del_o_id[i] <= 0) {
#ifdef TUX
        userlog ("DELIVERY: no new order for w_id: %d, d_id %d\n",
                w_id, i + 1);
#else
        fprintf (stderr, "DELIVERY: no new order for w_id: %d,
d_id %d\n",
                w_id, i + 1);
#endif
    }
    fprintf (lfp, " %d\n", retries);
    str->delout.terror = NOERR;
    str->delout.retry = retries;
//    fflush(lfp);
    return (0);
}

TPCsto (str)

struct stostruct *str;

{

w_id = str->stoin.w_id;
d_id = str->stoin.d_id;
threshold = str->stoin.threshold;
retries = 0;

if (str->stoout.terror = tkvcs ()) {
    if (str->stoout.terror != RECOVERERR)
        str->stoout.terror = IRRECERR;
    return (-1);
}

str->stoout.terror = NOERR;
str->stoout.low_stock = low_stock;
str->stoout.retry = retries;

```

```

return (0);
}

```

TPCCSVR.C

```

#ifdef RCSID
static char *RCSid =
    "$Header: tpccsvr.c 7030100.1 95/07/19 15:39:28 plai Generic<base>
$ Copyr (c) 1995 Oracle";
#endif /* RCSID */

/*=====+
|          Copyright (c) 1995 Oracle Corp, Redwood Shores, CA          |
|          OPEN SYSTEMS PERFORMANCE GROUP                              |
|          All Rights Reserved                                          |
+=====+
| FILENAME                                                              |
|   tpccsvr.c                                                            |
| DESCRIPTION                                                            |
|   Tuxedo server for TPC-C. use a #define TUX                          |
|   TOPEND server for TPC-C. use a #define TOP                          |
+=====*/

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

#ifdef TUX
#include <atmi.h>                // must occur prior to include of tpccapi.h
#include <stdlib.h>              // for generation of random seed for server
id
#include <time.h>                // for generation of random seed for server
id
#endif

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

#ifdef TUX

#include <tmenv.h>
#include <xa.h>
#include <userlog.h>

/* set up pointers for type casting */
struct newstruct *newinfo;
struct paystruct *payinfo;
struct ordstruct *ordinfo;
struct delstruct *delinfo;
struct stostruct *stoinfo;

extern void TMlog();

#endif

```

```

// Lifted from HP FDR since they did such a nice job
void Tmlog( char *format, ... )
{
    va_list args;
    char buf[4096];
    int len;
    va_start( args, format );
    _strtime( buf );
    strcat( buf, " " );
    len = strlen( buf );
    (void) _vsprintf( buf+ len, sizeof( buf) - len - 1, format,
args);
    buf[sizeof( buf )- 1]= '\0';
    va_end( args );
    userlog( buf );
}

/* FUNCTION: int tpsvrinit (int argc, char *argv[]);
 *
 * PURPOSE:      Connects into database
 * ARGUMENTS:    parameters passed in as int svrid, char *uid, char
 *               *pwd, int txntype
 *               do not check ordering, assume correct
 *               svrid:   an id number for server running
 *               uid:     the userid for the database
 *               pwd:     the password for the userid
 * RETURNS:      txntype: transaction type the server will be running
 *               None
 * COMMENTS:     None
 */

int tpsvrinit (int argc, char *argv[])
{
    int svrid, txntype;
    char *uid, *pwd;
    int svrcnt;

    /* pull out the values from argv */
    svrid = atoi(argv[0]);
    uid = argv[1];
    pwd = argv[2];
    txntype = atoi(argv[3]);

#ifdef TUX

    srand ( (unsigned)time( NULL ) );
    // svrcnt = rand();
    svrcnt = _getpid();

    /* send 6 for all txns to be initd */
    /* fix uid and pwd for now, pull out later */
    /* not passing parameters through TUX yet */
    if (TPCinit( svrcnt, "tpcc", "tpcc", 6)) {
        Tmlog( " FAILED to init all txns types");
        return (-1);
    }
#endif
}

```

```

    }
    return 0;
}
#else // for topend
    if (TPCinit( svrid, uid, pwd, txntype)) {
        fprintf(stderr, "Failed in TPCinit (probably connecting).");
        exit (1);
    }
    return (1);
#endif

}

void tpsvrdone ()
{
    TPCexit (0);
}

/* FUNCTION: int NEWORDER(CLIENTDATA *jobData, NewOrderData *neword,
int deadlock)
 *
 * PURPOSE:      This function handles the new order transaction.
 * ARGUMENTS:    deadlock : count of deadlocks encountered during txn
 *               jobData:  pointer to entire block of user data
 *               neword:   pointer to datastructure in jobData that
 *               contains the new order data
 * RETURNS:      int      TRUE      transaction committed
 *               FALSE     item number not valid
 *               -1        deadlock max
 *
 * retry reached
 *
 * COMMENTS:     None
 */

#ifdef TOP
int NEWORDER(CLIENTDATA *jobData, NewOrderData *neword, int deadlock)
#else
void NEWORDER (TPSVCINFO *msg)
#endif

{
#ifdef TOP
    int result;

    result = TPCnew(neword);

    return result;
#else // for Tuxedo

    newinfo = (struct newstruct *) msg->data;
    newinfo->retval = TPCnew (newinfo); // set return value to 0 or -1

```

```

    // always return tpreturn success - let client side poll retval for
actual error
    tpreturn (TPSUCCESS, 0, (char *) newinfo, sizeof (struct newstruct),
0);
#endif
}

/* FUNCTION: int PAYMENT(CLIENTDATA *jobData, PaymentData *paydata, int
deadlock)
*
* PURPOSE:      This function handles the new order transaction.
*
* ARGUMENTS:   deadlock :   count of deadlocks encountered during txn
*               jobData:   pointer to entire block of user data
*               paydata:   pointer to datastructure in jobData that
contains the new order data
* RETURNS:     int         TRUE   transaction committed
*               FALSE     item number not valid
*               -1         deadlock max
retry reached
*
* COMMENTS:    None
*/

#ifdef TOP
int PAYMENT(CLIENTDATA *jobData, PaymentData *paydata, int deadlock)
#else
void PAYMENT (TPSVCINFO *msg)
#endif

{
#ifdef TOP
    int result;

    result = TPCpay(paydata);

    return result;
#else
    payinfo = (struct paystruct *) msg->data;
    payinfo->retval = TPCpay (payinfo); // set return value to 1 or 0
or -1

    // always return tpreturn success - let client side poll retval for
actual error
    tpreturn (TPSUCCESS, 0, (char *) payinfo, sizeof (struct paystruct),
0);
#endif
}

/* FUNCTION: int ORDERSTATUS(CLIENTDATA *jobData, OrderStatusData
*orddata, int deadlock)
*
* PURPOSE:      This function handles the new order transaction.
*

```

```

* ARGUMENTS:   deadlock :   count of deadlocks encountered during txn
*               jobData:   pointer to entire block of user data
*               orddata:   pointer to datastructure in jobData that
contains the new order data
* RETURNS:     int         TRUE   transaction committed
*               FALSE     item number not valid
*               -1         deadlock max
retry reached
*
* COMMENTS:    None
*/

#ifdef TOP
int ORDERSTATUS(CLIENTDATA *jobData, OrderStatusData *orddata, int
deadlock)
#else
void ORDERSTATUS (TPSVCINFO *msg)
#endif

{
#ifdef TOP
    int result;

    result = TPCord(orddata);

    return result;
#else
    ordinfo = (struct ordstruct *) msg->data;
    ordinfo->retval = TPCord (ordinfo); // set return value to 0 or -1

    // always return tpreturn success - let client side poll retval for
actual error
    tpreturn (TPSUCCESS, 0, (char *) ordinfo, sizeof (struct ordstruct),
0);
#endif
}

/* FUNCTION: int DELIVERY(CLIENTDATA *jobData, DeliveryData *deldata,
int deadlock)
*
* PURPOSE:      This function handles the new order transaction.
*
* ARGUMENTS:   deadlock :   count of deadlocks encountered during txn
*               jobData:   pointer to entire block of user data
*               stodata:   pointer to datastructure in jobData that
contains the new order data
* RETURNS:     int         TRUE   transaction committed
*               FALSE     item number not valid
*               -1         deadlock max
retry reached
*
* COMMENTS:    None
*/

```

```

*/
#ifdef TOP
int DELIVERY(CLIENTDATA *jobData, DeliveryData *deldata, int deadlock)
#else
void DELIVERY (TPSVCINFO *msg)
#endif
{
#ifdef TOP
    int result;

    result = TPCdel(deldata);

    return result;
#else
    delinfo = (struct delstruct *) msg->data;
    delinfo->retval = TPCdel (delinfo); // set return value to 0 or -1

    // always return tpreturn success - let client side poll retval for
    actual error
    tpreturn (TPSUCCESS, 0, (char *) delinfo, sizeof (struct delstruct),
0);
#endif
}

/* FUNCTION: int STOCKLEVEL(CLIENTDATA *jobData, StockLevelData
*stodata, int deadlock)
*
* PURPOSE:      This function handles the new order transaction.
*
* ARGUMENTS:   deadlock :   count of deadlocks encountered during txn
*                jobData:   pointer to entire block of user data
*                stodata:   pointer to datastructure in jobData that
contains the new order data
* RETURNS:     int          TRUE      transaction committed
*                FALSE     item number not valid
*                -1         deadlock max
retry reached
*
* COMMENTS:    None
*/

#ifdef TOP
int STOCKLEVEL(CLIENTDATA *jobData, StockLevelData *stodata, int
deadlock)
#else
void STOCKLEVEL (TPSVCINFO *msg)
#endif
{
#ifdef TOP
    int result;

```

```

        result = TPCsto(stodata);

        return result;
#else
    stoinfo = (struct stostruct *) msg->data;
    stoinfo->retval = TPCsto (stoinfo); // set return value to 0 or -1

    // always return tpreturn success - let client side poll retval for
    actual error
    tpreturn (TPSUCCESS, 0, (char *) stoinfo, sizeof (struct stostruct),
0);
#endif
}

/* FUNCTION: int OPSTUXSERVER(CLIENTDATA *jobData, NewOrderData *neword,
int deadlock)
*
* PURPOSE:      This function handles all transactions.
*
* ARGUMENTS:   deadlock :   count of deadlocks encountered during txn
*                jobData:   pointer to entire block of user data
*                neword:    pointer to datastructure in jobData that
contains the new order data
* RETURNS:     int          TRUE      transaction committed
*                FALSE     item number not valid
*                -1         deadlock max
retry reached
*
* COMMENTS:    None
*/

#ifdef TOP
int OPSTUXSERVER(CLIENTDATA *jobData, NewOrderData *neword, int
deadlock)
#else
void OPSTUXSERVER (TPSVCINFO *msg)
#endif
{
#ifdef TOP
    int result;

    result = TPCnew(neword);

    return result;
#else
    // for Tuxedo

    if (msg->len == 928) { // len for neworder
        newinfo = (struct newstruct *) msg->data;
        newinfo->retval = TPCnew (newinfo); // set return value to 0
or -1 or 1

        // always return tpreturn success - let client side poll
retval for actual error

```

```

        tpreturn (TPSUCCESS, 0, (char *) newinfo, sizeof (struct
newstruct), 0);
    }
    else
        if (msg->len == 616) { // len for payment
            payinfo = (struct paystruct *) msg->data;
            payinfo->retval = TPCpay (payinfo); // set return value to
1 or 0 or -1

            // always return tpreturn success - let client side poll
retval for actual error
            tpreturn (TPSUCCESS, 0, (char *) payinfo, sizeof (struct
paystruct), 0);
        }
        else
            if (msg->len == 544) { // len for order status
                ordinfo = (struct ordstruct *) msg->data;
                ordinfo->retval = TPCord (ordinfo); // set return value
to 0 or -1 or 1

                // always return tpreturn success - let client side poll
retval for actual error
                tpreturn (TPSUCCESS, 0, (char *) ordinfo, sizeof (struct
ordstruct), 0);
            }
        }
        else
            if (msg->len == 40) { // len for delivery
                delinfo = (struct delstruct *) msg->data;
                delinfo->retval = TPCdel (delinfo); // set
return value to 0 or -1

                // always return tpreturn success - let client
side poll retval for actual error
                tpreturn (TPSUCCESS, 0, (char *) delinfo,
sizeof (struct delstruct), 0);
            }
        }
        else { // assume rest is stock level

            stoinfo = (struct stostruct *) msg->data;
            stoinfo->retval = TPCsto (stoinfo); // set
return value to 0 or -1

            // always return tpreturn success - let client
side poll retval for actual error
            tpreturn (TPSUCCESS, 0, (char *) stoinfo,
sizeof (struct stostruct), 0);
        }
    }

#endif
}

```

paynz.sql

```

DECLARE /* paynz */
--      cust_rowid          ROWID;
--      dist_name           VARCHAR2(11);
--      ware_name           VARCHAR2(11);
not_serializable          EXCEPTION;
PRAGMA EXCEPTION_INIT(not_serializable,-8177);
deadlock                   EXCEPTION;
PRAGMA EXCEPTION_INIT(deadlock,-60);

```

```

snapshot too old          EXCEPTION;
PRAGMA EXCEPTION_INIT(snapshot_too_old,-1555);
BEGIN
LOOP BEGIN
    UPDATE ware
        SET w_ytd = w_ytd + :h_amount
        WHERE w_id = :w_id
    RETURNING w_name, w_street_1, w_street_2, w_city, w_state,
w_zip
        INTO initpay.ware_name, :w_street_1, :w_street_2, :w_city,
:w_state, :w_zip;

    UPDATE cust
        SET c_balance = c_balance - :h_amount,
            c_ytd_payment = c_ytd_payment + :h_amount,
            c_payment_cnt = c_payment_cnt+1
        WHERE c_id = :c_id AND c_d_id = :c_d_id AND
            c_w_id = :c_w_id
    RETURNING rowid, c_first, c_middle, c_last, 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
        INTO
initpay.cust_rowid, :c_first, :c_middle, :c_last, :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;

    IF SQL%NOTFOUND THEN
        raise NO_DATA_FOUND;
    END IF;
--      insert into dummy values
(rowidtochar(initpay.cust_rowid));
--
:c_data := ' ';

    IF :c_credit = 'BC' THEN
        UPDATE cust
            SET c_data= substr ((to_char (:c_id) || ' ' ||
to_char (:c_d_id) || ' ' ||
to_char (:c_w_id) || ' ' ||
to_char (:d_id) || ' ' ||
to_char (:w_id) || ' ' ||
to_char (:h_amount/100, '9999.99')
|| ' | ')
                || c_data, 1, 500)
        WHERE rowid = initpay.cust_rowid
        WHERE C_W_ID=:C_W_ID AND C_D_ID=:C_D_ID AND C_ID=:C_ID
    RETURNING substr(c_data,1, 200)
        INTO :c_data;

    END IF;

    UPDATE dist
        SET d_ytd = d_ytd + :h_amount
        WHERE d_id = :d_id
        AND d_w_id = :w_id
    RETURNING d_name, d_street_1, d_street_2, d_city,d_state,
d_zip
        INTO
initpay.dist_name,:d_street_1,:d_street_2,:d_city,:d_state,
:d_zip;

    IF SQL%NOTFOUND THEN
        raise NO_DATA_FOUND;
    END IF;

```

```

INSERT INTO hist (h_c_id, h_c_d_id, h_c_w_id, h_d_id, h_w_id,
                 h_amount, h_date, h_data)
VALUES
(:c_id, :c_d_id, :c_w_id, :d_id, :w_id, :h_amount,
 :cr_date, initpay.ware_name || ' ' ||
initpay.dist_name);
-- COMMIT;
-- :h_date := to_char (:cr_date, 'DD-MM-YYYY.HH24:MI:SS');
EXIT;

EXCEPTION
WHEN not_serializable OR deadlock OR snapshot_too_old THEN
ROLLBACK;
:retry := :retry + 1;
END;

END LOOP;
END;

```

payz.sql

```

DECLARE /* payz */
not_serializable          EXCEPTION;
PRAGMA EXCEPTION_INIT(not_serializable,-8177);
deadlock                  EXCEPTION;
PRAGMA EXCEPTION_INIT(deadlock,-60);
snapshot_too_old         EXCEPTION;
PRAGMA EXCEPTION_INIT(snapshot_too_old,-1555);
BEGIN
LOOP BEGIN
UPDATE ware
SET w_ytd = w_ytd+h_amount
WHERE w_id = :w_id
RETURNING w_name,
          w_street_1, w_street_2, w_city, w_state, w_zip
INTO initpay.ware_name,
      :w_street_1, :w_street_2, :w_city, :w_state, :w_zip;

SELECT rowid
BULK COLLECT INTO initpay.row_id
FROM cust
WHERE c_d_id = :c_d_id AND c_w_id = :c_w_id AND c_last
= :c_last
ORDER BY c_last, c_d_id, c_w_id, c_first;

initpay.c_num := sql%rowcount;
initpay.cust_rowid := initpay.row_id((initpay.c_num+1) / 2);

UPDATE cust
SET c_balance = c_balance - :h_amount,
    c_ytd_payment = c_ytd_payment+ :h_amount,
    c_payment_cnt = c_payment_cnt+1
WHERE rowid = initpay.cust_rowid
RETURNING
c_id, c_first, c_middle, c_last, 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
INTO :c_id, :c_first, :c_middle, :c_last,

```

```

: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 := ' ';
IF :c_credit = 'BC' THEN
UPDATE cust
SET c_data = substr ((to_char (:c_id) || ' ' ||
to_char (:c_d_id) || ' ' ||
to_char (:c_w_id) || ' ' ||
to_char (:d_id) || ' ' ||
to_char (:w_id) || ' ' ||
to_char (:h_amount/100, '9999.99')
|| ' | ')
|| c_data, 1, 500)
--- WHERE rowid = initpay.cust_rowid
WHERE C_W_ID=:C_W_ID AND C_D_ID=:C_D_ID AND C_ID=:C_ID
RETURNING substr(c_data,1, 200)
INTO :c_data;

END IF;

UPDATE dist
SET d_ytd = d_ytd+h_amount
WHERE d_id = :d_id
AND d_w_id = :w_id
RETURNING d_name, d_street_1, d_street_2, d_city,
d_state, d_zip
INTO initpay.dist_name, :d_street_1, :d_street_2, :d_city,
:d_state, :d_zip;

IF SQL%NOTFOUND
THEN
raise NO_DATA_FOUND;
END IF;

INSERT INTO hist (h_c_id, h_c_d_id, h_c_w_id, h_d_id, h_w_id,
                 h_amount, h_date, h_data)
VALUES (:c_id, :c_d_id, :c_w_id, :d_id, :w_id, :h_amount,
 :cr_date, initpay.ware_name || ' ' ||
initpay.dist_name);

EXIT;

EXCEPTION
WHEN not_serializable OR deadlock OR snapshot_too_old THEN
ROLLBACK;
:retry := :retry + 1;
END;

END LOOP;
END;

```

tkvcpnew.sql

```

-- New Order Anonymous block

DECLARE
idx          BINARY_INTEGER;
dummy_local BINARY_INTEGER;
not_serializable EXCEPTION;
PRAGMA EXCEPTION_INIT(not_serializable,-8177);

```

```

deadlock          EXCEPTION;
PRAGMA EXCEPTION_INIT(deadlock,-60);
snapshot too old  EXCEPTION;
PRAGMA EXCEPTION_INIT(snapshot_too_old,-1555);
PROCEDURE u1 IS
BEGIN
  FORALL idx IN 1 .. :o_ol_cnt
    UPDATE stock_item
    SET s_order_cnt = s_order_cnt + 1,
    s_ytd = s_ytd + :ol_quantity(idx),
    s_remote_cnt = s_remote_cnt + :s_remote(idx),
    s_quantity = s_quantity - :ol_quantity(idx) +
      DECODE(sign(s_quantity - :ol_quantity(idx) -
10),-1,91,0)
    WHERE i_id = :ol_i_id(idx)
    AND s_w_id = :ol_supply_w_id(idx)
    RETURNING i_price, i_name, s_quantity, s_dist_01,
    DECODE (instr(i_data,'ORIGINAL'), 0, 'G',
    DECODE(instr(s_data,'ORIGINAL'), 0, 'G', 'B'))
    BULK COLLECT INTO :i_price, :i_name, :s_quantity,
initnew.s_dist,
:brand_generic;

END u1;

PROCEDURE u2 IS
BEGIN
  FORALL idx IN 1 .. :o_ol_cnt
    UPDATE stock_item
    SET s_order_cnt = s_order_cnt + 1,
    s_ytd = s_ytd + :ol_quantity(idx),
    s_remote_cnt = s_remote_cnt + :s_remote(idx),
    s_quantity = s_quantity - :ol_quantity(idx) +
      DECODE(sign(s_quantity - :ol_quantity(idx) -
10),-1,91,0)
    WHERE i_id = :ol_i_id(idx)
    AND s_w_id = :ol_supply_w_id(idx)
    RETURNING i_price, i_name, s_quantity, s_dist_02,
    DECODE (instr(i_data,'ORIGINAL'), 0, 'G',
    DECODE(instr(s_data,'ORIGINAL'), 0, 'G', 'B'))
    BULK COLLECT INTO :i_price, :i_name, :s_quantity,
initnew.s_dist,
:brand_generic;

END u2;

PROCEDURE u3 IS
BEGIN
  FORALL idx IN 1 .. :o_ol_cnt
    UPDATE stock_item
    SET s_order_cnt = s_order_cnt + 1,
    s_ytd = s_ytd + :ol_quantity(idx),
    s_remote_cnt = s_remote_cnt + :s_remote(idx),
    s_quantity = s_quantity - :ol_quantity(idx) +
      DECODE(sign(s_quantity - :ol_quantity(idx) -
10),-1,91,0)
    WHERE i_id = :ol_i_id(idx)
    AND s_w_id = :ol_supply_w_id(idx)
    RETURNING i_price, i_name, s_quantity, s_dist_03,
    DECODE (instr(i_data,'ORIGINAL'), 0, 'G',
    DECODE(instr(s_data,'ORIGINAL'), 0, 'G', 'B'))
    BULK COLLECT INTO :i_price, :i_name, :s_quantity,
initnew.s_dist,
:brand_generic;

```

```

END u3;

PROCEDURE u4 IS
BEGIN
  FORALL idx IN 1 .. :o_ol_cnt
    UPDATE stock_item
    SET s_order_cnt = s_order_cnt + 1,
    s_ytd = s_ytd + :ol_quantity(idx),
    s_remote_cnt = s_remote_cnt + :s_remote(idx),
    s_quantity = s_quantity - :ol_quantity(idx) +
      DECODE(sign(s_quantity - :ol_quantity(idx) -
10),-1,91,0)
    WHERE i_id = :ol_i_id(idx)
    AND s_w_id = :ol_supply_w_id(idx)
    RETURNING i_price, i_name, s_quantity, s_dist_04,
    DECODE (instr(i_data,'ORIGINAL'), 0, 'G',
    DECODE(instr(s_data,'ORIGINAL'), 0, 'G', 'B'))
    BULK COLLECT INTO :i_price, :i_name, :s_quantity,
initnew.s_dist,
:brand_generic;

END u4;

PROCEDURE u5 IS
BEGIN
  FORALL idx IN 1 .. :o_ol_cnt
    UPDATE stock_item
    SET s_order_cnt = s_order_cnt + 1,
    s_ytd = s_ytd + :ol_quantity(idx),
    s_remote_cnt = s_remote_cnt + :s_remote(idx),
    s_quantity = s_quantity - :ol_quantity(idx) +
      DECODE(sign(s_quantity - :ol_quantity(idx) -
10),-1,91,0)
    WHERE i_id = :ol_i_id(idx)
    AND s_w_id = :ol_supply_w_id(idx)
    RETURNING i_price, i_name, s_quantity, s_dist_05,
    DECODE (instr(i_data,'ORIGINAL'), 0, 'G',
    DECODE(instr(s_data,'ORIGINAL'), 0, 'G', 'B'))
    BULK COLLECT INTO :i_price, :i_name, :s_quantity,
initnew.s_dist,
:brand_generic;

END u5;

PROCEDURE u6 IS
BEGIN
  FORALL idx IN 1 .. :o_ol_cnt
    UPDATE stock_item
    SET s_order_cnt = s_order_cnt + 1,
    s_ytd = s_ytd + :ol_quantity(idx),
    s_remote_cnt = s_remote_cnt + :s_remote(idx),
    s_quantity = s_quantity - :ol_quantity(idx) +
      DECODE(sign(s_quantity - :ol_quantity(idx) -
10),-1,91,0)
    WHERE i_id = :ol_i_id(idx)
    AND s_w_id = :ol_supply_w_id(idx)
    RETURNING i_price, i_name, s_quantity, s_dist_06,
    DECODE (instr(i_data,'ORIGINAL'), 0, 'G',
    DECODE(instr(s_data,'ORIGINAL'), 0, 'G', 'B'))
    BULK COLLECT INTO :i_price, :i_name, :s_quantity,
initnew.s_dist,
:brand_generic;

END u6;

PROCEDURE u7 IS
BEGIN

```

```

FORALL idx IN 1 .. :o_ol_cnt
  UPDATE stock_item
  SET s_order_cnt = s_order_cnt + 1,
  s_ytd = s_ytd + :ol_quantity(idx),
  s_remote_cnt = s_remote_cnt + :s_remote(idx),
  s_quantity = s_quantity - :ol_quantity(idx) +
  DECODE(sign(s_quantity - :ol_quantity(idx) -
10),-1,91,0)
  WHERE i_id = :ol_i_id(idx)
  AND s_w_id = :ol_supply_w_id(idx)
  RETURNING i_price, i_name, s_quantity, s_dist_07,
  DECODE (instr(i_data,'ORIGINAL'), 0, 'G',
  DECODE(instr(s_data,'ORIGINAL'), 0, 'G', 'B'))
  BULK COLLECT INTO :i_price, :i_name, :s_quantity,
initnew.s_dist,
:brand_generic;
END u7;

PROCEDURE u8 IS
BEGIN
FORALL idx IN 1 .. :o_ol_cnt
  UPDATE stock_item
  SET s_order_cnt = s_order_cnt + 1,
  s_ytd = s_ytd + :ol_quantity(idx),
  s_remote_cnt = s_remote_cnt + :s_remote(idx),
  s_quantity = s_quantity - :ol_quantity(idx) +
  DECODE(sign(s_quantity - :ol_quantity(idx) -
10),-1,91,0)
  WHERE i_id = :ol_i_id(idx)
  AND s_w_id = :ol_supply_w_id(idx)
  RETURNING i_price, i_name, s_quantity, s_dist_08,
  DECODE (instr(i_data,'ORIGINAL'), 0, 'G',
  DECODE(instr(s_data,'ORIGINAL'), 0, 'G', 'B'))
  BULK COLLECT INTO :i_price, :i_name, :s_quantity,
initnew.s_dist,
:brand_generic;
END u8;

PROCEDURE u9 IS
BEGIN
FORALL idx IN 1 .. :o_ol_cnt
  UPDATE stock_item
  SET s_order_cnt = s_order_cnt + 1,
  s_ytd = s_ytd + :ol_quantity(idx),
  s_remote_cnt = s_remote_cnt + :s_remote(idx),
  s_quantity = s_quantity - :ol_quantity(idx) +
  DECODE(sign(s_quantity - :ol_quantity(idx) -
10),-1,91,0)
  WHERE i_id = :ol_i_id(idx)
  AND s_w_id = :ol_supply_w_id(idx)
  RETURNING i_price, i_name, s_quantity, s_dist_09,
  DECODE (instr(i_data,'ORIGINAL'), 0, 'G',
  DECODE(instr(s_data,'ORIGINAL'), 0, 'G', 'B'))
  BULK COLLECT INTO :i_price, :i_name, :s_quantity,
initnew.s_dist,
:brand_generic;
END u9;

PROCEDURE u10 IS
BEGIN
FORALL idx IN 1 .. :o_ol_cnt
  UPDATE stock_item
  SET s_order_cnt = s_order_cnt + 1,
  s_ytd = s_ytd + :ol_quantity(idx),

```

```

s_remote_cnt = s_remote_cnt + :s_remote(idx),
s_quantity = s_quantity - :ol_quantity(idx) +
  DECODE(sign(s_quantity - :ol_quantity(idx) -
10),-1,91,0)
  WHERE i_id = :ol_i_id(idx)
  AND s_w_id = :ol_supply_w_id(idx)
  RETURNING i_price, i_name, s_quantity, s_dist_10,
  DECODE (instr(i_data,'ORIGINAL'), 0, 'G',
  DECODE(instr(s_data,'ORIGINAL'), 0, 'G', 'B'))
  BULK COLLECT INTO :i_price, :i_name, :s_quantity,
initnew.s_dist,
:brand_generic;
END u10;

PROCEDURE fix_items IS
  rows_lost BINARY_INTEGER;
  max_index BINARY_INTEGER;
  temp_index BINARY_INTEGER;
BEGIN
-- gotta shift price, name, s_quantity, brand_generic, s_dist,
ol_amount
  idx := 1;
-- found 0 bad rows
  rows_lost := 0;
-- so many rows in out array to begin with
  max_index := sql%rowcount;

  WHILE (max_index != :o_ol_cnt) LOOP
-- find item where item ids dont match
  WHILE (idx <= sql%rowcount AND
  sql%bulk_rowcount(idx + rows_lost) = 1)
  LOOP
    idx := idx + 1;
  END LOOP;
-- shift the items please
  temp_index := max_index;
  WHILE (temp_index >= idx + rows_lost) LOOP
    :i_price(temp_index + 1) := :i_price(temp_index);
    :i_name(temp_index + 1) := :i_name(temp_index);
    :s_quantity(temp_index +
1) := :s_quantity(temp_index);
    initnew.s_dist(temp_index + 1) :=
initnew.s_dist(temp_index);
    :brand_generic(temp_index +
1) := :brand_generic(temp_index);
    temp_index := temp_index - 1;
  END LOOP;
-- values for the non-existent items if not at end
  IF (idx + rows_lost <= :o_ol_cnt) THEN
    :i_price(idx + rows_lost) := 0;
    :i_name(idx + rows_lost) := NULL;
    :s_quantity(idx + rows_lost) := 0;
    initnew.s_dist(idx + rows_lost) := NULL;
    :brand_generic(idx + rows_lost) := NULL;
-- one more bad row
    rows_lost := rows_lost + 1;
    max_index := max_index + 1;
  END IF;
END LOOP;

```

```

END fix_items;

BEGIN
  LOOP BEGIN
    UPDATE dist SET d_next_o_id = d_next_o_id + 1
      WHERE d_id = :d_id AND d_w_id = :w_id
      RETURNING d_tax, d_next_o_id-1
      INTO :d_tax, :o_id;

    SELECT c_discount, c_last, c_credit, w_tax
      INTO :c_discount, :c_last, :c_credit, :w_tax
      FROM cust, ware
      WHERE c_id = :c_id AND c_d_id = :d_id AND c_w_id = :w_id
      AND w_id = :w_id;

    INSERT INTO nord (no_o_id, no_d_id, no_w_id)
      VALUES (:o_id, :d_id, :w_id);
    INSERT INTO ord (o_id, o_d_id, o_w_id, o_c_id, o_entry_d,
      o_carrier_id, o_ol_cnt, o_all_local)
      VALUES (:o_id, :d_id, :w_id, :c_id,
      :cr_date, 11, :o_ol_cnt, :o_all_local);

    -- copying :d_id in local variable is important - lots of
instr. dummy_local := :d_id;

    IF (dummy_local = 1) THEN u1; END IF;
    IF (dummy_local = 2) THEN u2; END IF;
    IF (dummy_local = 3) THEN u3; END IF;
    IF (dummy_local = 4) THEN u4; END IF;
    IF (dummy_local = 5) THEN u5; END IF;
    IF (dummy_local = 6) THEN u6; END IF;
    IF (dummy_local = 7) THEN u7; END IF;
    IF (dummy_local = 8) THEN u8; END IF;
    IF (dummy_local = 9) THEN u9; END IF;
    IF (dummy_local = 10) THEN u10; END IF;

    -- cache the no of rows processed
    dummy_local := sql%rowcount;

    -- fix the rows if necessary
    IF (dummy_local != :o_ol_cnt ) THEN fix_items; END IF;

    -- calculate ol_amount

    FOR idx IN 1 ..:o_ol_cnt LOOP
      :ol_amount(idx) := :ol_quantity(idx) * :i_price(idx);
    END LOOP;

    FORALL idx IN 1..:o_ol_cnt
      INSERT INTO ordl
        (ol_o_id, ol_d_id, ol_w_id, ol_number, ol_delivery_d,
ol_i_id,

```

```

        ol_supply_w_id, ol_quantity, ol_amount, ol_dist_info)
      VALUES (:o_id, :d_id, :w_id, initnew.idx1arr(idx),
initnew.nulldate,
        :ol_i_id(idx), :ol_supply_w_id(idx),
        :ol_quantity(idx), :ol_amount(idx),
initnew.s_dist(idx));

    IF (dummy_local != :o_ol_cnt) THEN
      :o_ol_cnt := dummy_local;
      ROLLBACK;
    END IF;

    EXIT;

    EXCEPTION
      WHEN not serializable OR deadlock OR snapshot_too_old THEN
        ROLLBACK;
        :retry := :retry + 1;
      END;
    END LOOP;
  END;

```

DPBCORE.H

```

/* Copyright (c) Oracle Corporation 1993, 1992. All Rights Reserved.
*/

/*
NAME      DPBCORE.H

DESCRIPTION
  Header for CORE function

NOTES
  Desktop Performance Group

MODIFIED      (MM/DD/YY)
  B Moriarty  06/02/95 - add dpbetime() for accurate elapsed time
measure
  B Moriarty  05/26/95 - add dpboradt() for new reporting
  B Moriarty  05/10/95 - add dpbcpu() for tpcc
  C Kelly     04/21/94 - add dpbinpgm() and dpbxtpgm() for Netware
NLMs
  C Kelly     02/24/93 - add dpbfsync()
  B Moriarty  11/12/93 - add dpbgetprty()
  R Keller    10/18/93 - add dpbprty()
  R Keller    03/06/92 - initial version

*/

#ifdef __dpbcore__
#define __dpbcore__

#include <stdio.h>
#include "dpbpcnt1.h"

```

```

#ifdef __STDC__
*/
int dpbfsync(FILE *); /* fsync for ACID
*/
int dpbgetprty(char *,char *,int); /* get O/S priority
*/
void dpbinpgm(void); /* pgm. init. function
*/
unsigned long dpbpchk(pcntl *); /* check on forked
process */
unsigned long dpbproc(char *[], pcntl *); /* spawn/fork new
process */
int dpbprty(char *); /* set O/S priority
*/
//clock_t dpbtimef(void); /* get time
*/
double dpbtimef(void); /* get CPU time
clock_t dpbcpu(void);
*/
void dpbwait(clock_t); /* wait routine in
millisec */
void dpbxtpgm(void); /* pgm exit routine
*/
int dpboradt(char *); /* sys date time in ora
form*/
clock_t dpbetime(void); /* elapsed time
*/
#else
*/
int dpbfsync(); /* fsync for ACID
*/
int dpbgetprty(); /* get O/S priority
*/
void dpbinpgm(); /* pgm. init. function
*/
unsigned long dpbpchk(); /* check on forked
process */
unsigned long dpbproc(); /* spawn/fork new
process */
int dpbprty(); /* set O/S priority
*/
//clock_t dpbtimef(); /* get time
*/
double dpbtimef(void); /* get cpu time
clock_t dpbcpu();
*/
void dpbwait(); /* wait routine in
millisec */
void dpbxtpgm(); /* pgm exit routine
*/
int dpboradt(); /* sys date time in ora
form*/
clock_t dpbetime(); /* elapsed time
*/
#endif /* __STDC__ */

#endif /* __dpbcore__ */

```

DPBTIMEF.C

```

/* Copyright (c) Oracle Corporation 1993, 1992. All Rights Reserved.
*/
*/
NAME DPBTIMEF.C

DESCRIPTION
    Get time in seconds as a clock_t.

NOTES
    Desktop Performance Group

MODIFIED      (MM/DD/YY)
B Moriarty    02/14/95 - V4.6 fix NT & OS/2
C Kelly       01/20/94 - V4.4 added Netware support
C Kelly       02/05/93 - V3.1 added A/UX support
R Keller      03/02/92 - V3.0

*/

#ifdef ORA_OS2
# define ORA_PC
#endif /* ORA_OS2 */

#ifdef ORA_NT
# define ORA_PC
#endif /* ORA_NT */

#ifdef ORA_PC
# include <sys/types.h>
# include <sys/timeb.h>
# include <stdio.h>
# include <time.h>

# ifdef __STDC__
double dpbtimef(void) /*katsuf
# else
double dpbtimef() /*katsuf
# endif /* __STDC__ */
{
    struct timeb buf;

    ftime(&buf);
    return(((double)(buf.time + buf.millitm/1000.0)));
}
#endif /* ORA_PC */

#ifdef ORA_AUX
# include <sys/time.h>
double dpbtimef()
{
    struct timeval t;
    int rc;

    do
    {
        rc = gettimeofday(&t, (struct timezone *)0);
    } while (rc != 0);

    return (((double)t.tv_sec) + (((double)t.tv_usec)/1000000));
}

```

```

}
#endif

#ifdef ORA_NW
# include <time.h>
double dpbtimef()
{
    return (double)time(NULL);    /* there is no function with greater
precision */
}
#endif /* ORA_NW */

#ifdef ORA_MAC
# include <types.h>
# include <OSUtils.h>

double dpbtimef()
{
    unsigned long secs;
    GetDateTime(&secs);
    return((double) secs);
}
#endif /* ORA_MAC */

```

DELR.C

```

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

double get90per(const long *hist, long num, double interval)
{
    long i, sum, total;

    for (i = 0, total = 0; i < num; i++)
        total += hist[i];

    for (i = 0, sum = 0; i < num; i++)
        if ((sum += hist[i]) >= total - total / 10)
            return (double)i * interval;

    return -1;
}

void main (int argc, char **argv)
{
    struct
    {
        long        count_succ;
        long        count_err;
        double      rt_total;
    }

```

```

        double      rt_min;
        double      rt_max;
        long        *hist;
    } stats;

    double bucket_i;
    long    bucket_n;
    long    measure_from;
    long    measure_to;

    char    ibuf[1024];
    FILE    *fp;
    int     line;
    int     i;

    memset(&stats, 0, sizeof(stats));
    stats.rt_min = 1000*1000*1000;

    if (argc != 6)
    {
        fprintf(stderr, "Usage: %s <bucket interval sec> <# of
buckets> <measure from sec> <measure to sec> <filename>\n", argv[0]);
        exit(1);
    }

    if ((bucket_i = atof(argv[1])) <= 0)
    {
        fprintf (stderr, "Error: <bucket interval> must be > 0.0\n");
        exit(1);
    }

    if ((bucket_n = atol(argv[2])) <= 0)
    {
        fprintf (stderr, "Error: <# of buckets> must be > 0\n");
        exit(1);
    }

    if ((measure_from = atol(argv[3])) <= 0)
    {
        fprintf (stderr, "Error: <measure from sec> must be > 0\n");
        exit(1);
    }

    if ((measure_to = atol(argv[4])) <= 0)
    {
        fprintf (stderr, "Error: <measure to sec> must be > 0\n");
        exit(1);
    }

    if (measure_from > measure_to)
    {
        fprintf (stderr, "Error: <measure from sec> must be less or
equal to <measure to sec>\n");
        exit(1);
    }

    if (!(fp = fopen(argv[5], "r")))
    {

```

```

        fprintf(stderr, "Error: cannot open %s", argv[5]);
        exit(1);
    }

    if (!(stats.hist = (long*)calloc(bucket_n, sizeof(long))))
    {
        fprintf(stderr, "Error: can't allocate %d bytes memory\n",
            bucket_n * sizeof(long));
        exit(1);
    }

    for (line = 1; fgets(ibuf, sizeof(ibuf), fp); line++)
    {
        double rt;
        double rt_start;
        double rt_end;
        long w_id;
        long o_carrier_id;
        long del_o_id[10];
        long pid;

        if (sscanf(ibuf,
"%lf %lf %d %d %d %d %d %d %d %d %d %d",
            &rt_start, &rt_end, &w_id,
&o_carrier_id,
            &del_o_id[0], &del_o_id[1],
&del_o_id[2], &del_o_id[3], &del_o_id[4],
            &del_o_id[5], &del_o_id[6],
&del_o_id[7], &del_o_id[8], &del_o_id[9],
            &pid) != 15)
        {
            fprintf(stderr, "Warning: Encountered invalid
format in line %d\n", line);
            continue;
        }

        if (rt_start <= (double)measure_from || rt_end >=
(double)measure_to)
            continue;

        for (i = 0; i < 10 && del_o_id[i]; i++);
        if (i < 10)
        {
            stats.count_err++;
            continue;
        }

        rt = rt_end - rt_start;
        stats.count_succ++;
        stats.hist[ $\min((\text{long})\text{ceil}(rt / \text{bucket\_i}), \text{bucket\_n} -$ 
1)]++;

        stats.rt_total += rt;
        stats.rt_min =  $\min(\text{stats.rt\_min}, \text{rt})$ ;
        stats.rt_max =  $\max(\text{stats.rt\_max}, \text{rt})$ ;
    }

    printf("Measure start time          : %s",
ctime(&measure_from));

```

```

        printf("Measure end time          : %s",
ctime(&measure_to));
        printf("Total delivery transaction : %d\n",
stats.count_succ + stats.count_err);
        printf("Total succeeded delivery transaction : %d\n",
stats.count_succ);
        printf("Total skipped delivery transaction : %d\n",
stats.count_err);
        printf("Average response time          : %f\n",
stats.rt_total / (double)stats.count_succ);
        printf("90 percentile                  : %f\n",
get90per(stats.hist, bucket_n, bucket_i));
        printf("Min response time              : %f\n",
stats.rt_min);
        printf("Max response time              : %f\n",
stats.rt_max);

        exit(0);
    }

```

Appendix B : Database Design

B.1 Database build scripts

create_partition.bat

```
creapart -d PhysicalDrive1
creapart -x PhysicalDrive1
creapart -d PhysicalDrive2
creapart -x PhysicalDrive2
creapart -d PhysicalDrive3
creapart -x PhysicalDrive3
creapart -d PhysicalDrive4
creapart -x PhysicalDrive4
creapart -d PhysicalDrive5
creapart -x PhysicalDrive5
creapart -d PhysicalDrive7
creapart -x PhysicalDrive7
creapart -d PhysicalDrive8
creapart -x PhysicalDrive8
creapart -d PhysicalDrive9
creapart -x PhysicalDrive9
creapart -d PhysicalDrive10
creapart -x PhysicalDrive10
creapart -d PhysicalDrive11
creapart -x PhysicalDrive11
creapart -d PhysicalDrive13
creapart -x PhysicalDrive13
creapart -d PhysicalDrive14
creapart -x PhysicalDrive14
creapart -d PhysicalDrive16
creapart -x PhysicalDrive16
creapart -d PhysicalDrive17
creapart -x PhysicalDrive17
creapart -d PhysicalDrive18
creapart -x PhysicalDrive18
creapart -d PhysicalDrive19
creapart -x PhysicalDrive19
creapart -d PhysicalDrive20
creapart -x PhysicalDrive20
creapart -d PhysicalDrive22
creapart -x PhysicalDrive22
creapart -d PhysicalDrive23
creapart -x PhysicalDrive23
creapart -d PhysicalDrive24
creapart -x PhysicalDrive24
creapart -d PhysicalDrive25
creapart -x PhysicalDrive25
creapart -d PhysicalDrive26
creapart -x PhysicalDrive26
creapart -d PhysicalDrive28
creapart -x PhysicalDrive28
creapart -d PhysicalDrive29
```

```
creapart -x PhysicalDrive29
creapart -d PhysicalDrive30
creapart -x PhysicalDrive30
creapart -d PhysicalDrive31
creapart -x PhysicalDrive31
creapart -d PhysicalDrive32
creapart -x PhysicalDrive32
creapart -d PhysicalDrive33
creapart -x PhysicalDrive33
creapart -d PhysicalDrive34
creapart -x PhysicalDrive34
creapart -d PhysicalDrive36
creapart -x PhysicalDrive36
creapart -d PhysicalDrive37
creapart -x PhysicalDrive37
creapart -d PhysicalDrive38
creapart -x PhysicalDrive38
creapart -d PhysicalDrive39
creapart -x PhysicalDrive39
creapart -d PhysicalDrive40
creapart -x PhysicalDrive40
creapart -d PhysicalDrive42
creapart -x PhysicalDrive42
creapart -d PhysicalDrive43
creapart -x PhysicalDrive43
creapart -d PhysicalDrive44
creapart -x PhysicalDrive44
creapart -d PhysicalDrive45
creapart -x PhysicalDrive45
creapart -d PhysicalDrive46
creapart -x PhysicalDrive46
creapart -d PhysicalDrive47
creapart -x PhysicalDrive47
creapart -d PhysicalDrive48
creapart -x PhysicalDrive48
creapart -d PhysicalDrive50
creapart -x PhysicalDrive50
creapart -d PhysicalDrive51
creapart -x PhysicalDrive51
creapart -d PhysicalDrive52
creapart -x PhysicalDrive52
creapart -d PhysicalDrive53
creapart -x PhysicalDrive53
creapart -d PhysicalDrive54
creapart -x PhysicalDrive54
creapart -d PhysicalDrive56
creapart -x PhysicalDrive56
creapart -d PhysicalDrive57
creapart -x PhysicalDrive57
creapart -d PhysicalDrive12
creapart -x PhysicalDrive12
creapart -d PhysicalDrive27
creapart -x PhysicalDrive27
creapart -d PhysicalDrive41
creapart -x PhysicalDrive41
creapart -d PhysicalDrive55
creapart -x PhysicalDrive55
REM      log_111 drive: 12: partition: 1
creapart -l PhysicalDrive12      22533
REM      log_121 drive: 12: partition: 2
creapart -l PhysicalDrive12      22533
```

```

REM      log_231 drive: 27: partition: 1
creapart -1 PhysicalDrive27      22533

REM      log_241 drive: 27: partition: 2
creapart -1 PhysicalDrive27      22533

REM      log_351 drive: 41: partition: 1
creapart -1 PhysicalDrive41      22533

REM      log_361 drive: 41: partition: 2
creapart -1 PhysicalDrive41      22533

REM      log_471 drive: 55: partition: 1
creapart -1 PhysicalDrive55      22533

REM      log_481 drive: 55: partition: 2
creapart -1 PhysicalDrive55      22533

REM      stok_0_0 drive: 1: partition: 1
creapart -1 PhysicalDrive1       2355

REM      stok_0_1 drive: 2: partition: 1
creapart -1 PhysicalDrive2       2355

REM      stok_0_2 drive: 3: partition: 1
creapart -1 PhysicalDrive3       2355

REM      stok_0_3 drive: 4: partition: 1
creapart -1 PhysicalDrive4       2355

REM      stok_0_4 drive: 5: partition: 1
creapart -1 PhysicalDrive5       2355

REM      stok_0_5 drive: 7: partition: 1
creapart -1 PhysicalDrive7       2355

REM      stok_0_6 drive: 8: partition: 1
creapart -1 PhysicalDrive8       2355

REM      stok_0_7 drive: 9: partition: 1
creapart -1 PhysicalDrive9       2355

REM      stok_0_8 drive: 10: partition: 1
creapart -1 PhysicalDrive10      2355

REM      stok_0_9 drive: 11: partition: 1
creapart -1 PhysicalDrive11      2355

REM      stok_0_10 drive: 13: partition: 1
creapart -1 PhysicalDrive13      2355

REM      stok_0_11 drive: 14: partition: 1
creapart -1 PhysicalDrive14      2355

REM      stok_0_12 drive: 16: partition: 1
creapart -1 PhysicalDrive16      2355

REM      stok_0_13 drive: 17: partition: 1
creapart -1 PhysicalDrive17      2355

REM      stok_0_14 drive: 18: partition: 1
creapart -1 PhysicalDrive18      2355

```

```

REM      stok_0_15 drive: 19: partition: 1
creapart -1 PhysicalDrive19      2355

REM      stok_0_16 drive: 20: partition: 1
creapart -1 PhysicalDrive20      2355

REM      stok_0_17 drive: 22: partition: 1
creapart -1 PhysicalDrive22      2355

REM      stok_0_18 drive: 23: partition: 1
creapart -1 PhysicalDrive23      2355

REM      stok_0_19 drive: 24: partition: 1
creapart -1 PhysicalDrive24      2355

REM      stok_0_20 drive: 25: partition: 1
creapart -1 PhysicalDrive25      2355

REM      stok_0_21 drive: 26: partition: 1
creapart -1 PhysicalDrive26      2355

REM      stok_0_22 drive: 28: partition: 1
creapart -1 PhysicalDrive28      2355

REM      stok_0_23 drive: 29: partition: 1
creapart -1 PhysicalDrive29      2355

REM      stok_0_24 drive: 30: partition: 1
creapart -1 PhysicalDrive30      2355

REM      stok_0_25 drive: 31: partition: 1
creapart -1 PhysicalDrive31      2355

REM      stok_0_26 drive: 32: partition: 1
creapart -1 PhysicalDrive32      2355

REM      stok_0_27 drive: 33: partition: 1
creapart -1 PhysicalDrive33      2355

REM      stok_0_28 drive: 34: partition: 1
creapart -1 PhysicalDrive34      2355

REM      stok_0_29 drive: 36: partition: 1
creapart -1 PhysicalDrive36      2355

REM      stok_0_30 drive: 37: partition: 1
creapart -1 PhysicalDrive37      2355

REM      stok_0_31 drive: 38: partition: 1
creapart -1 PhysicalDrive38      2355

REM      stok_0_32 drive: 39: partition: 1
creapart -1 PhysicalDrive39      2355

REM      stok_0_33 drive: 40: partition: 1
creapart -1 PhysicalDrive40      2355

REM      stok_0_34 drive: 42: partition: 1
creapart -1 PhysicalDrive42      2355

REM      stok_0_35 drive: 43: partition: 1
creapart -1 PhysicalDrive43      2355

```

```

REM      stok_0_36 drive: 44: partition: 1
creapart -1 PhysicalDrive44      2355

REM      stok_0_37 drive: 45: partition: 1
creapart -1 PhysicalDrive45      2355

REM      stok_0_38 drive: 46: partition: 1
creapart -1 PhysicalDrive46      2355

REM      stok_0_39 drive: 47: partition: 1
creapart -1 PhysicalDrive47      2355

REM      stok_0_40 drive: 48: partition: 1
creapart -1 PhysicalDrive48      2355

REM      stok_0_41 drive: 50: partition: 1
creapart -1 PhysicalDrive50      2355

REM      stok_0_42 drive: 51: partition: 1
creapart -1 PhysicalDrive51      2355

REM      stok_0_43 drive: 52: partition: 1
creapart -1 PhysicalDrive52      2355

REM      stok_0_44 drive: 53: partition: 1
creapart -1 PhysicalDrive53      2355

REM      stok_0_45 drive: 54: partition: 1
creapart -1 PhysicalDrive54      2355

REM      stok_0_46 drive: 56: partition: 1
creapart -1 PhysicalDrive56      2355

REM      stok_0_47 drive: 57: partition: 1
creapart -1 PhysicalDrive57      2355

REM      stok_0_48 drive: 1: partition: 2
creapart -1 PhysicalDrive1       2355

REM      stok_0_49 drive: 2: partition: 2
creapart -1 PhysicalDrive2       2355

REM      stok_0_50 drive: 3: partition: 2
creapart -1 PhysicalDrive3       2355

REM      stok_0_51 drive: 4: partition: 2
creapart -1 PhysicalDrive4       2355

REM      stok_0_52 drive: 5: partition: 2
creapart -1 PhysicalDrive5       2355

REM      stok_0_53 drive: 7: partition: 2
creapart -1 PhysicalDrive7       2355

REM      stok_0_54 drive: 8: partition: 2
creapart -1 PhysicalDrive8       2355

REM      stok_0_55 drive: 9: partition: 2
creapart -1 PhysicalDrive9       2355

REM      stok_0_56 drive: 10: partition: 2
creapart -1 PhysicalDrive10      2355

```

```

REM      stok_0_57 drive: 11: partition: 2
creapart -1 PhysicalDrive11      2355

REM      stok_0_58 drive: 13: partition: 2
creapart -1 PhysicalDrive13      2355

REM      stok_0_59 drive: 14: partition: 2
creapart -1 PhysicalDrive14      2355

REM      stok_0_60 drive: 16: partition: 2
creapart -1 PhysicalDrive16      2355

REM      stok_0_61 drive: 17: partition: 2
creapart -1 PhysicalDrive17      2355

REM      stok_0_62 drive: 18: partition: 2
creapart -1 PhysicalDrive18      2355

REM      stok_0_63 drive: 19: partition: 2
creapart -1 PhysicalDrive19      2355

REM      stok_0_64 drive: 20: partition: 2
creapart -1 PhysicalDrive20      2355

REM      stok_0_65 drive: 22: partition: 2
creapart -1 PhysicalDrive22      2355

REM      stok_0_66 drive: 23: partition: 2
creapart -1 PhysicalDrive23      2355

REM      stok_0_67 drive: 24: partition: 2
creapart -1 PhysicalDrive24      2355

REM      stok_0_68 drive: 25: partition: 2
creapart -1 PhysicalDrive25      2355

REM      stok_0_69 drive: 26: partition: 2
creapart -1 PhysicalDrive26      2355

REM      stok_0_70 drive: 28: partition: 2
creapart -1 PhysicalDrive28      2355

REM      stok_0_71 drive: 29: partition: 2
creapart -1 PhysicalDrive29      2355

REM      stok_0_72 drive: 30: partition: 2
creapart -1 PhysicalDrive30      2355

REM      stok_0_73 drive: 31: partition: 2
creapart -1 PhysicalDrive31      2355

REM      stok_0_74 drive: 32: partition: 2
creapart -1 PhysicalDrive32      2355

REM      stok_0_75 drive: 33: partition: 2
creapart -1 PhysicalDrive33      2355

REM      stok_0_76 drive: 34: partition: 2
creapart -1 PhysicalDrive34      2355

REM      stok_0_77 drive: 36: partition: 2
creapart -1 PhysicalDrive36      2355

```

```

REM      stok_0_78 drive: 37: partition: 2
creapart -1 PhysicalDrive37          2355

REM      stok_0_79 drive: 38: partition: 2
creapart -1 PhysicalDrive38          2355

REM      stok_0_80 drive: 39: partition: 2
creapart -1 PhysicalDrive39          2355

REM      stok_0_81 drive: 40: partition: 2
creapart -1 PhysicalDrive40          2355

REM      stok_0_82 drive: 42: partition: 2
creapart -1 PhysicalDrive42          2355

REM      stok_0_83 drive: 43: partition: 2
creapart -1 PhysicalDrive43          2355

REM      stok_0_84 drive: 44: partition: 2
creapart -1 PhysicalDrive44          2355

REM      stok_0_85 drive: 45: partition: 2
creapart -1 PhysicalDrive45          2355

REM      stok_0_86 drive: 46: partition: 2
creapart -1 PhysicalDrive46          2355

REM      stok_0_87 drive: 47: partition: 2
creapart -1 PhysicalDrive47          2355

REM      stok_0_88 drive: 48: partition: 2
creapart -1 PhysicalDrive48          2355

REM      stok_0_89 drive: 50: partition: 2
creapart -1 PhysicalDrive50          2355

REM      stok_0_90 drive: 51: partition: 2
creapart -1 PhysicalDrive51          2355

REM      stok_0_91 drive: 52: partition: 2
creapart -1 PhysicalDrive52          2355

REM      stok_0_92 drive: 53: partition: 2
creapart -1 PhysicalDrive53          2355

REM      stok_0_93 drive: 54: partition: 2
creapart -1 PhysicalDrive54          2355

REM      stok_0_94 drive: 56: partition: 2
creapart -1 PhysicalDrive56          2355

REM      stok_0_95 drive: 57: partition: 2
creapart -1 PhysicalDrive57          2355

REM      cust_0_0 drive: 1: partition: 3
creapart -1 PhysicalDrive1           1785

REM      cust_0_1 drive: 2: partition: 3
creapart -1 PhysicalDrive2           1785

REM      cust_0_2 drive: 3: partition: 3
creapart -1 PhysicalDrive3           1785

```

```

REM      cust_0_3 drive: 4: partition: 3
creapart -1 PhysicalDrive4           1785

REM      cust_0_4 drive: 5: partition: 3
creapart -1 PhysicalDrive5           1785

REM      cust_0_5 drive: 7: partition: 3
creapart -1 PhysicalDrive7           1785

REM      cust_0_6 drive: 8: partition: 3
creapart -1 PhysicalDrive8           1785

REM      cust_0_7 drive: 9: partition: 3
creapart -1 PhysicalDrive9           1785

REM      cust_0_8 drive: 10: partition: 3
creapart -1 PhysicalDrive10          1785

REM      cust_0_9 drive: 11: partition: 3
creapart -1 PhysicalDrive11          1785

REM      cust_0_10 drive: 13: partition: 3
creapart -1 PhysicalDrive13          1785

REM      cust_0_11 drive: 14: partition: 3
creapart -1 PhysicalDrive14          1785

REM      cust_0_12 drive: 16: partition: 3
creapart -1 PhysicalDrive16          1785

REM      cust_0_13 drive: 17: partition: 3
creapart -1 PhysicalDrive17          1785

REM      cust_0_14 drive: 18: partition: 3
creapart -1 PhysicalDrive18          1785

REM      cust_0_15 drive: 19: partition: 3
creapart -1 PhysicalDrive19          1785

REM      cust_0_16 drive: 20: partition: 3
creapart -1 PhysicalDrive20          1785

REM      cust_0_17 drive: 22: partition: 3
creapart -1 PhysicalDrive22          1785

REM      cust_0_18 drive: 23: partition: 3
creapart -1 PhysicalDrive23          1785

REM      cust_0_19 drive: 24: partition: 3
creapart -1 PhysicalDrive24          1785

REM      cust_0_20 drive: 25: partition: 3
creapart -1 PhysicalDrive25          1785

REM      cust_0_21 drive: 26: partition: 3
creapart -1 PhysicalDrive26          1785

REM      cust_0_22 drive: 28: partition: 3
creapart -1 PhysicalDrive28          1785

REM      cust_0_23 drive: 29: partition: 3
creapart -1 PhysicalDrive29          1785

```

REM cust_0_24 drive: 30: partition: 3
creapart -1 PhysicalDrive30 1785

REM cust_0_25 drive: 31: partition: 3
creapart -1 PhysicalDrive31 1785

REM cust_0_26 drive: 32: partition: 3
creapart -1 PhysicalDrive32 1785

REM cust_0_27 drive: 33: partition: 3
creapart -1 PhysicalDrive33 1785

REM cust_0_28 drive: 34: partition: 3
creapart -1 PhysicalDrive34 1785

REM cust_0_29 drive: 36: partition: 3
creapart -1 PhysicalDrive36 1785

REM cust_0_30 drive: 37: partition: 3
creapart -1 PhysicalDrive37 1785

REM cust_0_31 drive: 38: partition: 3
creapart -1 PhysicalDrive38 1785

REM cust_0_32 drive: 39: partition: 3
creapart -1 PhysicalDrive39 1785

REM cust_0_33 drive: 40: partition: 3
creapart -1 PhysicalDrive40 1785

REM cust_0_34 drive: 42: partition: 3
creapart -1 PhysicalDrive42 1785

REM cust_0_35 drive: 43: partition: 3
creapart -1 PhysicalDrive43 1785

REM cust_0_36 drive: 44: partition: 3
creapart -1 PhysicalDrive44 1785

REM cust_0_37 drive: 45: partition: 3
creapart -1 PhysicalDrive45 1785

REM cust_0_38 drive: 46: partition: 3
creapart -1 PhysicalDrive46 1785

REM cust_0_39 drive: 47: partition: 3
creapart -1 PhysicalDrive47 1785

REM cust_0_40 drive: 48: partition: 3
creapart -1 PhysicalDrive48 1785

REM cust_0_41 drive: 50: partition: 3
creapart -1 PhysicalDrive50 1785

REM cust_0_42 drive: 51: partition: 3
creapart -1 PhysicalDrive51 1785

REM cust_0_43 drive: 52: partition: 3
creapart -1 PhysicalDrive52 1785

REM cust_0_44 drive: 53: partition: 3
creapart -1 PhysicalDrive53 1785

REM cust_0_45 drive: 54: partition: 3
creapart -1 PhysicalDrive54 1785

REM cust_0_46 drive: 56: partition: 3
creapart -1 PhysicalDrive56 1785

REM cust_0_47 drive: 57: partition: 3
creapart -1 PhysicalDrive57 1785

REM cust_0_48 drive: 1: partition: 4
creapart -1 PhysicalDrive1 1785

REM cust_0_49 drive: 2: partition: 4
creapart -1 PhysicalDrive2 1785

REM cust_0_50 drive: 3: partition: 4
creapart -1 PhysicalDrive3 1785

REM cust_0_51 drive: 4: partition: 4
creapart -1 PhysicalDrive4 1785

REM cust_0_52 drive: 5: partition: 4
creapart -1 PhysicalDrive5 1785

REM cust_0_53 drive: 7: partition: 4
creapart -1 PhysicalDrive7 1785

REM cust_0_54 drive: 8: partition: 4
creapart -1 PhysicalDrive8 1785

REM cust_0_55 drive: 9: partition: 4
creapart -1 PhysicalDrive9 1785

REM cust_0_56 drive: 10: partition: 4
creapart -1 PhysicalDrive10 1785

REM cust_0_57 drive: 11: partition: 4
creapart -1 PhysicalDrive11 1785

REM cust_0_58 drive: 13: partition: 4
creapart -1 PhysicalDrive13 1785

REM cust_0_59 drive: 14: partition: 4
creapart -1 PhysicalDrive14 1785

REM cust_0_60 drive: 16: partition: 4
creapart -1 PhysicalDrive16 1785

REM cust_0_61 drive: 17: partition: 4
creapart -1 PhysicalDrive17 1785

REM cust_0_62 drive: 18: partition: 4
creapart -1 PhysicalDrive18 1785

REM cust_0_63 drive: 19: partition: 4
creapart -1 PhysicalDrive19 1785

REM cust_0_64 drive: 20: partition: 4
creapart -1 PhysicalDrive20 1785

REM cust_0_65 drive: 22: partition: 4
creapart -1 PhysicalDrive22 1785

```

REM      cust_0_66 drive: 23: partition: 4
creapart -1 PhysicalDrive23          1785

REM      cust_0_67 drive: 24: partition: 4
creapart -1 PhysicalDrive24          1785

REM      cust_0_68 drive: 25: partition: 4
creapart -1 PhysicalDrive25          1785

REM      cust_0_69 drive: 26: partition: 4
creapart -1 PhysicalDrive26          1785

REM      cust_0_70 drive: 28: partition: 4
creapart -1 PhysicalDrive28          1785

REM      cust_0_71 drive: 29: partition: 4
creapart -1 PhysicalDrive29          1785

REM      cust_0_72 drive: 30: partition: 4
creapart -1 PhysicalDrive30          1785

REM      cust_0_73 drive: 31: partition: 4
creapart -1 PhysicalDrive31          1785

REM      cust_0_74 drive: 32: partition: 4
creapart -1 PhysicalDrive32          1785

REM      cust_0_75 drive: 33: partition: 4
creapart -1 PhysicalDrive33          1785

REM      cust_0_76 drive: 34: partition: 4
creapart -1 PhysicalDrive34          1785

REM      cust_0_77 drive: 36: partition: 4
creapart -1 PhysicalDrive36          1785

REM      cust_0_78 drive: 37: partition: 4
creapart -1 PhysicalDrive37          1785

REM      cust_0_79 drive: 38: partition: 4
creapart -1 PhysicalDrive38          1785

REM      cust_0_80 drive: 39: partition: 4
creapart -1 PhysicalDrive39          1785

REM      cust_0_81 drive: 40: partition: 4
creapart -1 PhysicalDrive40          1785

REM      cust_0_82 drive: 42: partition: 4
creapart -1 PhysicalDrive42          1785

REM      cust_0_83 drive: 43: partition: 4
creapart -1 PhysicalDrive43          1785

REM      cust_0_84 drive: 44: partition: 4
creapart -1 PhysicalDrive44          1785

REM      cust_0_85 drive: 45: partition: 4
creapart -1 PhysicalDrive45          1785

REM      cust_0_86 drive: 46: partition: 4
creapart -1 PhysicalDrive46          1785

```

```

REM      cust_0_87 drive: 47: partition: 4
creapart -1 PhysicalDrive47          1785

REM      cust_0_88 drive: 48: partition: 4
creapart -1 PhysicalDrive48          1785

REM      cust_0_89 drive: 50: partition: 4
creapart -1 PhysicalDrive50          1785

REM      cust_0_90 drive: 51: partition: 4
creapart -1 PhysicalDrive51          1785

REM      cust_0_91 drive: 52: partition: 4
creapart -1 PhysicalDrive52          1785

REM      cust_0_92 drive: 53: partition: 4
creapart -1 PhysicalDrive53          1785

REM      cust_0_93 drive: 54: partition: 4
creapart -1 PhysicalDrive54          1785

REM      cust_0_94 drive: 56: partition: 4
creapart -1 PhysicalDrive56          1785

REM      cust_0_95 drive: 57: partition: 4
creapart -1 PhysicalDrive57          1785

REM      icust2_0_0 drive: 1: partition: 5
creapart -1 PhysicalDrive1           245

REM      icust2_0_1 drive: 2: partition: 5
creapart -1 PhysicalDrive2           245

REM      icust2_0_2 drive: 3: partition: 5
creapart -1 PhysicalDrive3           245

REM      icust2_0_3 drive: 4: partition: 5
creapart -1 PhysicalDrive4           245

REM      icust2_0_4 drive: 5: partition: 5
creapart -1 PhysicalDrive5           245

REM      icust2_0_5 drive: 7: partition: 5
creapart -1 PhysicalDrive7           245

REM      icust2_0_6 drive: 8: partition: 5
creapart -1 PhysicalDrive8           245

REM      icust2_0_7 drive: 9: partition: 5
creapart -1 PhysicalDrive9           245

REM      icust2_0_8 drive: 10: partition: 5
creapart -1 PhysicalDrive10          245

REM      icust2_0_9 drive: 11: partition: 5
creapart -1 PhysicalDrive11          245

REM      icust2_0_10 drive: 13: partition: 5
creapart -1 PhysicalDrive13          245

REM      icust2_0_11 drive: 14: partition: 5
creapart -1 PhysicalDrive14          245

```

```

REM      icust2_0_12 drive: 16: partition: 5
      creapart -1 PhysicalDrive16          245
REM      icust2_0_13 drive: 17: partition: 5
      creapart -1 PhysicalDrive17          245
REM      icust2_0_14 drive: 18: partition: 5
      creapart -1 PhysicalDrive18          245
REM      icust2_0_15 drive: 19: partition: 5
      creapart -1 PhysicalDrive19          245
REM      icust2_0_16 drive: 20: partition: 5
      creapart -1 PhysicalDrive20          245
REM      icust2_0_17 drive: 22: partition: 5
      creapart -1 PhysicalDrive22          245
REM      icust2_0_18 drive: 23: partition: 5
      creapart -1 PhysicalDrive23          245
REM      icust2_0_19 drive: 24: partition: 5
      creapart -1 PhysicalDrive24          245
REM      icust2_0_20 drive: 25: partition: 5
      creapart -1 PhysicalDrive25          245
REM      icust2_0_21 drive: 26: partition: 5
      creapart -1 PhysicalDrive26          245
REM      icust2_0_22 drive: 28: partition: 5
      creapart -1 PhysicalDrive28          245
REM      icust2_0_23 drive: 29: partition: 5
      creapart -1 PhysicalDrive29          245
REM      icust2_0_24 drive: 30: partition: 5
      creapart -1 PhysicalDrive30          245
REM      icust2_0_25 drive: 31: partition: 5
      creapart -1 PhysicalDrive31          245
REM      icust2_0_26 drive: 32: partition: 5
      creapart -1 PhysicalDrive32          245
REM      icust2_0_27 drive: 33: partition: 5
      creapart -1 PhysicalDrive33          245
REM      icust2_0_28 drive: 34: partition: 5
      creapart -1 PhysicalDrive34          245
REM      icust2_0_29 drive: 36: partition: 5
      creapart -1 PhysicalDrive36          245
REM      icust2_0_30 drive: 37: partition: 5
      creapart -1 PhysicalDrive37          245
REM      icust2_0_31 drive: 38: partition: 5
      creapart -1 PhysicalDrive38          245
REM      icust2_0_32 drive: 39: partition: 5
      creapart -1 PhysicalDrive39          245

```

```

REM      icust2_0_33 drive: 40: partition: 5
      creapart -1 PhysicalDrive40          245
REM      icust2_0_34 drive: 42: partition: 5
      creapart -1 PhysicalDrive42          245
REM      icust2_0_35 drive: 43: partition: 5
      creapart -1 PhysicalDrive43          245
REM      icust2_0_36 drive: 44: partition: 5
      creapart -1 PhysicalDrive44          245
REM      icust2_0_37 drive: 45: partition: 5
      creapart -1 PhysicalDrive45          245
REM      icust2_0_38 drive: 46: partition: 5
      creapart -1 PhysicalDrive46          245
REM      icust2_0_39 drive: 47: partition: 5
      creapart -1 PhysicalDrive47          245
REM      icust2_0_40 drive: 48: partition: 5
      creapart -1 PhysicalDrive48          245
REM      icust2_0_41 drive: 50: partition: 5
      creapart -1 PhysicalDrive50          245
REM      icust2_0_42 drive: 51: partition: 5
      creapart -1 PhysicalDrive51          245
REM      icust2_0_43 drive: 52: partition: 5
      creapart -1 PhysicalDrive52          245
REM      icust2_0_44 drive: 53: partition: 5
      creapart -1 PhysicalDrive53          245
REM      icust2_0_45 drive: 54: partition: 5
      creapart -1 PhysicalDrive54          245
REM      icust2_0_46 drive: 56: partition: 5
      creapart -1 PhysicalDrive56          245
REM      icust2_0_47 drive: 57: partition: 5
      creapart -1 PhysicalDrive57          245
REM      item_0_0 drive: 1: partition: 6
      creapart -1 PhysicalDrive1           15
REM      item_0_1 drive: 2: partition: 6
      creapart -1 PhysicalDrive2           15
REM      item_0_2 drive: 3: partition: 6
      creapart -1 PhysicalDrive3           15
REM      item_0_3 drive: 4: partition: 6
      creapart -1 PhysicalDrive4           15
REM      item_0_4 drive: 5: partition: 6
      creapart -1 PhysicalDrive5           15
REM      item_0_5 drive: 7: partition: 6
      creapart -1 PhysicalDrive7           15

```

```

REM      item_0_6 drive: 8: partition: 6
creapart -l PhysicalDrive8          15

REM      item_0_7 drive: 9: partition: 6
creapart -l PhysicalDrive9          15

REM      item_0_8 drive: 10: partition: 6
creapart -l PhysicalDrive10         15

REM      item_0_9 drive: 11: partition: 6
creapart -l PhysicalDrive11         15

REM      item_0_10 drive: 13: partition: 6
creapart -l PhysicalDrive13         15

REM      item_0_11 drive: 14: partition: 6
creapart -l PhysicalDrive14         15

REM      item_0_12 drive: 16: partition: 6
creapart -l PhysicalDrive16         15

REM      item_0_13 drive: 17: partition: 6
creapart -l PhysicalDrive17         15

REM      item_0_14 drive: 18: partition: 6
creapart -l PhysicalDrive18         15

REM      item_0_15 drive: 19: partition: 6
creapart -l PhysicalDrive19         15

REM      item_0_16 drive: 20: partition: 6
creapart -l PhysicalDrive20         15

REM      item_0_17 drive: 22: partition: 6
creapart -l PhysicalDrive22         15

REM      item_0_18 drive: 23: partition: 6
creapart -l PhysicalDrive23         15

REM      item_0_19 drive: 24: partition: 6
creapart -l PhysicalDrive24         15

REM      item_0_20 drive: 25: partition: 6
creapart -l PhysicalDrive25         15

REM      item_0_21 drive: 26: partition: 6
creapart -l PhysicalDrive26         15

REM      item_0_22 drive: 28: partition: 6
creapart -l PhysicalDrive28         15

REM      item_0_23 drive: 29: partition: 6
creapart -l PhysicalDrive29         15

REM      item_0_24 drive: 30: partition: 6
creapart -l PhysicalDrive30         15

REM      item_0_25 drive: 31: partition: 6
creapart -l PhysicalDrive31         15

REM      item_0_26 drive: 32: partition: 6
creapart -l PhysicalDrive32         15

```

```

REM      item_0_27 drive: 33: partition: 6
creapart -l PhysicalDrive33         15

REM      item_0_28 drive: 34: partition: 6
creapart -l PhysicalDrive34         15

REM      item_0_29 drive: 36: partition: 6
creapart -l PhysicalDrive36         15

REM      item_0_30 drive: 37: partition: 6
creapart -l PhysicalDrive37         15

REM      item_0_31 drive: 38: partition: 6
creapart -l PhysicalDrive38         15

REM      item_0_32 drive: 39: partition: 6
creapart -l PhysicalDrive39         15

REM      item_0_33 drive: 40: partition: 6
creapart -l PhysicalDrive40         15

REM      item_0_34 drive: 42: partition: 6
creapart -l PhysicalDrive42         15

REM      item_0_35 drive: 43: partition: 6
creapart -l PhysicalDrive43         15

REM      item_0_36 drive: 44: partition: 6
creapart -l PhysicalDrive44         15

REM      item_0_37 drive: 45: partition: 6
creapart -l PhysicalDrive45         15

REM      item_0_38 drive: 46: partition: 6
creapart -l PhysicalDrive46         15

REM      item_0_39 drive: 47: partition: 6
creapart -l PhysicalDrive47         15

REM      item_0_40 drive: 48: partition: 6
creapart -l PhysicalDrive48         15

REM      item_0_41 drive: 50: partition: 6
creapart -l PhysicalDrive50         15

REM      item_0_42 drive: 51: partition: 6
creapart -l PhysicalDrive51         15

REM      item_0_43 drive: 52: partition: 6
creapart -l PhysicalDrive52         15

REM      item_0_44 drive: 53: partition: 6
creapart -l PhysicalDrive53         15

REM      item_0_45 drive: 54: partition: 6
creapart -l PhysicalDrive54         15

REM      item_0_46 drive: 56: partition: 6
creapart -l PhysicalDrive56         15

REM      item_0_47 drive: 57: partition: 6
creapart -l PhysicalDrive57         15

```

```

REM      ordl_0_0 drive: 1: partition: 7
creapart -1 PhysicalDrive1          4645

REM      ordl_0_1 drive: 2: partition: 7
creapart -1 PhysicalDrive2          4645

REM      ordl_0_2 drive: 3: partition: 7
creapart -1 PhysicalDrive3          4645

REM      ordl_1_0 drive: 4: partition: 7
creapart -1 PhysicalDrive4          4645

REM      ordl_1_1 drive: 5: partition: 7
creapart -1 PhysicalDrive5          4645

REM      ordl_1_2 drive: 7: partition: 7
creapart -1 PhysicalDrive7          4645

REM      ordl_2_0 drive: 8: partition: 7
creapart -1 PhysicalDrive8          4645

REM      ordl_2_1 drive: 9: partition: 7
creapart -1 PhysicalDrive9          4645

REM      ordl_2_2 drive: 10: partition: 7
creapart -1 PhysicalDrive10         4645

REM      ordl_3_0 drive: 11: partition: 7
creapart -1 PhysicalDrive11         4645

REM      ordl_3_1 drive: 13: partition: 7
creapart -1 PhysicalDrive13         4645

REM      ordl_3_2 drive: 14: partition: 7
creapart -1 PhysicalDrive14         4645

REM      ordl_0_3 drive: 16: partition: 7
creapart -1 PhysicalDrive16         4645

REM      ordl_0_4 drive: 17: partition: 7
creapart -1 PhysicalDrive17         4645

REM      ordl_0_5 drive: 18: partition: 7
creapart -1 PhysicalDrive18         4645

REM      ordl_1_3 drive: 19: partition: 7
creapart -1 PhysicalDrive19         4645

REM      ordl_1_4 drive: 20: partition: 7
creapart -1 PhysicalDrive20         4645

REM      ordl_1_5 drive: 22: partition: 7
creapart -1 PhysicalDrive22         4645

REM      ordl_2_3 drive: 23: partition: 7
creapart -1 PhysicalDrive23         4645

REM      ordl_2_4 drive: 24: partition: 7
creapart -1 PhysicalDrive24         4645

REM      ordl_2_5 drive: 25: partition: 7
creapart -1 PhysicalDrive25         4645

```

```

REM      ordl_3_3 drive: 26: partition: 7
creapart -1 PhysicalDrive26         4645

REM      ordl_3_4 drive: 28: partition: 7
creapart -1 PhysicalDrive28         4645

REM      ordl_3_5 drive: 29: partition: 7
creapart -1 PhysicalDrive29         4645

REM      ordl_0_6 drive: 30: partition: 7
creapart -1 PhysicalDrive30         4645

REM      ordl_0_7 drive: 31: partition: 7
creapart -1 PhysicalDrive31         4645

REM      ordl_0_8 drive: 32: partition: 7
creapart -1 PhysicalDrive32         4645

REM      ordl_1_6 drive: 33: partition: 7
creapart -1 PhysicalDrive33         4645

REM      ordl_1_7 drive: 34: partition: 7
creapart -1 PhysicalDrive34         4645

REM      ordl_1_8 drive: 36: partition: 7
creapart -1 PhysicalDrive36         4645

REM      ordl_2_6 drive: 37: partition: 7
creapart -1 PhysicalDrive37         4645

REM      ordl_2_7 drive: 38: partition: 7
creapart -1 PhysicalDrive38         4645

REM      ordl_2_8 drive: 39: partition: 7
creapart -1 PhysicalDrive39         4645

REM      ordl_3_6 drive: 40: partition: 7
creapart -1 PhysicalDrive40         4645

REM      ordl_3_7 drive: 42: partition: 7
creapart -1 PhysicalDrive42         4645

REM      ordl_3_8 drive: 43: partition: 7
creapart -1 PhysicalDrive43         4645

REM      ordl_0_9 drive: 44: partition: 7
creapart -1 PhysicalDrive44         4645

REM      ordl_0_10 drive: 45: partition: 7
creapart -1 PhysicalDrive45         4645

REM      ordl_0_11 drive: 46: partition: 7
creapart -1 PhysicalDrive46         4645

REM      ordl_1_9 drive: 47: partition: 7
creapart -1 PhysicalDrive47         4645

REM      ordl_1_10 drive: 48: partition: 7
creapart -1 PhysicalDrive48         4645

REM      ordl_1_11 drive: 50: partition: 7
creapart -1 PhysicalDrive50         4645

```

```

REM      ordl_2_9 drive: 51: partition: 7
creapart -1 PhysicalDrive51          4645

REM      ordl_2_10 drive: 52: partition: 7
creapart -1 PhysicalDrive52          4645

REM      ordl_2_11 drive: 53: partition: 7
creapart -1 PhysicalDrive53          4645

REM      ordl_3_9 drive: 54: partition: 7
creapart -1 PhysicalDrive54          4645

REM      ordl_3_10 drive: 56: partition: 7
creapart -1 PhysicalDrive56          4645

REM      ordl_3_11 drive: 57: partition: 7
creapart -1 PhysicalDrive57          4645

REM      ordr_0_0 drive: 1: partition: 8
creapart -1 PhysicalDrive1           375

REM      ordr_0_1 drive: 2: partition: 8
creapart -1 PhysicalDrive2           375

REM      ordr_0_2 drive: 3: partition: 8
creapart -1 PhysicalDrive3           375

REM      ordr_1_0 drive: 4: partition: 8
creapart -1 PhysicalDrive4           375

REM      ordr_1_1 drive: 5: partition: 8
creapart -1 PhysicalDrive5           375

REM      ordr_1_2 drive: 7: partition: 8
creapart -1 PhysicalDrive7           375

REM      ordr_2_0 drive: 8: partition: 8
creapart -1 PhysicalDrive8           375

REM      ordr_2_1 drive: 9: partition: 8
creapart -1 PhysicalDrive9           375

REM      ordr_2_2 drive: 10: partition: 8
creapart -1 PhysicalDrive10          375

REM      ordr_3_0 drive: 11: partition: 8
creapart -1 PhysicalDrive11          375

REM      ordr_3_1 drive: 13: partition: 8
creapart -1 PhysicalDrive13          375

REM      ordr_3_2 drive: 14: partition: 8
creapart -1 PhysicalDrive14          375

REM      ordr_0_3 drive: 16: partition: 8
creapart -1 PhysicalDrive16          375

REM      ordr_0_4 drive: 17: partition: 8
creapart -1 PhysicalDrive17          375

REM      ordr_0_5 drive: 18: partition: 8
creapart -1 PhysicalDrive18          375

```

```

REM      ordr_1_3 drive: 19: partition: 8
creapart -1 PhysicalDrive19          375

REM      ordr_1_4 drive: 20: partition: 8
creapart -1 PhysicalDrive20          375

REM      ordr_1_5 drive: 22: partition: 8
creapart -1 PhysicalDrive22          375

REM      ordr_2_3 drive: 23: partition: 8
creapart -1 PhysicalDrive23          375

REM      ordr_2_4 drive: 24: partition: 8
creapart -1 PhysicalDrive24          375

REM      ordr_2_5 drive: 25: partition: 8
creapart -1 PhysicalDrive25          375

REM      ordr_3_3 drive: 26: partition: 8
creapart -1 PhysicalDrive26          375

REM      ordr_3_4 drive: 28: partition: 8
creapart -1 PhysicalDrive28          375

REM      ordr_3_5 drive: 29: partition: 8
creapart -1 PhysicalDrive29          375

REM      ordr_0_6 drive: 30: partition: 8
creapart -1 PhysicalDrive30          375

REM      ordr_0_7 drive: 31: partition: 8
creapart -1 PhysicalDrive31          375

REM      ordr_0_8 drive: 32: partition: 8
creapart -1 PhysicalDrive32          375

REM      ordr_1_6 drive: 33: partition: 8
creapart -1 PhysicalDrive33          375

REM      ordr_1_7 drive: 34: partition: 8
creapart -1 PhysicalDrive34          375

REM      ordr_1_8 drive: 36: partition: 8
creapart -1 PhysicalDrive36          375

REM      ordr_2_6 drive: 37: partition: 8
creapart -1 PhysicalDrive37          375

REM      ordr_2_7 drive: 38: partition: 8
creapart -1 PhysicalDrive38          375

REM      ordr_2_8 drive: 39: partition: 8
creapart -1 PhysicalDrive39          375

REM      ordr_3_6 drive: 40: partition: 8
creapart -1 PhysicalDrive40          375

REM      ordr_3_7 drive: 42: partition: 8
creapart -1 PhysicalDrive42          375

REM      ordr_3_8 drive: 43: partition: 8
creapart -1 PhysicalDrive43          375

```

```

REM      ordr_0_9 drive: 44: partition: 8
creapart -1 PhysicalDrive44          375

REM      ordr_0_10 drive: 45: partition: 8
creapart -1 PhysicalDrive45          375

REM      ordr_0_11 drive: 46: partition: 8
creapart -1 PhysicalDrive46          375

REM      ordr_1_9 drive: 47: partition: 8
creapart -1 PhysicalDrive47          375

REM      ordr_1_10 drive: 48: partition: 8
creapart -1 PhysicalDrive48          375

REM      ordr_1_11 drive: 50: partition: 8
creapart -1 PhysicalDrive50          375

REM      ordr_2_9 drive: 51: partition: 8
creapart -1 PhysicalDrive51          375

REM      ordr_2_10 drive: 52: partition: 8
creapart -1 PhysicalDrive52          375

REM      ordr_2_11 drive: 53: partition: 8
creapart -1 PhysicalDrive53          375

REM      ordr_3_9 drive: 54: partition: 8
creapart -1 PhysicalDrive54          375

REM      ordr_3_10 drive: 56: partition: 8
creapart -1 PhysicalDrive56          375

REM      ordr_3_11 drive: 57: partition: 8
creapart -1 PhysicalDrive57          375

REM      iordr1_0_0 drive: 1: partition: 9
creapart -1 PhysicalDrive1           225

REM      iordr1_0_1 drive: 2: partition: 9
creapart -1 PhysicalDrive2           225

REM      iordr1_0_2 drive: 3: partition: 9
creapart -1 PhysicalDrive3           225

REM      iordr1_1_0 drive: 4: partition: 9
creapart -1 PhysicalDrive4           225

REM      iordr1_1_1 drive: 5: partition: 9
creapart -1 PhysicalDrive5           225

REM      iordr1_1_2 drive: 7: partition: 9
creapart -1 PhysicalDrive7           225

REM      iordr1_2_0 drive: 8: partition: 9
creapart -1 PhysicalDrive8           225

REM      iordr1_2_1 drive: 9: partition: 9
creapart -1 PhysicalDrive9           225

REM      iordr1_2_2 drive: 10: partition: 9
creapart -1 PhysicalDrive10          225

```

```

REM      iordr1_3_0 drive: 11: partition: 9
creapart -1 PhysicalDrive11          225

REM      iordr1_3_1 drive: 13: partition: 9
creapart -1 PhysicalDrive13          225

REM      iordr1_3_2 drive: 14: partition: 9
creapart -1 PhysicalDrive14          225

REM      iordr1_0_3 drive: 16: partition: 9
creapart -1 PhysicalDrive16          225

REM      iordr1_0_4 drive: 17: partition: 9
creapart -1 PhysicalDrive17          225

REM      iordr1_0_5 drive: 18: partition: 9
creapart -1 PhysicalDrive18          225

REM      iordr1_1_3 drive: 19: partition: 9
creapart -1 PhysicalDrive19          225

REM      iordr1_1_4 drive: 20: partition: 9
creapart -1 PhysicalDrive20          225

REM      iordr1_1_5 drive: 22: partition: 9
creapart -1 PhysicalDrive22          225

REM      iordr1_2_3 drive: 23: partition: 9
creapart -1 PhysicalDrive23          225

REM      iordr1_2_4 drive: 24: partition: 9
creapart -1 PhysicalDrive24          225

REM      iordr1_2_5 drive: 25: partition: 9
creapart -1 PhysicalDrive25          225

REM      iordr1_3_3 drive: 26: partition: 9
creapart -1 PhysicalDrive26          225

REM      iordr1_3_4 drive: 28: partition: 9
creapart -1 PhysicalDrive28          225

REM      iordr1_3_5 drive: 29: partition: 9
creapart -1 PhysicalDrive29          225

REM      iordr1_0_6 drive: 30: partition: 9
creapart -1 PhysicalDrive30          225

REM      iordr1_0_7 drive: 31: partition: 9
creapart -1 PhysicalDrive31          225

REM      iordr1_0_8 drive: 32: partition: 9
creapart -1 PhysicalDrive32          225

REM      iordr1_1_6 drive: 33: partition: 9
creapart -1 PhysicalDrive33          225

REM      iordr1_1_7 drive: 34: partition: 9
creapart -1 PhysicalDrive34          225

REM      iordr1_1_8 drive: 36: partition: 9
creapart -1 PhysicalDrive36          225

```

```

REM      iordr1_2_6 drive: 37: partition: 9
creapart -1 PhysicalDrive37          225

REM      iordr1_2_7 drive: 38: partition: 9
creapart -1 PhysicalDrive38          225

REM      iordr1_2_8 drive: 39: partition: 9
creapart -1 PhysicalDrive39          225

REM      iordr1_3_6 drive: 40: partition: 9
creapart -1 PhysicalDrive40          225

REM      iordr1_3_7 drive: 42: partition: 9
creapart -1 PhysicalDrive42          225

REM      iordr1_3_8 drive: 43: partition: 9
creapart -1 PhysicalDrive43          225

REM      iordr1_0_9 drive: 44: partition: 9
creapart -1 PhysicalDrive44          225

REM      iordr1_0_10 drive: 45: partition: 9
creapart -1 PhysicalDrive45          225

REM      iordr1_0_11 drive: 46: partition: 9
creapart -1 PhysicalDrive46          225

REM      iordr1_1_9 drive: 47: partition: 9
creapart -1 PhysicalDrive47          225

REM      iordr1_1_10 drive: 48: partition: 9
creapart -1 PhysicalDrive48          225

REM      iordr1_1_11 drive: 50: partition: 9
creapart -1 PhysicalDrive50          225

REM      iordr1_2_9 drive: 51: partition: 9
creapart -1 PhysicalDrive51          225

REM      iordr1_2_10 drive: 52: partition: 9
creapart -1 PhysicalDrive52          225

REM      iordr1_2_11 drive: 53: partition: 9
creapart -1 PhysicalDrive53          225

REM      iordr1_3_9 drive: 54: partition: 9
creapart -1 PhysicalDrive54          225

REM      iordr1_3_10 drive: 56: partition: 9
creapart -1 PhysicalDrive56          225

REM      iordr1_3_11 drive: 57: partition: 9
creapart -1 PhysicalDrive57          225

REM      iordr2_0_0 drive: 1: partition: 10
creapart -1 PhysicalDrive1           365

REM      iordr2_0_1 drive: 2: partition: 10
creapart -1 PhysicalDrive2           365

REM      iordr2_0_2 drive: 3: partition: 10
creapart -1 PhysicalDrive3           365

```

```

REM      iordr2_1_0 drive: 4: partition: 10
creapart -1 PhysicalDrive4           365

REM      iordr2_1_1 drive: 5: partition: 10
creapart -1 PhysicalDrive5           365

REM      iordr2_1_2 drive: 7: partition: 10
creapart -1 PhysicalDrive7           365

REM      iordr2_2_0 drive: 8: partition: 10
creapart -1 PhysicalDrive8           365

REM      iordr2_2_1 drive: 9: partition: 10
creapart -1 PhysicalDrive9           365

REM      iordr2_2_2 drive: 10: partition: 10
creapart -1 PhysicalDrive10          365

REM      iordr2_3_0 drive: 11: partition: 10
creapart -1 PhysicalDrive11          365

REM      iordr2_3_1 drive: 13: partition: 10
creapart -1 PhysicalDrive13          365

REM      iordr2_3_2 drive: 14: partition: 10
creapart -1 PhysicalDrive14          365

REM      iordr2_0_3 drive: 16: partition: 10
creapart -1 PhysicalDrive16          365

REM      iordr2_0_4 drive: 17: partition: 10
creapart -1 PhysicalDrive17          365

REM      iordr2_0_5 drive: 18: partition: 10
creapart -1 PhysicalDrive18          365

REM      iordr2_1_3 drive: 19: partition: 10
creapart -1 PhysicalDrive19          365

REM      iordr2_1_4 drive: 20: partition: 10
creapart -1 PhysicalDrive20          365

REM      iordr2_1_5 drive: 22: partition: 10
creapart -1 PhysicalDrive22          365

REM      iordr2_2_3 drive: 23: partition: 10
creapart -1 PhysicalDrive23          365

REM      iordr2_2_4 drive: 24: partition: 10
creapart -1 PhysicalDrive24          365

REM      iordr2_2_5 drive: 25: partition: 10
creapart -1 PhysicalDrive25          365

REM      iordr2_3_3 drive: 26: partition: 10
creapart -1 PhysicalDrive26          365

REM      iordr2_3_4 drive: 28: partition: 10
creapart -1 PhysicalDrive28          365

REM      iordr2_3_5 drive: 29: partition: 10
creapart -1 PhysicalDrive29          365

```

```

REM      iorder2_0_6 drive: 30: partition: 10
      creapart -1 PhysicalDrive30          365
REM      iorder2_0_7 drive: 31: partition: 10
      creapart -1 PhysicalDrive31          365
REM      iorder2_0_8 drive: 32: partition: 10
      creapart -1 PhysicalDrive32          365
REM      iorder2_1_6 drive: 33: partition: 10
      creapart -1 PhysicalDrive33          365
REM      iorder2_1_7 drive: 34: partition: 10
      creapart -1 PhysicalDrive34          365
REM      iorder2_1_8 drive: 36: partition: 10
      creapart -1 PhysicalDrive36          365
REM      iorder2_2_6 drive: 37: partition: 10
      creapart -1 PhysicalDrive37          365
REM      iorder2_2_7 drive: 38: partition: 10
      creapart -1 PhysicalDrive38          365
REM      iorder2_2_8 drive: 39: partition: 10
      creapart -1 PhysicalDrive39          365
REM      iorder2_3_6 drive: 40: partition: 10
      creapart -1 PhysicalDrive40          365
REM      iorder2_3_7 drive: 42: partition: 10
      creapart -1 PhysicalDrive42          365
REM      iorder2_3_8 drive: 43: partition: 10
      creapart -1 PhysicalDrive43          365
REM      iorder2_0_9 drive: 44: partition: 10
      creapart -1 PhysicalDrive44          365
REM      iorder2_0_10 drive: 45: partition: 10
      creapart -1 PhysicalDrive45          365
REM      iorder2_0_11 drive: 46: partition: 10
      creapart -1 PhysicalDrive46          365
REM      iorder2_1_9 drive: 47: partition: 10
      creapart -1 PhysicalDrive47          365
REM      iorder2_1_10 drive: 48: partition: 10
      creapart -1 PhysicalDrive48          365
REM      iorder2_1_11 drive: 50: partition: 10
      creapart -1 PhysicalDrive50          365
REM      iorder2_2_9 drive: 51: partition: 10
      creapart -1 PhysicalDrive51          365
REM      iorder2_2_10 drive: 52: partition: 10
      creapart -1 PhysicalDrive52          365
REM      iorder2_2_11 drive: 53: partition: 10
      creapart -1 PhysicalDrive53          365

```

```

REM      iorder2_3_9 drive: 54: partition: 10
      creapart -1 PhysicalDrive54          365
REM      iorder2_3_10 drive: 56: partition: 10
      creapart -1 PhysicalDrive56          365
REM      iorder2_3_11 drive: 57: partition: 10
      creapart -1 PhysicalDrive57          365
REM      nord_0_0 drive: 1: partition: 11
      creapart -1 PhysicalDrive1           55
REM      nord_0_1 drive: 2: partition: 11
      creapart -1 PhysicalDrive2           55
REM      nord_0_2 drive: 3: partition: 11
      creapart -1 PhysicalDrive3           55
REM      nord_1_0 drive: 4: partition: 11
      creapart -1 PhysicalDrive4           55
REM      nord_1_1 drive: 5: partition: 11
      creapart -1 PhysicalDrive5           55
REM      nord_1_2 drive: 7: partition: 11
      creapart -1 PhysicalDrive7           55
REM      nord_2_0 drive: 8: partition: 11
      creapart -1 PhysicalDrive8           55
REM      nord_2_1 drive: 9: partition: 11
      creapart -1 PhysicalDrive9           55
REM      nord_2_2 drive: 10: partition: 11
      creapart -1 PhysicalDrive10          55
REM      nord_3_0 drive: 11: partition: 11
      creapart -1 PhysicalDrive11          55
REM      nord_3_1 drive: 13: partition: 11
      creapart -1 PhysicalDrive13          55
REM      nord_3_2 drive: 14: partition: 11
      creapart -1 PhysicalDrive14          55
REM      nord_0_3 drive: 16: partition: 11
      creapart -1 PhysicalDrive16          55
REM      nord_0_4 drive: 17: partition: 11
      creapart -1 PhysicalDrive17          55
REM      nord_0_5 drive: 18: partition: 11
      creapart -1 PhysicalDrive18          55
REM      nord_1_3 drive: 19: partition: 11
      creapart -1 PhysicalDrive19          55
REM      nord_1_4 drive: 20: partition: 11
      creapart -1 PhysicalDrive20          55
REM      nord_1_5 drive: 22: partition: 11
      creapart -1 PhysicalDrive22          55

```

```

REM      nord_2_3 drive: 23: partition: 11
creapart -1 PhysicalDrive23          55

REM      nord_2_4 drive: 24: partition: 11
creapart -1 PhysicalDrive24          55

REM      nord_2_5 drive: 25: partition: 11
creapart -1 PhysicalDrive25          55

REM      nord_3_3 drive: 26: partition: 11
creapart -1 PhysicalDrive26          55

REM      nord_3_4 drive: 28: partition: 11
creapart -1 PhysicalDrive28          55

REM      nord_3_5 drive: 29: partition: 11
creapart -1 PhysicalDrive29          55

REM      nord_0_6 drive: 30: partition: 11
creapart -1 PhysicalDrive30          55

REM      nord_0_7 drive: 31: partition: 11
creapart -1 PhysicalDrive31          55

REM      nord_0_8 drive: 32: partition: 11
creapart -1 PhysicalDrive32          55

REM      nord_1_6 drive: 33: partition: 11
creapart -1 PhysicalDrive33          55

REM      nord_1_7 drive: 34: partition: 11
creapart -1 PhysicalDrive34          55

REM      nord_1_8 drive: 36: partition: 11
creapart -1 PhysicalDrive36          55

REM      nord_2_6 drive: 37: partition: 11
creapart -1 PhysicalDrive37          55

REM      nord_2_7 drive: 38: partition: 11
creapart -1 PhysicalDrive38          55

REM      nord_2_8 drive: 39: partition: 11
creapart -1 PhysicalDrive39          55

REM      nord_3_6 drive: 40: partition: 11
creapart -1 PhysicalDrive40          55

REM      nord_3_7 drive: 42: partition: 11
creapart -1 PhysicalDrive42          55

REM      nord_3_8 drive: 43: partition: 11
creapart -1 PhysicalDrive43          55

REM      nord_0_9 drive: 44: partition: 11
creapart -1 PhysicalDrive44          55

REM      nord_0_10 drive: 45: partition: 11
creapart -1 PhysicalDrive45          55

REM      nord_0_11 drive: 46: partition: 11
creapart -1 PhysicalDrive46          55

```

```

REM      nord_1_9 drive: 47: partition: 11
creapart -1 PhysicalDrive47          55

REM      nord_1_10 drive: 48: partition: 11
creapart -1 PhysicalDrive48          55

REM      nord_1_11 drive: 50: partition: 11
creapart -1 PhysicalDrive50          55

REM      nord_2_9 drive: 51: partition: 11
creapart -1 PhysicalDrive51          55

REM      nord_2_10 drive: 52: partition: 11
creapart -1 PhysicalDrive52          55

REM      nord_2_11 drive: 53: partition: 11
creapart -1 PhysicalDrive53          55

REM      nord_3_9 drive: 54: partition: 11
creapart -1 PhysicalDrive54          55

REM      nord_3_10 drive: 56: partition: 11
creapart -1 PhysicalDrive56          55

REM      nord_3_11 drive: 57: partition: 11
creapart -1 PhysicalDrive57          55

REM      hist_0_0 drive:  1: partition: 12
creapart -1 PhysicalDrive1           535

REM      hist_0_1 drive:  2: partition: 12
creapart -1 PhysicalDrive2           535

REM      hist_0_2 drive:  3: partition: 12
creapart -1 PhysicalDrive3           535

REM      hist_1_0 drive:  4: partition: 12
creapart -1 PhysicalDrive4           535

REM      hist_1_1 drive:  5: partition: 12
creapart -1 PhysicalDrive5           535

REM      hist_1_2 drive:  7: partition: 12
creapart -1 PhysicalDrive7           535

REM      hist_2_0 drive:  8: partition: 12
creapart -1 PhysicalDrive8           535

REM      hist_2_1 drive:  9: partition: 12
creapart -1 PhysicalDrive9           535

REM      hist_2_2 drive: 10: partition: 12
creapart -1 PhysicalDrive10          535

REM      hist_3_0 drive: 11: partition: 12
creapart -1 PhysicalDrive11          535

REM      hist_3_1 drive: 13: partition: 12
creapart -1 PhysicalDrive13          535

REM      hist_3_2 drive: 14: partition: 12
creapart -1 PhysicalDrive14          535

```

```

REM      hist_0_3 drive: 16: partition: 12
creapart -1 PhysicalDrive16          535

REM      hist_0_4 drive: 17: partition: 12
creapart -1 PhysicalDrive17          535

REM      hist_0_5 drive: 18: partition: 12
creapart -1 PhysicalDrive18          535

REM      hist_1_3 drive: 19: partition: 12
creapart -1 PhysicalDrive19          535

REM      hist_1_4 drive: 20: partition: 12
creapart -1 PhysicalDrive20          535

REM      hist_1_5 drive: 22: partition: 12
creapart -1 PhysicalDrive22          535

REM      hist_2_3 drive: 23: partition: 12
creapart -1 PhysicalDrive23          535

REM      hist_2_4 drive: 24: partition: 12
creapart -1 PhysicalDrive24          535

REM      hist_2_5 drive: 25: partition: 12
creapart -1 PhysicalDrive25          535

REM      hist_3_3 drive: 26: partition: 12
creapart -1 PhysicalDrive26          535

REM      hist_3_4 drive: 28: partition: 12
creapart -1 PhysicalDrive28          535

REM      hist_3_5 drive: 29: partition: 12
creapart -1 PhysicalDrive29          535

REM      hist_0_6 drive: 30: partition: 12
creapart -1 PhysicalDrive30          535

REM      hist_0_7 drive: 31: partition: 12
creapart -1 PhysicalDrive31          535

REM      hist_0_8 drive: 32: partition: 12
creapart -1 PhysicalDrive32          535

REM      hist_1_6 drive: 33: partition: 12
creapart -1 PhysicalDrive33          535

REM      hist_1_7 drive: 34: partition: 12
creapart -1 PhysicalDrive34          535

REM      hist_1_8 drive: 36: partition: 12
creapart -1 PhysicalDrive36          535

REM      hist_2_6 drive: 37: partition: 12
creapart -1 PhysicalDrive37          535

REM      hist_2_7 drive: 38: partition: 12
creapart -1 PhysicalDrive38          535

REM      hist_2_8 drive: 39: partition: 12
creapart -1 PhysicalDrive39          535

```

```

REM      hist_3_6 drive: 40: partition: 12
creapart -1 PhysicalDrive40          535

REM      hist_3_7 drive: 42: partition: 12
creapart -1 PhysicalDrive42          535

REM      hist_3_8 drive: 43: partition: 12
creapart -1 PhysicalDrive43          535

REM      hist_0_9 drive: 44: partition: 12
creapart -1 PhysicalDrive44          535

REM      hist_0_10 drive: 45: partition: 12
creapart -1 PhysicalDrive45          535

REM      hist_0_11 drive: 46: partition: 12
creapart -1 PhysicalDrive46          535

REM      hist_1_9 drive: 47: partition: 12
creapart -1 PhysicalDrive47          535

REM      hist_1_10 drive: 48: partition: 12
creapart -1 PhysicalDrive48          535

REM      hist_1_11 drive: 50: partition: 12
creapart -1 PhysicalDrive50          535

REM      hist_2_9 drive: 51: partition: 12
creapart -1 PhysicalDrive51          535

REM      hist_2_10 drive: 52: partition: 12
creapart -1 PhysicalDrive52          535

REM      hist_2_11 drive: 53: partition: 12
creapart -1 PhysicalDrive53          535

REM      hist_3_9 drive: 54: partition: 12
creapart -1 PhysicalDrive54          535

REM      hist_3_10 drive: 56: partition: 12
creapart -1 PhysicalDrive56          535

REM      hist_3_11 drive: 57: partition: 12
creapart -1 PhysicalDrive57          535

REM      dist_0_0 drive: 1: partition: 13
creapart -1 PhysicalDrive1           15

REM      dist_0_1 drive: 2: partition: 13
creapart -1 PhysicalDrive2           15

REM      dist_0_2 drive: 3: partition: 13
creapart -1 PhysicalDrive3           15

REM      dist_1_0 drive: 4: partition: 13
creapart -1 PhysicalDrive4           15

REM      dist_1_1 drive: 5: partition: 13
creapart -1 PhysicalDrive5           15

REM      dist_1_2 drive: 7: partition: 13
creapart -1 PhysicalDrive7           15

```

```

REM      dist_2_0 drive: 8: partition: 13
creapart -1 PhysicalDrive8          15

REM      dist_2_1 drive: 9: partition: 13
creapart -1 PhysicalDrive9          15

REM      dist_2_2 drive: 10: partition: 13
creapart -1 PhysicalDrive10         15

REM      dist_3_0 drive: 11: partition: 13
creapart -1 PhysicalDrive11         15

REM      dist_3_1 drive: 13: partition: 13
creapart -1 PhysicalDrive13         15

REM      dist_3_2 drive: 14: partition: 13
creapart -1 PhysicalDrive14         15

REM      dist_0_3 drive: 16: partition: 13
creapart -1 PhysicalDrive16         15

REM      dist_0_4 drive: 17: partition: 13
creapart -1 PhysicalDrive17         15

REM      dist_0_5 drive: 18: partition: 13
creapart -1 PhysicalDrive18         15

REM      dist_1_3 drive: 19: partition: 13
creapart -1 PhysicalDrive19         15

REM      dist_1_4 drive: 20: partition: 13
creapart -1 PhysicalDrive20         15

REM      dist_1_5 drive: 22: partition: 13
creapart -1 PhysicalDrive22         15

REM      dist_2_3 drive: 23: partition: 13
creapart -1 PhysicalDrive23         15

REM      dist_2_4 drive: 24: partition: 13
creapart -1 PhysicalDrive24         15

REM      dist_2_5 drive: 25: partition: 13
creapart -1 PhysicalDrive25         15

REM      dist_3_3 drive: 26: partition: 13
creapart -1 PhysicalDrive26         15

REM      dist_3_4 drive: 28: partition: 13
creapart -1 PhysicalDrive28         15

REM      dist_3_5 drive: 29: partition: 13
creapart -1 PhysicalDrive29         15

REM      dist_0_6 drive: 30: partition: 13
creapart -1 PhysicalDrive30         15

REM      dist_0_7 drive: 31: partition: 13
creapart -1 PhysicalDrive31         15

REM      dist_0_8 drive: 32: partition: 13
creapart -1 PhysicalDrive32         15

```

```

REM      dist_1_6 drive: 33: partition: 13
creapart -1 PhysicalDrive33         15

REM      dist_1_7 drive: 34: partition: 13
creapart -1 PhysicalDrive34         15

REM      dist_1_8 drive: 36: partition: 13
creapart -1 PhysicalDrive36         15

REM      dist_2_6 drive: 37: partition: 13
creapart -1 PhysicalDrive37         15

REM      dist_2_7 drive: 38: partition: 13
creapart -1 PhysicalDrive38         15

REM      dist_2_8 drive: 39: partition: 13
creapart -1 PhysicalDrive39         15

REM      dist_3_6 drive: 40: partition: 13
creapart -1 PhysicalDrive40         15

REM      dist_3_7 drive: 42: partition: 13
creapart -1 PhysicalDrive42         15

REM      dist_3_8 drive: 43: partition: 13
creapart -1 PhysicalDrive43         15

REM      dist_0_9 drive: 44: partition: 13
creapart -1 PhysicalDrive44         15

REM      dist_0_10 drive: 45: partition: 13
creapart -1 PhysicalDrive45         15

REM      dist_0_11 drive: 46: partition: 13
creapart -1 PhysicalDrive46         15

REM      dist_1_9 drive: 47: partition: 13
creapart -1 PhysicalDrive47         15

REM      dist_1_10 drive: 48: partition: 13
creapart -1 PhysicalDrive48         15

REM      dist_1_11 drive: 50: partition: 13
creapart -1 PhysicalDrive50         15

REM      dist_2_9 drive: 51: partition: 13
creapart -1 PhysicalDrive51         15

REM      dist_2_10 drive: 52: partition: 13
creapart -1 PhysicalDrive52         15

REM      dist_2_11 drive: 53: partition: 13
creapart -1 PhysicalDrive53         15

REM      dist_3_9 drive: 54: partition: 13
creapart -1 PhysicalDrive54         15

REM      dist_3_10 drive: 56: partition: 13
creapart -1 PhysicalDrive56         15

REM      dist_3_11 drive: 57: partition: 13
creapart -1 PhysicalDrive57         15

```

```

REM      ware_0_0 drive: 1: partition: 14
creapart -1 PhysicalDrive1      15

REM      ware_0_1 drive: 2: partition: 14
creapart -1 PhysicalDrive2      15

REM      ware_0_2 drive: 3: partition: 14
creapart -1 PhysicalDrive3      15

REM      ware_1_0 drive: 4: partition: 14
creapart -1 PhysicalDrive4      15

REM      ware_1_1 drive: 5: partition: 14
creapart -1 PhysicalDrive5      15

REM      ware_1_2 drive: 7: partition: 14
creapart -1 PhysicalDrive7      15

REM      ware_2_0 drive: 8: partition: 14
creapart -1 PhysicalDrive8      15

REM      ware_2_1 drive: 9: partition: 14
creapart -1 PhysicalDrive9      15

REM      ware_2_2 drive: 10: partition: 14
creapart -1 PhysicalDrive10     15

REM      ware_3_0 drive: 11: partition: 14
creapart -1 PhysicalDrive11     15

REM      ware_3_1 drive: 13: partition: 14
creapart -1 PhysicalDrive13     15

REM      ware_3_2 drive: 14: partition: 14
creapart -1 PhysicalDrive14     15

REM      ware_0_3 drive: 16: partition: 14
creapart -1 PhysicalDrive16     15

REM      ware_0_4 drive: 17: partition: 14
creapart -1 PhysicalDrive17     15

REM      ware_0_5 drive: 18: partition: 14
creapart -1 PhysicalDrive18     15

REM      ware_1_3 drive: 19: partition: 14
creapart -1 PhysicalDrive19     15

REM      ware_1_4 drive: 20: partition: 14
creapart -1 PhysicalDrive20     15

REM      ware_1_5 drive: 22: partition: 14
creapart -1 PhysicalDrive22     15

REM      ware_2_3 drive: 23: partition: 14
creapart -1 PhysicalDrive23     15

REM      ware_2_4 drive: 24: partition: 14
creapart -1 PhysicalDrive24     15

REM      ware_2_5 drive: 25: partition: 14
creapart -1 PhysicalDrive25     15

```

```

REM      ware_3_3 drive: 26: partition: 14
creapart -1 PhysicalDrive26     15

REM      ware_3_4 drive: 28: partition: 14
creapart -1 PhysicalDrive28     15

REM      ware_3_5 drive: 29: partition: 14
creapart -1 PhysicalDrive29     15

REM      ware_0_6 drive: 30: partition: 14
creapart -1 PhysicalDrive30     15

REM      ware_0_7 drive: 31: partition: 14
creapart -1 PhysicalDrive31     15

REM      ware_0_8 drive: 32: partition: 14
creapart -1 PhysicalDrive32     15

REM      ware_1_6 drive: 33: partition: 14
creapart -1 PhysicalDrive33     15

REM      ware_1_7 drive: 34: partition: 14
creapart -1 PhysicalDrive34     15

REM      ware_1_8 drive: 36: partition: 14
creapart -1 PhysicalDrive36     15

REM      ware_2_6 drive: 37: partition: 14
creapart -1 PhysicalDrive37     15

REM      ware_2_7 drive: 38: partition: 14
creapart -1 PhysicalDrive38     15

REM      ware_2_8 drive: 39: partition: 14
creapart -1 PhysicalDrive39     15

REM      ware_3_6 drive: 40: partition: 14
creapart -1 PhysicalDrive40     15

REM      ware_3_7 drive: 42: partition: 14
creapart -1 PhysicalDrive42     15

REM      ware_3_8 drive: 43: partition: 14
creapart -1 PhysicalDrive43     15

REM      ware_0_9 drive: 44: partition: 14
creapart -1 PhysicalDrive44     15

REM      ware_0_10 drive: 45: partition: 14
creapart -1 PhysicalDrive45     15

REM      ware_0_11 drive: 46: partition: 14
creapart -1 PhysicalDrive46     15

REM      ware_1_9 drive: 47: partition: 14
creapart -1 PhysicalDrive47     15

REM      ware_1_10 drive: 48: partition: 14
creapart -1 PhysicalDrive48     15

REM      ware_1_11 drive: 50: partition: 14
creapart -1 PhysicalDrive50     15

```

```

REM      ware_2_9 drive: 51: partition: 14
creapart -1 PhysicalDrive51          15

REM      ware_2_10 drive: 52: partition: 14
creapart -1 PhysicalDrive52          15

REM      ware_2_11 drive: 53: partition: 14
creapart -1 PhysicalDrive53          15

REM      ware_3_9 drive: 54: partition: 14
creapart -1 PhysicalDrive54          15

REM      ware_3_10 drive: 56: partition: 14
creapart -1 PhysicalDrive56          15

REM      ware_3_11 drive: 57: partition: 14
creapart -1 PhysicalDrive57          15

REM      roll_0_0 drive: 1: partition: 15
creapart -1 PhysicalDrive1           180

REM      roll_0_1 drive: 2: partition: 15
creapart -1 PhysicalDrive2           180

REM      roll_0_2 drive: 3: partition: 15
creapart -1 PhysicalDrive3           180

REM      roll_1_0 drive: 4: partition: 15
creapart -1 PhysicalDrive4           180

REM      roll_1_1 drive: 5: partition: 15
creapart -1 PhysicalDrive5           180

REM      roll_1_2 drive: 7: partition: 15
creapart -1 PhysicalDrive7           180

REM      roll_2_0 drive: 8: partition: 15
creapart -1 PhysicalDrive8           180

REM      roll_2_1 drive: 9: partition: 15
creapart -1 PhysicalDrive9           180

REM      roll_2_2 drive: 10: partition: 15
creapart -1 PhysicalDrive10          180

REM      roll_3_0 drive: 11: partition: 15
creapart -1 PhysicalDrive11          180

REM      roll_3_1 drive: 13: partition: 15
creapart -1 PhysicalDrive13          180

REM      roll_3_2 drive: 14: partition: 15
creapart -1 PhysicalDrive14          180

REM      roll_0_3 drive: 16: partition: 15
creapart -1 PhysicalDrive16          180

REM      roll_0_4 drive: 17: partition: 15
creapart -1 PhysicalDrive17          180

REM      roll_0_5 drive: 18: partition: 15
creapart -1 PhysicalDrive18          180

```

```

REM      roll_1_3 drive: 19: partition: 15
creapart -1 PhysicalDrive19          180

REM      roll_1_4 drive: 20: partition: 15
creapart -1 PhysicalDrive20          180

REM      roll_1_5 drive: 22: partition: 15
creapart -1 PhysicalDrive22          180

REM      roll_2_3 drive: 23: partition: 15
creapart -1 PhysicalDrive23          180

REM      roll_2_4 drive: 24: partition: 15
creapart -1 PhysicalDrive24          180

REM      roll_2_5 drive: 25: partition: 15
creapart -1 PhysicalDrive25          180

REM      roll_3_3 drive: 26: partition: 15
creapart -1 PhysicalDrive26          180

REM      roll_3_4 drive: 28: partition: 15
creapart -1 PhysicalDrive28          180

REM      roll_3_5 drive: 29: partition: 15
creapart -1 PhysicalDrive29          180

REM      roll_0_6 drive: 30: partition: 15
creapart -1 PhysicalDrive30          180

REM      roll_0_7 drive: 31: partition: 15
creapart -1 PhysicalDrive31          180

REM      roll_0_8 drive: 32: partition: 15
creapart -1 PhysicalDrive32          180

REM      roll_1_6 drive: 33: partition: 15
creapart -1 PhysicalDrive33          180

REM      roll_1_7 drive: 34: partition: 15
creapart -1 PhysicalDrive34          180

REM      roll_1_8 drive: 36: partition: 15
creapart -1 PhysicalDrive36          180

REM      roll_2_6 drive: 37: partition: 15
creapart -1 PhysicalDrive37          180

REM      roll_2_7 drive: 38: partition: 15
creapart -1 PhysicalDrive38          180

REM      roll_2_8 drive: 39: partition: 15
creapart -1 PhysicalDrive39          180

REM      roll_3_6 drive: 40: partition: 15
creapart -1 PhysicalDrive40          180

REM      roll_3_7 drive: 42: partition: 15
creapart -1 PhysicalDrive42          180

REM      roll_3_8 drive: 43: partition: 15
creapart -1 PhysicalDrive43          180

```

```

REM      roll_0_9 drive: 44: partition: 15
creapart -l PhysicalDrive44          180

REM      roll_0_10 drive: 45: partition: 15
creapart -l PhysicalDrive45          180

REM      roll_0_11 drive: 46: partition: 15
creapart -l PhysicalDrive46          180

REM      roll_1_9 drive: 47: partition: 15
creapart -l PhysicalDrive47          180

REM      roll_1_10 drive: 48: partition: 15
creapart -l PhysicalDrive48          180

REM      roll_1_11 drive: 50: partition: 15
creapart -l PhysicalDrive50          180

REM      roll_2_9 drive: 51: partition: 15
creapart -l PhysicalDrive51          180

REM      roll_2_10 drive: 52: partition: 15
creapart -l PhysicalDrive52          180

REM      roll_2_11 drive: 53: partition: 15
creapart -l PhysicalDrive53          180

REM      roll_3_9 drive: 54: partition: 15
creapart -l PhysicalDrive54          180

REM      roll_3_10 drive: 56: partition: 15
creapart -l PhysicalDrive56          180

REM      roll_3_11 drive: 57: partition: 15
creapart -l PhysicalDrive57          180

REM      temp_0_0 drive: 1: partition: 16
creapart -l PhysicalDrive1           505

REM      temp_0_1 drive: 2: partition: 16
creapart -l PhysicalDrive2           505

REM      temp_0_2 drive: 3: partition: 16
creapart -l PhysicalDrive3           505

REM      temp_0_3 drive: 4: partition: 16
creapart -l PhysicalDrive4           505

REM      temp_0_4 drive: 5: partition: 16
creapart -l PhysicalDrive5           505

REM      temp_0_5 drive: 7: partition: 16
creapart -l PhysicalDrive7           505

REM      temp_0_6 drive: 8: partition: 16
creapart -l PhysicalDrive8           505

REM      temp_0_7 drive: 9: partition: 16
creapart -l PhysicalDrive9           505

REM      temp_0_8 drive: 10: partition: 16
creapart -l PhysicalDrive10          505

```

```

REM      temp_0_9 drive: 11: partition: 16
creapart -l PhysicalDrive11          505

REM      temp_0_10 drive: 13: partition: 16
creapart -l PhysicalDrive13          505

REM      temp_0_11 drive: 14: partition: 16
creapart -l PhysicalDrive14          505

REM      temp_0_12 drive: 16: partition: 16
creapart -l PhysicalDrive16          505

REM      temp_0_13 drive: 17: partition: 16
creapart -l PhysicalDrive17          505

REM      temp_0_14 drive: 18: partition: 16
creapart -l PhysicalDrive18          505

REM      temp_0_15 drive: 19: partition: 16
creapart -l PhysicalDrive19          505

REM      temp_0_16 drive: 20: partition: 16
creapart -l PhysicalDrive20          505

REM      temp_0_17 drive: 22: partition: 16
creapart -l PhysicalDrive22          505

REM      temp_0_18 drive: 23: partition: 16
creapart -l PhysicalDrive23          505

REM      temp_0_19 drive: 24: partition: 16
creapart -l PhysicalDrive24          505

REM      temp_0_20 drive: 25: partition: 16
creapart -l PhysicalDrive25          505

REM      temp_0_21 drive: 26: partition: 16
creapart -l PhysicalDrive26          505

REM      temp_0_22 drive: 28: partition: 16
creapart -l PhysicalDrive28          505

REM      temp_0_23 drive: 29: partition: 16
creapart -l PhysicalDrive29          505

REM      temp_0_24 drive: 30: partition: 16
creapart -l PhysicalDrive30          505

REM      temp_0_25 drive: 31: partition: 16
creapart -l PhysicalDrive31          505

REM      temp_0_26 drive: 32: partition: 16
creapart -l PhysicalDrive32          505

REM      temp_0_27 drive: 33: partition: 16
creapart -l PhysicalDrive33          505

REM      temp_0_28 drive: 34: partition: 16
creapart -l PhysicalDrive34          505

REM      temp_0_29 drive: 36: partition: 16
creapart -l PhysicalDrive36          505

```

```

REM      temp_0_30 drive: 37: partition: 16
creapart -l PhysicalDrive37          505

REM      temp_0_31 drive: 38: partition: 16
creapart -l PhysicalDrive38          505

REM      temp_0_32 drive: 39: partition: 16
creapart -l PhysicalDrive39          505

REM      temp_0_33 drive: 40: partition: 16
creapart -l PhysicalDrive40          505

REM      temp_0_34 drive: 42: partition: 16
creapart -l PhysicalDrive42          505

REM      temp_0_35 drive: 43: partition: 16
creapart -l PhysicalDrive43          505

REM      temp_0_36 drive: 44: partition: 16
creapart -l PhysicalDrive44          505

REM      temp_0_37 drive: 45: partition: 16
creapart -l PhysicalDrive45          505

REM      temp_0_38 drive: 46: partition: 16
creapart -l PhysicalDrive46          505

REM      temp_0_39 drive: 47: partition: 16
creapart -l PhysicalDrive47          505

REM      temp_0_40 drive: 48: partition: 16
creapart -l PhysicalDrive48          505

REM      temp_0_41 drive: 50: partition: 16
creapart -l PhysicalDrive50          505

REM      temp_0_42 drive: 51: partition: 16
creapart -l PhysicalDrive51          505

REM      temp_0_43 drive: 52: partition: 16
creapart -l PhysicalDrive52          505

REM      temp_0_44 drive: 53: partition: 16
creapart -l PhysicalDrive53          505

REM      temp_0_45 drive: 54: partition: 16
creapart -l PhysicalDrive54          505

REM      temp_0_46 drive: 56: partition: 16
creapart -l PhysicalDrive56          505

REM      temp_0_47 drive: 57: partition: 16
creapart -l PhysicalDrive57          505

REM      iware_0_0 drive:  1: partition: 17
creapart -l PhysicalDrive1           25

REM      idist_0_0 drive:  2: partition: 17
creapart -l PhysicalDrive2           35

REM      iitem_0_0 drive:  3: partition: 17
creapart -l PhysicalDrive3           15

```

```

REM      inord_0_0 drive:  4: partition: 17
creapart -l PhysicalDrive4           713

REM      icust1_0_0 drive:  5: partition: 17
creapart -l PhysicalDrive5          1005

REM      istok_0_0 drive:  7: partition: 17
creapart -l PhysicalDrive7          3065

REM      iordl_0_0 drive:  8: partition: 17
creapart -l PhysicalDrive8          8030

REM      iordl_0_1 drive:  9: partition: 17
creapart -l PhysicalDrive9          8030

REM      iordl_0_2 drive: 10: partition: 17
creapart -l PhysicalDrive10         8030

REM      control_001 drive: 11: partition: 17
creapart -l PhysicalDrive11         510

REM      iware_1_0 drive: 16: partition: 17
creapart -l PhysicalDrive16          25

REM      idist_1_0 drive: 17: partition: 17
creapart -l PhysicalDrive17          35

REM      iitem_0_1 drive: 18: partition: 17
creapart -l PhysicalDrive18          15

REM      inord_0_1 drive: 19: partition: 17
creapart -l PhysicalDrive19          713

REM      icust1_0_1 drive: 20: partition: 17
creapart -l PhysicalDrive20         1005

REM      istok_0_1 drive: 22: partition: 17
creapart -l PhysicalDrive22         3065

REM      iordl_0_3 drive: 23: partition: 17
creapart -l PhysicalDrive23         8030

REM      iordl_0_4 drive: 24: partition: 17
creapart -l PhysicalDrive24         8030

REM      iordl_0_5 drive: 25: partition: 17
creapart -l PhysicalDrive25         8030

REM      system_001 drive: 26: partition: 17
creapart -l PhysicalDrive26         510

REM      iware_2_0 drive: 30: partition: 17
creapart -l PhysicalDrive30          25

REM      idist_2_0 drive: 31: partition: 17
creapart -l PhysicalDrive31          35

REM      iitem_0_2 drive: 32: partition: 17
creapart -l PhysicalDrive32          15

REM      inord_0_2 drive: 33: partition: 17
creapart -l PhysicalDrive33         713

```

```

REM      icust1_0_2 drive: 34: partition: 17
      creapart -1 PhysicalDrive34      1005
REM      istok_0_2 drive: 36: partition: 17
      creapart -1 PhysicalDrive36      3065
REM      iordl_0_6 drive: 37: partition: 17
      creapart -1 PhysicalDrive37      8030
REM      iordl_0_7 drive: 38: partition: 17
      creapart -1 PhysicalDrive38      8030
REM      iordl_0_8 drive: 39: partition: 17
      creapart -1 PhysicalDrive39      8030
REM      control_002 drive: 40: partition: 17
      creapart -1 PhysicalDrive40      510
REM      iware_3_0 drive: 44: partition: 17
      creapart -1 PhysicalDrive44      25
REM      idist_3_0 drive: 45: partition: 17
      creapart -1 PhysicalDrive45      35
REM      iitem_0_3 drive: 46: partition: 17
      creapart -1 PhysicalDrive46      15
REM      inord_0_3 drive: 47: partition: 17
      creapart -1 PhysicalDrive47      713
REM      icust1_0_3 drive: 48: partition: 17
      creapart -1 PhysicalDrive48      1005
REM      istok_0_3 drive: 50: partition: 17
      creapart -1 PhysicalDrive50      3065
REM      iordl_0_9 drive: 51: partition: 17
      creapart -1 PhysicalDrive51      8030
REM      iordl_0_10 drive: 52: partition: 17
      creapart -1 PhysicalDrive52      8030
REM      iordl_0_11 drive: 53: partition: 17
      creapart -1 PhysicalDrive53      8030

```

```

      stok_0_5      \device\harddisk7\partition1
      stok_0_6      \device\harddisk8\partition1
      stok_0_7      \device\harddisk9\partition1
      stok_0_8      \device\harddisk10\partition1
      stok_0_9      \device\harddisk11\partition1
      stok_0_10     \device\harddisk13\partition1
      stok_0_11     \device\harddisk14\partition1
      stok_0_12     \device\harddisk16\partition1
      stok_0_13     \device\harddisk17\partition1
      stok_0_14     \device\harddisk18\partition1
      stok_0_15     \device\harddisk19\partition1
      stok_0_16     \device\harddisk20\partition1
      stok_0_17     \device\harddisk22\partition1
      stok_0_18     \device\harddisk23\partition1
      stok_0_19     \device\harddisk24\partition1
      stok_0_20     \device\harddisk25\partition1
      stok_0_21     \device\harddisk26\partition1
      stok_0_22     \device\harddisk28\partition1
      stok_0_23     \device\harddisk29\partition1
      stok_0_24     \device\harddisk30\partition1
      stok_0_25     \device\harddisk31\partition1
      stok_0_26     \device\harddisk32\partition1
      stok_0_27     \device\harddisk33\partition1
      stok_0_28     \device\harddisk34\partition1
      stok_0_29     \device\harddisk36\partition1
      stok_0_30     \device\harddisk37\partition1
      stok_0_31     \device\harddisk38\partition1
      stok_0_32     \device\harddisk39\partition1
      stok_0_33     \device\harddisk40\partition1
      stok_0_34     \device\harddisk42\partition1
      stok_0_35     \device\harddisk43\partition1
      stok_0_36     \device\harddisk44\partition1
      stok_0_37     \device\harddisk45\partition1
      stok_0_38     \device\harddisk46\partition1
      stok_0_39     \device\harddisk47\partition1
      stok_0_40     \device\harddisk48\partition1
      stok_0_41     \device\harddisk50\partition1
      stok_0_42     \device\harddisk51\partition1
      stok_0_43     \device\harddisk52\partition1
      stok_0_44     \device\harddisk53\partition1
      stok_0_45     \device\harddisk54\partition1
      stok_0_46     \device\harddisk56\partition1
      stok_0_47     \device\harddisk57\partition1
      stok_0_48     \device\harddisk1\partition2
      stok_0_49     \device\harddisk2\partition2
      stok_0_50     \device\harddisk3\partition2
      stok_0_51     \device\harddisk4\partition2
      stok_0_52     \device\harddisk5\partition2
      stok_0_53     \device\harddisk7\partition2
      stok_0_54     \device\harddisk8\partition2
      stok_0_55     \device\harddisk9\partition2
      stok_0_56     \device\harddisk10\partition2
      stok_0_57     \device\harddisk11\partition2
      stok_0_58     \device\harddisk13\partition2
      stok_0_59     \device\harddisk14\partition2
      stok_0_60     \device\harddisk16\partition2
      stok_0_61     \device\harddisk17\partition2
      stok_0_62     \device\harddisk18\partition2
      stok_0_63     \device\harddisk19\partition2
      stok_0_64     \device\harddisk20\partition2
      stok_0_65     \device\harddisk22\partition2
      stok_0_66     \device\harddisk23\partition2
      stok_0_67     \device\harddisk24\partition2

```

links.prn

```

log_111      \device\harddisk12\partition1
log_121      \device\harddisk12\partition2
log_231      \device\harddisk27\partition1
log_241      \device\harddisk27\partition2
log_351      \device\harddisk41\partition1
log_361      \device\harddisk41\partition2
log_471      \device\harddisk55\partition1
log_481      \device\harddisk55\partition2
stok_0_0     \device\harddisk1\partition1
stok_0_1     \device\harddisk2\partition1
stok_0_2     \device\harddisk3\partition1
stok_0_3     \device\harddisk4\partition1
stok_0_4     \device\harddisk5\partition1

```

```

stok_0_68 \device\harddisk25\partition2
stok_0_69 \device\harddisk26\partition2
stok_0_70 \device\harddisk28\partition2
stok_0_71 \device\harddisk29\partition2
stok_0_72 \device\harddisk30\partition2
stok_0_73 \device\harddisk31\partition2
stok_0_74 \device\harddisk32\partition2
stok_0_75 \device\harddisk33\partition2
stok_0_76 \device\harddisk34\partition2
stok_0_77 \device\harddisk36\partition2
stok_0_78 \device\harddisk37\partition2
stok_0_79 \device\harddisk38\partition2
stok_0_80 \device\harddisk39\partition2
stok_0_81 \device\harddisk40\partition2
stok_0_82 \device\harddisk42\partition2
stok_0_83 \device\harddisk43\partition2
stok_0_84 \device\harddisk44\partition2
stok_0_85 \device\harddisk45\partition2
stok_0_86 \device\harddisk46\partition2
stok_0_87 \device\harddisk47\partition2
stok_0_88 \device\harddisk48\partition2
stok_0_89 \device\harddisk50\partition2
stok_0_90 \device\harddisk51\partition2
stok_0_91 \device\harddisk52\partition2
stok_0_92 \device\harddisk53\partition2
stok_0_93 \device\harddisk54\partition2
stok_0_94 \device\harddisk56\partition2
stok_0_95 \device\harddisk57\partition2
cust_0_0 \device\harddisk1\partition3
cust_0_1 \device\harddisk2\partition3
cust_0_2 \device\harddisk3\partition3
cust_0_3 \device\harddisk4\partition3
cust_0_4 \device\harddisk5\partition3
cust_0_5 \device\harddisk7\partition3
cust_0_6 \device\harddisk8\partition3
cust_0_7 \device\harddisk9\partition3
cust_0_8 \device\harddisk10\partition3
cust_0_9 \device\harddisk11\partition3
cust_0_10 \device\harddisk13\partition3
cust_0_11 \device\harddisk14\partition3
cust_0_12 \device\harddisk16\partition3
cust_0_13 \device\harddisk17\partition3
cust_0_14 \device\harddisk18\partition3
cust_0_15 \device\harddisk19\partition3
cust_0_16 \device\harddisk20\partition3
cust_0_17 \device\harddisk22\partition3
cust_0_18 \device\harddisk23\partition3
cust_0_19 \device\harddisk24\partition3
cust_0_20 \device\harddisk25\partition3
cust_0_21 \device\harddisk26\partition3
cust_0_22 \device\harddisk28\partition3
cust_0_23 \device\harddisk29\partition3
cust_0_24 \device\harddisk30\partition3
cust_0_25 \device\harddisk31\partition3
cust_0_26 \device\harddisk32\partition3
cust_0_27 \device\harddisk33\partition3
cust_0_28 \device\harddisk34\partition3
cust_0_29 \device\harddisk36\partition3
cust_0_30 \device\harddisk37\partition3
cust_0_31 \device\harddisk38\partition3
cust_0_32 \device\harddisk39\partition3
cust_0_33 \device\harddisk40\partition3
cust_0_34 \device\harddisk42\partition3

```

```

cust_0_35 \device\harddisk43\partition3
cust_0_36 \device\harddisk44\partition3
cust_0_37 \device\harddisk45\partition3
cust_0_38 \device\harddisk46\partition3
cust_0_39 \device\harddisk47\partition3
cust_0_40 \device\harddisk48\partition3
cust_0_41 \device\harddisk50\partition3
cust_0_42 \device\harddisk51\partition3
cust_0_43 \device\harddisk52\partition3
cust_0_44 \device\harddisk53\partition3
cust_0_45 \device\harddisk54\partition3
cust_0_46 \device\harddisk56\partition3
cust_0_47 \device\harddisk57\partition3
cust_0_48 \device\harddisk1\partition4
cust_0_49 \device\harddisk2\partition4
cust_0_50 \device\harddisk3\partition4
cust_0_51 \device\harddisk4\partition4
cust_0_52 \device\harddisk5\partition4
cust_0_53 \device\harddisk7\partition4
cust_0_54 \device\harddisk8\partition4
cust_0_55 \device\harddisk9\partition4
cust_0_56 \device\harddisk10\partition4
cust_0_57 \device\harddisk11\partition4
cust_0_58 \device\harddisk13\partition4
cust_0_59 \device\harddisk14\partition4
cust_0_60 \device\harddisk16\partition4
cust_0_61 \device\harddisk17\partition4
cust_0_62 \device\harddisk18\partition4
cust_0_63 \device\harddisk19\partition4
cust_0_64 \device\harddisk20\partition4
cust_0_65 \device\harddisk22\partition4
cust_0_66 \device\harddisk23\partition4
cust_0_67 \device\harddisk24\partition4
cust_0_68 \device\harddisk25\partition4
cust_0_69 \device\harddisk26\partition4
cust_0_70 \device\harddisk28\partition4
cust_0_71 \device\harddisk29\partition4
cust_0_72 \device\harddisk30\partition4
cust_0_73 \device\harddisk31\partition4
cust_0_74 \device\harddisk32\partition4
cust_0_75 \device\harddisk33\partition4
cust_0_76 \device\harddisk34\partition4
cust_0_77 \device\harddisk36\partition4
cust_0_78 \device\harddisk37\partition4
cust_0_79 \device\harddisk38\partition4
cust_0_80 \device\harddisk39\partition4
cust_0_81 \device\harddisk40\partition4
cust_0_82 \device\harddisk42\partition4
cust_0_83 \device\harddisk43\partition4
cust_0_84 \device\harddisk44\partition4
cust_0_85 \device\harddisk45\partition4
cust_0_86 \device\harddisk46\partition4
cust_0_87 \device\harddisk47\partition4
cust_0_88 \device\harddisk48\partition4
cust_0_89 \device\harddisk50\partition4
cust_0_90 \device\harddisk51\partition4
cust_0_91 \device\harddisk52\partition4
cust_0_92 \device\harddisk53\partition4
cust_0_93 \device\harddisk54\partition4
cust_0_94 \device\harddisk56\partition4
cust_0_95 \device\harddisk57\partition4
icust2_0_0 \device\harddisk1\partition5
icust2_0_1 \device\harddisk2\partition5

```

```

icust2_0_2    \device\harddisk3\partition5
icust2_0_3    \device\harddisk4\partition5
icust2_0_4    \device\harddisk5\partition5
icust2_0_5    \device\harddisk7\partition5
icust2_0_6    \device\harddisk8\partition5
icust2_0_7    \device\harddisk9\partition5
icust2_0_8    \device\harddisk10\partition5
icust2_0_9    \device\harddisk11\partition5
icust2_0_10   \device\harddisk13\partition5
icust2_0_11   \device\harddisk14\partition5
icust2_0_12   \device\harddisk16\partition5
icust2_0_13   \device\harddisk17\partition5
icust2_0_14   \device\harddisk18\partition5
icust2_0_15   \device\harddisk19\partition5
icust2_0_16   \device\harddisk20\partition5
icust2_0_17   \device\harddisk22\partition5
icust2_0_18   \device\harddisk23\partition5
icust2_0_19   \device\harddisk24\partition5
icust2_0_20   \device\harddisk25\partition5
icust2_0_21   \device\harddisk26\partition5
icust2_0_22   \device\harddisk28\partition5
icust2_0_23   \device\harddisk29\partition5
icust2_0_24   \device\harddisk30\partition5
icust2_0_25   \device\harddisk31\partition5
icust2_0_26   \device\harddisk32\partition5
icust2_0_27   \device\harddisk33\partition5
icust2_0_28   \device\harddisk34\partition5
icust2_0_29   \device\harddisk36\partition5
icust2_0_30   \device\harddisk37\partition5
icust2_0_31   \device\harddisk38\partition5
icust2_0_32   \device\harddisk39\partition5
icust2_0_33   \device\harddisk40\partition5
icust2_0_34   \device\harddisk42\partition5
icust2_0_35   \device\harddisk43\partition5
icust2_0_36   \device\harddisk44\partition5
icust2_0_37   \device\harddisk45\partition5
icust2_0_38   \device\harddisk46\partition5
icust2_0_39   \device\harddisk47\partition5
icust2_0_40   \device\harddisk48\partition5
icust2_0_41   \device\harddisk50\partition5
icust2_0_42   \device\harddisk51\partition5
icust2_0_43   \device\harddisk52\partition5
icust2_0_44   \device\harddisk53\partition5
icust2_0_45   \device\harddisk54\partition5
icust2_0_46   \device\harddisk56\partition5
icust2_0_47   \device\harddisk57\partition5
item_0_0      \device\harddisk1\partition6
item_0_1      \device\harddisk2\partition6
item_0_2      \device\harddisk3\partition6
item_0_3      \device\harddisk4\partition6
item_0_4      \device\harddisk5\partition6
item_0_5      \device\harddisk7\partition6
item_0_6      \device\harddisk8\partition6
item_0_7      \device\harddisk9\partition6
item_0_8      \device\harddisk10\partition6
item_0_9      \device\harddisk11\partition6
item_0_10     \device\harddisk13\partition6
item_0_11     \device\harddisk14\partition6
item_0_12     \device\harddisk16\partition6
item_0_13     \device\harddisk17\partition6
item_0_14     \device\harddisk18\partition6
item_0_15     \device\harddisk19\partition6
item_0_16     \device\harddisk20\partition6

```

```

item_0_17     \device\harddisk22\partition6
item_0_18     \device\harddisk23\partition6
item_0_19     \device\harddisk24\partition6
item_0_20     \device\harddisk25\partition6
item_0_21     \device\harddisk26\partition6
item_0_22     \device\harddisk28\partition6
item_0_23     \device\harddisk29\partition6
item_0_24     \device\harddisk30\partition6
item_0_25     \device\harddisk31\partition6
item_0_26     \device\harddisk32\partition6
item_0_27     \device\harddisk33\partition6
item_0_28     \device\harddisk34\partition6
item_0_29     \device\harddisk36\partition6
item_0_30     \device\harddisk37\partition6
item_0_31     \device\harddisk38\partition6
item_0_32     \device\harddisk39\partition6
item_0_33     \device\harddisk40\partition6
item_0_34     \device\harddisk42\partition6
item_0_35     \device\harddisk43\partition6
item_0_36     \device\harddisk44\partition6
item_0_37     \device\harddisk45\partition6
item_0_38     \device\harddisk46\partition6
item_0_39     \device\harddisk47\partition6
item_0_40     \device\harddisk48\partition6
item_0_41     \device\harddisk50\partition6
item_0_42     \device\harddisk51\partition6
item_0_43     \device\harddisk52\partition6
item_0_44     \device\harddisk53\partition6
item_0_45     \device\harddisk54\partition6
item_0_46     \device\harddisk56\partition6
item_0_47     \device\harddisk57\partition6
ordl_0_0      \device\harddisk1\partition7
ordl_0_1      \device\harddisk2\partition7
ordl_0_2      \device\harddisk3\partition7
ordl_1_0      \device\harddisk4\partition7
ordl_1_1      \device\harddisk5\partition7
ordl_1_2      \device\harddisk7\partition7
ordl_2_0      \device\harddisk8\partition7
ordl_2_1      \device\harddisk9\partition7
ordl_2_2      \device\harddisk10\partition7
ordl_3_0      \device\harddisk11\partition7
ordl_3_1      \device\harddisk13\partition7
ordl_3_2      \device\harddisk14\partition7
ordl_0_3      \device\harddisk16\partition7
ordl_0_4      \device\harddisk17\partition7
ordl_0_5      \device\harddisk18\partition7
ordl_1_3      \device\harddisk19\partition7
ordl_1_4      \device\harddisk20\partition7
ordl_1_5      \device\harddisk22\partition7
ordl_2_3      \device\harddisk23\partition7
ordl_2_4      \device\harddisk24\partition7
ordl_2_5      \device\harddisk25\partition7
ordl_3_3      \device\harddisk26\partition7
ordl_3_4      \device\harddisk28\partition7
ordl_3_5      \device\harddisk29\partition7
ordl_0_6      \device\harddisk30\partition7
ordl_0_7      \device\harddisk31\partition7
ordl_0_8      \device\harddisk32\partition7
ordl_1_6      \device\harddisk33\partition7
ordl_1_7      \device\harddisk34\partition7
ordl_1_8      \device\harddisk36\partition7
ordl_2_6      \device\harddisk37\partition7
ordl_2_7      \device\harddisk38\partition7

```

```

ordl_2_8    \device\harddisk39\partition7
ordl_3_6    \device\harddisk40\partition7
ordl_3_7    \device\harddisk42\partition7
ordl_3_8    \device\harddisk43\partition7
ordl_0_9    \device\harddisk44\partition7
ordl_0_10   \device\harddisk45\partition7
ordl_0_11   \device\harddisk46\partition7
ordl_1_9    \device\harddisk47\partition7
ordl_1_10   \device\harddisk48\partition7
ordl_1_11   \device\harddisk50\partition7
ordl_2_9    \device\harddisk51\partition7
ordl_2_10   \device\harddisk52\partition7
ordl_2_11   \device\harddisk53\partition7
ordl_3_9    \device\harddisk54\partition7
ordl_3_10   \device\harddisk56\partition7
ordl_3_11   \device\harddisk57\partition7
ordr_0_0    \device\harddisk1\partition8
ordr_0_1    \device\harddisk2\partition8
ordr_0_2    \device\harddisk3\partition8
ordr_1_0    \device\harddisk4\partition8
ordr_1_1    \device\harddisk5\partition8
ordr_1_2    \device\harddisk7\partition8
ordr_2_0    \device\harddisk8\partition8
ordr_2_1    \device\harddisk9\partition8
ordr_2_2    \device\harddisk10\partition8
ordr_3_0    \device\harddisk11\partition8
ordr_3_1    \device\harddisk13\partition8
ordr_3_2    \device\harddisk14\partition8
ordr_0_3    \device\harddisk16\partition8
ordr_0_4    \device\harddisk17\partition8
ordr_0_5    \device\harddisk18\partition8
ordr_1_3    \device\harddisk19\partition8
ordr_1_4    \device\harddisk20\partition8
ordr_1_5    \device\harddisk22\partition8
ordr_2_3    \device\harddisk23\partition8
ordr_2_4    \device\harddisk24\partition8
ordr_2_5    \device\harddisk25\partition8
ordr_3_3    \device\harddisk26\partition8
ordr_3_4    \device\harddisk28\partition8
ordr_3_5    \device\harddisk29\partition8
ordr_0_6    \device\harddisk30\partition8
ordr_0_7    \device\harddisk31\partition8
ordr_0_8    \device\harddisk32\partition8
ordr_1_6    \device\harddisk33\partition8
ordr_1_7    \device\harddisk34\partition8
ordr_1_8    \device\harddisk36\partition8
ordr_2_6    \device\harddisk37\partition8
ordr_2_7    \device\harddisk38\partition8
ordr_2_8    \device\harddisk39\partition8
ordr_3_6    \device\harddisk40\partition8
ordr_3_7    \device\harddisk42\partition8
ordr_3_8    \device\harddisk43\partition8
ordr_0_9    \device\harddisk44\partition8
ordr_0_10   \device\harddisk45\partition8
ordr_0_11   \device\harddisk46\partition8
ordr_1_9    \device\harddisk47\partition8
ordr_1_10   \device\harddisk48\partition8
ordr_1_11   \device\harddisk50\partition8
ordr_2_9    \device\harddisk51\partition8
ordr_2_10   \device\harddisk52\partition8
ordr_2_11   \device\harddisk53\partition8
ordr_3_9    \device\harddisk54\partition8
ordr_3_10   \device\harddisk56\partition8

```

```

ordr_3_11   \device\harddisk57\partition8
iordr_0_0   \device\harddisk1\partition9
iordr_0_1   \device\harddisk2\partition9
iordr_0_2   \device\harddisk3\partition9
iordr_1_0   \device\harddisk4\partition9
iordr_1_1   \device\harddisk5\partition9
iordr_1_2   \device\harddisk7\partition9
iordr_2_0   \device\harddisk8\partition9
iordr_2_1   \device\harddisk9\partition9
iordr_2_2   \device\harddisk10\partition9
iordr_3_0   \device\harddisk11\partition9
iordr_3_1   \device\harddisk13\partition9
iordr_3_2   \device\harddisk14\partition9
iordr_0_3   \device\harddisk16\partition9
iordr_0_4   \device\harddisk17\partition9
iordr_0_5   \device\harddisk18\partition9
iordr_1_3   \device\harddisk19\partition9
iordr_1_4   \device\harddisk20\partition9
iordr_1_5   \device\harddisk22\partition9
iordr_2_3   \device\harddisk23\partition9
iordr_2_4   \device\harddisk24\partition9
iordr_2_5   \device\harddisk25\partition9
iordr_3_3   \device\harddisk26\partition9
iordr_3_4   \device\harddisk28\partition9
iordr_3_5   \device\harddisk29\partition9
iordr_0_6   \device\harddisk30\partition9
iordr_0_7   \device\harddisk31\partition9
iordr_0_8   \device\harddisk32\partition9
iordr_1_6   \device\harddisk33\partition9
iordr_1_7   \device\harddisk34\partition9
iordr_1_8   \device\harddisk36\partition9
iordr_2_6   \device\harddisk37\partition9
iordr_2_7   \device\harddisk38\partition9
iordr_2_8   \device\harddisk39\partition9
iordr_3_6   \device\harddisk40\partition9
iordr_3_7   \device\harddisk42\partition9
iordr_3_8   \device\harddisk43\partition9
iordr_0_9   \device\harddisk44\partition9
iordr_0_10  \device\harddisk45\partition9
iordr_0_11  \device\harddisk46\partition9
iordr_1_9   \device\harddisk47\partition9
iordr_1_10  \device\harddisk48\partition9
iordr_1_11  \device\harddisk50\partition9
iordr_2_9   \device\harddisk51\partition9
iordr_2_10  \device\harddisk52\partition9
iordr_2_11  \device\harddisk53\partition9
iordr_3_9   \device\harddisk54\partition9
iordr_3_10  \device\harddisk56\partition9
iordr_3_11  \device\harddisk57\partition9
iordr_0_0   \device\harddisk1\partition10
iordr_2_0_1 \device\harddisk2\partition10
iordr_2_0_2 \device\harddisk3\partition10
iordr_2_1_0 \device\harddisk4\partition10
iordr_2_1_1 \device\harddisk5\partition10
iordr_2_1_2 \device\harddisk7\partition10
iordr_2_2_0 \device\harddisk8\partition10
iordr_2_2_1 \device\harddisk9\partition10
iordr_2_2_2 \device\harddisk10\partition10
iordr_2_3_0 \device\harddisk11\partition10
iordr_2_3_1 \device\harddisk13\partition10
iordr_2_3_2 \device\harddisk14\partition10
iordr_2_0_3 \device\harddisk16\partition10
iordr_2_0_4 \device\harddisk17\partition10

```

```

iordr2_0_5 \device\harddisk18\partition10
iordr2_1_3 \device\harddisk19\partition10
iordr2_1_4 \device\harddisk20\partition10
iordr2_1_5 \device\harddisk22\partition10
iordr2_2_3 \device\harddisk23\partition10
iordr2_2_4 \device\harddisk24\partition10
iordr2_2_5 \device\harddisk25\partition10
iordr2_3_3 \device\harddisk26\partition10
iordr2_3_4 \device\harddisk28\partition10
iordr2_3_5 \device\harddisk29\partition10
iordr2_0_6 \device\harddisk30\partition10
iordr2_0_7 \device\harddisk31\partition10
iordr2_0_8 \device\harddisk32\partition10
iordr2_1_6 \device\harddisk33\partition10
iordr2_1_7 \device\harddisk34\partition10
iordr2_1_8 \device\harddisk36\partition10
iordr2_2_6 \device\harddisk37\partition10
iordr2_2_7 \device\harddisk38\partition10
iordr2_2_8 \device\harddisk39\partition10
iordr2_3_6 \device\harddisk40\partition10
iordr2_3_7 \device\harddisk42\partition10
iordr2_3_8 \device\harddisk43\partition10
iordr2_0_9 \device\harddisk44\partition10
iordr2_0_10 \device\harddisk45\partition10
iordr2_0_11 \device\harddisk46\partition10
iordr2_1_9 \device\harddisk47\partition10
iordr2_1_10 \device\harddisk48\partition10
iordr2_1_11 \device\harddisk50\partition10
iordr2_2_9 \device\harddisk51\partition10
iordr2_2_10 \device\harddisk52\partition10
iordr2_2_11 \device\harddisk53\partition10
iordr2_3_9 \device\harddisk54\partition10
iordr2_3_10 \device\harddisk56\partition10
iordr2_3_11 \device\harddisk57\partition10
nord_0_0 \device\harddisk1\partition11
nord_0_1 \device\harddisk2\partition11
nord_0_2 \device\harddisk3\partition11
nord_1_0 \device\harddisk4\partition11
nord_1_1 \device\harddisk5\partition11
nord_1_2 \device\harddisk7\partition11
nord_2_0 \device\harddisk8\partition11
nord_2_1 \device\harddisk9\partition11
nord_2_2 \device\harddisk10\partition11
nord_3_0 \device\harddisk11\partition11
nord_3_1 \device\harddisk13\partition11
nord_3_2 \device\harddisk14\partition11
nord_0_3 \device\harddisk16\partition11
nord_0_4 \device\harddisk17\partition11
nord_0_5 \device\harddisk18\partition11
nord_1_3 \device\harddisk19\partition11
nord_1_4 \device\harddisk20\partition11
nord_1_5 \device\harddisk22\partition11
nord_2_3 \device\harddisk23\partition11
nord_2_4 \device\harddisk24\partition11
nord_2_5 \device\harddisk25\partition11
nord_3_3 \device\harddisk26\partition11
nord_3_4 \device\harddisk28\partition11
nord_3_5 \device\harddisk29\partition11
nord_0_6 \device\harddisk30\partition11
nord_0_7 \device\harddisk31\partition11
nord_0_8 \device\harddisk32\partition11
nord_1_6 \device\harddisk33\partition11
nord_1_7 \device\harddisk34\partition11

```

```

nord_1_8 \device\harddisk36\partition11
nord_2_6 \device\harddisk37\partition11
nord_2_7 \device\harddisk38\partition11
nord_2_8 \device\harddisk39\partition11
nord_3_6 \device\harddisk40\partition11
nord_3_7 \device\harddisk42\partition11
nord_3_8 \device\harddisk43\partition11
nord_0_9 \device\harddisk44\partition11
nord_0_10 \device\harddisk45\partition11
nord_0_11 \device\harddisk46\partition11
nord_1_9 \device\harddisk47\partition11
nord_1_10 \device\harddisk48\partition11
nord_1_11 \device\harddisk50\partition11
nord_2_9 \device\harddisk51\partition11
nord_2_10 \device\harddisk52\partition11
nord_2_11 \device\harddisk53\partition11
nord_3_9 \device\harddisk54\partition11
nord_3_10 \device\harddisk56\partition11
nord_3_11 \device\harddisk57\partition11
hist_0_0 \device\harddisk1\partition12
hist_0_1 \device\harddisk2\partition12
hist_0_2 \device\harddisk3\partition12
hist_1_0 \device\harddisk4\partition12
hist_1_1 \device\harddisk5\partition12
hist_1_2 \device\harddisk7\partition12
hist_2_0 \device\harddisk8\partition12
hist_2_1 \device\harddisk9\partition12
hist_2_2 \device\harddisk10\partition12
hist_3_0 \device\harddisk11\partition12
hist_3_1 \device\harddisk13\partition12
hist_3_2 \device\harddisk14\partition12
hist_0_3 \device\harddisk16\partition12
hist_0_4 \device\harddisk17\partition12
hist_0_5 \device\harddisk18\partition12
hist_1_3 \device\harddisk19\partition12
hist_1_4 \device\harddisk20\partition12
hist_1_5 \device\harddisk22\partition12
hist_2_3 \device\harddisk23\partition12
hist_2_4 \device\harddisk24\partition12
hist_2_5 \device\harddisk25\partition12
hist_3_3 \device\harddisk26\partition12
hist_3_4 \device\harddisk28\partition12
hist_3_5 \device\harddisk29\partition12
hist_0_6 \device\harddisk30\partition12
hist_0_7 \device\harddisk31\partition12
hist_0_8 \device\harddisk32\partition12
hist_1_6 \device\harddisk33\partition12
hist_1_7 \device\harddisk34\partition12
hist_1_8 \device\harddisk36\partition12
hist_2_6 \device\harddisk37\partition12
hist_2_7 \device\harddisk38\partition12
hist_2_8 \device\harddisk39\partition12
hist_3_6 \device\harddisk40\partition12
hist_3_7 \device\harddisk42\partition12
hist_3_8 \device\harddisk43\partition12
hist_0_9 \device\harddisk44\partition12
hist_0_10 \device\harddisk45\partition12
hist_0_11 \device\harddisk46\partition12
hist_1_9 \device\harddisk47\partition12
hist_1_10 \device\harddisk48\partition12
hist_1_11 \device\harddisk50\partition12
hist_2_9 \device\harddisk51\partition12
hist_2_10 \device\harddisk52\partition12

```



```

roll_0_8      \device\harddisk32\partition15
roll_1_6      \device\harddisk33\partition15
roll_1_7      \device\harddisk34\partition15
roll_1_8      \device\harddisk36\partition15
roll_2_6      \device\harddisk37\partition15
roll_2_7      \device\harddisk38\partition15
roll_2_8      \device\harddisk39\partition15
roll_3_6      \device\harddisk40\partition15
roll_3_7      \device\harddisk42\partition15
roll_3_8      \device\harddisk43\partition15
roll_0_9      \device\harddisk44\partition15
roll_0_10     \device\harddisk45\partition15
roll_0_11     \device\harddisk46\partition15
roll_1_9      \device\harddisk47\partition15
roll_1_10     \device\harddisk48\partition15
roll_1_11     \device\harddisk50\partition15
roll_2_9      \device\harddisk51\partition15
roll_2_10     \device\harddisk52\partition15
roll_2_11     \device\harddisk53\partition15
roll_3_9      \device\harddisk54\partition15
roll_3_10     \device\harddisk56\partition15
roll_3_11     \device\harddisk57\partition15
temp_0_0      \device\harddisk1\partition16
temp_0_1      \device\harddisk2\partition16
temp_0_2      \device\harddisk3\partition16
temp_0_3      \device\harddisk4\partition16
temp_0_4      \device\harddisk5\partition16
temp_0_5      \device\harddisk7\partition16
temp_0_6      \device\harddisk8\partition16
temp_0_7      \device\harddisk9\partition16
temp_0_8      \device\harddisk10\partition16
temp_0_9      \device\harddisk11\partition16
temp_0_10     \device\harddisk13\partition16
temp_0_11     \device\harddisk14\partition16
temp_0_12     \device\harddisk16\partition16
temp_0_13     \device\harddisk17\partition16
temp_0_14     \device\harddisk18\partition16
temp_0_15     \device\harddisk19\partition16
temp_0_16     \device\harddisk20\partition16
temp_0_17     \device\harddisk22\partition16
temp_0_18     \device\harddisk23\partition16
temp_0_19     \device\harddisk24\partition16
temp_0_20     \device\harddisk25\partition16
temp_0_21     \device\harddisk26\partition16
temp_0_22     \device\harddisk28\partition16
temp_0_23     \device\harddisk29\partition16
temp_0_24     \device\harddisk30\partition16
temp_0_25     \device\harddisk31\partition16
temp_0_26     \device\harddisk32\partition16
temp_0_27     \device\harddisk33\partition16
temp_0_28     \device\harddisk34\partition16
temp_0_29     \device\harddisk36\partition16
temp_0_30     \device\harddisk37\partition16
temp_0_31     \device\harddisk38\partition16
temp_0_32     \device\harddisk39\partition16
temp_0_33     \device\harddisk40\partition16
temp_0_34     \device\harddisk42\partition16
temp_0_35     \device\harddisk43\partition16
temp_0_36     \device\harddisk44\partition16
temp_0_37     \device\harddisk45\partition16
temp_0_38     \device\harddisk46\partition16
temp_0_39     \device\harddisk47\partition16
temp_0_40     \device\harddisk48\partition16

```

```

temp_0_41     \device\harddisk50\partition16
temp_0_42     \device\harddisk51\partition16
temp_0_43     \device\harddisk52\partition16
temp_0_44     \device\harddisk53\partition16
temp_0_45     \device\harddisk54\partition16
temp_0_46     \device\harddisk56\partition16
temp_0_47     \device\harddisk57\partition16
iware_0_0     \device\harddisk1\partition17
idist_0_0     \device\harddisk2\partition17
iitem_0_0     \device\harddisk3\partition17
inord_0_0     \device\harddisk4\partition17
icust1_0_0    \device\harddisk5\partition17
istok_0_0     \device\harddisk7\partition17
iord1_0_0     \device\harddisk8\partition17
iord1_0_1     \device\harddisk9\partition17
iord1_0_2     \device\harddisk10\partition17
control_001   \device\harddisk11\partition17
iware_1_0     \device\harddisk16\partition17
idist_1_0     \device\harddisk17\partition17
iitem_0_1     \device\harddisk18\partition17
inord_0_1     \device\harddisk19\partition17
icust1_0_1    \device\harddisk20\partition17
istok_0_1     \device\harddisk22\partition17
iord1_0_3     \device\harddisk23\partition17
iord1_0_4     \device\harddisk24\partition17
iord1_0_5     \device\harddisk25\partition17
system_001   \device\harddisk26\partition17
iware_2_0     \device\harddisk30\partition17
idist_2_0     \device\harddisk31\partition17
iitem_0_2     \device\harddisk32\partition17
inord_0_2     \device\harddisk33\partition17
icust1_0_2    \device\harddisk34\partition17
istok_0_2     \device\harddisk36\partition17
iord1_0_6     \device\harddisk37\partition17
iord1_0_7     \device\harddisk38\partition17
iord1_0_8     \device\harddisk39\partition17
control_002   \device\harddisk40\partition17
iware_3_0     \device\harddisk44\partition17
idist_3_0     \device\harddisk45\partition17
iitem_0_3     \device\harddisk46\partition17
inord_0_3     \device\harddisk47\partition17
icust1_0_3    \device\harddisk48\partition17
istok_0_3     \device\harddisk50\partition17
iord1_0_9     \device\harddisk51\partition17
iord1_0_10    \device\harddisk52\partition17
iord1_0_11    \device\harddisk53\partition17

```

Driver.sh

```

#!/sh

STEP=1
START=0
END=0
CONTINUE=1
PROGRAMME="driver.sh"
STOPFILE="stop"
TRACEFILE="log/trace.log"
JUNKFILE="junk"

```

```

usage()
{
  echo ""
  echo "Usage: $PROGNAME [<startstepno> <stopstepno>] [<startstepno>]
[-step <stepno>]"
  echo "      [<startstepno> <stopstepno>] - allows user to run a
specified"
  echo "      range of steps."
  echo "      [<startstepno>] - runs from step number
<start> till"
  echo "      the end of the script."
  echo "      [-step <stepno>] - runs only step number
<stepno> and"
  echo "      then stops."
  echo ""
  echo "      STEP    FUNCTION"
  echo "-----"
  echo "      1      Create Database."
  echo "      2      Create Rollback Segments for Build."
  echo "      3      Shutdown Database."
  echo "      4      Startup Database for Build"
  echo "      5      Create User TPCC."
  echo "      6      Create Tablespaces."
  echo "      7      Assign Temporary Tablespace to user
TPCC."
  echo "      8      Create Data Dictionary Views."
  echo "      9      Create Warehouse."
  echo "     10      Create District."
  echo "     11      Create Customer."
  echo "     12      Create History."
  echo "     13      Create Order."
  echo "     14      Create Neworder."
  echo "     15      Create Orderline."
  echo "     16      Create Stock."
  echo "     17      Create Item."
  echo "     18      Load Warehouse."
  echo "     19      Load District."
  echo "     20      Load Item."
  echo "     21      Load History"
  echo "     22      Load Neworder"
  echo "     23      Load Order/Orderline"
  echo "     24      Load Customer"
  echo "     25      Load Stock"
  echo "     26      Create Warehouse Index."
  echo "     27      Create District Index."
  echo "     28      Create Item Index."
  echo "     29      Create Customer1 Index."
  echo "     30      Create Customer2 Index."
  echo "     31      Create Stock Index."
  echo "     32      Create Orders1 Index."
  echo "     33      Create Orders2 Index."
  echo "     34      Create Neworder Index."
  echo "     35      Create Orderline Index."
  echo "     36      Analyze Tables/Clusters/Indexes."
  echo "     37      Create Statistics Tables."
  echo "     38      Load Stored Procedures."
  echo "     39      Create Rollback Segments for Runs."
  echo "     40      Generate Space Report."
  echo "     41      Misc."
  echo "     42      Offline Rollback segments for Build."

```

```

  echo "-----"
  exit 1;
}

torun()
{
  mv -f *.log* log > junk 2>&1
  rm -f $JUNKFILE*

  if test -f $STOPFILE
  then
    echo "An error has ocured, please look into the $TRACEFILE file."
    echo "OR you need to remove the 'stop' file found in the current
directory."
    echo "The 'stop' file need to be removed after an error occurs and
before resuming."
    exit 1;
  fi
  if test $STEP -ge $START
  then
    if test $STEP -le $END
    then
      STEP=`expr $STEP + 1`
      return 0
    else
      if test $CONTINUE -eq 0
      then
        STEP=`expr $STEP + 1`
        return 0
      fi
    fi
  fi
  STEP=`expr $STEP + 1`
  return 1
}

case $# in
  0)    usage;
        ;;
  1)    case $1 in
        -h)    usage
                ;;
        [0-9]*)
                START=$1
                CONTINUE=0
                ;;
        *)    usage
                ;;
        esac
        ;;
  2)    case $1 in
        -step) shift
                case $1 in
                [0-9]*) ;;
                *) usage
                        ;;
                esac
                START=$1
                END=$1
                CONTINUE=1
                ;;
        [0-9]*)
                START=$1

```

```

        shift
        case $1 in
            [0-9]*) ;;
            *) usage
               ;;
        esac
        END=$1
        CONTINUE=1
        ;;
    *) usage
       ;;
esac
*) usage
;;
esac
*) usage
;;
esac
if torun
then
    echo "Creating the Database ..."
    echo "Creating the Database ..." `date` "\n" >> $TRACEFILE
    ./step2createdb.sh
    if test $? -ne 0
    then
        echo "Creating the Database failed." `date` "\n" >> $TRACEFILE
        echo "Look at log/step2createdb.log for more details." `date` "\n"
    >> $TRACEFILE
        echo "Stopped" >> $STOPFILE
        echo "Creating the Database failed ..."
    else
        echo "Creating the Database done." `date` "\n" >> $TRACEFILE
        echo "Creating the Database done ..."
    fi
fi
if torun
then
    echo "Creating the Rollback Segments ..."
    echo "Creating the Rollback Segments ..." `date` "\n" >> $TRACEFILE
    ./step3createrollback.sh
    if test $? -ne 0
    then
        echo "Creating the Rollback Segments failed." `date` "\n" >>
$TRACEFILE
        echo "Look at log/step3createrollback.log for more details." `date`
"\n" >> $TRACEFILE
        echo "Stopped" >> $STOPFILE
        echo "Creating the Rollback Segments failed ..."
    else
        echo "Creating the Rollback Segments done." `date` "\n" >>
$TRACEFILE
        echo "Creating the Rollback Segments done ..."
    fi
fi
if torun
then
    echo "Shutting down the Database ..."
    echo "Shutting down the Database ..." `date` "\n" >> $TRACEFILE
    ./stepshut.sh
    if test $? -ne 0
    then
        echo "Shutting down the Database failed." `date` "\n" >> $TRACEFILE

```

```

        echo "Look at log/stepshut.log for more details." `date` "\n" >>
$TRACEFILE
        echo "Stopped" >> $STOPFILE
        echo "Shutting down the Database failed ..."
    else
        echo "Shutting down the Database done." `date` "\n" >> $TRACEFILE
        echo "Shutting down the Database done ..."
    fi
fi
if torun
then
    echo "Start up Database for Build ..."
    echo "Start up Database for Build ..." `date` "\n" >> $TRACEFILE
    ./stepstartb.sh
    if test $? -ne 0
    then
        echo "Start up Database for Build failed." `date` "\n" >>
$TRACEFILE
        echo "Look at log/stepstartb.log for more details." `date` "\n" >>
$TRACEFILE
        echo "Stopped" >> $STOPFILE
        echo "Start up Database for Build failed ..."
    else
        echo "Start up Database for Build done." `date` "\n" >> $TRACEFILE
        echo "Start up Database for Build done ..."
    fi
fi
if torun
then
    echo "Create User TPCC ..."
    echo "Create User TPCC ..." `date` "\n" >> $TRACEFILE
    ./stepcreateuser.sh
    if test $? -ne 0
    then
        echo "Create User TPCC failed." `date` "\n" >> $TRACEFILE
        echo "Look at log/stepcreateuser.log for more details." `date` "\n"
    >> $TRACEFILE
        echo "Stopped" >> $STOPFILE
        echo "Create User TPCC failed ..."
    else
        echo "Create User TPCC done." `date` "\n" >> $TRACEFILE
        echo "Create User TPCC done ..."
    fi
fi
if torun
then
    echo "Create Tablespaces ..."
    echo "Create Tablespaces ..." `date` "\n" >> $TRACEFILE
    ./step5createts.sh
    if test $? -ne 0
    then
        echo "Create Tablespaces failed." `date` "\n" >> $TRACEFILE
        echo "Look at log/step5createts*.log for more details." `date` "\n"
    >> $TRACEFILE
        echo "Stopped" >> $STOPFILE
        echo "Create Tablespaces failed ..."
    else
        echo "Create Tablespaces done." `date` "\n" >> $TRACEFILE
        echo "Create Tablespaces done ..."
    fi
fi

```

```

fi
if torun
then
  echo "Assign Temporary Tablespace to user TPCC ..."
  echo "Assign Temporary Tablespace to user TPCC ..." `date` "\n" >>
$TRACEFILE
  ./stepusertemp.sh
  if test $? -ne 0
  then
    echo "Assign Temporary Tablespace to user TPCC failed." `date` "\n"
>> $TRACEFILE
    echo "Look at log/stepusertemp.log for more details." `date` "\n"
>> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Assign Temporary Tablespace to user TPCC failed ..."
  else
    echo "Assign Temporary Tablespace to user TPCC done." `date` "\n"
>> $TRACEFILE
    echo "Assign Temporary Tablespace to user TPCC done ..."
  fi
fi
if torun
then
  echo "Create Data Dictionary views ..."
  echo "Create Data Dictionary views ..." `date` "\n" >> $TRACEFILE
  ./step6createddviews.sh
  if test $? -ne 0
  then
    echo "Create Data Dictionary views failed." `date` "\n" >>
$TRACEFILE
    echo "Look at log/step6createddviews.log for more details." `date`
"\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create Data Dictionary views failed ..."
  else
    echo "Create Data Dictionary views done." `date` "\n" >> $TRACEFILE
    echo "Create Data Dictionary views done ..."
  fi
fi
if torun
then
  echo "Create Warehouse ..."
  echo "Create Warehouse ..." `date` "\n" >> $TRACEFILE
  ./step9createware.sh
  if test $? -ne 0
  then
    echo "Create Warehouse failed." `date` "\n" >> $TRACEFILE
    echo "Look at log/step9createware.log for more details." `date`
"\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create Warehouse failed ..."
  else
    echo "Create Warehouse done." `date` "\n" >> $TRACEFILE
    echo "Create Warehouse done ..."
  fi
fi
if torun
then
  echo "Create District ..."

```

```

  echo "Create District ..." `date` "\n" >> $TRACEFILE
  ./step10createdist.sh
  if test $? -ne 0
  then
    echo "Create District failed." `date` "\n" >> $TRACEFILE
    echo "Look at log/step10createdist.log for more details." `date`
"\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create District failed ..."
  else
    echo "Create District done." `date` "\n" >> $TRACEFILE
    echo "Create District done ..."
  fi
fi
if torun
then
  echo "Create Customer ..."
  echo "Create Customer ..." `date` "\n" >> $TRACEFILE
  ./step11createcust.sh
  if test $? -ne 0
  then
    echo "Create Customer failed.\n" `date` "\n" >> $TRACEFILE
    echo "Look at log/step11createcust.log for more details." `date`
"\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create Customer failed ..."
  else
    echo "Create Customer done." `date` "\n" >> $TRACEFILE
    echo "Create Customer done ..."
  fi
fi
if torun
then
  echo "Create History ..."
  echo "Create History ..." `date` "\n" >> $TRACEFILE
  ./step12createhist.sh
  if test $? -ne 0
  then
    echo "Create History failed." `date` "\n" >> $TRACEFILE
    echo "Look at log/step12createhist.log for more details." `date`
"\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create History failed ..."
  else
    echo "Create History done." `date` "\n" >> $TRACEFILE
    echo "Create History done ..."
  fi
fi
if torun
then
  echo "Create Order ..."
  echo "Create Order ..." `date` "\n" >> $TRACEFILE
  ./step13createordr.sh
  if test $? -ne 0
  then
    echo "Create Order failed." `date` "\n" >> $TRACEFILE
    echo "Look at log/step13createordr.log for more details." `date`
"\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create Order failed ..."

```

```

else
  echo "Create Order done." `date` "\n" >> $TRACEFILE
  echo "Create Order done ..."
fi
fi
if torun
then
  echo "Create Neworder ..."
  echo "Create Neworder ..." `date` "\n" >> $TRACEFILE
  ./step14createnord.sh
  if test $? -ne 0
  then
    echo "Create Neworder failed." `date` "\n" >> $TRACEFILE
    echo "Look at log/step14createnord.log for more details." `date`
"\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create Neworder failed ..."
  else
    echo "Create Neworder done." `date` "\n" >> $TRACEFILE
    echo "Create Neworder done ..."
  fi
fi
if torun
then
  echo "Create Orderline ..."
  echo "Create Orderline ..." `date` "\n" >> $TRACEFILE
  ./step15createordl.sh
  if test $? -ne 0
  then
    echo "Create Orderline failed." `date` "\n" >> $TRACEFILE
    echo "Look at log/step15createordl.log for more details." `date`
"\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create Orderline failed ..."
  else
    echo "Create Orderline done." `date` "\n" >> $TRACEFILE
    echo "Create Orderline done ..."
  fi
fi
if torun
then
  echo "Create Stock ..."
  echo "Create Stock ..." `date` "\n" >> $TRACEFILE
  ./step16createtok.sh
  if test $? -ne 0
  then
    echo "Create Stock failed." `date` "\n" >> $TRACEFILE
    echo "Look at log/step16createtok.log for more details." `date`
"\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create Stock failed ..."
  else
    echo "Create Stock done." `date` "\n" >> $TRACEFILE
    echo "Create Stock done ..."
  fi
fi
if torun
then
  echo "Create Item ..."

```

```

echo "Create Item ..." `date` "\n" >> $TRACEFILE
./step17createitem.sh
if test $? -ne 0
then
  echo "Create Item failed." `date` "\n" >> $TRACEFILE
  echo "Look at log/step17createitem.log for more details." `date`
"\n" >> $TRACEFILE
  echo "Stopped" >> $STOPFILE
  echo "Create Item failed ..."
else
  echo "Create Item done." `date` "\n" >> $TRACEFILE
  echo "Create Item done ..."
fi
fi
if torun
then
  echo "Load Warehouse ..."
  echo "Load Warehouse ..." `date` "\n" >> $TRACEFILE
  ./step20loadware.sh
  if test $? -ne 0
  then
    echo "Load Warehouse failed." `date` "\n" >> $TRACEFILE
    echo "Look at log/step20loadware.log for more details." `date` "\n"
>> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Load Warehouse failed ..."
  else
    echo "Load Warehouse done." `date` "\n" >> $TRACEFILE
    echo "Load Warehouse done ..."
  fi
fi
if torun
then
  echo "Load District ..."
  echo "Load District ..." `date` "\n" >> $TRACEFILE
  ./step21loaddist.sh
  if test $? -ne 0
  then
    echo "Load District failed." `date` "\n" >> $TRACEFILE
    echo "Look at log/step21loaddist.log for more details." `date` "\n"
>> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Load District failed ..."
  else
    echo "Load District done." `date` "\n" >> $TRACEFILE
    echo "Load District done ..."
  fi
fi
if torun
then
  echo "Load Item ..."
  echo "Load Item ..." `date` "\n" >> $TRACEFILE
  ./step22loaditem.sh
  if test $? -ne 0
  then
    echo "Load Item failed." `date` "\n" >> $TRACEFILE
    echo "Look at log/step22loaditem.log for more details." `date` "\n"
>> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Load Item failed ..."

```

```

else
  echo "Load Item done." `date` "\n" >> $TRACEFILE
  echo "Load Item done ..."
fi
fi
if torun
then
  echo "Load History ..."
  echo "Load History ..." `date` "\n" >> $TRACEFILE
  ./step23loadhist.sh
  if test $? -ne 0
  then
    echo "Load History failed." `date` "\n" >> $TRACEFILE
    echo "Look at log/step23loadhist*.log for more details." `date`
"\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Load History failed ..."
  else
    echo "Load History done." `date` "\n" >> $TRACEFILE
    echo "Load History done ..."
  fi
fi
if torun
then
  echo "Load Neworder ..."
  echo "Load Neworder ..." `date` "\n" >> $TRACEFILE
  ./step24loadnord.sh
  if test $? -ne 0
  then
    echo "Load Neworder failed." `date` "\n" >> $TRACEFILE
    echo "Look at log/step24loadnord*.log for more details." `date`
"\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Load Neworder failed ..."
  else
    echo "Load Neworder done." `date` "\n" >> $TRACEFILE
    echo "Load Neworder done ..."
  fi
fi
if torun
then
  echo "Load Order/Orderline ..."
  echo "Load Order/Orderline ..." `date` "\n" >> $TRACEFILE
  ./step25loadordrordl.sh
  if test $? -ne 0
  then
    echo "Load Order/Orderline failed." `date` "\n" >> $TRACEFILE
    echo "Look at log/step25loadordrordl*.log for more details." `date`
"\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Load Order/Orderline failed ..."
  else
    echo "Load Order/Orderline done." `date` "\n" >> $TRACEFILE
    echo "Load Order/Orderline done ..."
  fi
fi
if torun
then
  echo "Load Customer ..."

```

```

echo "Load Customer ..." `date` "\n" >> $TRACEFILE
./step26loadcust.sh
if test $? -ne 0
then
  echo "Load Customer failed." `date` "\n" >> $TRACEFILE
  echo "Look at log/step26loadcust*.log for more details." `date`
"\n" >> $TRACEFILE
  echo "Stopped" >> $STOPFILE
  echo "Load Customer failed ..."
else
  echo "Load Customer done." `date` "\n" >> $TRACEFILE
  echo "Load Customer done ..."
fi
fi
if torun
then
  echo "Load Stock ..."
  echo "Load Stock ..." `date` "\n" >> $TRACEFILE
  ./step27loadstok.sh
  if test $? -ne 0
  then
    echo "Load Stock failed." `date` "\n" >> $TRACEFILE
    echo "Look at log/step27loadstok*.log for more details." `date`
"\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Load Stock failed ..."
  else
    echo "Load Stock done." `date` "\n" >> $TRACEFILE
    echo "Load Stock done ..."
  fi
fi
if torun
then
  echo "Create Warehouse Index ..."
  echo "Create Warehouse Index ..." `date` "\n" >> $TRACEFILE
  ./step29createiware.sh
  if test $? -ne 0
  then
    echo "Create Warehouse Index failed." `date` "\n" >> $TRACEFILE
    echo "Look at log/step29createiware.log for more details." `date`
"\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create Warehouse Index failed ..."
  else
    echo "Create Warehouse Index done." `date` "\n" >> $TRACEFILE
    echo "Create Warehouse Index done ..."
  fi
fi
if torun
then
  echo "Create District Index ..."
  echo "Create District Index ..." `date` "\n" >> $TRACEFILE
  ./step30createidist.sh
  if test $? -ne 0
  then
    echo "Create District Index failed." `date` "\n" >> $TRACEFILE
    echo "Look at log/step30createidist.log for more details." `date`
"\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create District Index failed ..."

```

```

else
  echo "Create District Index done." `date` "\n" >> $TRACEFILE
  echo "Create District Index done ..."
fi
fi
if torun
then
  echo "Create Item Index ..."
  echo "Create Item Index ..." `date` "\n" >> $TRACEFILE
  ./step31createiitem.sh
  if test $? -ne 0
  then
    echo "Create Item Index failed." `date` "\n" >> $TRACEFILE
    echo "Look at log/step31createiitem.log for more details." `date`
"\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create Item Index failed ..."
  else
    echo "Create Item Index done." `date` "\n" >> $TRACEFILE
    echo "Create Item Index done ..."
  fi
fi
if torun
then
  echo "Create Customer1 Index ..."
  echo "Create Customer1 Index ..." `date` "\n" >> $TRACEFILE
  ./step32createicust1.sh
  if test $? -ne 0
  then
    echo "Create Customer1 Index failed." `date` "\n" >> $TRACEFILE
    echo "Look at log/step32createicust1.log for more details." `date`
"\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create Customer1 Index failed ..."
  else
    echo "Create Customer1 Index done." `date` "\n" >> $TRACEFILE
    echo "Create Customer1 Index done ..."
  fi
fi
if torun
then
  echo "Create Customer2 Index ..."
  echo "Create Customer2 Index ..." `date` "\n" >> $TRACEFILE
  ./step33createicust2.sh
  if test $? -ne 0
  then
    echo "Create Customer2 Index failed." `date` "\n" >> $TRACEFILE
    echo "Look at log/step33createicust2.log for more details." `date`
"\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create Customer2 Index failed ..."
  else
    echo "Create Customer2 Index done." `date` "\n" >> $TRACEFILE
    echo "Create Customer2 Index done ..."
  fi
fi
if torun
then
  echo "Create Stock Index ..."

```

```

echo "Create Stock Index ..." `date` "\n" >> $TRACEFILE
./step34createistok.sh
if test $? -ne 0
then
  echo "Create Stock Index failed." `date` "\n" >> $TRACEFILE
  echo "Look at log/step34createistok.log for more details." `date`
"\n" >> $TRACEFILE
  echo "Stopped" >> $STOPFILE
  echo "Create Stock Index failed ..."
else
  echo "Create Stock Index done." `date` "\n" >> $TRACEFILE
  echo "Create Stock Index done ..."
fi
fi
if torun
then
  echo "Create Order1 Index ..."
  echo "Create Order1 Index ..." `date` "\n" >> $TRACEFILE
  ./step35createiordr1.sh
  if test $? -ne 0
  then
    echo "Create Order1 Index failed." `date` "\n" >> $TRACEFILE
    echo "Look at log/step35createiordr1.log for more details." `date`
"\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create Order1 Index failed ..."
  else
    echo "Create Order1 Index done." `date` "\n" >> $TRACEFILE
    echo "Create Order1 Index done ..."
  fi
fi
if torun
then
  echo "Create Order2 Index ..."
  echo "Create Order2 Index ..." `date` "\n" >> $TRACEFILE
  ./step36createiordr2.sh
  if test $? -ne 0
  then
    echo "Create Order2 Index failed." `date` "\n" >> $TRACEFILE
    echo "Look at log/step36createiordr2.log for more details." `date`
"\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create Order2 Index failed ..."
  else
    echo "Create Order2 Index done." `date` "\n" >> $TRACEFILE
    echo "Create Order2 Index done ..."
  fi
fi
if torun
then
  echo "Create Neworder Index ..."
  echo "Create Neworder Index ..." `date` "\n" >> $TRACEFILE
  ./step37createinord.sh
  if test $? -ne 0
  then
    echo "Create Neworder Index failed." `date` "\n" >> $TRACEFILE
    echo "Look at log/step37createinord.log for more details." `date`
"\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create Neworder Index failed ..."

```

```

else
  echo "Create Neworder Index done." `date` "\n" >> $TRACEFILE
  echo "Create Neworder Index done ..."
fi
fi
if torun
then
  echo "Create Orderline Index ..."
  echo "Create Orderline Index ..." `date` "\n" >> $TRACEFILE
  ./step38createiordl.sh
  if test $? -ne 0
  then
    echo "Create Orderline Index failed." `date` "\n" >> $TRACEFILE
    echo "Look at log/step38createiordl.log for more details." `date`
"\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create Orderline Index failed ..."
  else
    echo "Create Orderline Index done." `date` "\n" >> $TRACEFILE
    echo "Create Orderline Index done ..."
  fi
fi
if torun
then
  echo "Analyze Tables/Clusters/Indexes ..."
  echo "Analyze Tables/Clusters/Indexes ..." `date` "\n" >> $TRACEFILE
  ./step39analyze.sh
  if test $? -ne 0
  then
    echo "Analyze Tables/Clusters/Indexes failed." `date` "\n" >>
$TRACEFILE
    echo "Look at log/step39analyze.log for more details." `date` "\n"
>> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Analyze Tables/Clusters/Indexes failed ..."
  else
    echo "Analyze Tables/Clusters/Indexes done." `date` "\n" >>
$TRACEFILE
    echo "Analyze Tables/Clusters/Indexes done ..."
  fi
fi
if torun
then
  echo "Create Statistics Tables ..."
  echo "Create Statistics Tables ..." `date` "\n" >> $TRACEFILE
  ./step40createstats.sh
  if test $? -ne 0
  then
    echo "Create Statistics Tables failed." `date` "\n" >> $TRACEFILE
    echo "Look at log/step40createstats.log for more details." `date`
"\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create Statistics Tables failed ..."
  else
    echo "Create Statistics Tables done." `date` "\n" >> $TRACEFILE
    echo "Create Statistics Tables done ..."
  fi
fi
if torun

```

```

then
  echo "Load Stored Procedures ..."
  echo "Load Stored Procedures ..." `date` "\n" >> $TRACEFILE
  ./step41createstoredprocs.sh
  if test $? -ne 0
  then
    echo "Load Stored Procedures failed." `date` "\n" >> $TRACEFILE
    echo "Look at log/step41createstoredprocs.log for more details."
`date` "\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Load Stored Procedures failed ..."
  else
    echo "Load Stored Procedures done." `date` "\n" >> $TRACEFILE
    echo "Load Stored Procedures done ..."
  fi
fi
if torun
then
  echo "Create Rollback Segments for Runs ..."
  echo "Create Rollback Segments for Runs ..." `date` "\n" >>
$TRACEFILE
  ./step18createrollsegs.sh
  if test $? -ne 0
  then
    echo "Create Rollback Segments for Runs failed." `date` "\n" >>
$TRACEFILE
    echo "Look at log/step18createrollsegs.log for more details."
`date` "\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create Rollback Segments for Runs failed ..."
  else
    echo "Create Rollback Segments for Runs done." `date` "\n" >>
$TRACEFILE
    echo "Create Rollback Segments for Runs done ..."
  fi
fi
if torun
then
  echo "Generate Space Reports ..."
  echo "Generate Space Reports ..." `date` "\n" >> $TRACEFILE
  ./step42createspacestats.sh
  if test $? -ne 0
  then
    echo "Generate Space Reports failed." `date` "\n" >> $TRACEFILE
    echo "Look at log/step42createspacestats.log for more details."
`date` "\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Generate Space Reports failed ..."
  else
    echo "Generate Space Reports done." `date` "\n" >> $TRACEFILE
    echo "Generate Space Reports done ..."
  fi
fi
if torun
then
  echo "Misc ..."
  echo "Misc ..." `date` "\n" >> $TRACEFILE
  ./step43createmisc.sh
  if test $? -ne 0
  then

```

```

    echo "Misc failed." `date` "\n" >> $TRACEFILE
    echo "Look at log/step43createmisc.log for more details." `date`
"\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Misc failed ..."
    else
    echo "Misc done." `date` "\n" >> $TRACEFILE
    echo "Misc done ..."
    fi
fi

if torun
then
    echo "Offline Rollback Segments for Build ..."
    echo "Offline Rollback Segments for Build ..." `date` "\n" >>
$TRACEFILE
    ./step47offlinerollsegs.sh
    if test $? -ne 0
    then
        echo "Offline Rollback Segments for Build failed." `date` "\n" >>
$TRACEFILE
        echo "Look at log/step47offlinerollsegs.log for more details."
`date` "\n" >> $TRACEFILE
        echo "Stopped" >> $STOPFILE
        echo "Offline Rollback Segments for Build failed ..."
    else
        echo "Offline Rollback Segments for Build done." `date` "\n" >>
$TRACEFILE
        echo "Offline Rollback Segments for Build done ..."
    fi
fi

torun

```

addfile.sh

```

#!/sh

$SQLPLUS tpcc/tpcc <<!
    spool step5addfile_$.log
    set echo on
    alter tablespace $1 add datafile '$2' size $3 reuse;
    set echo off
    spool off
    exit sql.sqlcode;
!

```

addtempfile.sh

```

#!/sh

$SQLPLUS tpcc/tpcc <<!
    spool step5addfile_$.log
    set echo on
    alter tablespace $1 add tempfile '$2' size $3 reuse;
    set echo off
    spool off
    exit sql.sqlcode;
!

```

addtempts.sh

```

#!/sh

$SQLPLUS tpcc/tpcc <<!
    spool step5createts_$.log
    set echo on
    drop tablespace $1 including contents;
    create temporary tablespace $1 tempfile '$2' size $3 reuse extent
management local uniform size $4;
    set echo off
    spool off
    exit sql.sqlcode;
!

```

addts.sh

```

#!/sh

$SQLPLUS tpcc/tpcc <<!
    spool step5createts_$.log
    set echo on
    drop tablespace $1 including contents;
    create tablespace $1 datafile '$2' size $3 reuse extent management
local uniform size $4 nologging ;
    set echo off
    spool off
    exit sql.sqlcode;
!

```

startb.sh

```

#!/sh

$SQLPLUS internal/internal @stepstartb > junk 2>&1

if test $? -eq 0
then
    exit 1;
else
    exit 0;
fi

```

stepstart.b.sql

```

spool stepstartb.log;

set echo on;

startup pfile=p_build.ora open;

set echo off;
spool off;

exit sql.sqlcode;

```

step1tpccenv.sh

```
#!/sh
SQLDBA=svrmgr1
export SQLDBA
SQLPLUS=sqplus
export SQLPLUS
TPCCLOAD=tpccload.exe
export TPCCLOAD
ORACLE_SID=tpcc
export ORACLE_SID
```

step2createdb.sh

```
#!/sh
$SQLPLUS internal/internal @step2createdb > junk 2>&1

if test $? -ne 0
then
    exit 1;
else
    exit 0;
fi
```

step2createdb.sql

```
spool step2createdb.log

set echo on

startup pfile=p_create.ora nomount
create database tpcc
    controlfile reuse
    maxdatafiles 1000
    maxinstances 4
    datafile '\\.\system_001' size 505M reuse
    logfile group 1 ('\\.\log_111') size 22528M reuse,
    group 2 ('\\.\log_121') size 22528M reuse;
alter database add logfile thread 2 group 3 ('\\.\log_231') size 22528M
reuse,
    group 4 ('\\.\log_241') size 22528M reuse;
alter database enable public thread 2;
alter database add logfile thread 3 group 5 ('\\.\log_351') size 22528M
reuse,
    group 6 ('\\.\log_361') size 22528M reuse;
alter database enable public thread 3;
alter database add logfile thread 4 group 7 ('\\.\log_471') size 22528M
reuse,
    group 8 ('\\.\log_481') size 22528M reuse;
alter database enable public thread 4;
spool off
set echo off
exit sql.sqlcode
```

step3createrollback.sh

```
#!/sh
$SQLPLUS internal/internal @step3createrollback > junk 2>&1

if test $? -ne 0
then
    exit 1;
else
    exit 0;
fi
```

step3createrollback.sql

```
spool step3createrollback.log
set echo on

alter rollback segment t1 offline;
drop public rollback segment t1;
create public rollback segment t1
    storage (initial 200K minextents 2 next 200K);
alter rollback segment t1 online;
alter rollback segment t2 offline;
drop public rollback segment t2;
create public rollback segment t2
    storage (initial 200K minextents 2 next 200K);
alter rollback segment t2 online;
alter rollback segment t3 offline;
drop public rollback segment t3;
create public rollback segment t3
    storage (initial 200K minextents 2 next 200K);
alter rollback segment t3 online;
alter rollback segment t4 offline;
drop public rollback segment t4;
create public rollback segment t4
    storage (initial 200K minextents 2 next 200K);
alter rollback segment t4 online;
alter rollback segment t5 offline;
drop public rollback segment t5;
create public rollback segment t5
    storage (initial 200K minextents 2 next 200K);
alter rollback segment t5 online;
alter rollback segment t6 offline;
drop public rollback segment t6;
create public rollback segment t6
    storage (initial 200K minextents 2 next 200K);
alter rollback segment t6 online;
alter rollback segment t7 offline;
drop public rollback segment t7;
create public rollback segment t7
    storage (initial 200K minextents 2 next 200K);
alter rollback segment t7 online;
alter rollback segment t8 offline;
drop public rollback segment t8;
create public rollback segment t8
    storage (initial 200K minextents 2 next 200K);
alter rollback segment t8 online;
alter rollback segment t9 offline;
drop public rollback segment t9;
create public rollback segment t9
    storage (initial 200K minextents 2 next 200K);
alter rollback segment t9 online;
```

```

alter rollback segment t10 offline;
drop public rollback segment t10;
create public rollback segment t10
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t10 online;
alter rollback segment t11 offline;
drop public rollback segment t11;
create public rollback segment t11
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t11 online;
alter rollback segment t12 offline;
drop public rollback segment t12;
create public rollback segment t12
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t12 online;
alter rollback segment t13 offline;
drop public rollback segment t13;
create public rollback segment t13
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t13 online;
alter rollback segment t14 offline;
drop public rollback segment t14;
create public rollback segment t14
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t14 online;
alter rollback segment t15 offline;
drop public rollback segment t15;
create public rollback segment t15
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t15 online;
alter rollback segment t16 offline;
drop public rollback segment t16;
create public rollback segment t16
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t16 online;
alter rollback segment t17 offline;
drop public rollback segment t17;
create public rollback segment t17
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t17 online;
alter rollback segment t18 offline;
drop public rollback segment t18;
create public rollback segment t18
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t18 online;
alter rollback segment t19 offline;
drop public rollback segment t19;
create public rollback segment t19
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t19 online;
alter rollback segment t20 offline;
drop public rollback segment t20;
create public rollback segment t20
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t20 online;
alter rollback segment t21 offline;
drop public rollback segment t21;
create public rollback segment t21
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t21 online;
alter rollback segment t22 offline;
drop public rollback segment t22;
create public rollback segment t22

```

```

  storage (initial 200K minextents 2 next 200K);
alter rollback segment t22 online;
alter rollback segment t23 offline;
drop public rollback segment t23;
create public rollback segment t23
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t23 online;
alter rollback segment t24 offline;
drop public rollback segment t24;
create public rollback segment t24
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t24 online;
alter rollback segment t25 offline;
drop public rollback segment t25;
create public rollback segment t25
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t25 online;
alter rollback segment t26 offline;
drop public rollback segment t26;
create public rollback segment t26
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t26 online;
alter rollback segment t27 offline;
drop public rollback segment t27;
create public rollback segment t27
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t27 online;
alter rollback segment t28 offline;
drop public rollback segment t28;
create public rollback segment t28
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t28 online;
alter rollback segment t29 offline;
drop public rollback segment t29;
create public rollback segment t29
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t29 online;
alter rollback segment t30 offline;
drop public rollback segment t30;
create public rollback segment t30
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t30 online;

spool off
set echo off
exit sql.sqlcode

```

step5createts.sh

```

#!/sh
procsum=0
addts.sh stok_0 \\\.\stok_0_0 2350M 2005308K > junk1 2>&1 &
procl=$!
addts.sh istok_0 \\\.\istok_0_0 3060M 10240K > junk2 2>&1 &
proc2=$!
addts.sh ware_0 \\\.\ware_0_0 10M 210K > junk3 2>&1 &
proc3=$!
addts.sh item_0 \\\.\item_0_0 10M 282K > junk7 2>&1 &
proc7=$!
addts.sh ordl_0 \\\.\ordl_0_0 4640M 5120K > junk8 2>&1 &

```

```

proc8=$!
wait $proc1
proc1=$?
procsum=`expr $procsum + $proc1`
wait $proc2
proc2=$?
procsum=`expr $procsum + $proc2`
wait $proc3
proc3=$?
procsum=`expr $procsum + $proc3`
wait $proc7
proc7=$?
procsum=`expr $procsum + $proc7`
wait $proc8
proc8=$?
procsum=`expr $procsum + $proc8`

adts.sh ordl_1 \\\\.\\ordl_1_0 4640M 5120K > junk9 2>&1 &
proc9=$!
adts.sh ordl_2 \\\\.\\ordl_2_0 4640M 5120K > junk10 2>&1 &
proc10=$!
adts.sh ordl_3 \\\\.\\ordl_3_0 4640M 5120K > junk11 2>&1 &
proc11=$!
adts.sh iordr1_0 \\\\.\\iordr1_0_0 220M 5120K > junk12 2>&1 &
proc12=$!
adts.sh iordr1_1 \\\\.\\iordr1_1_0 220M 5120K > junk13 2>&1 &
proc13=$!
adts.sh iordr1_2 \\\\.\\iordr1_2_0 220M 5120K > junk14 2>&1 &
proc14=$!
adts.sh iordr1_3 \\\\.\\iordr1_3_0 220M 5120K > junk15 2>&1 &
proc15=$!
adts.sh iordr2_0 \\\\.\\iordr2_0_0 360M 5120K > junk16 2>&1 &
proc16=$!
wait $proc9
proc9=$?
procsum=`expr $procsum + $proc9`
wait $proc10
proc10=$?
procsum=`expr $procsum + $proc10`
wait $proc11
proc11=$?
procsum=`expr $procsum + $proc11`
wait $proc12
proc12=$?
procsum=`expr $procsum + $proc12`
wait $proc13
proc13=$?
procsum=`expr $procsum + $proc13`
wait $proc14
proc14=$?
procsum=`expr $procsum + $proc14`
wait $proc15
proc15=$?
procsum=`expr $procsum + $proc15`
wait $proc16
proc16=$?
procsum=`expr $procsum + $proc16`

adts.sh iordr2_1 \\\\.\\iordr2_1_0 360M 5120K > junk17 2>&1 &
proc17=$!
adts.sh iordr2_2 \\\\.\\iordr2_2_0 360M 5120K > junk18 2>&1 &
proc18=$!
adts.sh iordr2_3 \\\\.\\iordr2_3_0 360M 5120K > junk19 2>&1 &

```

```

proc19=$!
adts.sh nord_0 \\\\.\\nord_0_0 50M 5120K > junk20 2>&1 &
proc20=$!
adts.sh nord_1 \\\\.\\nord_1_0 50M 5120K > junk21 2>&1 &
proc21=$!
adts.sh nord_2 \\\\.\\nord_2_0 50M 5120K > junk22 2>&1 &
proc22=$!
adts.sh nord_3 \\\\.\\nord_3_0 50M 5120K > junk23 2>&1 &
proc23=$!
adts.sh dist_0 \\\\.\\dist_0_0 10M 2100K > junk24 2>&1 &
proc24=$!
wait $proc17
proc17=$?
procsum=`expr $procsum + $proc17`
wait $proc18
proc18=$?
procsum=`expr $procsum + $proc18`
wait $proc19
proc19=$?
procsum=`expr $procsum + $proc19`
wait $proc20
proc20=$?
procsum=`expr $procsum + $proc20`
wait $proc21
proc21=$?
procsum=`expr $procsum + $proc21`
wait $proc22
proc22=$?
procsum=`expr $procsum + $proc22`
wait $proc23
proc23=$?
procsum=`expr $procsum + $proc23`
wait $proc24
proc24=$?
procsum=`expr $procsum + $proc24`

adtemptps.sh temp_0 \\\\.\\temp_0_0 500M 25M > junk28 2>&1 &
proc28=$!
adts.sh ordr_0 \\\\.\\ordr_0_0 370M 1024K > junk29 2>&1 &
proc29=$!
adts.sh ordr_1 \\\\.\\ordr_1_0 370M 1024K > junk30 2>&1 &
proc30=$!
adts.sh ordr_2 \\\\.\\ordr_2_0 370M 1024K > junk31 2>&1 &
proc31=$!
adts.sh ordr_3 \\\\.\\ordr_3_0 370M 1024K > junk32 2>&1 &
proc32=$!
wait $proc28
proc28=$?
procsum=`expr $procsum + $proc28`
wait $proc29
proc29=$?
procsum=`expr $procsum + $proc29`
wait $proc30
proc30=$?
procsum=`expr $procsum + $proc30`
wait $proc31
proc31=$?
procsum=`expr $procsum + $proc31`
wait $proc32
proc32=$?
procsum=`expr $procsum + $proc32`

adts.sh hist_0 \\\\.\\hist_0_0 530M 10240K > junk33 2>&1 &

```

```

proc33=$!
addts.sh hist_1 \\\\.\\hist_1_0 530M 10240K > junk34 2>&1 &
proc34=$!
addts.sh hist_2 \\\\.\\hist_2_0 530M 10240K > junk35 2>&1 &
proc35=$!
addts.sh hist_3 \\\\.\\hist_3_0 530M 10240K > junk36 2>&1 &
proc36=$!
addts.sh iitem_0 \\\\.\\iitem_0_0 10M 624K > junk37 2>&1 &
proc37=$!
addts.sh iware_0 \\\\.\\iware_0_0 20M 512K > junk38 2>&1 &
proc38=$!
wait $proc33
proc33=$?
procsum=`expr $procsum + $proc33`
wait $proc34
proc34=$?
procsum=`expr $procsum + $proc34`
wait $proc35
proc35=$?
procsum=`expr $procsum + $proc35`
wait $proc36
proc36=$?
procsum=`expr $procsum + $proc36`
wait $proc37
proc37=$?
procsum=`expr $procsum + $proc37`
wait $proc38
proc38=$?
procsum=`expr $procsum + $proc38`
wait $proc39
proc39=$?
procsum=`expr $procsum + $proc39`
wait $proc40
proc40=$?
procsum=`expr $procsum + $proc40`

addts.sh icust1_0 \\\\.\\icust1_0_0 1000M 10240K > junk42 2>&1 &
proc42=$!
addts.sh icust2_0 \\\\.\\icust2_0_0 240M 10240K > junk43 2>&1 &
proc43=$!
addts.sh iord1_0 \\\\.\\iord1_0_0 8025M 10240K > junk44 2>&1 &
proc44=$!
addts.sh inord_0 \\\\.\\inord_0_0 708M 10240K > junk45 2>&1 &
proc45=$!
addts.sh idist_0 \\\\.\\idist_0_0 30M 1024K > junk46 2>&1 &
proc46=$!
addts.sh cust_0 \\\\.\\cust_0_0 1780M 1512200K > junk50 2>&1 &
proc50=$!
wait $proc41
proc41=$?
procsum=`expr $procsum + $proc41`
wait $proc42
proc42=$?
procsum=`expr $procsum + $proc42`
wait $proc43
proc43=$?
procsum=`expr $procsum + $proc43`
wait $proc44
proc44=$?
procsum=`expr $procsum + $proc44`
wait $proc45
proc45=$?
procsum=`expr $procsum + $proc45`

```

```

wait $proc46
proc46=$?
procsum=`expr $procsum + $proc46`
wait $proc47
proc47=$?
procsum=`expr $procsum + $proc47`
wait $proc48
proc48=$?
procsum=`expr $procsum + $proc48`
wait $proc49
proc49=$?
procsum=`expr $procsum + $proc49`
wait $proc50
proc50=$?
procsum=`expr $procsum + $proc50`

addfile.sh iware_0 \\\\.\\iware_1_0 20M > junk39 2>&1 &
proc39=$!
addfile.sh iware_0 \\\\.\\iware_2_0 20M > junk40 2>&1 &
proc40=$!
addfile.sh iware_0 \\\\.\\iware_3_0 20M > junk41 2>&1 &
proc41=$!
addfile.sh idist_0 \\\\.\\idist_1_0 30M > junk47 2>&1 &
proc47=$!
addfile.sh idist_0 \\\\.\\idist_2_0 30M > junk48 2>&1 &
proc48=$!
addfile.sh idist_0 \\\\.\\idist_3_0 30M > junk49 2>&1 &
proc49=$!
addfile.sh stok_0 \\\\.\\stok_0_1 2350M > junk51 2>&1 &
proc51=$!
addfile.sh stok_0 \\\\.\\stok_0_2 2350M > junk52 2>&1 &
proc52=$!
addfile.sh stok_0 \\\\.\\stok_0_3 2350M > junk53 2>&1 &
proc53=$!
addfile.sh stok_0 \\\\.\\stok_0_4 2350M > junk54 2>&1 &
proc54=$!
addfile.sh stok_0 \\\\.\\stok_0_5 2350M > junk55 2>&1 &
proc55=$!
addfile.sh stok_0 \\\\.\\stok_0_6 2350M > junk56 2>&1 &
proc56=$!
addfile.sh stok_0 \\\\.\\stok_0_7 2350M > junk57 2>&1 &
proc57=$!
addfile.sh stok_0 \\\\.\\stok_0_8 2350M > junk58 2>&1 &
proc58=$!
addfile.sh stok_0 \\\\.\\stok_0_9 2350M > junk59 2>&1 &
proc59=$!
addfile.sh stok_0 \\\\.\\stok_0_10 2350M > junk60 2>&1 &
proc60=$!
addfile.sh stok_0 \\\\.\\stok_0_11 2350M > junk61 2>&1 &
proc61=$!
addfile.sh stok_0 \\\\.\\stok_0_12 2350M > junk62 2>&1 &
proc62=$!
addfile.sh stok_0 \\\\.\\stok_0_13 2350M > junk63 2>&1 &
proc63=$!
addfile.sh stok_0 \\\\.\\stok_0_14 2350M > junk64 2>&1 &
proc64=$!
addfile.sh stok_0 \\\\.\\stok_0_15 2350M > junk65 2>&1 &
proc65=$!
addfile.sh stok_0 \\\\.\\stok_0_16 2350M > junk66 2>&1 &
proc66=$!
addfile.sh stok_0 \\\\.\\stok_0_17 2350M > junk67 2>&1 &
proc67=$!
addfile.sh stok_0 \\\\.\\stok_0_18 2350M > junk68 2>&1 &

```

```

proc68=$!
addfile.sh stok_0 \\\\.\\stok_0_19 2350M > junk69 2>&1 &
proc69=$!
addfile.sh stok_0 \\\\.\\stok_0_20 2350M > junk70 2>&1 &
proc70=$!
addfile.sh stok_0 \\\\.\\stok_0_21 2350M > junk71 2>&1 &
proc71=$!
addfile.sh stok_0 \\\\.\\stok_0_22 2350M > junk72 2>&1 &
proc72=$!
addfile.sh stok_0 \\\\.\\stok_0_23 2350M > junk73 2>&1 &
proc73=$!
addfile.sh stok_0 \\\\.\\stok_0_24 2350M > junk74 2>&1 &
proc74=$!
addfile.sh stok_0 \\\\.\\stok_0_25 2350M > junk75 2>&1 &
proc75=$!
addfile.sh stok_0 \\\\.\\stok_0_26 2350M > junk76 2>&1 &
proc76=$!
addfile.sh stok_0 \\\\.\\stok_0_27 2350M > junk77 2>&1 &
proc77=$!
addfile.sh stok_0 \\\\.\\stok_0_28 2350M > junk78 2>&1 &
proc78=$!
addfile.sh stok_0 \\\\.\\stok_0_29 2350M > junk79 2>&1 &
proc79=$!
addfile.sh stok_0 \\\\.\\stok_0_30 2350M > junk80 2>&1 &
proc80=$!
addfile.sh stok_0 \\\\.\\stok_0_31 2350M > junk81 2>&1 &
proc81=$!
addfile.sh stok_0 \\\\.\\stok_0_32 2350M > junk82 2>&1 &
proc82=$!
addfile.sh stok_0 \\\\.\\stok_0_33 2350M > junk83 2>&1 &
proc83=$!
addfile.sh stok_0 \\\\.\\stok_0_34 2350M > junk84 2>&1 &
proc84=$!
addfile.sh stok_0 \\\\.\\stok_0_35 2350M > junk85 2>&1 &
proc85=$!
addfile.sh stok_0 \\\\.\\stok_0_36 2350M > junk86 2>&1 &
proc86=$!
addfile.sh stok_0 \\\\.\\stok_0_37 2350M > junk87 2>&1 &
proc87=$!
addfile.sh stok_0 \\\\.\\stok_0_38 2350M > junk88 2>&1 &
proc88=$!
addfile.sh stok_0 \\\\.\\stok_0_39 2350M > junk89 2>&1 &
proc89=$!
addfile.sh stok_0 \\\\.\\stok_0_40 2350M > junk90 2>&1 &
proc90=$!
addfile.sh stok_0 \\\\.\\stok_0_41 2350M > junk91 2>&1 &
proc91=$!
addfile.sh stok_0 \\\\.\\stok_0_42 2350M > junk92 2>&1 &
proc92=$!
addfile.sh stok_0 \\\\.\\stok_0_43 2350M > junk93 2>&1 &
proc93=$!
addfile.sh stok_0 \\\\.\\stok_0_44 2350M > junk94 2>&1 &
proc94=$!
addfile.sh stok_0 \\\\.\\stok_0_45 2350M > junk95 2>&1 &
proc95=$!
addfile.sh stok_0 \\\\.\\stok_0_46 2350M > junk96 2>&1 &
proc96=$!
addfile.sh stok_0 \\\\.\\stok_0_47 2350M > junk97 2>&1 &
proc97=$!

wait $proc39
proc39=$?
procsum=`expr $procsum + $proc39`

```

```

wait $proc40
proc40=$?
procsum=`expr $procsum + $proc40`
wait $proc41
proc41=$?
procsum=`expr $procsum + $proc41`
wait $proc47
proc47=$?
procsum=`expr $procsum + $proc47`
wait $proc48
proc48=$?
procsum=`expr $procsum + $proc48`
wait $proc49
proc49=$?
procsum=`expr $procsum + $proc49`
wait $proc51
proc51=$?
procsum=`expr $procsum + $proc51`
wait $proc52
proc52=$?
procsum=`expr $procsum + $proc52`
wait $proc53
proc53=$?
procsum=`expr $procsum + $proc53`
wait $proc54
proc54=$?
procsum=`expr $procsum + $proc54`
wait $proc55
proc55=$?
procsum=`expr $procsum + $proc55`
wait $proc56
proc56=$?
procsum=`expr $procsum + $proc56`
wait $proc57
proc57=$?
procsum=`expr $procsum + $proc57`
wait $proc58
proc58=$?
procsum=`expr $procsum + $proc58`

wait $proc59
proc59=$?
procsum=`expr $procsum + $proc59`
wait $proc60
proc60=$?
procsum=`expr $procsum + $proc60`
wait $proc61
proc61=$?
procsum=`expr $procsum + $proc61`
wait $proc62
proc62=$?
procsum=`expr $procsum + $proc62`
wait $proc63
proc63=$?
procsum=`expr $procsum + $proc63`
wait $proc64
proc64=$?
procsum=`expr $procsum + $proc64`
wait $proc65
proc65=$?
procsum=`expr $procsum + $proc65`
wait $proc66
proc66=$?

```

```

procsum=`expr $procsum + $proc66`
wait $proc67
proc67=$?
procsum=`expr $procsum + $proc67`
wait $proc68
proc68=$?
procsum=`expr $procsum + $proc68`
wait $proc69
proc69=$?
procsum=`expr $procsum + $proc69`
wait $proc70
proc70=$?
procsum=`expr $procsum + $proc70`
wait $proc71
proc71=$?
procsum=`expr $procsum + $proc71`
wait $proc72
proc72=$?
procsum=`expr $procsum + $proc72`
wait $proc73
proc73=$?
procsum=`expr $procsum + $proc73`
wait $proc74
proc74=$?
procsum=`expr $procsum + $proc74`

wait $proc75
proc75=$?
procsum=`expr $procsum + $proc75`
wait $proc76
proc76=$?
procsum=`expr $procsum + $proc76`
wait $proc77
proc77=$?
procsum=`expr $procsum + $proc77`
wait $proc78
proc78=$?
procsum=`expr $procsum + $proc78`
wait $proc79
proc79=$?
procsum=`expr $procsum + $proc79`
wait $proc80
proc80=$?
procsum=`expr $procsum + $proc80`
wait $proc81
proc81=$?
procsum=`expr $procsum + $proc81`
wait $proc82
proc82=$?
procsum=`expr $procsum + $proc82`

wait $proc83
proc83=$?
procsum=`expr $procsum + $proc83`
wait $proc84
proc84=$?
procsum=`expr $procsum + $proc84`
wait $proc85
proc85=$?
procsum=`expr $procsum + $proc85`
wait $proc86
proc86=$?

```

```

procsum=`expr $procsum + $proc86`
wait $proc87
proc87=$?
procsum=`expr $procsum + $proc87`
wait $proc88
proc88=$?
procsum=`expr $procsum + $proc88`
wait $proc89
proc89=$?
procsum=`expr $procsum + $proc89`
wait $proc90
proc90=$?
procsum=`expr $procsum + $proc90`
wait $proc91
proc91=$?
procsum=`expr $procsum + $proc91`
wait $proc92
proc92=$?
procsum=`expr $procsum + $proc92`
wait $proc93
proc93=$?
procsum=`expr $procsum + $proc93`
wait $proc94
proc94=$?
procsum=`expr $procsum + $proc94`
wait $proc95
proc95=$?
procsum=`expr $procsum + $proc95`
wait $proc96
proc96=$?
procsum=`expr $procsum + $proc96`
wait $proc97
proc97=$?
procsum=`expr $procsum + $proc97`

addfile.sh stok_0 \\\\.\\stok_0_48 2350M > junk98 2>&1 &
proc98=$!
addfile.sh stok_0 \\\\.\\stok_0_49 2350M > junk99 2>&1 &
proc99=$!
addfile.sh stok_0 \\\\.\\stok_0_50 2350M > junk100 2>&1 &
proc100=$!
addfile.sh stok_0 \\\\.\\stok_0_51 2350M > junk101 2>&1 &
proc101=$!
addfile.sh stok_0 \\\\.\\stok_0_52 2350M > junk102 2>&1 &
proc102=$!
addfile.sh stok_0 \\\\.\\stok_0_53 2350M > junk103 2>&1 &
proc103=$!
addfile.sh stok_0 \\\\.\\stok_0_54 2350M > junk104 2>&1 &
proc104=$!
addfile.sh stok_0 \\\\.\\stok_0_55 2350M > junk105 2>&1 &
proc105=$!
addfile.sh stok_0 \\\\.\\stok_0_56 2350M > junk106 2>&1 &
proc106=$!
addfile.sh stok_0 \\\\.\\stok_0_57 2350M > junk107 2>&1 &
proc107=$!
addfile.sh stok_0 \\\\.\\stok_0_58 2350M > junk108 2>&1 &
proc108=$!
addfile.sh stok_0 \\\\.\\stok_0_59 2350M > junk109 2>&1 &
proc109=$!
addfile.sh stok_0 \\\\.\\stok_0_60 2350M > junk110 2>&1 &
proc110=$!
addfile.sh stok_0 \\\\.\\stok_0_61 2350M > junk111 2>&1 &
proc111=$!

```

```

addfile.sh stok_0 \\\\.\\stok_0_62 2350M > junk112 2>&1 &
proc112=$!
addfile.sh stok_0 \\\\.\\stok_0_63 2350M > junk113 2>&1 &
proc113=$!
addfile.sh stok_0 \\\\.\\stok_0_64 2350M > junk114 2>&1 &
proc114=$!
addfile.sh stok_0 \\\\.\\stok_0_65 2350M > junk115 2>&1 &
proc115=$!
addfile.sh stok_0 \\\\.\\stok_0_66 2350M > junk116 2>&1 &
proc116=$!
addfile.sh stok_0 \\\\.\\stok_0_67 2350M > junk117 2>&1 &
proc117=$!
addfile.sh stok_0 \\\\.\\stok_0_68 2350M > junk118 2>&1 &
proc118=$!
addfile.sh stok_0 \\\\.\\stok_0_69 2350M > junk119 2>&1 &
proc119=$!
addfile.sh stok_0 \\\\.\\stok_0_70 2350M > junk120 2>&1 &
proc120=$!
addfile.sh stok_0 \\\\.\\stok_0_71 2350M > junk121 2>&1 &
proc121=$!
addfile.sh stok_0 \\\\.\\stok_0_72 2350M > junk122 2>&1 &
proc122=$!
addfile.sh stok_0 \\\\.\\stok_0_73 2350M > junk123 2>&1 &
proc123=$!
addfile.sh stok_0 \\\\.\\stok_0_74 2350M > junk124 2>&1 &
proc124=$!
addfile.sh stok_0 \\\\.\\stok_0_75 2350M > junk125 2>&1 &
proc125=$!
addfile.sh stok_0 \\\\.\\stok_0_76 2350M > junk126 2>&1 &
proc126=$!
addfile.sh stok_0 \\\\.\\stok_0_77 2350M > junk127 2>&1 &
proc127=$!
addfile.sh stok_0 \\\\.\\stok_0_78 2350M > junk128 2>&1 &
proc128=$!
addfile.sh stok_0 \\\\.\\stok_0_79 2350M > junk129 2>&1 &
proc129=$!
addfile.sh stok_0 \\\\.\\stok_0_80 2350M > junk130 2>&1 &
proc130=$!
addfile.sh stok_0 \\\\.\\stok_0_81 2350M > junk131 2>&1 &
proc131=$!
addfile.sh stok_0 \\\\.\\stok_0_82 2350M > junk132 2>&1 &
proc132=$!
addfile.sh stok_0 \\\\.\\stok_0_83 2350M > junk133 2>&1 &
proc133=$!
addfile.sh stok_0 \\\\.\\stok_0_84 2350M > junk134 2>&1 &
proc134=$!
addfile.sh stok_0 \\\\.\\stok_0_85 2350M > junk135 2>&1 &
proc135=$!
addfile.sh stok_0 \\\\.\\stok_0_86 2350M > junk136 2>&1 &
proc136=$!
addfile.sh stok_0 \\\\.\\stok_0_87 2350M > junk137 2>&1 &
proc137=$!
addfile.sh stok_0 \\\\.\\stok_0_88 2350M > junk138 2>&1 &
proc138=$!
addfile.sh stok_0 \\\\.\\stok_0_89 2350M > junk139 2>&1 &
proc139=$!
addfile.sh stok_0 \\\\.\\stok_0_90 2350M > junk140 2>&1 &
proc140=$!
addfile.sh stok_0 \\\\.\\stok_0_91 2350M > junk141 2>&1 &
proc141=$!
addfile.sh stok_0 \\\\.\\stok_0_92 2350M > junk142 2>&1 &
proc142=$!
addfile.sh stok_0 \\\\.\\stok_0_93 2350M > junk143 2>&1 &

```

```

proc143=$!
addfile.sh stok_0 \\\\.\\stok_0_94 2350M > junk144 2>&1 &
proc144=$!
addfile.sh stok_0 \\\\.\\stok_0_95 2350M > junk145 2>&1 &
proc145=$!

wait $proc98
proc98=$?
procsum=`expr $procsum + $proc98`

wait $proc99
proc99=$?
procsum=`expr $procsum + $proc99`
wait $proc100
proc100=$?
procsum=`expr $procsum + $proc100`
wait $proc101
proc101=$?
procsum=`expr $procsum + $proc101`
wait $proc102
proc102=$?
procsum=`expr $procsum + $proc102`
wait $proc103
proc103=$?
procsum=`expr $procsum + $proc103`
wait $proc104
proc104=$?
procsum=`expr $procsum + $proc104`
wait $proc105
proc105=$?
procsum=`expr $procsum + $proc105`
wait $proc106
proc106=$?
procsum=`expr $procsum + $proc106`

wait $proc107
proc107=$?
procsum=`expr $procsum + $proc107`
wait $proc108
proc108=$?
procsum=`expr $procsum + $proc108`
wait $proc109
proc109=$?
procsum=`expr $procsum + $proc109`
wait $proc110
proc110=$?
procsum=`expr $procsum + $proc110`
wait $proc111
proc111=$?
procsum=`expr $procsum + $proc111`
wait $proc112
proc112=$?
procsum=`expr $procsum + $proc112`
wait $proc113
proc113=$?
procsum=`expr $procsum + $proc113`
wait $proc114
proc114=$?
procsum=`expr $procsum + $proc114`

wait $proc115
proc115=$?
procsum=`expr $procsum + $proc115`

```

```

wait $proc116
proc116=$?
procsum=`expr $procsum + $proc116`
wait $proc117
proc117=$?
procsum=`expr $procsum + $proc117`
wait $proc118
proc118=$?
procsum=`expr $procsum + $proc118`
wait $proc119
proc119=$?
procsum=`expr $procsum + $proc119`
wait $proc120
proc120=$?
procsum=`expr $procsum + $proc120`
wait $proc121
proc121=$?
procsum=`expr $procsum + $proc121`
wait $proc122
proc122=$?
procsum=`expr $procsum + $proc122`

wait $proc123
proc123=$?
procsum=`expr $procsum + $proc123`
wait $proc124
proc124=$?
procsum=`expr $procsum + $proc124`
wait $proc125
proc125=$?
procsum=`expr $procsum + $proc125`
wait $proc126
proc126=$?
procsum=`expr $procsum + $proc126`
wait $proc127
proc127=$?
procsum=`expr $procsum + $proc127`
wait $proc128
proc128=$?
procsum=`expr $procsum + $proc128`
wait $proc129
proc129=$?
procsum=`expr $procsum + $proc129`
wait $proc130
proc130=$?
procsum=`expr $procsum + $proc130`

wait $proc131
proc131=$?
procsum=`expr $procsum + $proc131`
wait $proc132
proc132=$?
procsum=`expr $procsum + $proc132`
wait $proc133
proc133=$?
procsum=`expr $procsum + $proc133`
wait $proc134
proc134=$?
procsum=`expr $procsum + $proc134`
wait $proc135
proc135=$?
procsum=`expr $procsum + $proc135`
wait $proc136

```

```

proc136=$?
procsum=`expr $procsum + $proc136`
wait $proc137
proc137=$?
procsum=`expr $procsum + $proc137`
wait $proc138
proc138=$?
procsum=`expr $procsum + $proc138`

wait $proc139
proc139=$?
procsum=`expr $procsum + $proc139`
wait $proc140
proc140=$?
procsum=`expr $procsum + $proc140`
wait $proc141
proc141=$?
procsum=`expr $procsum + $proc141`
wait $proc142
proc142=$?
procsum=`expr $procsum + $proc142`
wait $proc143
proc143=$?
procsum=`expr $procsum + $proc143`
wait $proc144
proc144=$?
procsum=`expr $procsum + $proc144`
wait $proc145
proc145=$?
procsum=`expr $procsum + $proc145`

```

```

addfile.sh istok_0 \\\\.\\istok_0_1 3060M > junk194 2>&1 &
proc194=$!
addfile.sh istok_0 \\\\.\\istok_0_2 3060M > junk195 2>&1 &
proc195=$!
addfile.sh istok_0 \\\\.\\istok_0_3 3060M > junk196 2>&1 &
proc196=$!
addfile.sh item_0 \\\\.\\item_0_1 10M > junk197 2>&1 &
proc197=$!
addfile.sh item_0 \\\\.\\item_0_2 10M > junk198 2>&1 &
proc198=$!
addfile.sh item_0 \\\\.\\item_0_3 10M > junk199 2>&1 &
proc199=$!
addfile.sh item_0 \\\\.\\item_0_4 10M > junk200 2>&1 &
proc200=$!
addfile.sh item_0 \\\\.\\item_0_5 10M > junk201 2>&1 &
proc201=$!
addfile.sh item_0 \\\\.\\item_0_6 10M > junk202 2>&1 &
proc202=$!
wait $proc194
proc194=$?
procsum=`expr $procsum + $proc194`
wait $proc195
proc195=$?
procsum=`expr $procsum + $proc195`
wait $proc196
proc196=$?
procsum=`expr $procsum + $proc196`
wait $proc197
proc197=$?
procsum=`expr $procsum + $proc197`
wait $proc198

```

```

proc198=$?
procsum=`expr $procsum + $proc198`
wait $proc199
proc199=$?
procsum=`expr $procsum + $proc199`
wait $proc200
proc200=$?
procsum=`expr $procsum + $proc200`
wait $proc201
proc201=$?
procsum=`expr $procsum + $proc201`
wait $proc202
proc202=$?
procsum=`expr $procsum + $proc202`

addfile.sh item_0 \\\item_0_7 10M > junk203 2>&1 &
proc203=$!
addfile.sh item_0 \\\item_0_8 10M > junk204 2>&1 &
proc204=$!
addfile.sh item_0 \\\item_0_9 10M > junk205 2>&1 &
proc205=$!
addfile.sh item_0 \\\item_0_10 10M > junk206 2>&1 &
proc206=$!
addfile.sh item_0 \\\item_0_11 10M > junk207 2>&1 &
proc207=$!
addfile.sh item_0 \\\item_0_12 10M > junk208 2>&1 &
proc208=$!
addfile.sh item_0 \\\item_0_13 10M > junk209 2>&1 &
proc209=$!
addfile.sh item_0 \\\item_0_14 10M > junk210 2>&1 &
proc210=$!
wait $proc203
proc203=$?
procsum=`expr $procsum + $proc203`
wait $proc204
proc204=$?
procsum=`expr $procsum + $proc204`
wait $proc205
proc205=$?
procsum=`expr $procsum + $proc205`
wait $proc206
proc206=$?
procsum=`expr $procsum + $proc206`
wait $proc207
proc207=$?
procsum=`expr $procsum + $proc207`
wait $proc208
proc208=$?
procsum=`expr $procsum + $proc208`
wait $proc209
proc209=$?
procsum=`expr $procsum + $proc209`
wait $proc210
proc210=$?
procsum=`expr $procsum + $proc210`

addfile.sh item_0 \\\item_0_15 10M > junk211 2>&1 &
proc211=$!
addfile.sh item_0 \\\item_0_16 10M > junk212 2>&1 &
proc212=$!
addfile.sh item_0 \\\item_0_17 10M > junk213 2>&1 &
proc213=$!
addfile.sh item_0 \\\item_0_18 10M > junk214 2>&1 &

```

```

proc214=$!
addfile.sh item_0 \\\item_0_19 10M > junk215 2>&1 &
proc215=$!
addfile.sh item_0 \\\item_0_20 10M > junk216 2>&1 &
proc216=$!
addfile.sh item_0 \\\item_0_21 10M > junk217 2>&1 &
proc217=$!
addfile.sh item_0 \\\item_0_22 10M > junk218 2>&1 &
proc218=$!
wait $proc211
proc211=$?
procsum=`expr $procsum + $proc211`
wait $proc212
proc212=$?
procsum=`expr $procsum + $proc212`
wait $proc213
proc213=$?
procsum=`expr $procsum + $proc213`
wait $proc214
proc214=$?
procsum=`expr $procsum + $proc214`
wait $proc215
proc215=$?
procsum=`expr $procsum + $proc215`
wait $proc216
proc216=$?
procsum=`expr $procsum + $proc216`
wait $proc217
proc217=$?
procsum=`expr $procsum + $proc217`
wait $proc218
proc218=$?
procsum=`expr $procsum + $proc218`

addfile.sh item_0 \\\item_0_23 10M > junk219 2>&1 &
proc219=$!
addfile.sh item_0 \\\item_0_24 10M > junk220 2>&1 &
proc220=$!
addfile.sh item_0 \\\item_0_25 10M > junk221 2>&1 &
proc221=$!
addfile.sh item_0 \\\item_0_26 10M > junk222 2>&1 &
proc222=$!
addfile.sh item_0 \\\item_0_27 10M > junk223 2>&1 &
proc223=$!
addfile.sh item_0 \\\item_0_28 10M > junk224 2>&1 &
proc224=$!
addfile.sh item_0 \\\item_0_29 10M > junk225 2>&1 &
proc225=$!
addfile.sh item_0 \\\item_0_30 10M > junk226 2>&1 &
proc226=$!
wait $proc219
proc219=$?
procsum=`expr $procsum + $proc219`
wait $proc220
proc220=$?
procsum=`expr $procsum + $proc220`
wait $proc221
proc221=$?
procsum=`expr $procsum + $proc221`
wait $proc222
proc222=$?
procsum=`expr $procsum + $proc222`
wait $proc223

```

```

proc223=$?
procsum=`expr $procsum + $proc223`
wait $proc224
proc224=$?
procsum=`expr $procsum + $proc224`
wait $proc225
proc225=$?
procsum=`expr $procsum + $proc225`
wait $proc226
proc226=$?
procsum=`expr $procsum + $proc226`

addfile.sh item_0 \\\item_0_31 10M > junk227 2>&1 &
proc227=$!
addfile.sh item_0 \\\item_0_32 10M > junk228 2>&1 &
proc228=$!
addfile.sh item_0 \\\item_0_33 10M > junk229 2>&1 &
proc229=$!
addfile.sh item_0 \\\item_0_34 10M > junk230 2>&1 &
proc230=$!
addfile.sh item_0 \\\item_0_35 10M > junk231 2>&1 &
proc231=$!
addfile.sh item_0 \\\item_0_36 10M > junk232 2>&1 &
proc232=$!
addfile.sh item_0 \\\item_0_37 10M > junk233 2>&1 &
proc233=$!
addfile.sh item_0 \\\item_0_38 10M > junk234 2>&1 &
proc234=$!
wait $proc227
proc227=$?
procsum=`expr $procsum + $proc227`
wait $proc228
proc228=$?
procsum=`expr $procsum + $proc228`
wait $proc229
proc229=$?
procsum=`expr $procsum + $proc229`
wait $proc230
proc230=$?
procsum=`expr $procsum + $proc230`
wait $proc231
proc231=$?
procsum=`expr $procsum + $proc231`
wait $proc232
proc232=$?
procsum=`expr $procsum + $proc232`
wait $proc233
proc233=$?
procsum=`expr $procsum + $proc233`
wait $proc234
proc234=$?
procsum=`expr $procsum + $proc234`

addfile.sh item_0 \\\item_0_39 10M > junk235 2>&1 &
proc235=$!
addfile.sh item_0 \\\item_0_40 10M > junk236 2>&1 &
proc236=$!
addfile.sh item_0 \\\item_0_41 10M > junk237 2>&1 &
proc237=$!
addfile.sh item_0 \\\item_0_42 10M > junk238 2>&1 &
proc238=$!
addfile.sh item_0 \\\item_0_43 10M > junk239 2>&1 &
proc239=$!

```

```

addfile.sh item_0 \\\item_0_44 10M > junk240 2>&1 &
proc240=$!
addfile.sh item_0 \\\item_0_45 10M > junk241 2>&1 &
proc241=$!
addfile.sh item_0 \\\item_0_46 10M > junk242 2>&1 &
proc242=$!
addfile.sh item_0 \\\item_0_47 10M > junk243 2>&1 &
proc243=$!
wait $proc235
proc235=$?
procsum=`expr $procsum + $proc235`
wait $proc236
proc236=$?
procsum=`expr $procsum + $proc236`
wait $proc237
proc237=$?
procsum=`expr $procsum + $proc237`
wait $proc238
proc238=$?
procsum=`expr $procsum + $proc238`
wait $proc239
proc239=$?
procsum=`expr $procsum + $proc239`
wait $proc240
proc240=$?
procsum=`expr $procsum + $proc240`
wait $proc241
proc241=$?
procsum=`expr $procsum + $proc241`
wait $proc242
proc242=$?
procsum=`expr $procsum + $proc242`
wait $proc243
proc243=$?
procsum=`expr $procsum + $proc243`

addfile.sh iitem_0 \\\iitem_0_1 10M > junk279 2>&1 &
proc279=$!
addfile.sh iitem_0 \\\iitem_0_2 10M > junk280 2>&1 &
proc280=$!
addfile.sh iitem_0 \\\iitem_0_3 10M > junk281 2>&1 &
proc281=$!
addfile.sh icust1_0 \\\icust1_0_1 1000M > junk282 2>&1 &
proc282=$!
wait $proc279
proc279=$?
procsum=`expr $procsum + $proc279`
wait $proc280
proc280=$?
procsum=`expr $procsum + $proc280`
wait $proc281
proc281=$?
procsum=`expr $procsum + $proc281`
wait $proc282
proc282=$?
procsum=`expr $procsum + $proc282`

addfile.sh icust1_0 \\\icust1_0_2 1000M > junk283 2>&1 &
proc283=$!
addfile.sh icust1_0 \\\icust1_0_3 1000M > junk284 2>&1 &
proc284=$!
addfile.sh iordl_0 \\\iordl_0_1 8025M > junk288 2>&1 &
proc288=$!

```

```

addfile.sh iordl_0 \\\.\iordl_0_2 8025M > junk289 2>&1 &
proc289=$!
addfile.sh iordl_0 \\\.\iordl_0_3 8025M > junk290 2>&1 &
proc290=$!
wait $proc283
proc283=$?
procsum=`expr $procsum + $proc283`
wait $proc284
proc284=$?
procsum=`expr $procsum + $proc284`
wait $proc288
proc288=$?
procsum=`expr $procsum + $proc288`
wait $proc289
proc289=$?
procsum=`expr $procsum + $proc289`
wait $proc290
proc290=$?
procsum=`expr $procsum + $proc290`

addfile.sh iordl_0 \\\.\iordl_0_4 8025M > junk291 2>&1 &
proc291=$!
addfile.sh iordl_0 \\\.\iordl_0_5 8025M > junk292 2>&1 &
proc292=$!
addfile.sh iordl_0 \\\.\iordl_0_6 8025M > junk293 2>&1 &
proc293=$!
addfile.sh iordl_0 \\\.\iordl_0_7 8025M > junk294 2>&1 &
proc294=$!
addfile.sh iordl_0 \\\.\iordl_0_8 8025M > junk295 2>&1 &
proc295=$!
addfile.sh iordl_0 \\\.\iordl_0_9 8025M > junk296 2>&1 &
proc296=$!
addfile.sh iordl_0 \\\.\iordl_0_10 8025M > junk297 2>&1 &
proc297=$!
addfile.sh iordl_0 \\\.\iordl_0_11 8025M > junk298 2>&1 &
proc298=$!
wait $proc291
proc291=$?
procsum=`expr $procsum + $proc291`
wait $proc292
proc292=$?
procsum=`expr $procsum + $proc292`
wait $proc293
proc293=$?
procsum=`expr $procsum + $proc293`
wait $proc294
proc294=$?
procsum=`expr $procsum + $proc294`
wait $proc295
proc295=$?
procsum=`expr $procsum + $proc295`
wait $proc296
proc296=$?
procsum=`expr $procsum + $proc296`
wait $proc297
proc297=$?
procsum=`expr $procsum + $proc297`
wait $proc298
proc298=$?
procsum=`expr $procsum + $proc298`

addfile.sh inord_0 \\\.\inord_0_1 708M > junk299 2>&1 &
proc299=$!

```

```

addfile.sh inord_0 \\\.\inord_0_2 708M > junk300 2>&1 &
proc300=$!
addfile.sh inord_0 \\\.\inord_0_3 708M > junk301 2>&1 &
proc301=$!
wait $proc299
proc299=$?
procsum=`expr $procsum + $proc299`
wait $proc300
proc300=$?
procsum=`expr $procsum + $proc300`
wait $proc301
proc301=$?
procsum=`expr $procsum + $proc301`

addfile.sh cust_0 \\\.\cust_0_1 1780M > junk398 2>&1 &
proc398=$!
addfile.sh cust_0 \\\.\cust_0_2 1780M > junk399 2>&1 &
proc399=$!
addfile.sh cust_0 \\\.\cust_0_3 1780M > junk400 2>&1 &
proc400=$!
addfile.sh cust_0 \\\.\cust_0_4 1780M > junk401 2>&1 &
proc401=$!
addfile.sh cust_0 \\\.\cust_0_5 1780M > junk402 2>&1 &
proc402=$!
addfile.sh cust_0 \\\.\cust_0_6 1780M > junk403 2>&1 &
proc403=$!
addfile.sh cust_0 \\\.\cust_0_7 1780M > junk404 2>&1 &
proc404=$!
addfile.sh cust_0 \\\.\cust_0_8 1780M > junk405 2>&1 &
proc405=$!
addfile.sh cust_0 \\\.\cust_0_9 1780M > junk406 2>&1 &
proc406=$!
addfile.sh cust_0 \\\.\cust_0_10 1780M > junk407 2>&1 &
proc407=$!
addfile.sh cust_0 \\\.\cust_0_11 1780M > junk408 2>&1 &
proc408=$!
addfile.sh cust_0 \\\.\cust_0_12 1780M > junk409 2>&1 &
proc409=$!
addfile.sh cust_0 \\\.\cust_0_13 1780M > junk410 2>&1 &
proc410=$!
addfile.sh cust_0 \\\.\cust_0_14 1780M > junk411 2>&1 &
proc411=$!
addfile.sh cust_0 \\\.\cust_0_15 1780M > junk412 2>&1 &
proc412=$!
addfile.sh cust_0 \\\.\cust_0_16 1780M > junk413 2>&1 &
proc413=$!
addfile.sh cust_0 \\\.\cust_0_17 1780M > junk414 2>&1 &
proc414=$!
addfile.sh cust_0 \\\.\cust_0_18 1780M > junk415 2>&1 &
proc415=$!
addfile.sh cust_0 \\\.\cust_0_19 1780M > junk416 2>&1 &
proc416=$!
addfile.sh cust_0 \\\.\cust_0_20 1780M > junk417 2>&1 &
proc417=$!
addfile.sh cust_0 \\\.\cust_0_21 1780M > junk418 2>&1 &
proc418=$!
addfile.sh cust_0 \\\.\cust_0_22 1780M > junk419 2>&1 &
proc419=$!
addfile.sh cust_0 \\\.\cust_0_23 1780M > junk420 2>&1 &
proc420=$!
addfile.sh cust_0 \\\.\cust_0_24 1780M > junk421 2>&1 &
proc421=$!
addfile.sh cust_0 \\\.\cust_0_25 1780M > junk422 2>&1 &

```

```

proc422=$!
addfile.sh cust_0 \\\\.\\cust_0_26 1780M > junk423 2>&1 &
proc423=$!
addfile.sh cust_0 \\\\.\\cust_0_27 1780M > junk424 2>&1 &
proc424=$!
addfile.sh cust_0 \\\\.\\cust_0_28 1780M > junk425 2>&1 &
proc425=$!
addfile.sh cust_0 \\\\.\\cust_0_29 1780M > junk426 2>&1 &
proc426=$!
addfile.sh cust_0 \\\\.\\cust_0_30 1780M > junk427 2>&1 &
proc427=$!
addfile.sh cust_0 \\\\.\\cust_0_31 1780M > junk428 2>&1 &
proc428=$!
addfile.sh cust_0 \\\\.\\cust_0_32 1780M > junk429 2>&1 &
proc429=$!
addfile.sh cust_0 \\\\.\\cust_0_33 1780M > junk430 2>&1 &
proc430=$!
addfile.sh cust_0 \\\\.\\cust_0_34 1780M > junk431 2>&1 &
proc431=$!
addfile.sh cust_0 \\\\.\\cust_0_35 1780M > junk432 2>&1 &
proc432=$!
addfile.sh cust_0 \\\\.\\cust_0_36 1780M > junk433 2>&1 &
proc433=$!
addfile.sh cust_0 \\\\.\\cust_0_37 1780M > junk434 2>&1 &
proc434=$!
addfile.sh cust_0 \\\\.\\cust_0_38 1780M > junk435 2>&1 &
proc435=$!
addfile.sh cust_0 \\\\.\\cust_0_39 1780M > junk436 2>&1 &
proc436=$!
addfile.sh cust_0 \\\\.\\cust_0_40 1780M > junk437 2>&1 &
proc437=$!
addfile.sh cust_0 \\\\.\\cust_0_41 1780M > junk438 2>&1 &
proc438=$!
addfile.sh cust_0 \\\\.\\cust_0_42 1780M > junk439 2>&1 &
proc439=$!
addfile.sh cust_0 \\\\.\\cust_0_43 1780M > junk440 2>&1 &
proc440=$!
addfile.sh cust_0 \\\\.\\cust_0_44 1780M > junk441 2>&1 &
proc441=$!
addfile.sh cust_0 \\\\.\\cust_0_45 1780M > junk442 2>&1 &
proc442=$!
addfile.sh cust_0 \\\\.\\cust_0_46 1780M > junk443 2>&1 &
proc443=$!
addfile.sh cust_0 \\\\.\\cust_0_47 1780M > junk444 2>&1 &
proc444=$!

```

```

wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`

```

```
wait $proc403
```

```

proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`
wait $proc409
proc409=$?
procsum=`expr $procsum + $proc409`
wait $proc410
proc410=$?
procsum=`expr $procsum + $proc410`

```

```

wait $proc411
proc411=$?
procsum=`expr $procsum + $proc411`
wait $proc412
proc412=$?
procsum=`expr $procsum + $proc412`
wait $proc413
proc413=$?
procsum=`expr $procsum + $proc413`
wait $proc414
proc414=$?
procsum=`expr $procsum + $proc414`
wait $proc415
proc415=$?
procsum=`expr $procsum + $proc415`
wait $proc416
proc416=$?
procsum=`expr $procsum + $proc416`
wait $proc417
proc417=$?
procsum=`expr $procsum + $proc417`
wait $proc418
proc418=$?
procsum=`expr $procsum + $proc418`

```

```

wait $proc419
proc419=$?
procsum=`expr $procsum + $proc419`
wait $proc420
proc420=$?
procsum=`expr $procsum + $proc420`
wait $proc421
proc421=$?
procsum=`expr $procsum + $proc421`
wait $proc422
proc422=$?
procsum=`expr $procsum + $proc422`
wait $proc423
proc423=$?

```

```

procsum=`expr $procsum + $proc423`
wait $proc424
proc424=$?
procsum=`expr $procsum + $proc424`
wait $proc425
proc425=$?
procsum=`expr $procsum + $proc425`
wait $proc426
proc426=$?
procsum=`expr $procsum + $proc426`

wait $proc427
proc427=$?
procsum=`expr $procsum + $proc427`
wait $proc428
proc428=$?
procsum=`expr $procsum + $proc428`
wait $proc429
proc429=$?
procsum=`expr $procsum + $proc429`
wait $proc430
proc430=$?
procsum=`expr $procsum + $proc430`
wait $proc431
proc431=$?
procsum=`expr $procsum + $proc431`
wait $proc432
proc432=$?
procsum=`expr $procsum + $proc432`
wait $proc433
proc433=$?
procsum=`expr $procsum + $proc433`
wait $proc434
proc434=$?
procsum=`expr $procsum + $proc434`

wait $proc435
proc435=$?
procsum=`expr $procsum + $proc435`
wait $proc436
proc436=$?
procsum=`expr $procsum + $proc436`
wait $proc437
proc437=$?
procsum=`expr $procsum + $proc437`
wait $proc438
proc438=$?
procsum=`expr $procsum + $proc438`
wait $proc439
proc439=$?
procsum=`expr $procsum + $proc439`
wait $proc440
proc440=$?
procsum=`expr $procsum + $proc440`
wait $proc441
proc441=$?
procsum=`expr $procsum + $proc441`
wait $proc442
proc442=$?
procsum=`expr $procsum + $proc442`
wait $proc443
proc443=$?
procsum=`expr $procsum + $proc443`

```

```

wait $proc444
proc444=$?
procsum=`expr $procsum + $proc444`

addfile.sh cust_0 \\\\.\\cust_0_48 1780M > junk445 2>&1 &
proc445=$!
addfile.sh cust_0 \\\\.\\cust_0_49 1780M > junk446 2>&1 &
proc446=$!
addfile.sh cust_0 \\\\.\\cust_0_50 1780M > junk447 2>&1 &
proc447=$!
addfile.sh cust_0 \\\\.\\cust_0_51 1780M > junk448 2>&1 &
proc448=$!
addfile.sh cust_0 \\\\.\\cust_0_52 1780M > junk449 2>&1 &
proc449=$!
addfile.sh cust_0 \\\\.\\cust_0_53 1780M > junk450 2>&1 &
proc450=$!
addfile.sh cust_0 \\\\.\\cust_0_54 1780M > junk451 2>&1 &
proc451=$!
addfile.sh cust_0 \\\\.\\cust_0_55 1780M > junk452 2>&1 &
proc452=$!
addfile.sh cust_0 \\\\.\\cust_0_56 1780M > junk453 2>&1 &
proc453=$!
addfile.sh cust_0 \\\\.\\cust_0_57 1780M > junk454 2>&1 &
proc454=$!
addfile.sh cust_0 \\\\.\\cust_0_58 1780M > junk455 2>&1 &
proc455=$!
addfile.sh cust_0 \\\\.\\cust_0_59 1780M > junk456 2>&1 &
proc456=$!
addfile.sh cust_0 \\\\.\\cust_0_60 1780M > junk457 2>&1 &
proc457=$!
addfile.sh cust_0 \\\\.\\cust_0_61 1780M > junk458 2>&1 &
proc458=$!
addfile.sh cust_0 \\\\.\\cust_0_62 1780M > junk459 2>&1 &
proc459=$!
addfile.sh cust_0 \\\\.\\cust_0_63 1780M > junk460 2>&1 &
proc460=$!
addfile.sh cust_0 \\\\.\\cust_0_64 1780M > junk461 2>&1 &
proc461=$!
addfile.sh cust_0 \\\\.\\cust_0_65 1780M > junk462 2>&1 &
proc462=$!
addfile.sh cust_0 \\\\.\\cust_0_66 1780M > junk463 2>&1 &
proc463=$!
addfile.sh cust_0 \\\\.\\cust_0_67 1780M > junk464 2>&1 &
proc464=$!
addfile.sh cust_0 \\\\.\\cust_0_68 1780M > junk465 2>&1 &
proc465=$!
addfile.sh cust_0 \\\\.\\cust_0_69 1780M > junk466 2>&1 &
proc466=$!
addfile.sh cust_0 \\\\.\\cust_0_70 1780M > junk467 2>&1 &
proc467=$!
addfile.sh cust_0 \\\\.\\cust_0_71 1780M > junk468 2>&1 &
proc468=$!
addfile.sh cust_0 \\\\.\\cust_0_72 1780M > junk469 2>&1 &
proc469=$!
addfile.sh cust_0 \\\\.\\cust_0_73 1780M > junk470 2>&1 &
proc470=$!
addfile.sh cust_0 \\\\.\\cust_0_74 1780M > junk471 2>&1 &
proc471=$!
addfile.sh cust_0 \\\\.\\cust_0_75 1780M > junk472 2>&1 &
proc472=$!
addfile.sh cust_0 \\\\.\\cust_0_76 1780M > junk473 2>&1 &
proc473=$!
addfile.sh cust_0 \\\\.\\cust_0_77 1780M > junk474 2>&1 &

```

```

proc474=$!
addfile.sh cust_0 \\\\.\\cust_0_78 1780M > junk475 2>&1 &
proc475=$!
addfile.sh cust_0 \\\\.\\cust_0_79 1780M > junk476 2>&1 &
proc476=$!
addfile.sh cust_0 \\\\.\\cust_0_80 1780M > junk477 2>&1 &
proc477=$!
addfile.sh cust_0 \\\\.\\cust_0_81 1780M > junk478 2>&1 &
proc478=$!
addfile.sh cust_0 \\\\.\\cust_0_82 1780M > junk479 2>&1 &
proc479=$!
addfile.sh cust_0 \\\\.\\cust_0_83 1780M > junk480 2>&1 &
proc480=$!
addfile.sh cust_0 \\\\.\\cust_0_84 1780M > junk481 2>&1 &
proc481=$!
addfile.sh cust_0 \\\\.\\cust_0_85 1780M > junk482 2>&1 &
proc482=$!
addfile.sh cust_0 \\\\.\\cust_0_86 1780M > junk483 2>&1 &
proc483=$!
addfile.sh cust_0 \\\\.\\cust_0_87 1780M > junk484 2>&1 &
proc484=$!
addfile.sh cust_0 \\\\.\\cust_0_88 1780M > junk485 2>&1 &
proc485=$!
addfile.sh cust_0 \\\\.\\cust_0_89 1780M > junk486 2>&1 &
proc486=$!
addfile.sh cust_0 \\\\.\\cust_0_90 1780M > junk487 2>&1 &
proc487=$!
addfile.sh cust_0 \\\\.\\cust_0_91 1780M > junk488 2>&1 &
proc488=$!
addfile.sh cust_0 \\\\.\\cust_0_92 1780M > junk489 2>&1 &
proc489=$!
addfile.sh cust_0 \\\\.\\cust_0_93 1780M > junk490 2>&1 &
proc490=$!
addfile.sh cust_0 \\\\.\\cust_0_94 1780M > junk491 2>&1 &
proc491=$!
addfile.sh cust_0 \\\\.\\cust_0_95 1780M > junk492 2>&1 &
proc492=$!

```

```

wait $proc445
proc445=$?
procsum=`expr $procsum + $proc445`
wait $proc446
proc446=$?
procsum=`expr $procsum + $proc446`
wait $proc447
proc447=$?
procsum=`expr $procsum + $proc447`
wait $proc448
proc448=$?
procsum=`expr $procsum + $proc448`
wait $proc449
proc449=$?
procsum=`expr $procsum + $proc449`
wait $proc450
proc450=$?
procsum=`expr $procsum + $proc450`
wait $proc451
proc451=$?
procsum=`expr $procsum + $proc451`
wait $proc452
proc452=$?
procsum=`expr $procsum + $proc452`

```

```

wait $proc453
proc453=$?
procsum=`expr $procsum + $proc453`
wait $proc454
proc454=$?
procsum=`expr $procsum + $proc454`
wait $proc455
proc455=$?
procsum=`expr $procsum + $proc455`
wait $proc456
proc456=$?
procsum=`expr $procsum + $proc456`
wait $proc457
proc457=$?
procsum=`expr $procsum + $proc457`
wait $proc458
proc458=$?
procsum=`expr $procsum + $proc458`
wait $proc459
proc459=$?
procsum=`expr $procsum + $proc459`
wait $proc460
proc460=$?
procsum=`expr $procsum + $proc460`
wait $proc461
proc461=$?
procsum=`expr $procsum + $proc461`
wait $proc462
proc462=$?
procsum=`expr $procsum + $proc462`
wait $proc463
proc463=$?
procsum=`expr $procsum + $proc463`
wait $proc464
proc464=$?
procsum=`expr $procsum + $proc464`
wait $proc465
proc465=$?
procsum=`expr $procsum + $proc465`
wait $proc466
proc466=$?
procsum=`expr $procsum + $proc466`
wait $proc467
proc467=$?
procsum=`expr $procsum + $proc467`
wait $proc468
proc468=$?
procsum=`expr $procsum + $proc468`
wait $proc469
proc469=$?
procsum=`expr $procsum + $proc469`
wait $proc470
proc470=$?
procsum=`expr $procsum + $proc470`
wait $proc471
proc471=$?
procsum=`expr $procsum + $proc471`
wait $proc472
proc472=$?
procsum=`expr $procsum + $proc472`
wait $proc473
proc473=$?
procsum=`expr $procsum + $proc473`

```

```

wait $proc474
proc474=$?
procsum=`expr $procsum + $proc474`
wait $proc475
proc475=$?
procsum=`expr $procsum + $proc475`
wait $proc476
proc476=$?
procsum=`expr $procsum + $proc476`
wait $proc477
proc477=$?
procsum=`expr $procsum + $proc477`
wait $proc478
proc478=$?
procsum=`expr $procsum + $proc478`
wait $proc479
proc479=$?
procsum=`expr $procsum + $proc479`
wait $proc480
proc480=$?
procsum=`expr $procsum + $proc480`
wait $proc481
proc481=$?
procsum=`expr $procsum + $proc481`
wait $proc482
proc482=$?
procsum=`expr $procsum + $proc482`
wait $proc483
proc483=$?
procsum=`expr $procsum + $proc483`
wait $proc484
proc484=$?
procsum=`expr $procsum + $proc484`
wait $proc485
proc485=$?
procsum=`expr $procsum + $proc485`
wait $proc486
proc486=$?
procsum=`expr $procsum + $proc486`
wait $proc487
proc487=$?
procsum=`expr $procsum + $proc487`
wait $proc488
proc488=$?
procsum=`expr $procsum + $proc488`
wait $proc489
proc489=$?
procsum=`expr $procsum + $proc489`
wait $proc490
proc490=$?
procsum=`expr $procsum + $proc490`
wait $proc491
proc491=$?
procsum=`expr $procsum + $proc491`
wait $proc492
proc492=$?
procsum=`expr $procsum + $proc492`

addfile.sh iorder2_0 \\\\.\\iorder2_0_1 360M > junk398 2>&1 &
proc398=$!
addfile.sh iorder2_0 \\\\.\\iorder2_0_2 360M > junk399 2>&1 &
proc399=$!
addfile.sh iorder2_0 \\\\.\\iorder2_0_3 360M > junk400 2>&1 &

```

```

proc400=$!
addfile.sh iorder2_0 \\\\.\\iorder2_0_4 360M > junk401 2>&1 &
proc401=$!
addfile.sh iorder2_0 \\\\.\\iorder2_0_5 360M > junk402 2>&1 &
proc402=$!
addfile.sh iorder2_0 \\\\.\\iorder2_0_6 360M > junk403 2>&1 &
proc403=$!
addfile.sh iorder2_0 \\\\.\\iorder2_0_7 360M > junk404 2>&1 &
proc404=$!
addfile.sh iorder2_0 \\\\.\\iorder2_0_8 360M > junk405 2>&1 &
proc405=$!
addfile.sh iorder2_0 \\\\.\\iorder2_0_9 360M > junk406 2>&1 &
proc406=$!
addfile.sh iorder2_0 \\\\.\\iorder2_0_10 360M > junk407 2>&1 &
proc407=$!
addfile.sh iorder2_0 \\\\.\\iorder2_0_11 360M > junk408 2>&1 &
proc408=$!
wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

addfile.sh iorder2_1 \\\\.\\iorder2_1_1 360M > junk398 2>&1 &
proc398=$!
addfile.sh iorder2_1 \\\\.\\iorder2_1_2 360M > junk399 2>&1 &
proc399=$!
addfile.sh iorder2_1 \\\\.\\iorder2_1_3 360M > junk400 2>&1 &
proc400=$!
addfile.sh iorder2_1 \\\\.\\iorder2_1_4 360M > junk401 2>&1 &
proc401=$!
addfile.sh iorder2_1 \\\\.\\iorder2_1_5 360M > junk402 2>&1 &
proc402=$!
addfile.sh iorder2_1 \\\\.\\iorder2_1_6 360M > junk403 2>&1 &
proc403=$!

```

```

addfile.sh iordr2_1 \\\.\iordr2_1_7 360M > junk404 2>&1 &
proc404=$!
addfile.sh iordr2_1 \\\.\iordr2_1_8 360M > junk405 2>&1 &
proc405=$!
addfile.sh iordr2_1 \\\.\iordr2_1_9 360M > junk406 2>&1 &
proc406=$!
addfile.sh iordr2_1 \\\.\iordr2_1_10 360M > junk407 2>&1 &
proc407=$!
addfile.sh iordr2_1 \\\.\iordr2_1_11 360M > junk408 2>&1 &
proc408=$!
wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

addfile.sh iordr2_2 \\\.\iordr2_2_1 360M > junk398 2>&1 &
proc398=$!
addfile.sh iordr2_2 \\\.\iordr2_2_2 360M > junk399 2>&1 &
proc399=$!
addfile.sh iordr2_2 \\\.\iordr2_2_3 360M > junk400 2>&1 &
proc400=$!
addfile.sh iordr2_2 \\\.\iordr2_2_4 360M > junk401 2>&1 &
proc401=$!
addfile.sh iordr2_2 \\\.\iordr2_2_5 360M > junk402 2>&1 &
proc402=$!
addfile.sh iordr2_2 \\\.\iordr2_2_6 360M > junk403 2>&1 &
proc403=$!
addfile.sh iordr2_2 \\\.\iordr2_2_7 360M > junk404 2>&1 &
proc404=$!
addfile.sh iordr2_2 \\\.\iordr2_2_8 360M > junk405 2>&1 &
proc405=$!
addfile.sh iordr2_2 \\\.\iordr2_2_9 360M > junk406 2>&1 &
proc406=$!
addfile.sh iordr2_2 \\\.\iordr2_2_10 360M > junk407 2>&1 &

```

```

proc407=$!
addfile.sh iordr2_2 \\\.\iordr2_2_11 360M > junk408 2>&1 &
proc408=$!

wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

addfile.sh iordr2_3 \\\.\iordr2_3_1 360M > junk398 2>&1 &
proc398=$!
addfile.sh iordr2_3 \\\.\iordr2_3_2 360M > junk399 2>&1 &
proc399=$!
addfile.sh iordr2_3 \\\.\iordr2_3_3 360M > junk400 2>&1 &
proc400=$!
addfile.sh iordr2_3 \\\.\iordr2_3_4 360M > junk401 2>&1 &
proc401=$!
addfile.sh iordr2_3 \\\.\iordr2_3_5 360M > junk402 2>&1 &
proc402=$!
addfile.sh iordr2_3 \\\.\iordr2_3_6 360M > junk403 2>&1 &
proc403=$!
addfile.sh iordr2_3 \\\.\iordr2_3_7 360M > junk404 2>&1 &
proc404=$!
addfile.sh iordr2_3 \\\.\iordr2_3_8 360M > junk405 2>&1 &
proc405=$!
addfile.sh iordr2_3 \\\.\iordr2_3_9 360M > junk406 2>&1 &
proc406=$!
addfile.sh iordr2_3 \\\.\iordr2_3_10 360M > junk407 2>&1 &
proc407=$!
addfile.sh iordr2_3 \\\.\iordr2_3_11 360M > junk408 2>&1 &
proc408=$!

wait $proc398
proc398=$?

```

```

procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

addfile.sh icust2_0 \\\\.\\icust2_0_1 240M > junk398 2>&1 &
proc398=$!
addfile.sh icust2_0 \\\\.\\icust2_0_2 240M > junk399 2>&1 &
proc399=$!
addfile.sh icust2_0 \\\\.\\icust2_0_3 240M > junk400 2>&1 &
proc400=$!
addfile.sh icust2_0 \\\\.\\icust2_0_4 240M > junk401 2>&1 &
proc401=$!
addfile.sh icust2_0 \\\\.\\icust2_0_5 240M > junk402 2>&1 &
proc402=$!
addfile.sh icust2_0 \\\\.\\icust2_0_6 240M > junk403 2>&1 &
proc403=$!
addfile.sh icust2_0 \\\\.\\icust2_0_7 240M > junk404 2>&1 &
proc404=$!
addfile.sh icust2_0 \\\\.\\icust2_0_8 240M > junk405 2>&1 &
proc405=$!
addfile.sh icust2_0 \\\\.\\icust2_0_9 240M > junk406 2>&1 &
proc406=$!
addfile.sh icust2_0 \\\\.\\icust2_0_10 240M > junk407 2>&1 &
proc407=$!
addfile.sh icust2_0 \\\\.\\icust2_0_11 240M > junk408 2>&1 &
proc408=$!
addfile.sh icust2_0 \\\\.\\icust2_0_12 240M > junk409 2>&1 &
proc409=$!
addfile.sh icust2_0 \\\\.\\icust2_0_13 240M > junk410 2>&1 &
proc410=$!
addfile.sh icust2_0 \\\\.\\icust2_0_14 240M > junk411 2>&1 &
proc411=$!
addfile.sh icust2_0 \\\\.\\icust2_0_15 240M > junk412 2>&1 &
proc412=$!
addfile.sh icust2_0 \\\\.\\icust2_0_16 240M > junk413 2>&1 &

```

```

proc413=$!
addfile.sh icust2_0 \\\\.\\icust2_0_17 240M > junk414 2>&1 &
proc414=$!
addfile.sh icust2_0 \\\\.\\icust2_0_18 240M > junk415 2>&1 &
proc415=$!
addfile.sh icust2_0 \\\\.\\icust2_0_19 240M > junk416 2>&1 &
proc416=$!
addfile.sh icust2_0 \\\\.\\icust2_0_20 240M > junk417 2>&1 &
proc417=$!
addfile.sh icust2_0 \\\\.\\icust2_0_21 240M > junk418 2>&1 &
proc418=$!
addfile.sh icust2_0 \\\\.\\icust2_0_22 240M > junk419 2>&1 &
proc419=$!
addfile.sh icust2_0 \\\\.\\icust2_0_23 240M > junk420 2>&1 &
proc420=$!
addfile.sh icust2_0 \\\\.\\icust2_0_24 240M > junk421 2>&1 &
proc421=$!
addfile.sh icust2_0 \\\\.\\icust2_0_25 240M > junk422 2>&1 &
proc422=$!
addfile.sh icust2_0 \\\\.\\icust2_0_26 240M > junk423 2>&1 &
proc423=$!
addfile.sh icust2_0 \\\\.\\icust2_0_27 240M > junk424 2>&1 &
proc424=$!
addfile.sh icust2_0 \\\\.\\icust2_0_28 240M > junk425 2>&1 &
proc425=$!
addfile.sh icust2_0 \\\\.\\icust2_0_29 240M > junk426 2>&1 &
proc426=$!
addfile.sh icust2_0 \\\\.\\icust2_0_30 240M > junk427 2>&1 &
proc427=$!
addfile.sh icust2_0 \\\\.\\icust2_0_31 240M > junk428 2>&1 &
proc428=$!
addfile.sh icust2_0 \\\\.\\icust2_0_32 240M > junk429 2>&1 &
proc429=$!
addfile.sh icust2_0 \\\\.\\icust2_0_33 240M > junk430 2>&1 &
proc430=$!
addfile.sh icust2_0 \\\\.\\icust2_0_34 240M > junk431 2>&1 &
proc431=$!
addfile.sh icust2_0 \\\\.\\icust2_0_35 240M > junk432 2>&1 &
proc432=$!
addfile.sh icust2_0 \\\\.\\icust2_0_36 240M > junk433 2>&1 &
proc433=$!
addfile.sh icust2_0 \\\\.\\icust2_0_37 240M > junk434 2>&1 &
proc434=$!
addfile.sh icust2_0 \\\\.\\icust2_0_38 240M > junk435 2>&1 &
proc435=$!
addfile.sh icust2_0 \\\\.\\icust2_0_39 240M > junk436 2>&1 &
proc436=$!
addfile.sh icust2_0 \\\\.\\icust2_0_40 240M > junk437 2>&1 &
proc437=$!
addfile.sh icust2_0 \\\\.\\icust2_0_41 240M > junk438 2>&1 &
proc438=$!
addfile.sh icust2_0 \\\\.\\icust2_0_42 240M > junk439 2>&1 &
proc439=$!
addfile.sh icust2_0 \\\\.\\icust2_0_43 240M > junk440 2>&1 &
proc440=$!
addfile.sh icust2_0 \\\\.\\icust2_0_44 240M > junk441 2>&1 &
proc441=$!
addfile.sh icust2_0 \\\\.\\icust2_0_45 240M > junk442 2>&1 &
proc442=$!
addfile.sh icust2_0 \\\\.\\icust2_0_46 240M > junk443 2>&1 &
proc443=$!
addfile.sh icust2_0 \\\\.\\icust2_0_47 240M > junk444 2>&1 &
proc444=$!

```

```

wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`
wait $proc409
proc409=$?
procsum=`expr $procsum + $proc409`
wait $proc410
proc410=$?
procsum=`expr $procsum + $proc410`
wait $proc411
proc411=$?
procsum=`expr $procsum + $proc411`
wait $proc412
proc412=$?
procsum=`expr $procsum + $proc412`
wait $proc413
proc413=$?
procsum=`expr $procsum + $proc413`
wait $proc414
proc414=$?
procsum=`expr $procsum + $proc414`
wait $proc415
proc415=$?
procsum=`expr $procsum + $proc415`
wait $proc416
proc416=$?
procsum=`expr $procsum + $proc416`
wait $proc417
proc417=$?
procsum=`expr $procsum + $proc417`
wait $proc418
proc418=$?

```

```

procsum=`expr $procsum + $proc418`
wait $proc419
proc419=$?
procsum=`expr $procsum + $proc419`
wait $proc420
proc420=$?
procsum=`expr $procsum + $proc420`
wait $proc421
proc421=$?
procsum=`expr $procsum + $proc421`
wait $proc422
proc422=$?
procsum=`expr $procsum + $proc422`
wait $proc423
proc423=$?
procsum=`expr $procsum + $proc423`
wait $proc424
proc424=$?
procsum=`expr $procsum + $proc424`
wait $proc425
proc425=$?
procsum=`expr $procsum + $proc425`
wait $proc426
proc426=$?
procsum=`expr $procsum + $proc426`
wait $proc427
proc427=$?
procsum=`expr $procsum + $proc427`
wait $proc428
proc428=$?
procsum=`expr $procsum + $proc428`
wait $proc429
proc429=$?
procsum=`expr $procsum + $proc429`
wait $proc430
proc430=$?
procsum=`expr $procsum + $proc430`
wait $proc431
proc431=$?
procsum=`expr $procsum + $proc431`
wait $proc432
proc432=$?
procsum=`expr $procsum + $proc432`
wait $proc433
proc433=$?
procsum=`expr $procsum + $proc433`
wait $proc434
proc434=$?
procsum=`expr $procsum + $proc434`
wait $proc435
proc435=$?
procsum=`expr $procsum + $proc435`
wait $proc436
proc436=$?
procsum=`expr $procsum + $proc436`
wait $proc437
proc437=$?
procsum=`expr $procsum + $proc437`
wait $proc438
proc438=$?
procsum=`expr $procsum + $proc438`
wait $proc439
proc439=$?

```

```

procsum=`expr $procsum + $proc439`
wait $proc440
proc440=$?
procsum=`expr $procsum + $proc440`
wait $proc441
proc441=$?
procsum=`expr $procsum + $proc441`
wait $proc442
proc442=$?
procsum=`expr $procsum + $proc442`
wait $proc443
proc443=$?
procsum=`expr $procsum + $proc443`
wait $proc444
proc444=$?
procsum=`expr $procsum + $proc444`

```

```

addfile.sh ordl_0 \\\\.\\ordl_0_1 4640M > junk398 2>&1 &
proc398=$!
addfile.sh ordl_0 \\\\.\\ordl_0_2 4640M > junk399 2>&1 &
proc399=$!
addfile.sh ordl_0 \\\\.\\ordl_0_3 4640M > junk400 2>&1 &
proc400=$!
addfile.sh ordl_0 \\\\.\\ordl_0_4 4640M > junk401 2>&1 &
proc401=$!
addfile.sh ordl_0 \\\\.\\ordl_0_5 4640M > junk402 2>&1 &
proc402=$!
addfile.sh ordl_0 \\\\.\\ordl_0_6 4640M > junk403 2>&1 &
proc403=$!
addfile.sh ordl_0 \\\\.\\ordl_0_7 4640M > junk404 2>&1 &
proc404=$!
addfile.sh ordl_0 \\\\.\\ordl_0_8 4640M > junk405 2>&1 &
proc405=$!
addfile.sh ordl_0 \\\\.\\ordl_0_9 4640M > junk406 2>&1 &
proc406=$!
addfile.sh ordl_0 \\\\.\\ordl_0_10 4640M > junk407 2>&1 &
proc407=$!
addfile.sh ordl_0 \\\\.\\ordl_0_11 4640M > junk408 2>&1 &
proc408=$!
wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405

```

```

proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

```

```

addfile.sh ordl_1 \\\\.\\ordl_1_1 4640M > junk398 2>&1 &
proc398=$!
addfile.sh ordl_1 \\\\.\\ordl_1_2 4640M > junk399 2>&1 &
proc399=$!
addfile.sh ordl_1 \\\\.\\ordl_1_3 4640M > junk400 2>&1 &
proc400=$!
addfile.sh ordl_1 \\\\.\\ordl_1_4 4640M > junk401 2>&1 &
proc401=$!
addfile.sh ordl_1 \\\\.\\ordl_1_5 4640M > junk402 2>&1 &
proc402=$!
addfile.sh ordl_1 \\\\.\\ordl_1_6 4640M > junk403 2>&1 &
proc403=$!
addfile.sh ordl_1 \\\\.\\ordl_1_7 4640M > junk404 2>&1 &
proc404=$!
addfile.sh ordl_1 \\\\.\\ordl_1_8 4640M > junk405 2>&1 &
proc405=$!
addfile.sh ordl_1 \\\\.\\ordl_1_9 4640M > junk406 2>&1 &
proc406=$!
addfile.sh ordl_1 \\\\.\\ordl_1_10 4640M > junk407 2>&1 &
proc407=$!
addfile.sh ordl_1 \\\\.\\ordl_1_11 4640M > junk408 2>&1 &
proc408=$!
wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?

```

```

procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

addfile.sh ordl_2 \\\\.\\ordl_2_1 4640M > junk398 2>&1 &
proc398=$!
addfile.sh ordl_2 \\\\.\\ordl_2_2 4640M > junk399 2>&1 &
proc399=$!
addfile.sh ordl_2 \\\\.\\ordl_2_3 4640M > junk400 2>&1 &
proc400=$!
addfile.sh ordl_2 \\\\.\\ordl_2_4 4640M > junk401 2>&1 &
proc401=$!
addfile.sh ordl_2 \\\\.\\ordl_2_5 4640M > junk402 2>&1 &
proc402=$!
addfile.sh ordl_2 \\\\.\\ordl_2_6 4640M > junk403 2>&1 &
proc403=$!
addfile.sh ordl_2 \\\\.\\ordl_2_7 4640M > junk404 2>&1 &
proc404=$!
addfile.sh ordl_2 \\\\.\\ordl_2_8 4640M > junk405 2>&1 &
proc405=$!
addfile.sh ordl_2 \\\\.\\ordl_2_9 4640M > junk406 2>&1 &
proc406=$!
addfile.sh ordl_2 \\\\.\\ordl_2_10 4640M > junk407 2>&1 &
proc407=$!
addfile.sh ordl_2 \\\\.\\ordl_2_11 4640M > junk408 2>&1 &
proc408=$!

wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

addfile.sh ordl_3 \\\\.\\ordl_3_1 4640M > junk398 2>&1 &

```

```

proc398=$!
addfile.sh ordl_3 \\\\.\\ordl_3_2 4640M > junk399 2>&1 &
proc399=$!
addfile.sh ordl_3 \\\\.\\ordl_3_3 4640M > junk400 2>&1 &
proc400=$!
addfile.sh ordl_3 \\\\.\\ordl_3_4 4640M > junk401 2>&1 &
proc401=$!
addfile.sh ordl_3 \\\\.\\ordl_3_5 4640M > junk402 2>&1 &
proc402=$!
addfile.sh ordl_3 \\\\.\\ordl_3_6 4640M > junk403 2>&1 &
proc403=$!
addfile.sh ordl_3 \\\\.\\ordl_3_7 4640M > junk404 2>&1 &
proc404=$!
addfile.sh ordl_3 \\\\.\\ordl_3_8 4640M > junk405 2>&1 &
proc405=$!
addfile.sh ordl_3 \\\\.\\ordl_3_9 4640M > junk406 2>&1 &
proc406=$!
addfile.sh ordl_3 \\\\.\\ordl_3_10 4640M > junk407 2>&1 &
proc407=$!
addfile.sh ordl_3 \\\\.\\ordl_3_11 4640M > junk408 2>&1 &
proc408=$!

wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

addfile.sh iordr1_0 \\\\.\\iordr1_0_1 220M > junk398 2>&1 &
proc398=$!
addfile.sh iordr1_0 \\\\.\\iordr1_0_2 220M > junk399 2>&1 &
proc399=$!
addfile.sh iordr1_0 \\\\.\\iordr1_0_3 220M > junk400 2>&1 &
proc400=$!
addfile.sh iordr1_0 \\\\.\\iordr1_0_4 220M > junk401 2>&1 &

```

```

proc401=$!
addfile.sh iordr1_0 \\\\.\\iordr1_0_5 220M > junk402 2>&1 &
proc402=$!
addfile.sh iordr1_0 \\\\.\\iordr1_0_6 220M > junk403 2>&1 &
proc403=$!
addfile.sh iordr1_0 \\\\.\\iordr1_0_7 220M > junk404 2>&1 &
proc404=$!
addfile.sh iordr1_0 \\\\.\\iordr1_0_8 220M > junk405 2>&1 &
proc405=$!
addfile.sh iordr1_0 \\\\.\\iordr1_0_9 220M > junk406 2>&1 &
proc406=$!
addfile.sh iordr1_0 \\\\.\\iordr1_0_10 220M > junk407 2>&1 &
proc407=$!
addfile.sh iordr1_0 \\\\.\\iordr1_0_11 220M > junk408 2>&1 &
proc408=$!
wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

addfile.sh iordr1_1 \\\\.\\iordr1_1_1 220M > junk398 2>&1 &
proc398=$!
addfile.sh iordr1_1 \\\\.\\iordr1_1_2 220M > junk399 2>&1 &
proc399=$!
addfile.sh iordr1_1 \\\\.\\iordr1_1_3 220M > junk400 2>&1 &
proc400=$!
addfile.sh iordr1_1 \\\\.\\iordr1_1_4 220M > junk401 2>&1 &
proc401=$!
addfile.sh iordr1_1 \\\\.\\iordr1_1_5 220M > junk402 2>&1 &
proc402=$!
addfile.sh iordr1_1 \\\\.\\iordr1_1_6 220M > junk403 2>&1 &
proc403=$!
addfile.sh iordr1_1 \\\\.\\iordr1_1_7 220M > junk404 2>&1 &
proc404=$!

```

```

addfile.sh iordr1_1 \\\\.\\iordr1_1_8 220M > junk405 2>&1 &
proc405=$!
addfile.sh iordr1_1 \\\\.\\iordr1_1_9 220M > junk406 2>&1 &
proc406=$!
addfile.sh iordr1_1 \\\\.\\iordr1_1_10 220M > junk407 2>&1 &
proc407=$!
addfile.sh iordr1_1 \\\\.\\iordr1_1_11 220M > junk408 2>&1 &
proc408=$!
wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

addfile.sh iordr1_2 \\\\.\\iordr1_2_1 220M > junk398 2>&1 &
proc398=$!
addfile.sh iordr1_2 \\\\.\\iordr1_2_2 220M > junk399 2>&1 &
proc399=$!
addfile.sh iordr1_2 \\\\.\\iordr1_2_3 220M > junk400 2>&1 &
proc400=$!
addfile.sh iordr1_2 \\\\.\\iordr1_2_4 220M > junk401 2>&1 &
proc401=$!
addfile.sh iordr1_2 \\\\.\\iordr1_2_5 220M > junk402 2>&1 &
proc402=$!
addfile.sh iordr1_2 \\\\.\\iordr1_2_6 220M > junk403 2>&1 &
proc403=$!
addfile.sh iordr1_2 \\\\.\\iordr1_2_7 220M > junk404 2>&1 &
proc404=$!
addfile.sh iordr1_2 \\\\.\\iordr1_2_8 220M > junk405 2>&1 &
proc405=$!
addfile.sh iordr1_2 \\\\.\\iordr1_2_9 220M > junk406 2>&1 &
proc406=$!
addfile.sh iordr1_2 \\\\.\\iordr1_2_10 220M > junk407 2>&1 &
proc407=$!
addfile.sh iordr1_2 \\\\.\\iordr1_2_11 220M > junk408 2>&1 &

```

```

proc408=$!

wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

addfile.sh iordr1_3 \\\\.\\iordr1_3_1 220M > junk398 2>&1 &
proc398=$!
addfile.sh iordr1_3 \\\\.\\iordr1_3_2 220M > junk399 2>&1 &
proc399=$!
addfile.sh iordr1_3 \\\\.\\iordr1_3_3 220M > junk400 2>&1 &
proc400=$!
addfile.sh iordr1_3 \\\\.\\iordr1_3_4 220M > junk401 2>&1 &
proc401=$!
addfile.sh iordr1_3 \\\\.\\iordr1_3_5 220M > junk402 2>&1 &
proc402=$!
addfile.sh iordr1_3 \\\\.\\iordr1_3_6 220M > junk403 2>&1 &
proc403=$!
addfile.sh iordr1_3 \\\\.\\iordr1_3_7 220M > junk404 2>&1 &
proc404=$!
addfile.sh iordr1_3 \\\\.\\iordr1_3_8 220M > junk405 2>&1 &
proc405=$!
addfile.sh iordr1_3 \\\\.\\iordr1_3_9 220M > junk406 2>&1 &
proc406=$!
addfile.sh iordr1_3 \\\\.\\iordr1_3_10 220M > junk407 2>&1 &
proc407=$!
addfile.sh iordr1_3 \\\\.\\iordr1_3_11 220M > junk408 2>&1 &
proc408=$!

wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399

```

```

proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

addfile.sh hist_0 \\\\.\\hist_0_1 530M > junk398 2>&1 &
proc398=$!
addfile.sh hist_0 \\\\.\\hist_0_2 530M > junk399 2>&1 &
proc399=$!
addfile.sh hist_0 \\\\.\\hist_0_3 530M > junk400 2>&1 &
proc400=$!
addfile.sh hist_0 \\\\.\\hist_0_4 530M > junk401 2>&1 &
proc401=$!
addfile.sh hist_0 \\\\.\\hist_0_5 530M > junk402 2>&1 &
proc402=$!
addfile.sh hist_0 \\\\.\\hist_0_6 530M > junk403 2>&1 &
proc403=$!
addfile.sh hist_0 \\\\.\\hist_0_7 530M > junk404 2>&1 &
proc404=$!
addfile.sh hist_0 \\\\.\\hist_0_8 530M > junk405 2>&1 &
proc405=$!
addfile.sh hist_0 \\\\.\\hist_0_9 530M > junk406 2>&1 &
proc406=$!
addfile.sh hist_0 \\\\.\\hist_0_10 530M > junk407 2>&1 &
proc407=$!
addfile.sh hist_0 \\\\.\\hist_0_11 530M > junk408 2>&1 &
proc408=$!
wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?

```

```

procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

```

```

addfile.sh hist_1 \\\\.\\hist_1_1 530M > junk398 2>&1 &
proc398=$!
addfile.sh hist_1 \\\\.\\hist_1_2 530M > junk399 2>&1 &
proc399=$!
addfile.sh hist_1 \\\\.\\hist_1_3 530M > junk400 2>&1 &
proc400=$!
addfile.sh hist_1 \\\\.\\hist_1_4 530M > junk401 2>&1 &
proc401=$!
addfile.sh hist_1 \\\\.\\hist_1_5 530M > junk402 2>&1 &
proc402=$!
addfile.sh hist_1 \\\\.\\hist_1_6 530M > junk403 2>&1 &
proc403=$!
addfile.sh hist_1 \\\\.\\hist_1_7 530M > junk404 2>&1 &
proc404=$!
addfile.sh hist_1 \\\\.\\hist_1_8 530M > junk405 2>&1 &
proc405=$!
addfile.sh hist_1 \\\\.\\hist_1_9 530M > junk406 2>&1 &
proc406=$!
addfile.sh hist_1 \\\\.\\hist_1_10 530M > junk407 2>&1 &
proc407=$!
addfile.sh hist_1 \\\\.\\hist_1_11 530M > junk408 2>&1 &
proc408=$!
wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`

```

```

wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

```

```

addfile.sh hist_2 \\\\.\\hist_2_1 530M > junk398 2>&1 &
proc398=$!
addfile.sh hist_2 \\\\.\\hist_2_2 530M > junk399 2>&1 &
proc399=$!
addfile.sh hist_2 \\\\.\\hist_2_3 530M > junk400 2>&1 &
proc400=$!
addfile.sh hist_2 \\\\.\\hist_2_4 530M > junk401 2>&1 &
proc401=$!
addfile.sh hist_2 \\\\.\\hist_2_5 530M > junk402 2>&1 &
proc402=$!
addfile.sh hist_2 \\\\.\\hist_2_6 530M > junk403 2>&1 &
proc403=$!
addfile.sh hist_2 \\\\.\\hist_2_7 530M > junk404 2>&1 &
proc404=$!
addfile.sh hist_2 \\\\.\\hist_2_8 530M > junk405 2>&1 &
proc405=$!
addfile.sh hist_2 \\\\.\\hist_2_9 530M > junk406 2>&1 &
proc406=$!
addfile.sh hist_2 \\\\.\\hist_2_10 530M > junk407 2>&1 &
proc407=$!
addfile.sh hist_2 \\\\.\\hist_2_11 530M > junk408 2>&1 &
proc408=$!

```

```

wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`

```

```

wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

addfile.sh hist_3 \\\\.\\hist_3_1 530M > junk398 2>&1 &
proc398=$!
addfile.sh hist_3 \\\\.\\hist_3_2 530M > junk399 2>&1 &
proc399=$!
addfile.sh hist_3 \\\\.\\hist_3_3 530M > junk400 2>&1 &
proc400=$!
addfile.sh hist_3 \\\\.\\hist_3_4 530M > junk401 2>&1 &
proc401=$!
addfile.sh hist_3 \\\\.\\hist_3_5 530M > junk402 2>&1 &
proc402=$!
addfile.sh hist_3 \\\\.\\hist_3_6 530M > junk403 2>&1 &
proc403=$!
addfile.sh hist_3 \\\\.\\hist_3_7 530M > junk404 2>&1 &
proc404=$!
addfile.sh hist_3 \\\\.\\hist_3_8 530M > junk405 2>&1 &
proc405=$!
addfile.sh hist_3 \\\\.\\hist_3_9 530M > junk406 2>&1 &
proc406=$!
addfile.sh hist_3 \\\\.\\hist_3_10 530M > junk407 2>&1 &
proc407=$!
addfile.sh hist_3 \\\\.\\hist_3_11 530M > junk408 2>&1 &
proc408=$!

wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`

```

```

wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

addfile.sh nord_0 \\\\.\\nord_0_1 50M > junk398 2>&1 &
proc398=$!
addfile.sh nord_0 \\\\.\\nord_0_2 50M > junk399 2>&1 &
proc399=$!
addfile.sh nord_0 \\\\.\\nord_0_3 50M > junk400 2>&1 &
proc400=$!
addfile.sh nord_0 \\\\.\\nord_0_4 50M > junk401 2>&1 &
proc401=$!
addfile.sh nord_0 \\\\.\\nord_0_5 50M > junk402 2>&1 &
proc402=$!
addfile.sh nord_0 \\\\.\\nord_0_6 50M > junk403 2>&1 &
proc403=$!
addfile.sh nord_0 \\\\.\\nord_0_7 50M > junk404 2>&1 &
proc404=$!
addfile.sh nord_0 \\\\.\\nord_0_8 50M > junk405 2>&1 &
proc405=$!
addfile.sh nord_0 \\\\.\\nord_0_9 50M > junk406 2>&1 &
proc406=$!
addfile.sh nord_0 \\\\.\\nord_0_10 50M > junk407 2>&1 &
proc407=$!
addfile.sh nord_0 \\\\.\\nord_0_11 50M > junk408 2>&1 &
proc408=$!
wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

addfile.sh nord_1 \\\\.\\nord_1_1 50M > junk398 2>&1 &
proc398=$!
addfile.sh nord_1 \\\\.\\nord_1_2 50M > junk399 2>&1 &

```

```

proc399=$!
addfile.sh nord_1 \\\\.\\nord_1_3 50M > junk400 2>&1 &
proc400=$!
addfile.sh nord_1 \\\\.\\nord_1_4 50M > junk401 2>&1 &
proc401=$!
addfile.sh nord_1 \\\\.\\nord_1_5 50M > junk402 2>&1 &
proc402=$!
addfile.sh nord_1 \\\\.\\nord_1_6 50M > junk403 2>&1 &
proc403=$!
addfile.sh nord_1 \\\\.\\nord_1_7 50M > junk404 2>&1 &
proc404=$!
addfile.sh nord_1 \\\\.\\nord_1_8 50M > junk405 2>&1 &
proc405=$!
addfile.sh nord_1 \\\\.\\nord_1_9 50M > junk406 2>&1 &
proc406=$!
addfile.sh nord_1 \\\\.\\nord_1_10 50M > junk407 2>&1 &
proc407=$!
addfile.sh nord_1 \\\\.\\nord_1_11 50M > junk408 2>&1 &
proc408=$!
wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

addfile.sh nord_2 \\\\.\\nord_2_1 50M > junk398 2>&1 &
proc398=$!
addfile.sh nord_2 \\\\.\\nord_2_2 50M > junk399 2>&1 &
proc399=$!
addfile.sh nord_2 \\\\.\\nord_2_3 50M > junk400 2>&1 &
proc400=$!
addfile.sh nord_2 \\\\.\\nord_2_4 50M > junk401 2>&1 &
proc401=$!
addfile.sh nord_2 \\\\.\\nord_2_5 50M > junk402 2>&1 &
proc402=$!

```

```

addfile.sh nord_2 \\\\.\\nord_2_6 50M > junk403 2>&1 &
proc403=$!
addfile.sh nord_2 \\\\.\\nord_2_7 50M > junk404 2>&1 &
proc404=$!
addfile.sh nord_2 \\\\.\\nord_2_8 50M > junk405 2>&1 &
proc405=$!
addfile.sh nord_2 \\\\.\\nord_2_9 50M > junk406 2>&1 &
proc406=$!
addfile.sh nord_2 \\\\.\\nord_2_10 50M > junk407 2>&1 &
proc407=$!
addfile.sh nord_2 \\\\.\\nord_2_11 50M > junk408 2>&1 &
proc408=$!

wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

addfile.sh nord_3 \\\\.\\nord_3_1 50M > junk398 2>&1 &
proc398=$!
addfile.sh nord_3 \\\\.\\nord_3_2 50M > junk399 2>&1 &
proc399=$!
addfile.sh nord_3 \\\\.\\nord_3_3 50M > junk400 2>&1 &
proc400=$!
addfile.sh nord_3 \\\\.\\nord_3_4 50M > junk401 2>&1 &
proc401=$!
addfile.sh nord_3 \\\\.\\nord_3_5 50M > junk402 2>&1 &
proc402=$!
addfile.sh nord_3 \\\\.\\nord_3_6 50M > junk403 2>&1 &
proc403=$!
addfile.sh nord_3 \\\\.\\nord_3_7 50M > junk404 2>&1 &
proc404=$!
addfile.sh nord_3 \\\\.\\nord_3_8 50M > junk405 2>&1 &
proc405=$!

```

```

addfile.sh nord_3 \\\\.\\nord_3_9 50M > junk406 2>&1 &
proc406=$!
addfile.sh nord_3 \\\\.\\nord_3_10 50M > junk407 2>&1 &
proc407=$!
addfile.sh nord_3 \\\\.\\nord_3_11 50M > junk408 2>&1 &
proc408=$!

wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

addtempfile.sh temp_0 \\\\.\\temp_0_1 500M > junk398 2>&1 &
proc398=$!
addtempfile.sh temp_0 \\\\.\\temp_0_2 500M > junk399 2>&1 &
proc399=$!
addtempfile.sh temp_0 \\\\.\\temp_0_3 500M > junk400 2>&1 &
proc400=$!
addtempfile.sh temp_0 \\\\.\\temp_0_4 500M > junk401 2>&1 &
proc401=$!
addtempfile.sh temp_0 \\\\.\\temp_0_5 500M > junk402 2>&1 &
proc402=$!
addtempfile.sh temp_0 \\\\.\\temp_0_6 500M > junk403 2>&1 &
proc403=$!
addtempfile.sh temp_0 \\\\.\\temp_0_7 500M > junk404 2>&1 &
proc404=$!
addtempfile.sh temp_0 \\\\.\\temp_0_8 500M > junk405 2>&1 &
proc405=$!
addtempfile.sh temp_0 \\\\.\\temp_0_9 500M > junk406 2>&1 &
proc406=$!
addtempfile.sh temp_0 \\\\.\\temp_0_10 500M > junk407 2>&1 &
proc407=$!
addtempfile.sh temp_0 \\\\.\\temp_0_11 500M > junk408 2>&1 &
proc408=$!

```

```

addtempfile.sh temp_0 \\\\.\\temp_0_12 500M > junk409 2>&1 &
proc409=$!
addtempfile.sh temp_0 \\\\.\\temp_0_13 500M > junk410 2>&1 &
proc410=$!
addtempfile.sh temp_0 \\\\.\\temp_0_14 500M > junk411 2>&1 &
proc411=$!
addtempfile.sh temp_0 \\\\.\\temp_0_15 500M > junk412 2>&1 &
proc412=$!
addtempfile.sh temp_0 \\\\.\\temp_0_16 500M > junk413 2>&1 &
proc413=$!
addtempfile.sh temp_0 \\\\.\\temp_0_17 500M > junk414 2>&1 &
proc414=$!
addtempfile.sh temp_0 \\\\.\\temp_0_18 500M > junk415 2>&1 &
proc415=$!
addtempfile.sh temp_0 \\\\.\\temp_0_19 500M > junk416 2>&1 &
proc416=$!
addtempfile.sh temp_0 \\\\.\\temp_0_20 500M > junk417 2>&1 &
proc417=$!
addtempfile.sh temp_0 \\\\.\\temp_0_21 500M > junk418 2>&1 &
proc418=$!
addtempfile.sh temp_0 \\\\.\\temp_0_22 500M > junk419 2>&1 &
proc419=$!
addtempfile.sh temp_0 \\\\.\\temp_0_23 500M > junk420 2>&1 &
proc420=$!
addtempfile.sh temp_0 \\\\.\\temp_0_24 500M > junk421 2>&1 &
proc421=$!
addtempfile.sh temp_0 \\\\.\\temp_0_25 500M > junk422 2>&1 &
proc422=$!
addtempfile.sh temp_0 \\\\.\\temp_0_26 500M > junk423 2>&1 &
proc423=$!
addtempfile.sh temp_0 \\\\.\\temp_0_27 500M > junk424 2>&1 &
proc424=$!
addtempfile.sh temp_0 \\\\.\\temp_0_28 500M > junk425 2>&1 &
proc425=$!
addtempfile.sh temp_0 \\\\.\\temp_0_29 500M > junk426 2>&1 &
proc426=$!
addtempfile.sh temp_0 \\\\.\\temp_0_30 500M > junk427 2>&1 &
proc427=$!
addtempfile.sh temp_0 \\\\.\\temp_0_31 500M > junk428 2>&1 &
proc428=$!
addtempfile.sh temp_0 \\\\.\\temp_0_32 500M > junk429 2>&1 &
proc429=$!
addtempfile.sh temp_0 \\\\.\\temp_0_33 500M > junk430 2>&1 &
proc430=$!
addtempfile.sh temp_0 \\\\.\\temp_0_34 500M > junk431 2>&1 &
proc431=$!
addtempfile.sh temp_0 \\\\.\\temp_0_35 500M > junk432 2>&1 &
proc432=$!
addtempfile.sh temp_0 \\\\.\\temp_0_36 500M > junk433 2>&1 &
proc433=$!
addtempfile.sh temp_0 \\\\.\\temp_0_37 500M > junk434 2>&1 &
proc434=$!
addtempfile.sh temp_0 \\\\.\\temp_0_38 500M > junk435 2>&1 &
proc435=$!
addtempfile.sh temp_0 \\\\.\\temp_0_39 500M > junk436 2>&1 &
proc436=$!
addtempfile.sh temp_0 \\\\.\\temp_0_40 500M > junk437 2>&1 &
proc437=$!
addtempfile.sh temp_0 \\\\.\\temp_0_41 500M > junk438 2>&1 &
proc438=$!
addtempfile.sh temp_0 \\\\.\\temp_0_42 500M > junk439 2>&1 &
proc439=$!
addtempfile.sh temp_0 \\\\.\\temp_0_43 500M > junk440 2>&1 &

```

```
proc440=$!  
addtempfile.sh temp_0 \\\\.\\temp_0_44 500M > junk441 2>&1 &  
proc441=$!  
addtempfile.sh temp_0 \\\\.\\temp_0_45 500M > junk442 2>&1 &  
proc442=$!  
addtempfile.sh temp_0 \\\\.\\temp_0_46 500M > junk443 2>&1 &  
proc443=$!  
addtempfile.sh temp_0 \\\\.\\temp_0_47 500M > junk444 2>&1 &  
proc444=$!
```

```
wait $proc398  
proc398=$?  
procsum=`expr $procsum + $proc398`  
wait $proc399  
proc399=$?  
procsum=`expr $procsum + $proc399`  
wait $proc400  
proc400=$?  
procsum=`expr $procsum + $proc400`  
wait $proc401  
proc401=$?  
procsum=`expr $procsum + $proc401`  
wait $proc402  
proc402=$?  
procsum=`expr $procsum + $proc402`  
wait $proc403  
proc403=$?  
procsum=`expr $procsum + $proc403`  
wait $proc404  
proc404=$?  
procsum=`expr $procsum + $proc404`  
wait $proc405  
proc405=$?  
procsum=`expr $procsum + $proc405`  
wait $proc406  
proc406=$?  
procsum=`expr $procsum + $proc406`  
wait $proc407  
proc407=$?  
procsum=`expr $procsum + $proc407`  
wait $proc408  
proc408=$?  
procsum=`expr $procsum + $proc408`  
wait $proc409  
proc409=$?  
procsum=`expr $procsum + $proc409`  
wait $proc410  
proc410=$?  
procsum=`expr $procsum + $proc410`  
wait $proc411  
proc411=$?  
procsum=`expr $procsum + $proc411`  
wait $proc412  
proc412=$?  
procsum=`expr $procsum + $proc412`  
wait $proc413  
proc413=$?  
procsum=`expr $procsum + $proc413`  
wait $proc414  
proc414=$?  
procsum=`expr $procsum + $proc414`  
wait $proc415  
proc415=$?
```

```
procsum=`expr $procsum + $proc415`  
wait $proc416  
proc416=$?  
procsum=`expr $procsum + $proc416`  
wait $proc417  
proc417=$?  
procsum=`expr $procsum + $proc417`  
wait $proc418  
proc418=$?  
procsum=`expr $procsum + $proc418`  
wait $proc419  
proc419=$?  
procsum=`expr $procsum + $proc419`  
wait $proc420  
proc420=$?  
procsum=`expr $procsum + $proc420`  
wait $proc421  
proc421=$?  
procsum=`expr $procsum + $proc421`  
wait $proc422  
proc422=$?  
procsum=`expr $procsum + $proc422`  
wait $proc423  
proc423=$?  
procsum=`expr $procsum + $proc423`  
wait $proc424  
proc424=$?  
procsum=`expr $procsum + $proc424`  
wait $proc425  
proc425=$?  
procsum=`expr $procsum + $proc425`  
wait $proc426  
proc426=$?  
procsum=`expr $procsum + $proc426`  
wait $proc427  
proc427=$?  
procsum=`expr $procsum + $proc427`  
wait $proc428  
proc428=$?  
procsum=`expr $procsum + $proc428`  
wait $proc429  
proc429=$?  
procsum=`expr $procsum + $proc429`  
wait $proc430  
proc430=$?  
procsum=`expr $procsum + $proc430`  
wait $proc431  
proc431=$?  
procsum=`expr $procsum + $proc431`  
wait $proc432  
proc432=$?  
procsum=`expr $procsum + $proc432`  
wait $proc433  
proc433=$?  
procsum=`expr $procsum + $proc433`  
wait $proc434  
proc434=$?  
procsum=`expr $procsum + $proc434`  
wait $proc435  
proc435=$?  
procsum=`expr $procsum + $proc435`  
wait $proc436  
proc436=$?
```

```

procsum=`expr $procsum + $proc436`
wait $proc437
proc437=$?
procsum=`expr $procsum + $proc437`
wait $proc438
proc438=$?
procsum=`expr $procsum + $proc438`
wait $proc439
proc439=$?
procsum=`expr $procsum + $proc439`
wait $proc440
proc440=$?
procsum=`expr $procsum + $proc440`
wait $proc441
proc441=$?
procsum=`expr $procsum + $proc441`
wait $proc442
proc442=$?
procsum=`expr $procsum + $proc442`
wait $proc443
proc443=$?
procsum=`expr $procsum + $proc443`
wait $proc444
proc444=$?
procsum=`expr $procsum + $proc444`

addfile.sh ord_0 \\\\.\\ordr_0_1 370M > junk398 2>&1 &
proc398=$!
addfile.sh ord_0 \\\\.\\ordr_0_2 370M > junk399 2>&1 &
proc399=$!
addfile.sh ord_0 \\\\.\\ordr_0_3 370M > junk400 2>&1 &
proc400=$!
addfile.sh ord_0 \\\\.\\ordr_0_4 370M > junk401 2>&1 &
proc401=$!
addfile.sh ord_0 \\\\.\\ordr_0_5 370M > junk402 2>&1 &
proc402=$!
addfile.sh ord_0 \\\\.\\ordr_0_6 370M > junk403 2>&1 &
proc403=$!
addfile.sh ord_0 \\\\.\\ordr_0_7 370M > junk404 2>&1 &
proc404=$!
addfile.sh ord_0 \\\\.\\ordr_0_8 370M > junk405 2>&1 &
proc405=$!
addfile.sh ord_0 \\\\.\\ordr_0_9 370M > junk406 2>&1 &
proc406=$!
addfile.sh ord_0 \\\\.\\ordr_0_10 370M > junk407 2>&1 &
proc407=$!
addfile.sh ord_0 \\\\.\\ordr_0_11 370M > junk408 2>&1 &
proc408=$!
wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`

```

```

wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

addfile.sh ord_1 \\\\.\\ordr_1_1 370M > junk398 2>&1 &
proc398=$!
addfile.sh ord_1 \\\\.\\ordr_1_2 370M > junk399 2>&1 &
proc399=$!
addfile.sh ord_1 \\\\.\\ordr_1_3 370M > junk400 2>&1 &
proc400=$!
addfile.sh ord_1 \\\\.\\ordr_1_4 370M > junk401 2>&1 &
proc401=$!
addfile.sh ord_1 \\\\.\\ordr_1_5 370M > junk402 2>&1 &
proc402=$!
addfile.sh ord_1 \\\\.\\ordr_1_6 370M > junk403 2>&1 &
proc403=$!
addfile.sh ord_1 \\\\.\\ordr_1_7 370M > junk404 2>&1 &
proc404=$!
addfile.sh ord_1 \\\\.\\ordr_1_8 370M > junk405 2>&1 &
proc405=$!
addfile.sh ord_1 \\\\.\\ordr_1_9 370M > junk406 2>&1 &
proc406=$!
addfile.sh ord_1 \\\\.\\ordr_1_10 370M > junk407 2>&1 &
proc407=$!
addfile.sh ord_1 \\\\.\\ordr_1_11 370M > junk408 2>&1 &
proc408=$!
wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405

```

```

proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

addfile.sh ord_2 \\\\.\\ordr_2_1 370M > junk398 2>&1 &
proc398=$!
addfile.sh ord_2 \\\\.\\ordr_2_2 370M > junk399 2>&1 &
proc399=$!
addfile.sh ord_2 \\\\.\\ordr_2_3 370M > junk400 2>&1 &
proc400=$!
addfile.sh ord_2 \\\\.\\ordr_2_4 370M > junk401 2>&1 &
proc401=$!
addfile.sh ord_2 \\\\.\\ordr_2_5 370M > junk402 2>&1 &
proc402=$!
addfile.sh ord_2 \\\\.\\ordr_2_6 370M > junk403 2>&1 &
proc403=$!
addfile.sh ord_2 \\\\.\\ordr_2_7 370M > junk404 2>&1 &
proc404=$!
addfile.sh ord_2 \\\\.\\ordr_2_8 370M > junk405 2>&1 &
proc405=$!
addfile.sh ord_2 \\\\.\\ordr_2_9 370M > junk406 2>&1 &
proc406=$!
addfile.sh ord_2 \\\\.\\ordr_2_10 370M > junk407 2>&1 &
proc407=$!
addfile.sh ord_2 \\\\.\\ordr_2_11 370M > junk408 2>&1 &
proc408=$!

wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407

```

```

proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

addfile.sh ord_3 \\\\.\\ordr_3_1 370M > junk398 2>&1 &
proc398=$!
addfile.sh ord_3 \\\\.\\ordr_3_2 370M > junk399 2>&1 &
proc399=$!
addfile.sh ord_3 \\\\.\\ordr_3_3 370M > junk400 2>&1 &
proc400=$!
addfile.sh ord_3 \\\\.\\ordr_3_4 370M > junk401 2>&1 &
proc401=$!
addfile.sh ord_3 \\\\.\\ordr_3_5 370M > junk402 2>&1 &
proc402=$!
addfile.sh ord_3 \\\\.\\ordr_3_6 370M > junk403 2>&1 &
proc403=$!
addfile.sh ord_3 \\\\.\\ordr_3_7 370M > junk404 2>&1 &
proc404=$!
addfile.sh ord_3 \\\\.\\ordr_3_8 370M > junk405 2>&1 &
proc405=$!
addfile.sh ord_3 \\\\.\\ordr_3_9 370M > junk406 2>&1 &
proc406=$!
addfile.sh ord_3 \\\\.\\ordr_3_10 370M > junk407 2>&1 &
proc407=$!
addfile.sh ord_3 \\\\.\\ordr_3_11 370M > junk408 2>&1 &
proc408=$!

wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

```

```

addfile.sh ware_0 \\\\.\\ware_1_0 10M > junk4 2>&1 &
proc4=$!
addfile.sh ware_0 \\\\.\\ware_2_0 10M > junk5 2>&1 &
proc5=$!
addfile.sh ware_0 \\\\.\\ware_3_0 10M > junk6 2>&1 &
proc6=$!
addfile.sh ware_0 \\\\.\\ware_0_1 10M > junk398 2>&1 &
proc398=$!
addfile.sh ware_0 \\\\.\\ware_0_2 10M > junk399 2>&1 &
proc399=$!
addfile.sh ware_0 \\\\.\\ware_0_3 10M > junk400 2>&1 &
proc400=$!
addfile.sh ware_0 \\\\.\\ware_0_4 10M > junk401 2>&1 &
proc401=$!
addfile.sh ware_0 \\\\.\\ware_0_5 10M > junk402 2>&1 &
proc402=$!
addfile.sh ware_0 \\\\.\\ware_0_6 10M > junk403 2>&1 &
proc403=$!
addfile.sh ware_0 \\\\.\\ware_0_7 10M > junk404 2>&1 &
proc404=$!
addfile.sh ware_0 \\\\.\\ware_0_8 10M > junk405 2>&1 &
proc405=$!
addfile.sh ware_0 \\\\.\\ware_0_9 10M > junk406 2>&1 &
proc406=$!
addfile.sh ware_0 \\\\.\\ware_0_10 10M > junk407 2>&1 &
proc407=$!
addfile.sh ware_0 \\\\.\\ware_0_11 10M > junk408 2>&1 &
proc408=$!
wait $proc4
proc4=$?
procsum=`expr $procsum + $proc4`
wait $proc5
proc5=$?
procsum=`expr $procsum + $proc5`
wait $proc6
proc6=$?
procsum=`expr $procsum + $proc6`
wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?

```

```

procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

addfile.sh ware_0 \\\\.\\ware_1_1 10M > junk398 2>&1 &
proc398=$!
addfile.sh ware_0 \\\\.\\ware_1_2 10M > junk399 2>&1 &
proc399=$!
addfile.sh ware_0 \\\\.\\ware_1_3 10M > junk400 2>&1 &
proc400=$!
addfile.sh ware_0 \\\\.\\ware_1_4 10M > junk401 2>&1 &
proc401=$!
addfile.sh ware_0 \\\\.\\ware_1_5 10M > junk402 2>&1 &
proc402=$!
addfile.sh ware_0 \\\\.\\ware_1_6 10M > junk403 2>&1 &
proc403=$!
addfile.sh ware_0 \\\\.\\ware_1_7 10M > junk404 2>&1 &
proc404=$!
addfile.sh ware_0 \\\\.\\ware_1_8 10M > junk405 2>&1 &
proc405=$!
addfile.sh ware_0 \\\\.\\ware_1_9 10M > junk406 2>&1 &
proc406=$!
addfile.sh ware_0 \\\\.\\ware_1_10 10M > junk407 2>&1 &
proc407=$!
addfile.sh ware_0 \\\\.\\ware_1_11 10M > junk408 2>&1 &
proc408=$!
wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

```

```

addfile.sh ware_0 \\\\.\\ware_2_1 10M > junk398 2>&1 &
proc398=$!
addfile.sh ware_0 \\\\.\\ware_2_2 10M > junk399 2>&1 &
proc399=$!
addfile.sh ware_0 \\\\.\\ware_2_3 10M > junk400 2>&1 &
proc400=$!
addfile.sh ware_0 \\\\.\\ware_2_4 10M > junk401 2>&1 &
proc401=$!
addfile.sh ware_0 \\\\.\\ware_2_5 10M > junk402 2>&1 &
proc402=$!
addfile.sh ware_0 \\\\.\\ware_2_6 10M > junk403 2>&1 &
proc403=$!
addfile.sh ware_0 \\\\.\\ware_2_7 10M > junk404 2>&1 &
proc404=$!
addfile.sh ware_0 \\\\.\\ware_2_8 10M > junk405 2>&1 &
proc405=$!
addfile.sh ware_0 \\\\.\\ware_2_9 10M > junk406 2>&1 &
proc406=$!
addfile.sh ware_0 \\\\.\\ware_2_10 10M > junk407 2>&1 &
proc407=$!
addfile.sh ware_0 \\\\.\\ware_2_11 10M > junk408 2>&1 &
proc408=$!

```

```

wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

```

```

addfile.sh ware_0 \\\\.\\ware_3_1 10M > junk398 2>&1 &
proc398=$!
addfile.sh ware_0 \\\\.\\ware_3_2 10M > junk399 2>&1 &
proc399=$!
addfile.sh ware_0 \\\\.\\ware_3_3 10M > junk400 2>&1 &

```

```

proc400=$!
addfile.sh ware_0 \\\\.\\ware_3_4 10M > junk401 2>&1 &
proc401=$!
addfile.sh ware_0 \\\\.\\ware_3_5 10M > junk402 2>&1 &
proc402=$!
addfile.sh ware_0 \\\\.\\ware_3_6 10M > junk403 2>&1 &
proc403=$!
addfile.sh ware_0 \\\\.\\ware_3_7 10M > junk404 2>&1 &
proc404=$!
addfile.sh ware_0 \\\\.\\ware_3_8 10M > junk405 2>&1 &
proc405=$!
addfile.sh ware_0 \\\\.\\ware_3_9 10M > junk406 2>&1 &
proc406=$!
addfile.sh ware_0 \\\\.\\ware_3_10 10M > junk407 2>&1 &
proc407=$!
addfile.sh ware_0 \\\\.\\ware_3_11 10M > junk408 2>&1 &
proc408=$!

```

```

wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

```

```

addfile.sh dist_0 \\\\.\\dist_1_0 10M > junk25 2>&1 &
proc25=$!
addfile.sh dist_0 \\\\.\\dist_2_0 10M > junk26 2>&1 &
proc26=$!
addfile.sh dist_0 \\\\.\\dist_3_0 10M > junk27 2>&1 &
proc27=$!
addfile.sh dist_0 \\\\.\\dist_0_1 10M > junk398 2>&1 &
proc398=$!
addfile.sh dist_0 \\\\.\\dist_0_2 10M > junk399 2>&1 &
proc399=$!
addfile.sh dist_0 \\\\.\\dist_0_3 10M > junk400 2>&1 &

```

```

proc400=$!
addfile.sh dist_0 \\\\.\\dist_0_4 10M > junk401 2>&1 &
proc401=$!
addfile.sh dist_0 \\\\.\\dist_0_5 10M > junk402 2>&1 &
proc402=$!
addfile.sh dist_0 \\\\.\\dist_0_6 10M > junk403 2>&1 &
proc403=$!
addfile.sh dist_0 \\\\.\\dist_0_7 10M > junk404 2>&1 &
proc404=$!
addfile.sh dist_0 \\\\.\\dist_0_8 10M > junk405 2>&1 &
proc405=$!
addfile.sh dist_0 \\\\.\\dist_0_9 10M > junk406 2>&1 &
proc406=$!
addfile.sh dist_0 \\\\.\\dist_0_10 10M > junk407 2>&1 &
proc407=$!
addfile.sh dist_0 \\\\.\\dist_0_11 10M > junk408 2>&1 &
proc408=$!
wait $proc25
proc25=$?
procsum=`expr $procsum + $proc25`
wait $proc26
proc26=$?
procsum=`expr $procsum + $proc26`
wait $proc27
proc27=$?
procsum=`expr $procsum + $proc27`
wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

addfile.sh dist_0 \\\\.\\dist_1_1 10M > junk398 2>&1 &
proc398=$!
addfile.sh dist_0 \\\\.\\dist_1_2 10M > junk399 2>&1 &

```

```

proc399=$!
addfile.sh dist_0 \\\\.\\dist_1_3 10M > junk400 2>&1 &
proc400=$!
addfile.sh dist_0 \\\\.\\dist_1_4 10M > junk401 2>&1 &
proc401=$!
addfile.sh dist_0 \\\\.\\dist_1_5 10M > junk402 2>&1 &
proc402=$!
addfile.sh dist_0 \\\\.\\dist_1_6 10M > junk403 2>&1 &
proc403=$!
addfile.sh dist_0 \\\\.\\dist_1_7 10M > junk404 2>&1 &
proc404=$!
addfile.sh dist_0 \\\\.\\dist_1_8 10M > junk405 2>&1 &
proc405=$!
addfile.sh dist_0 \\\\.\\dist_1_9 10M > junk406 2>&1 &
proc406=$!
addfile.sh dist_0 \\\\.\\dist_1_10 10M > junk407 2>&1 &
proc407=$!
addfile.sh dist_0 \\\\.\\dist_1_11 10M > junk408 2>&1 &
proc408=$!
wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

addfile.sh dist_0 \\\\.\\dist_2_1 10M > junk398 2>&1 &
proc398=$!
addfile.sh dist_0 \\\\.\\dist_2_2 10M > junk399 2>&1 &
proc399=$!
addfile.sh dist_0 \\\\.\\dist_2_3 10M > junk400 2>&1 &
proc400=$!
addfile.sh dist_0 \\\\.\\dist_2_4 10M > junk401 2>&1 &
proc401=$!
addfile.sh dist_0 \\\\.\\dist_2_5 10M > junk402 2>&1 &
proc402=$!

```

```

addfile.sh dist_0 \\\\.\\dist_2_6 10M > junk403 2>&1 &
proc403=$!
addfile.sh dist_0 \\\\.\\dist_2_7 10M > junk404 2>&1 &
proc404=$!
addfile.sh dist_0 \\\\.\\dist_2_8 10M > junk405 2>&1 &
proc405=$!
addfile.sh dist_0 \\\\.\\dist_2_9 10M > junk406 2>&1 &
proc406=$!
addfile.sh dist_0 \\\\.\\dist_2_10 10M > junk407 2>&1 &
proc407=$!
addfile.sh dist_0 \\\\.\\dist_2_11 10M > junk408 2>&1 &
proc408=$!

```

```

wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

```

```

addfile.sh dist_0 \\\\.\\dist_3_1 10M > junk398 2>&1 &
proc398=$!
addfile.sh dist_0 \\\\.\\dist_3_2 10M > junk399 2>&1 &
proc399=$!
addfile.sh dist_0 \\\\.\\dist_3_3 10M > junk400 2>&1 &
proc400=$!
addfile.sh dist_0 \\\\.\\dist_3_4 10M > junk401 2>&1 &
proc401=$!
addfile.sh dist_0 \\\\.\\dist_3_5 10M > junk402 2>&1 &
proc402=$!
addfile.sh dist_0 \\\\.\\dist_3_6 10M > junk403 2>&1 &
proc403=$!
addfile.sh dist_0 \\\\.\\dist_3_7 10M > junk404 2>&1 &
proc404=$!
addfile.sh dist_0 \\\\.\\dist_3_8 10M > junk405 2>&1 &
proc405=$!

```

```

addfile.sh dist_0 \\\\.\\dist_3_9 10M > junk406 2>&1 &
proc406=$!
addfile.sh dist_0 \\\\.\\dist_3_10 10M > junk407 2>&1 &
proc407=$!
addfile.sh dist_0 \\\\.\\dist_3_11 10M > junk408 2>&1 &
proc408=$!

```

```

wait $proc398
proc398=$?
procsum=`expr $procsum + $proc398`
wait $proc399
proc399=$?
procsum=`expr $procsum + $proc399`
wait $proc400
proc400=$?
procsum=`expr $procsum + $proc400`
wait $proc401
proc401=$?
procsum=`expr $procsum + $proc401`
wait $proc402
proc402=$?
procsum=`expr $procsum + $proc402`
wait $proc403
proc403=$?
procsum=`expr $procsum + $proc403`
wait $proc404
proc404=$?
procsum=`expr $procsum + $proc404`
wait $proc405
proc405=$?
procsum=`expr $procsum + $proc405`
wait $proc406
proc406=$?
procsum=`expr $procsum + $proc406`
wait $proc407
proc407=$?
procsum=`expr $procsum + $proc407`
wait $proc408
proc408=$?
procsum=`expr $procsum + $proc408`

```

```

proc=`expr $procsum + 0`
proc=`expr $proc % 127`

```

```

if test $proc -ne 0
then
    exit 1;
else
    exit 0;
fi

```

step6createddviews.sh

```

#!/sh
$SQLPLUS internal/internal @step6createddviews > junk 2>&1

if test $? -ne 0
then
    exit 1;
else
    exit 0;
fi

```

step6createddviews.sql

```
spool step6createddviews.log
@@ORACLE_HOME%\rdbms\admin\catalog
@@ORACLE_HOME%\rdbms\admin\catproc
@@ORACLE_HOME%\rdbms\admin\catparr
spool off
exit sql.sqlcode;
```

step18createrollsegs.sh

```
#!/sh
$SQLPLUS tpcc/tpcc @step18createrollsegs > junk 2>&1

if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi
```

step18createrollsegs.sql

```
spool create_rollsegs.log
set echo on

alter rollback segment t_0_1 offline;
alter rollback segment t_0_2 offline;
alter rollback segment t_0_3 offline;
alter rollback segment t_0_4 offline;
alter rollback segment t_0_5 offline;
alter rollback segment t_0_6 offline;
alter rollback segment t_0_7 offline;
alter rollback segment t_0_8 offline;
alter rollback segment t_0_9 offline;
alter rollback segment t_0_10 offline;
alter rollback segment t_0_11 offline;
alter rollback segment t_0_12 offline;
alter rollback segment t_0_13 offline;
alter rollback segment t_0_14 offline;
alter rollback segment t_0_15 offline;
alter rollback segment t_0_16 offline;
```

```
alter rollback segment t_0_17 offline;
alter rollback segment t_0_18 offline;
alter rollback segment t_0_19 offline;
alter rollback segment t_0_20 offline;
alter rollback segment t_0_21 offline;
alter rollback segment t_0_22 offline;
alter rollback segment t_0_23 offline;
alter rollback segment t_0_24 offline;
alter rollback segment t_0_25 offline;
alter rollback segment t_0_26 offline;
alter rollback segment t_0_27 offline;
alter rollback segment t_0_28 offline;
alter rollback segment t_0_29 offline;
alter rollback segment t_0_30 offline;
alter rollback segment t_0_31 offline;
alter rollback segment t_0_32 offline;
alter rollback segment t_0_33 offline;
alter rollback segment t_0_34 offline;
alter rollback segment t_0_35 offline;
alter rollback segment t_0_36 offline;
alter rollback segment t_0_37 offline;
alter rollback segment t_0_38 offline;
alter rollback segment t_0_39 offline;
alter rollback segment t_0_40 offline;
alter rollback segment t_0_41 offline;
alter rollback segment t_0_42 offline;
alter rollback segment t_0_43 offline;
alter rollback segment t_0_44 offline;
alter rollback segment t_0_45 offline;
alter rollback segment t_0_46 offline;
alter rollback segment t_0_47 offline;
```

```
alter rollback segment t_0_48 offline;
alter rollback segment t_0_49 offline;
alter rollback segment t_0_50 offline;
alter rollback segment t_0_51 offline;
alter rollback segment t_0_52 offline;
alter rollback segment t_0_53 offline;
alter rollback segment t_0_54 offline;
alter rollback segment t_0_55 offline;
alter rollback segment t_0_56 offline;
alter rollback segment t_0_57 offline;
alter rollback segment t_0_58 offline;
alter rollback segment t_0_59 offline;
alter rollback segment t_0_60 offline;
alter rollback segment t_0_61 offline;
alter rollback segment t_0_62 offline;
alter rollback segment t_0_63 offline;
alter rollback segment t_0_64 offline;
alter rollback segment t_0_65 offline;
alter rollback segment t_0_66 offline;
alter rollback segment t_0_67 offline;
alter rollback segment t_0_68 offline;
alter rollback segment t_0_69 offline;
alter rollback segment t_0_70 offline;
alter rollback segment t_0_71 offline;
alter rollback segment t_0_72 offline;
alter rollback segment t_0_73 offline;
alter rollback segment t_0_74 offline;
alter rollback segment t_0_75 offline;
alter rollback segment t_0_76 offline;
alter rollback segment t_0_77 offline;
alter rollback segment t_0_78 offline;
alter rollback segment t_0_79 offline;
```

```
alter rollback segment t_0_80 offline;
alter rollback segment t_0_81 offline;
alter rollback segment t_0_82 offline;
alter rollback segment t_0_83 offline;
alter rollback segment t_0_84 offline;
alter rollback segment t_0_85 offline;
alter rollback segment t_0_86 offline;
alter rollback segment t_0_87 offline;
alter rollback segment t_0_88 offline;
alter rollback segment t_0_89 offline;
alter rollback segment t_0_90 offline;
alter rollback segment t_0_91 offline;
alter rollback segment t_0_92 offline;
alter rollback segment t_0_93 offline;
alter rollback segment t_0_94 offline;
alter rollback segment t_0_95 offline;
alter rollback segment t_0_96 offline;
alter rollback segment t_0_97 offline;
alter rollback segment t_0_98 offline;
alter rollback segment t_0_99 offline;
alter rollback segment t_0_100 offline;
alter rollback segment t_0_101 offline;
alter rollback segment t_0_102 offline;
alter rollback segment t_0_103 offline;
alter rollback segment t_0_104 offline;
alter rollback segment t_0_105 offline;
alter rollback segment t_0_106 offline;
alter rollback segment t_0_107 offline;
alter rollback segment t_0_108 offline;
alter rollback segment t_0_109 offline;
alter rollback segment t_0_110 offline;
```

```
alter rollback segment t_0_111 offline;
alter rollback segment t_0_112 offline;
alter rollback segment t_0_113 offline;
alter rollback segment t_0_114 offline;
alter rollback segment t_0_115 offline;
alter rollback segment t_0_116 offline;
alter rollback segment t_0_117 offline;
alter rollback segment t_0_118 offline;
alter rollback segment t_0_119 offline;
alter rollback segment t_0_120 offline;
alter rollback segment t_0_121 offline;
alter rollback segment t_0_122 offline;
alter rollback segment t_0_123 offline;
alter rollback segment t_0_124 offline;
alter rollback segment t_0_125 offline;
alter rollback segment t_0_126 offline;
alter rollback segment t_0_127 offline;
alter rollback segment t_0_128 offline;
alter rollback segment t_0_129 offline;
alter rollback segment t_0_130 offline;
alter rollback segment t_0_131 offline;
alter rollback segment t_0_132 offline;
alter rollback segment t_0_133 offline;
alter rollback segment t_0_134 offline;
alter rollback segment t_0_135 offline;
alter rollback segment t_0_136 offline;
alter rollback segment t_0_137 offline;
alter rollback segment t_0_138 offline;
alter rollback segment t_0_139 offline;
alter rollback segment t_0_140 offline;
alter rollback segment t_0_141 offline;
alter rollback segment t_0_142 offline;
```

```
alter rollback segment t_0_143 offline;
alter rollback segment t_0_144 offline;
alter rollback segment t_0_145 offline;
alter rollback segment t_0_146 offline;
alter rollback segment t_0_147 offline;
alter rollback segment t_0_148 offline;
alter rollback segment t_0_149 offline;
alter rollback segment t_0_150 offline;
alter rollback segment t_0_151 offline;
alter rollback segment t_0_152 offline;
alter rollback segment t_0_153 offline;
alter rollback segment t_0_154 offline;
alter rollback segment t_0_155 offline;
alter rollback segment t_0_156 offline;
alter rollback segment t_0_157 offline;
alter rollback segment t_0_158 offline;
alter rollback segment t_0_159 offline;
alter rollback segment t_0_160 offline;
alter rollback segment t_0_161 offline;
alter rollback segment t_0_162 offline;
alter rollback segment t_0_163 offline;
alter rollback segment t_0_164 offline;
alter rollback segment t_0_165 offline;
alter rollback segment t_0_166 offline;
alter rollback segment t_0_167 offline;
alter rollback segment t_0_168 offline;
alter rollback segment t_0_169 offline;
alter rollback segment t_0_170 offline;
alter rollback segment t_0_171 offline;
alter rollback segment t_0_172 offline;
alter rollback segment t_0_173 offline;
```

```

alter rollback segment t_0_174 offline;
alter rollback segment t_0_175 offline;
alter rollback segment t_0_176 offline;
alter rollback segment t_0_177 offline;
alter rollback segment t_0_178 offline;
alter rollback segment t_0_179 offline;
alter rollback segment t_0_180 offline;
alter rollback segment t_0_181 offline;
alter rollback segment t_0_182 offline;
alter rollback segment t_0_183 offline;
alter rollback segment t_0_184 offline;
alter rollback segment t_0_185 offline;
alter rollback segment t_0_186 offline;
alter rollback segment t_0_187 offline;
alter rollback segment t_0_188 offline;
alter rollback segment t_0_189 offline;
alter rollback segment t_0_190 offline;
alter rollback segment t_0_191 offline;
alter rollback segment t_0_192 offline;
alter rollback segment t_0_193 offline;
alter rollback segment t_0_194 offline;
alter rollback segment t_0_195 offline;
alter rollback segment t_0_196 offline;
alter rollback segment t_0_197 offline;
alter rollback segment t_0_198 offline;
alter rollback segment t_0_199 offline;
alter rollback segment t_0_200 offline;
drop tablespace roll_0 including contents;

create tablespace roll_0 datafile '\\.\roll_0_0' size 170M reuse extent
management local uniform size 200K nologging ;
alter tablespace roll_0 add datafile '\\.\roll_0_1' size      170M
reuse;
alter tablespace roll_0 add datafile '\\.\roll_0_2' size      170M
reuse;

```

```

alter tablespace roll_0 add datafile '\\.\roll_0_3' size      170M
reuse;
alter tablespace roll_0 add datafile '\\.\roll_0_4' size      170M
reuse;
alter tablespace roll_0 add datafile '\\.\roll_0_5' size      170M
reuse;
alter tablespace roll_0 add datafile '\\.\roll_0_6' size      170M
reuse;
alter tablespace roll_0 add datafile '\\.\roll_0_7' size      170M
reuse;
alter tablespace roll_0 add datafile '\\.\roll_0_8' size      170M
reuse;
alter tablespace roll_0 add datafile '\\.\roll_0_9' size      170M
reuse;
alter tablespace roll_0 add datafile '\\.\roll_0_10' size     170M
reuse;
alter tablespace roll_0 add datafile '\\.\roll_0_11' size     170M
reuse;

create rollback segment t_0_1 tablespace roll_0;
create rollback segment t_0_2 tablespace roll_0;
create rollback segment t_0_3 tablespace roll_0;
create rollback segment t_0_4 tablespace roll_0;
create rollback segment t_0_5 tablespace roll_0;
create rollback segment t_0_6 tablespace roll_0;
create rollback segment t_0_7 tablespace roll_0;
create rollback segment t_0_8 tablespace roll_0;
create rollback segment t_0_9 tablespace roll_0;
create rollback segment t_0_10 tablespace roll_0;
create rollback segment t_0_11 tablespace roll_0;
create rollback segment t_0_12 tablespace roll_0;
create rollback segment t_0_13 tablespace roll_0;
create rollback segment t_0_14 tablespace roll_0;
create rollback segment t_0_15 tablespace roll_0;
create rollback segment t_0_16 tablespace roll_0;
create rollback segment t_0_17 tablespace roll_0;
create rollback segment t_0_18 tablespace roll_0;
create rollback segment t_0_19 tablespace roll_0;
create rollback segment t_0_20 tablespace roll_0;
create rollback segment t_0_21 tablespace roll_0;
create rollback segment t_0_22 tablespace roll_0;
create rollback segment t_0_23 tablespace roll_0;
create rollback segment t_0_24 tablespace roll_0;
create rollback segment t_0_25 tablespace roll_0;
create rollback segment t_0_26 tablespace roll_0;
create rollback segment t_0_27 tablespace roll_0;
create rollback segment t_0_28 tablespace roll_0;
create rollback segment t_0_29 tablespace roll_0;
create rollback segment t_0_30 tablespace roll_0;
create rollback segment t_0_31 tablespace roll_0;
create rollback segment t_0_32 tablespace roll_0;
create rollback segment t_0_33 tablespace roll_0;
create rollback segment t_0_34 tablespace roll_0;
create rollback segment t_0_35 tablespace roll_0;
create rollback segment t_0_36 tablespace roll_0;
create rollback segment t_0_37 tablespace roll_0;
create rollback segment t_0_38 tablespace roll_0;
create rollback segment t_0_39 tablespace roll_0;
create rollback segment t_0_40 tablespace roll_0;
create rollback segment t_0_41 tablespace roll_0;
create rollback segment t_0_42 tablespace roll_0;
create rollback segment t_0_43 tablespace roll_0;
create rollback segment t_0_44 tablespace roll_0;

```



```
create rollback segment t_0_171 tablespace roll_0;
create rollback segment t_0_172 tablespace roll_0;
create rollback segment t_0_173 tablespace roll_0;
create rollback segment t_0_174 tablespace roll_0;
create rollback segment t_0_175 tablespace roll_0;
create rollback segment t_0_176 tablespace roll_0;
create rollback segment t_0_177 tablespace roll_0;
create rollback segment t_0_178 tablespace roll_0;
create rollback segment t_0_179 tablespace roll_0;
create rollback segment t_0_180 tablespace roll_0;
create rollback segment t_0_181 tablespace roll_0;
create rollback segment t_0_182 tablespace roll_0;
create rollback segment t_0_183 tablespace roll_0;
create rollback segment t_0_184 tablespace roll_0;
create rollback segment t_0_185 tablespace roll_0;
create rollback segment t_0_186 tablespace roll_0;
create rollback segment t_0_187 tablespace roll_0;
create rollback segment t_0_188 tablespace roll_0;
create rollback segment t_0_189 tablespace roll_0;
create rollback segment t_0_190 tablespace roll_0;
create rollback segment t_0_191 tablespace roll_0;
create rollback segment t_0_192 tablespace roll_0;
create rollback segment t_0_193 tablespace roll_0;
create rollback segment t_0_194 tablespace roll_0;
create rollback segment t_0_195 tablespace roll_0;
create rollback segment t_0_196 tablespace roll_0;
create rollback segment t_0_197 tablespace roll_0;
create rollback segment t_0_198 tablespace roll_0;
create rollback segment t_0_199 tablespace roll_0;
create rollback segment t_0_200 tablespace roll_0;
```

```
alter rollback segment t_1_1 offline;
alter rollback segment t_1_2 offline;
alter rollback segment t_1_3 offline;
alter rollback segment t_1_4 offline;
alter rollback segment t_1_5 offline;
alter rollback segment t_1_6 offline;
alter rollback segment t_1_7 offline;
alter rollback segment t_1_8 offline;
alter rollback segment t_1_9 offline;
alter rollback segment t_1_10 offline;
alter rollback segment t_1_11 offline;
alter rollback segment t_1_12 offline;
alter rollback segment t_1_13 offline;
alter rollback segment t_1_14 offline;
alter rollback segment t_1_15 offline;
alter rollback segment t_1_16 offline;
```

```
alter rollback segment t_1_17 offline;
alter rollback segment t_1_18 offline;
alter rollback segment t_1_19 offline;
alter rollback segment t_1_20 offline;
alter rollback segment t_1_21 offline;
alter rollback segment t_1_22 offline;
alter rollback segment t_1_23 offline;
alter rollback segment t_1_24 offline;
alter rollback segment t_1_25 offline;
alter rollback segment t_1_26 offline;
alter rollback segment t_1_27 offline;
alter rollback segment t_1_28 offline;
alter rollback segment t_1_29 offline;
alter rollback segment t_1_30 offline;
alter rollback segment t_1_31 offline;
alter rollback segment t_1_32 offline;
alter rollback segment t_1_33 offline;
alter rollback segment t_1_34 offline;
alter rollback segment t_1_35 offline;
alter rollback segment t_1_36 offline;
alter rollback segment t_1_37 offline;
alter rollback segment t_1_38 offline;
alter rollback segment t_1_39 offline;
alter rollback segment t_1_40 offline;
alter rollback segment t_1_41 offline;
alter rollback segment t_1_42 offline;
alter rollback segment t_1_43 offline;
alter rollback segment t_1_44 offline;
alter rollback segment t_1_45 offline;
alter rollback segment t_1_46 offline;
alter rollback segment t_1_47 offline;
alter rollback segment t_1_48 offline;
```

```
alter rollback segment t_1_49 offline;
alter rollback segment t_1_50 offline;
alter rollback segment t_1_51 offline;
alter rollback segment t_1_52 offline;
alter rollback segment t_1_53 offline;
alter rollback segment t_1_54 offline;
alter rollback segment t_1_55 offline;
alter rollback segment t_1_56 offline;
alter rollback segment t_1_57 offline;
alter rollback segment t_1_58 offline;
alter rollback segment t_1_59 offline;
alter rollback segment t_1_60 offline;
alter rollback segment t_1_61 offline;
alter rollback segment t_1_62 offline;
alter rollback segment t_1_63 offline;
alter rollback segment t_1_64 offline;
alter rollback segment t_1_65 offline;
alter rollback segment t_1_66 offline;
alter rollback segment t_1_67 offline;
alter rollback segment t_1_68 offline;
alter rollback segment t_1_69 offline;
alter rollback segment t_1_70 offline;
alter rollback segment t_1_71 offline;
alter rollback segment t_1_72 offline;
alter rollback segment t_1_73 offline;
alter rollback segment t_1_74 offline;
alter rollback segment t_1_75 offline;
alter rollback segment t_1_76 offline;
alter rollback segment t_1_77 offline;
alter rollback segment t_1_78 offline;
alter rollback segment t_1_79 offline;
```

```
alter rollback segment t_1_80 offline;
alter rollback segment t_1_81 offline;
alter rollback segment t_1_82 offline;
alter rollback segment t_1_83 offline;
alter rollback segment t_1_84 offline;
alter rollback segment t_1_85 offline;
alter rollback segment t_1_86 offline;
alter rollback segment t_1_87 offline;
alter rollback segment t_1_88 offline;
alter rollback segment t_1_89 offline;
alter rollback segment t_1_90 offline;
alter rollback segment t_1_91 offline;
alter rollback segment t_1_92 offline;
alter rollback segment t_1_93 offline;
alter rollback segment t_1_94 offline;
alter rollback segment t_1_95 offline;
alter rollback segment t_1_96 offline;
alter rollback segment t_1_97 offline;
alter rollback segment t_1_98 offline;
alter rollback segment t_1_99 offline;
alter rollback segment t_1_100 offline;
alter rollback segment t_1_101 offline;
alter rollback segment t_1_102 offline;
alter rollback segment t_1_103 offline;
alter rollback segment t_1_104 offline;
alter rollback segment t_1_105 offline;
alter rollback segment t_1_106 offline;
alter rollback segment t_1_107 offline;
alter rollback segment t_1_108 offline;
alter rollback segment t_1_109 offline;
alter rollback segment t_1_110 offline;
alter rollback segment t_1_111 offline;
```

```
alter rollback segment t_1_112 offline;
alter rollback segment t_1_113 offline;
alter rollback segment t_1_114 offline;
alter rollback segment t_1_115 offline;
alter rollback segment t_1_116 offline;
alter rollback segment t_1_117 offline;
alter rollback segment t_1_118 offline;
alter rollback segment t_1_119 offline;
alter rollback segment t_1_120 offline;
alter rollback segment t_1_121 offline;
alter rollback segment t_1_122 offline;
alter rollback segment t_1_123 offline;
alter rollback segment t_1_124 offline;
alter rollback segment t_1_125 offline;
alter rollback segment t_1_126 offline;
alter rollback segment t_1_127 offline;
alter rollback segment t_1_128 offline;
alter rollback segment t_1_129 offline;
alter rollback segment t_1_130 offline;
alter rollback segment t_1_131 offline;
alter rollback segment t_1_132 offline;
alter rollback segment t_1_133 offline;
alter rollback segment t_1_134 offline;
alter rollback segment t_1_135 offline;
alter rollback segment t_1_136 offline;
alter rollback segment t_1_137 offline;
alter rollback segment t_1_138 offline;
alter rollback segment t_1_139 offline;
alter rollback segment t_1_140 offline;
alter rollback segment t_1_141 offline;
alter rollback segment t_1_142 offline;
```

```
alter rollback segment t_1_143 offline;
alter rollback segment t_1_144 offline;
alter rollback segment t_1_145 offline;
alter rollback segment t_1_146 offline;
alter rollback segment t_1_147 offline;
alter rollback segment t_1_148 offline;
alter rollback segment t_1_149 offline;
alter rollback segment t_1_150 offline;
alter rollback segment t_1_151 offline;
alter rollback segment t_1_152 offline;
alter rollback segment t_1_153 offline;
alter rollback segment t_1_154 offline;
alter rollback segment t_1_155 offline;
alter rollback segment t_1_156 offline;
alter rollback segment t_1_157 offline;
alter rollback segment t_1_158 offline;
alter rollback segment t_1_159 offline;
alter rollback segment t_1_160 offline;
alter rollback segment t_1_161 offline;
alter rollback segment t_1_162 offline;
alter rollback segment t_1_163 offline;
alter rollback segment t_1_164 offline;
alter rollback segment t_1_165 offline;
alter rollback segment t_1_166 offline;
alter rollback segment t_1_167 offline;
alter rollback segment t_1_168 offline;
alter rollback segment t_1_169 offline;
alter rollback segment t_1_170 offline;
alter rollback segment t_1_171 offline;
alter rollback segment t_1_172 offline;
alter rollback segment t_1_173 offline;
alter rollback segment t_1_174 offline;
```

```

alter rollback segment t_1_175 offline;
alter rollback segment t_1_176 offline;
alter rollback segment t_1_177 offline;
alter rollback segment t_1_178 offline;
alter rollback segment t_1_179 offline;
alter rollback segment t_1_180 offline;
alter rollback segment t_1_181 offline;
alter rollback segment t_1_182 offline;
alter rollback segment t_1_183 offline;
alter rollback segment t_1_184 offline;
alter rollback segment t_1_185 offline;
alter rollback segment t_1_186 offline;
alter rollback segment t_1_187 offline;
alter rollback segment t_1_188 offline;
alter rollback segment t_1_189 offline;
alter rollback segment t_1_190 offline;
alter rollback segment t_1_191 offline;
alter rollback segment t_1_192 offline;
alter rollback segment t_1_193 offline;
alter rollback segment t_1_194 offline;
alter rollback segment t_1_195 offline;
alter rollback segment t_1_196 offline;
alter rollback segment t_1_197 offline;
alter rollback segment t_1_198 offline;
alter rollback segment t_1_199 offline;
alter rollback segment t_1_200 offline;

drop tablespace roll_1 including contents;

create tablespace roll_1 datafile '\\.\roll_1_0' size 170M reuse extent
management local uniform size 200K nologging ;
alter tablespace roll_1 add datafile '\\.\roll_1_1' size      170M
reuse;
alter tablespace roll_1 add datafile '\\.\roll_1_2' size      170M
reuse;
alter tablespace roll_1 add datafile '\\.\roll_1_3' size      170M
reuse;

```

```

alter tablespace roll_1 add datafile '\\.\roll_1_4' size      170M
reuse;
alter tablespace roll_1 add datafile '\\.\roll_1_5' size      170M
reuse;
alter tablespace roll_1 add datafile '\\.\roll_1_6' size      170M
reuse;
alter tablespace roll_1 add datafile '\\.\roll_1_7' size      170M
reuse;
alter tablespace roll_1 add datafile '\\.\roll_1_8' size      170M
reuse;
alter tablespace roll_1 add datafile '\\.\roll_1_9' size      170M
reuse;
alter tablespace roll_1 add datafile '\\.\roll_1_10' size     170M
reuse;
alter tablespace roll_1 add datafile '\\.\roll_1_11' size     170M
reuse;

create rollback segment t_1_1 tablespace roll_1;
create rollback segment t_1_2 tablespace roll_1;
create rollback segment t_1_3 tablespace roll_1;
create rollback segment t_1_4 tablespace roll_1;
create rollback segment t_1_5 tablespace roll_1;
create rollback segment t_1_6 tablespace roll_1;
create rollback segment t_1_7 tablespace roll_1;
create rollback segment t_1_8 tablespace roll_1;
create rollback segment t_1_9 tablespace roll_1;
create rollback segment t_1_10 tablespace roll_1;
create rollback segment t_1_11 tablespace roll_1;
create rollback segment t_1_12 tablespace roll_1;
create rollback segment t_1_13 tablespace roll_1;
create rollback segment t_1_14 tablespace roll_1;
create rollback segment t_1_15 tablespace roll_1;
create rollback segment t_1_16 tablespace roll_1;
create rollback segment t_1_17 tablespace roll_1;
create rollback segment t_1_18 tablespace roll_1;
create rollback segment t_1_19 tablespace roll_1;
create rollback segment t_1_20 tablespace roll_1;
create rollback segment t_1_21 tablespace roll_1;
create rollback segment t_1_22 tablespace roll_1;
create rollback segment t_1_23 tablespace roll_1;
create rollback segment t_1_24 tablespace roll_1;
create rollback segment t_1_25 tablespace roll_1;
create rollback segment t_1_26 tablespace roll_1;
create rollback segment t_1_27 tablespace roll_1;
create rollback segment t_1_28 tablespace roll_1;
create rollback segment t_1_29 tablespace roll_1;
create rollback segment t_1_30 tablespace roll_1;
create rollback segment t_1_31 tablespace roll_1;
create rollback segment t_1_32 tablespace roll_1;
create rollback segment t_1_33 tablespace roll_1;
create rollback segment t_1_34 tablespace roll_1;
create rollback segment t_1_35 tablespace roll_1;
create rollback segment t_1_36 tablespace roll_1;
create rollback segment t_1_37 tablespace roll_1;
create rollback segment t_1_38 tablespace roll_1;
create rollback segment t_1_39 tablespace roll_1;
create rollback segment t_1_40 tablespace roll_1;
create rollback segment t_1_41 tablespace roll_1;
create rollback segment t_1_42 tablespace roll_1;
create rollback segment t_1_43 tablespace roll_1;
create rollback segment t_1_44 tablespace roll_1;
create rollback segment t_1_45 tablespace roll_1;
create rollback segment t_1_46 tablespace roll_1;

```



```
create rollback segment t_1_173 tablespace roll_1;
create rollback segment t_1_174 tablespace roll_1;
create rollback segment t_1_175 tablespace roll_1;
create rollback segment t_1_176 tablespace roll_1;
create rollback segment t_1_177 tablespace roll_1;
create rollback segment t_1_178 tablespace roll_1;
create rollback segment t_1_179 tablespace roll_1;
create rollback segment t_1_180 tablespace roll_1;
create rollback segment t_1_181 tablespace roll_1;
create rollback segment t_1_182 tablespace roll_1;
create rollback segment t_1_183 tablespace roll_1;
create rollback segment t_1_184 tablespace roll_1;
create rollback segment t_1_185 tablespace roll_1;
create rollback segment t_1_186 tablespace roll_1;
create rollback segment t_1_187 tablespace roll_1;
create rollback segment t_1_188 tablespace roll_1;
create rollback segment t_1_189 tablespace roll_1;
create rollback segment t_1_190 tablespace roll_1;
create rollback segment t_1_191 tablespace roll_1;
create rollback segment t_1_192 tablespace roll_1;
create rollback segment t_1_193 tablespace roll_1;
create rollback segment t_1_194 tablespace roll_1;
create rollback segment t_1_195 tablespace roll_1;
create rollback segment t_1_196 tablespace roll_1;
create rollback segment t_1_197 tablespace roll_1;
create rollback segment t_1_198 tablespace roll_1;
create rollback segment t_1_199 tablespace roll_1;
create rollback segment t_1_200 tablespace roll_1;
```

```
alter rollback segment t_2_1 offline;
alter rollback segment t_2_2 offline;
alter rollback segment t_2_3 offline;
alter rollback segment t_2_4 offline;
alter rollback segment t_2_5 offline;
alter rollback segment t_2_6 offline;
alter rollback segment t_2_7 offline;
alter rollback segment t_2_8 offline;
alter rollback segment t_2_9 offline;
alter rollback segment t_2_10 offline;
alter rollback segment t_2_11 offline;
alter rollback segment t_2_12 offline;
alter rollback segment t_2_13 offline;
alter rollback segment t_2_14 offline;
alter rollback segment t_2_15 offline;
alter rollback segment t_2_16 offline;
alter rollback segment t_2_17 offline;
```

```
alter rollback segment t_2_18 offline;
alter rollback segment t_2_19 offline;
alter rollback segment t_2_20 offline;
alter rollback segment t_2_21 offline;
alter rollback segment t_2_22 offline;
alter rollback segment t_2_23 offline;
alter rollback segment t_2_24 offline;
alter rollback segment t_2_25 offline;
alter rollback segment t_2_26 offline;
alter rollback segment t_2_27 offline;
alter rollback segment t_2_28 offline;
alter rollback segment t_2_29 offline;
alter rollback segment t_2_30 offline;
alter rollback segment t_2_31 offline;
alter rollback segment t_2_32 offline;
alter rollback segment t_2_33 offline;
alter rollback segment t_2_34 offline;
alter rollback segment t_2_35 offline;
alter rollback segment t_2_36 offline;
alter rollback segment t_2_37 offline;
alter rollback segment t_2_38 offline;
alter rollback segment t_2_39 offline;
alter rollback segment t_2_40 offline;
alter rollback segment t_2_41 offline;
alter rollback segment t_2_42 offline;
alter rollback segment t_2_43 offline;
alter rollback segment t_2_44 offline;
alter rollback segment t_2_45 offline;
alter rollback segment t_2_46 offline;
alter rollback segment t_2_47 offline;
alter rollback segment t_2_48 offline;
alter rollback segment t_2_49 offline;
```

```
alter rollback segment t_2_50 offline;
alter rollback segment t_2_51 offline;
alter rollback segment t_2_52 offline;
alter rollback segment t_2_53 offline;
alter rollback segment t_2_54 offline;
alter rollback segment t_2_55 offline;
alter rollback segment t_2_56 offline;
alter rollback segment t_2_57 offline;
alter rollback segment t_2_58 offline;
alter rollback segment t_2_59 offline;
alter rollback segment t_2_60 offline;
alter rollback segment t_2_61 offline;
alter rollback segment t_2_62 offline;
alter rollback segment t_2_63 offline;
alter rollback segment t_2_64 offline;
alter rollback segment t_2_65 offline;
alter rollback segment t_2_66 offline;
alter rollback segment t_2_67 offline;
alter rollback segment t_2_68 offline;
alter rollback segment t_2_69 offline;
alter rollback segment t_2_70 offline;
alter rollback segment t_2_71 offline;
alter rollback segment t_2_72 offline;
alter rollback segment t_2_73 offline;
alter rollback segment t_2_74 offline;
alter rollback segment t_2_75 offline;
alter rollback segment t_2_76 offline;
alter rollback segment t_2_77 offline;
alter rollback segment t_2_78 offline;
alter rollback segment t_2_79 offline;
alter rollback segment t_2_80 offline;
```

```
alter rollback segment t_2_81 offline;
alter rollback segment t_2_82 offline;
alter rollback segment t_2_83 offline;
alter rollback segment t_2_84 offline;
alter rollback segment t_2_85 offline;
alter rollback segment t_2_86 offline;
alter rollback segment t_2_87 offline;
alter rollback segment t_2_88 offline;
alter rollback segment t_2_89 offline;
alter rollback segment t_2_90 offline;
alter rollback segment t_2_91 offline;
alter rollback segment t_2_92 offline;
alter rollback segment t_2_93 offline;
alter rollback segment t_2_94 offline;
alter rollback segment t_2_95 offline;
alter rollback segment t_2_96 offline;
alter rollback segment t_2_97 offline;
alter rollback segment t_2_98 offline;
alter rollback segment t_2_99 offline;
alter rollback segment t_2_100 offline;
alter rollback segment t_2_101 offline;
alter rollback segment t_2_102 offline;
alter rollback segment t_2_103 offline;
alter rollback segment t_2_104 offline;
alter rollback segment t_2_105 offline;
alter rollback segment t_2_106 offline;
alter rollback segment t_2_107 offline;
alter rollback segment t_2_108 offline;
alter rollback segment t_2_109 offline;
alter rollback segment t_2_110 offline;
alter rollback segment t_2_111 offline;
alter rollback segment t_2_112 offline;
```

```
alter rollback segment t_2_113 offline;
alter rollback segment t_2_114 offline;
alter rollback segment t_2_115 offline;
alter rollback segment t_2_116 offline;
alter rollback segment t_2_117 offline;
alter rollback segment t_2_118 offline;
alter rollback segment t_2_119 offline;
alter rollback segment t_2_120 offline;
alter rollback segment t_2_121 offline;
alter rollback segment t_2_122 offline;
alter rollback segment t_2_123 offline;
alter rollback segment t_2_124 offline;
alter rollback segment t_2_125 offline;
alter rollback segment t_2_126 offline;
alter rollback segment t_2_127 offline;
alter rollback segment t_2_128 offline;
alter rollback segment t_2_129 offline;
alter rollback segment t_2_130 offline;
alter rollback segment t_2_131 offline;
alter rollback segment t_2_132 offline;
alter rollback segment t_2_133 offline;
alter rollback segment t_2_134 offline;
alter rollback segment t_2_135 offline;
alter rollback segment t_2_136 offline;
alter rollback segment t_2_137 offline;
alter rollback segment t_2_138 offline;
alter rollback segment t_2_139 offline;
alter rollback segment t_2_140 offline;
alter rollback segment t_2_141 offline;
alter rollback segment t_2_142 offline;
alter rollback segment t_2_143 offline;
```

```
alter rollback segment t_2_144 offline;
alter rollback segment t_2_145 offline;
alter rollback segment t_2_146 offline;
alter rollback segment t_2_147 offline;
alter rollback segment t_2_148 offline;
alter rollback segment t_2_149 offline;
alter rollback segment t_2_150 offline;
alter rollback segment t_2_151 offline;
alter rollback segment t_2_152 offline;
alter rollback segment t_2_153 offline;
alter rollback segment t_2_154 offline;
alter rollback segment t_2_155 offline;
alter rollback segment t_2_156 offline;
alter rollback segment t_2_157 offline;
alter rollback segment t_2_158 offline;
alter rollback segment t_2_159 offline;
alter rollback segment t_2_160 offline;
alter rollback segment t_2_161 offline;
alter rollback segment t_2_162 offline;
alter rollback segment t_2_163 offline;
alter rollback segment t_2_164 offline;
alter rollback segment t_2_165 offline;
alter rollback segment t_2_166 offline;
alter rollback segment t_2_167 offline;
alter rollback segment t_2_168 offline;
alter rollback segment t_2_169 offline;
alter rollback segment t_2_170 offline;
alter rollback segment t_2_171 offline;
alter rollback segment t_2_172 offline;
alter rollback segment t_2_173 offline;
alter rollback segment t_2_174 offline;
alter rollback segment t_2_175 offline;
```

```

alter rollback segment t_2_176 offline;
alter rollback segment t_2_177 offline;
alter rollback segment t_2_178 offline;
alter rollback segment t_2_179 offline;
alter rollback segment t_2_180 offline;
alter rollback segment t_2_181 offline;
alter rollback segment t_2_182 offline;
alter rollback segment t_2_183 offline;
alter rollback segment t_2_184 offline;
alter rollback segment t_2_185 offline;
alter rollback segment t_2_186 offline;
alter rollback segment t_2_187 offline;
alter rollback segment t_2_188 offline;
alter rollback segment t_2_189 offline;
alter rollback segment t_2_190 offline;
alter rollback segment t_2_191 offline;
alter rollback segment t_2_192 offline;
alter rollback segment t_2_193 offline;
alter rollback segment t_2_194 offline;
alter rollback segment t_2_195 offline;
alter rollback segment t_2_196 offline;
alter rollback segment t_2_197 offline;
alter rollback segment t_2_198 offline;
alter rollback segment t_2_199 offline;
alter rollback segment t_2_200 offline;
drop tablespace roll_2 including contents;

create tablespace roll_2 datafile '\\.\roll_2_0' size 170M reuse extent
management local uniform size 200K nologging ;
alter tablespace roll_2 add datafile '\\.\roll_2_1' size      170M
reuse;
alter tablespace roll_2 add datafile '\\.\roll_2_2' size      170M
reuse;
alter tablespace roll_2 add datafile '\\.\roll_2_3' size      170M
reuse;
alter tablespace roll_2 add datafile '\\.\roll_2_4' size      170M
reuse;

```

```

alter tablespace roll_2 add datafile '\\.\roll_2_5' size      170M
reuse;
alter tablespace roll_2 add datafile '\\.\roll_2_6' size      170M
reuse;
alter tablespace roll_2 add datafile '\\.\roll_2_7' size      170M
reuse;
alter tablespace roll_2 add datafile '\\.\roll_2_8' size      170M
reuse;
alter tablespace roll_2 add datafile '\\.\roll_2_9' size      170M
reuse;
alter tablespace roll_2 add datafile '\\.\roll_2_10' size     170M
reuse;
alter tablespace roll_2 add datafile '\\.\roll_2_11' size     170M
reuse;

create rollback segment t_2_1 tablespace roll_2;
create rollback segment t_2_2 tablespace roll_2;
create rollback segment t_2_3 tablespace roll_2;
create rollback segment t_2_4 tablespace roll_2;
create rollback segment t_2_5 tablespace roll_2;
create rollback segment t_2_6 tablespace roll_2;
create rollback segment t_2_7 tablespace roll_2;
create rollback segment t_2_8 tablespace roll_2;
create rollback segment t_2_9 tablespace roll_2;
create rollback segment t_2_10 tablespace roll_2;
create rollback segment t_2_11 tablespace roll_2;
create rollback segment t_2_12 tablespace roll_2;
create rollback segment t_2_13 tablespace roll_2;
create rollback segment t_2_14 tablespace roll_2;
create rollback segment t_2_15 tablespace roll_2;
create rollback segment t_2_16 tablespace roll_2;
create rollback segment t_2_17 tablespace roll_2;
create rollback segment t_2_18 tablespace roll_2;
create rollback segment t_2_19 tablespace roll_2;
create rollback segment t_2_20 tablespace roll_2;
create rollback segment t_2_21 tablespace roll_2;
create rollback segment t_2_22 tablespace roll_2;
create rollback segment t_2_23 tablespace roll_2;
create rollback segment t_2_24 tablespace roll_2;
create rollback segment t_2_25 tablespace roll_2;
create rollback segment t_2_26 tablespace roll_2;
create rollback segment t_2_27 tablespace roll_2;
create rollback segment t_2_28 tablespace roll_2;
create rollback segment t_2_29 tablespace roll_2;
create rollback segment t_2_30 tablespace roll_2;
create rollback segment t_2_31 tablespace roll_2;
create rollback segment t_2_32 tablespace roll_2;
create rollback segment t_2_33 tablespace roll_2;
create rollback segment t_2_34 tablespace roll_2;
create rollback segment t_2_35 tablespace roll_2;
create rollback segment t_2_36 tablespace roll_2;
create rollback segment t_2_37 tablespace roll_2;
create rollback segment t_2_38 tablespace roll_2;
create rollback segment t_2_39 tablespace roll_2;
create rollback segment t_2_40 tablespace roll_2;
create rollback segment t_2_41 tablespace roll_2;
create rollback segment t_2_42 tablespace roll_2;
create rollback segment t_2_43 tablespace roll_2;
create rollback segment t_2_44 tablespace roll_2;
create rollback segment t_2_45 tablespace roll_2;
create rollback segment t_2_46 tablespace roll_2;
create rollback segment t_2_47 tablespace roll_2;
create rollback segment t_2_48 tablespace roll_2;

```



```
create rollback segment t_2_175 tablespace roll_2;
create rollback segment t_2_176 tablespace roll_2;
create rollback segment t_2_177 tablespace roll_2;
create rollback segment t_2_178 tablespace roll_2;
create rollback segment t_2_179 tablespace roll_2;
create rollback segment t_2_180 tablespace roll_2;
create rollback segment t_2_181 tablespace roll_2;
create rollback segment t_2_182 tablespace roll_2;
create rollback segment t_2_183 tablespace roll_2;
create rollback segment t_2_184 tablespace roll_2;
create rollback segment t_2_185 tablespace roll_2;
create rollback segment t_2_186 tablespace roll_2;
create rollback segment t_2_187 tablespace roll_2;
create rollback segment t_2_188 tablespace roll_2;
create rollback segment t_2_189 tablespace roll_2;
create rollback segment t_2_190 tablespace roll_2;
create rollback segment t_2_191 tablespace roll_2;
create rollback segment t_2_192 tablespace roll_2;
create rollback segment t_2_193 tablespace roll_2;
create rollback segment t_2_194 tablespace roll_2;
create rollback segment t_2_195 tablespace roll_2;
create rollback segment t_2_196 tablespace roll_2;
create rollback segment t_2_197 tablespace roll_2;
create rollback segment t_2_198 tablespace roll_2;
create rollback segment t_2_199 tablespace roll_2;
create rollback segment t_2_200 tablespace roll_2;
```

```
alter rollback segment t_3_1 offline;
alter rollback segment t_3_2 offline;
alter rollback segment t_3_3 offline;
alter rollback segment t_3_4 offline;
alter rollback segment t_3_5 offline;
alter rollback segment t_3_6 offline;
alter rollback segment t_3_7 offline;
alter rollback segment t_3_8 offline;
alter rollback segment t_3_9 offline;
alter rollback segment t_3_10 offline;
alter rollback segment t_3_11 offline;
alter rollback segment t_3_12 offline;
alter rollback segment t_3_13 offline;
alter rollback segment t_3_14 offline;
alter rollback segment t_3_15 offline;
alter rollback segment t_3_16 offline;
alter rollback segment t_3_17 offline;
alter rollback segment t_3_18 offline;
```

```
alter rollback segment t_3_19 offline;
alter rollback segment t_3_20 offline;
alter rollback segment t_3_21 offline;
alter rollback segment t_3_22 offline;
alter rollback segment t_3_23 offline;
alter rollback segment t_3_24 offline;
alter rollback segment t_3_25 offline;
alter rollback segment t_3_26 offline;
alter rollback segment t_3_27 offline;
alter rollback segment t_3_28 offline;
alter rollback segment t_3_29 offline;
alter rollback segment t_3_30 offline;
alter rollback segment t_3_31 offline;
alter rollback segment t_3_32 offline;
alter rollback segment t_3_33 offline;
alter rollback segment t_3_34 offline;
alter rollback segment t_3_35 offline;
alter rollback segment t_3_36 offline;
alter rollback segment t_3_37 offline;
alter rollback segment t_3_38 offline;
alter rollback segment t_3_39 offline;
alter rollback segment t_3_40 offline;
alter rollback segment t_3_41 offline;
alter rollback segment t_3_42 offline;
alter rollback segment t_3_43 offline;
alter rollback segment t_3_44 offline;
alter rollback segment t_3_45 offline;
alter rollback segment t_3_46 offline;
alter rollback segment t_3_47 offline;
alter rollback segment t_3_48 offline;
alter rollback segment t_3_49 offline;
alter rollback segment t_3_50 offline;
```

```
alter rollback segment t_3_51 offline;
alter rollback segment t_3_52 offline;
alter rollback segment t_3_53 offline;
alter rollback segment t_3_54 offline;
alter rollback segment t_3_55 offline;
alter rollback segment t_3_56 offline;
alter rollback segment t_3_57 offline;
alter rollback segment t_3_58 offline;
alter rollback segment t_3_59 offline;
alter rollback segment t_3_60 offline;
alter rollback segment t_3_61 offline;
alter rollback segment t_3_62 offline;
alter rollback segment t_3_63 offline;
alter rollback segment t_3_64 offline;
alter rollback segment t_3_65 offline;
alter rollback segment t_3_66 offline;
alter rollback segment t_3_67 offline;
alter rollback segment t_3_68 offline;
alter rollback segment t_3_69 offline;
alter rollback segment t_3_70 offline;
alter rollback segment t_3_71 offline;
alter rollback segment t_3_72 offline;
alter rollback segment t_3_73 offline;
alter rollback segment t_3_74 offline;
alter rollback segment t_3_75 offline;
alter rollback segment t_3_76 offline;
alter rollback segment t_3_77 offline;
alter rollback segment t_3_78 offline;
alter rollback segment t_3_79 offline;
alter rollback segment t_3_80 offline;
alter rollback segment t_3_81 offline;
```

```
alter rollback segment t_3_82 offline;
alter rollback segment t_3_83 offline;
alter rollback segment t_3_84 offline;
alter rollback segment t_3_85 offline;
alter rollback segment t_3_86 offline;
alter rollback segment t_3_87 offline;
alter rollback segment t_3_88 offline;
alter rollback segment t_3_89 offline;
alter rollback segment t_3_90 offline;
alter rollback segment t_3_91 offline;
alter rollback segment t_3_92 offline;
alter rollback segment t_3_93 offline;
alter rollback segment t_3_94 offline;
alter rollback segment t_3_95 offline;
alter rollback segment t_3_96 offline;
alter rollback segment t_3_97 offline;
alter rollback segment t_3_98 offline;
alter rollback segment t_3_99 offline;
alter rollback segment t_3_100 offline;
alter rollback segment t_3_101 offline;
alter rollback segment t_3_102 offline;
alter rollback segment t_3_103 offline;
alter rollback segment t_3_104 offline;
alter rollback segment t_3_105 offline;
alter rollback segment t_3_106 offline;
alter rollback segment t_3_107 offline;
alter rollback segment t_3_108 offline;
alter rollback segment t_3_109 offline;
alter rollback segment t_3_110 offline;
alter rollback segment t_3_111 offline;
alter rollback segment t_3_112 offline;
alter rollback segment t_3_113 offline;
```

```
alter rollback segment t_3_114 offline;
alter rollback segment t_3_115 offline;
alter rollback segment t_3_116 offline;
alter rollback segment t_3_117 offline;
alter rollback segment t_3_118 offline;
alter rollback segment t_3_119 offline;
alter rollback segment t_3_120 offline;
alter rollback segment t_3_121 offline;
alter rollback segment t_3_122 offline;
alter rollback segment t_3_123 offline;
alter rollback segment t_3_124 offline;
alter rollback segment t_3_125 offline;
alter rollback segment t_3_126 offline;
alter rollback segment t_3_127 offline;
alter rollback segment t_3_128 offline;
alter rollback segment t_3_129 offline;
alter rollback segment t_3_130 offline;
alter rollback segment t_3_131 offline;
alter rollback segment t_3_132 offline;
alter rollback segment t_3_133 offline;
alter rollback segment t_3_134 offline;
alter rollback segment t_3_135 offline;
alter rollback segment t_3_136 offline;
alter rollback segment t_3_137 offline;
alter rollback segment t_3_138 offline;
alter rollback segment t_3_139 offline;
alter rollback segment t_3_140 offline;
alter rollback segment t_3_141 offline;
alter rollback segment t_3_142 offline;
alter rollback segment t_3_143 offline;
alter rollback segment t_3_144 offline;
```

```
alter rollback segment t_3_145 offline;
alter rollback segment t_3_146 offline;
alter rollback segment t_3_147 offline;
alter rollback segment t_3_148 offline;
alter rollback segment t_3_149 offline;
alter rollback segment t_3_150 offline;
alter rollback segment t_3_151 offline;
alter rollback segment t_3_152 offline;
alter rollback segment t_3_153 offline;
alter rollback segment t_3_154 offline;
alter rollback segment t_3_155 offline;
alter rollback segment t_3_156 offline;
alter rollback segment t_3_157 offline;
alter rollback segment t_3_158 offline;
alter rollback segment t_3_159 offline;
alter rollback segment t_3_160 offline;
alter rollback segment t_3_161 offline;
alter rollback segment t_3_162 offline;
alter rollback segment t_3_163 offline;
alter rollback segment t_3_164 offline;
alter rollback segment t_3_165 offline;
alter rollback segment t_3_166 offline;
alter rollback segment t_3_167 offline;
alter rollback segment t_3_168 offline;
alter rollback segment t_3_169 offline;
alter rollback segment t_3_170 offline;
alter rollback segment t_3_171 offline;
alter rollback segment t_3_172 offline;
alter rollback segment t_3_173 offline;
alter rollback segment t_3_174 offline;
alter rollback segment t_3_175 offline;
alter rollback segment t_3_176 offline;
```

```

alter rollback segment t_3_177 offline;
alter rollback segment t_3_178 offline;
alter rollback segment t_3_179 offline;
alter rollback segment t_3_180 offline;
alter rollback segment t_3_181 offline;
alter rollback segment t_3_182 offline;
alter rollback segment t_3_183 offline;
alter rollback segment t_3_184 offline;
alter rollback segment t_3_185 offline;
alter rollback segment t_3_186 offline;
alter rollback segment t_3_187 offline;
alter rollback segment t_3_188 offline;
alter rollback segment t_3_189 offline;
alter rollback segment t_3_190 offline;
alter rollback segment t_3_191 offline;
alter rollback segment t_3_192 offline;
alter rollback segment t_3_193 offline;
alter rollback segment t_3_194 offline;
alter rollback segment t_3_195 offline;
alter rollback segment t_3_196 offline;
alter rollback segment t_3_197 offline;
alter rollback segment t_3_198 offline;
alter rollback segment t_3_199 offline;
alter rollback segment t_3_200 offline;

drop tablespace roll_3 including contents;

create tablespace roll_3 datafile '\\.\roll_3_0' size 170M reuse extent
management local uniform size 200K nologging ;
alter tablespace roll_3 add datafile '\\.\roll_3_1' size 170M
reuse;
alter tablespace roll_3 add datafile '\\.\roll_3_2' size 170M
reuse;
alter tablespace roll_3 add datafile '\\.\roll_3_3' size 170M
reuse;
alter tablespace roll_3 add datafile '\\.\roll_3_4' size 170M
reuse;
alter tablespace roll_3 add datafile '\\.\roll_3_5' size 170M
reuse;

```

```

alter tablespace roll_3 add datafile '\\.\roll_3_6' size 170M
reuse;
alter tablespace roll_3 add datafile '\\.\roll_3_7' size 170M
reuse;
alter tablespace roll_3 add datafile '\\.\roll_3_8' size 170M
reuse;
alter tablespace roll_3 add datafile '\\.\roll_3_9' size 170M
reuse;
alter tablespace roll_3 add datafile '\\.\roll_3_10' size 170M
reuse;
alter tablespace roll_3 add datafile '\\.\roll_3_11' size 170M
reuse;

create rollback segment t_3_1 tablespace roll_3;
create rollback segment t_3_2 tablespace roll_3;
create rollback segment t_3_3 tablespace roll_3;
create rollback segment t_3_4 tablespace roll_3;
create rollback segment t_3_5 tablespace roll_3;
create rollback segment t_3_6 tablespace roll_3;
create rollback segment t_3_7 tablespace roll_3;
create rollback segment t_3_8 tablespace roll_3;
create rollback segment t_3_9 tablespace roll_3;
create rollback segment t_3_10 tablespace roll_3;
create rollback segment t_3_11 tablespace roll_3;
create rollback segment t_3_12 tablespace roll_3;
create rollback segment t_3_13 tablespace roll_3;
create rollback segment t_3_14 tablespace roll_3;
create rollback segment t_3_15 tablespace roll_3;
create rollback segment t_3_16 tablespace roll_3;
create rollback segment t_3_17 tablespace roll_3;
create rollback segment t_3_18 tablespace roll_3;
create rollback segment t_3_19 tablespace roll_3;
create rollback segment t_3_20 tablespace roll_3;
create rollback segment t_3_21 tablespace roll_3;
create rollback segment t_3_22 tablespace roll_3;
create rollback segment t_3_23 tablespace roll_3;
create rollback segment t_3_24 tablespace roll_3;
create rollback segment t_3_25 tablespace roll_3;
create rollback segment t_3_26 tablespace roll_3;
create rollback segment t_3_27 tablespace roll_3;
create rollback segment t_3_28 tablespace roll_3;
create rollback segment t_3_29 tablespace roll_3;
create rollback segment t_3_30 tablespace roll_3;
create rollback segment t_3_31 tablespace roll_3;
create rollback segment t_3_32 tablespace roll_3;
create rollback segment t_3_33 tablespace roll_3;
create rollback segment t_3_34 tablespace roll_3;
create rollback segment t_3_35 tablespace roll_3;
create rollback segment t_3_36 tablespace roll_3;
create rollback segment t_3_37 tablespace roll_3;
create rollback segment t_3_38 tablespace roll_3;
create rollback segment t_3_39 tablespace roll_3;
create rollback segment t_3_40 tablespace roll_3;
create rollback segment t_3_41 tablespace roll_3;
create rollback segment t_3_42 tablespace roll_3;
create rollback segment t_3_43 tablespace roll_3;
create rollback segment t_3_44 tablespace roll_3;
create rollback segment t_3_45 tablespace roll_3;
create rollback segment t_3_46 tablespace roll_3;
create rollback segment t_3_47 tablespace roll_3;
create rollback segment t_3_48 tablespace roll_3;
create rollback segment t_3_49 tablespace roll_3;
create rollback segment t_3_50 tablespace roll_3;

```



```

create rollback segment t_3_177 tablespace roll_3;
create rollback segment t_3_178 tablespace roll_3;
create rollback segment t_3_179 tablespace roll_3;
create rollback segment t_3_180 tablespace roll_3;
create rollback segment t_3_181 tablespace roll_3;
create rollback segment t_3_182 tablespace roll_3;
create rollback segment t_3_183 tablespace roll_3;
create rollback segment t_3_184 tablespace roll_3;
create rollback segment t_3_185 tablespace roll_3;
create rollback segment t_3_186 tablespace roll_3;
create rollback segment t_3_187 tablespace roll_3;
create rollback segment t_3_188 tablespace roll_3;
create rollback segment t_3_189 tablespace roll_3;
create rollback segment t_3_190 tablespace roll_3;
create rollback segment t_3_191 tablespace roll_3;
create rollback segment t_3_192 tablespace roll_3;
create rollback segment t_3_193 tablespace roll_3;
create rollback segment t_3_194 tablespace roll_3;
create rollback segment t_3_195 tablespace roll_3;
create rollback segment t_3_196 tablespace roll_3;
create rollback segment t_3_197 tablespace roll_3;
create rollback segment t_3_198 tablespace roll_3;
create rollback segment t_3_199 tablespace roll_3;
create rollback segment t_3_200 tablespace roll_3;

```

```

spool off
set echo off
exit sql.sqlcode

```

cre_tab.sh

```

#!/sh

$SQLPLUS tpcc/tpcc @step9createware > step9.log 2>&1 &
$SQLPLUS tpcc/tpcc @step10createdist > step10.log 2>&1 &
$SQLPLUS tpcc/tpcc @step11createcust > step11.log 2>&1 &
$SQLPLUS tpcc/tpcc @step12createhist > step12.log 2>&1 &
$SQLPLUS tpcc/tpcc @step13createordr > step13.log 2>&1 &
$SQLPLUS tpcc/tpcc @step14createnord > step14.log 2>&1 &
$SQLPLUS tpcc/tpcc @step15createordl > step15.log 2>&1 &
$SQLPLUS tpcc/tpcc @step16createstok > step16.log 2>&1 &
$SQLPLUS tpcc/tpcc @step17createitem > step17.log 2>&1 &

wait

```

step9createware.sql

```

spool step9createware.log;
set echo on;
drop table ware;
drop cluster warecluster including tables;

set timing on;

create cluster warecluster (
    w_id          number (5,0)
)
single          table

```

```

hashkeys          4800
hash is           w_id
size              1536
initrans          3
pctfree           0
storage ( freelists 22 freelist groups 43 )
tablespace        ware_0;

```

```

create table ware (
    w_id          number(5,0),
    w_ytd         number,
    w_tax         number,
    w_name        varchar2(10),
    w_street_1    varchar2(20),
    w_street_2    varchar2(20),
    w_city        varchar2(20),
    w_state       char(2),
    w_zip         char(9)
)
cluster warecluster (w_id);

```

```

spool off;
set echo off;
exit sql.sqlcode;

```

step10createdist.sql

```

spool step10createdist.log;
set echo on;
drop table dist;
drop cluster distcluster including tables;

set timing on;

```

```

create cluster distcluster (
    d_w_id        number(5,0),
    d_id          number(2,0)
)
single          table
hashkeys        48000
hash is         (d_w_id) * 10 + d_id
size            1536
initrans        3
pctfree         0
storage ( freelists 22 freelist groups 43 )
tablespace      dist_0;

create table dist (
    d_id          number(2,0),
    d_w_id        number(5,0),
    d_ytd         number,
    d_tax         number,
    d_next_o_id   number,
    d_name        varchar2(10),
    d_street_1    varchar2(20),
    d_street_2    varchar2(20),
    d_city        varchar2(20),
    d_state       char(2),
    d_zip         char(9)
)
cluster distcluster (d_w_id, d_id);

```

```

spool off;
set echo off;
exit sql.sqlcode;

```

step11createcust.sql

```

spool step11createcust.log;
set echo on;
drop table cust;
drop cluster custcluster including tables;

set timing on;

create cluster custcluster (
  c_id      number(5,0)
,c_d_id    number(2,0)
,c_w_id    number(5,0)
)
  single      table
  hashkeys   144000000
  hash is    (c_w_id * 30000 + c_id * 10 + c_d_id - 30011)
  size      850
  initrans  3
  pctfree   0
  storage ( buffer_pool recycle freelists 22 freelist groups 43 )
  tablespace cust_0;

create table cust (
  c_id      number(5,0),
  c_d_id    number(2,0),
  c_w_id    number(5,0),
  c_discount number,
  c_credit  char(2),
  c_last    varchar2(16),
  c_first   varchar2(16),
  c_credit_lim number,
  c_balance number,
  c_ytd_payment number,
  c_payment_cnt number,
  c_delivery_cnt number,
  c_street_1 varchar2(20),
  c_street_2 varchar2(20),
  c_city     varchar2(20),
  c_state    char(2),
  c_zip      char(9),
  c_phone    char(16),
  c_since    date,
  c_middle   char(2),
  c_data     varchar2(500)
)
cluster custcluster (c_id
,c_d_id
,c_w_id
);
spool off;
set echo off;
exit sql.sqlcode;

```

step12createhist.sql

```

spool step12createhist.log;
set echo on;
drop table hist;

set timing on;

create table hist (
  h_c_id      number,
  h_c_d_id    number,
  h_c_w_id    number,
  h_d_id      number,
  h_w_id      number,
  h_date      date,
  h_amount    number,
  h_data      varchar2(24)
)
  partition by range(h_w_id)
  (
    partition hist_0 values less than (1201)
      tablespace hist_0,
    partition hist_1 values less than (2401)
      tablespace hist_1,
    partition hist_2 values less than (3601)
      tablespace hist_2,
    partition hist_3 values less than (maxvalue)
      tablespace hist_3
  )
  initrans 4
  pctfree 5
  storage ( freelists 22 freelist groups 43 );
spool off;
set echo off;
exit sql.sqlcode;

```

step13createordr.sql

```

spool step13createordr.log;
set echo on;
drop table ordr;

set timing on;

create table ordr (
  o_id      number,
  o_w_id    number,
  o_d_id    number,
  o_c_id    number,
  o_carrier_id number,
  o_ol_cnt  number,
  o_all_local number,
  o_entry_d date
)
  partition by range(o_w_id)
  (
    partition ordr_0 values less than (1201)
      tablespace ordr_0,
    partition ordr_1 values less than (2401)
      tablespace ordr_1,
    partition ordr_2 values less than (3601)
      tablespace ordr_2,
  )

```

```

        partition ordr_3 values less than (maxvalue)
            tablespace ordr_3
    )
    initrans      4
    pctfree      5
    storage ( freelists 22 freelist groups 43 );
spool off;
set echo off;
exit sql.sqlcode;

```

step14createnord.sql

```

spool step14createnord.log;
set echo on;
drop table nord;

set timing on;

create table nord (
    no_w_id      number,
    no_d_id      number,
    no_o_id      number,
    constraint inord primary key (no_w_id, no_d_id, no_o_id)
)
organization index
partition by range(no_w_id)
(
    partition nord_0 values less than (1201)
        tablespace nörd_0,
    partition nord_1 values less than (2401)
        tablespace nörd_1,
    partition nord_2 values less than (3601)
        tablespace nörd_2,
    partition nord_3 values less than (maxvalue)
        tablespace nörd_3
)
initrans      4
pctfree      5
storage ( freelists 22 freelist groups 43 );
spool off;
set echo off;
exit sql.sqlcode;

```

step15createordl.sql

```

spool step15createordl.log;
set echo on;
drop table ordl;

set timing on;

create table ordl (
    ol_w_id      number,
    ol_d_id      number,
    ol_o_id      number,
    ol_number    number,
    ol_i_id      number,
    ol_delivery_d date,
    ol_amount    number,
    ol_supply_w_id number,

```

```

    ol_quantity  number,
    ol_dist_info char(24),
    constraint iordl primary key (ol_w_id, ol_d_id, ol_o_id,
ol_number)
)
organization index
partition by range(ol_w_id)
(
    partition ordl_0 values less than (1201)
        tablespace ordl_0,
    partition ordl_1 values less than (2401)
        tablespace ordl_1,
    partition ordl_2 values less than (3601)
        tablespace ordl_2,
    partition ordl_3 values less than (maxvalue)
        tablespace ordl_3
)
initrans      4
pctfree      5
storage ( freelists 22 freelist groups 43 );
spool off;
set echo off;
exit sql.sqlcode;

```

step16createstok.sql

```

spool step16createstok.log;
set echo on;
drop table stok;
drop cluster stokcluster including tables;

set timing on;

create cluster stokcluster (
    s_i_id      number(6,0)
    ,s_w_id      number(5,0)
)
single table
hashkeys      480000000
hash is      (abs(s_i_id - 1) * 1200 + mod((s_w_id - 1), 1200) + trunc
((s_w_id - 1) / 1200) * 120000000)
size          350
initrans      3
pctfree      0
storage ( buffer_pool keep freelists 22 freelist groups 43 )
tablespace     stok_0;

create table stok (
    s_i_id      number(6,0),
    s_w_id      number(5,0),
    s_quantity  number,
    s_ytd       number,
    s_order_cnt number,
    s_remote_cnt number,
    s_data      varchar2(50),
    s_dist_01   char(24),
    s_dist_02   char(24),
    s_dist_03   char(24),
    s_dist_04   char(24),
    s_dist_05   char(24),
    s_dist_06   char(24),

```

```

        s_dist_07      char(24),
        s_dist_08      char(24),
        s_dist_09      char(24),
        s_dist_10      char(24)
    )
    cluster stokcluster (s_i_id
    ,s_w_id
    );
    spool off;
    set echo off;
    exit sql.sqlcode;

```

step17createitem.sql

```

spool step17createitem.log;
set echo on;
drop table item;
drop cluster itemcluster including tables;

set timing on;

create cluster itemcluster (
i_id      number(6,0)
)
  single      table
  hashkeys   100000
  hash is    (i_id + 1)
  size      120
  initrans  3
  pctfree   0
  storage ( buffer_pool keep freelists 22 freelist groups 43 )
  tablespace item_0;

create table item (
  i_id      number(6,0),
  i_name    varchar2(24),
  i_price   number,
  i_data    varchar2(50),
  i_im_id   number
)
cluster itemcluster (i_id
);
spool off;
set echo off;
exit sql.sqlcode;

```

cre_idx.sh

```

#!/sh

$SQLPLUS tpcc/tpcc @step29createiware > step29.log 2>&1 &
$SQLPLUS tpcc/tpcc @step30createidist > step30.log 2>&1 &
$SQLPLUS tpcc/tpcc @step31createiitem > step31.log 2>&1 &
$SQLPLUS tpcc/tpcc @step32createicust1 > step32.log 2>&1 &
$SQLPLUS tpcc/tpcc @step33createicust2 > step33.log 2>&1 &
$SQLPLUS tpcc/tpcc @step34createistok > step34.log 2>&1 &
$SQLPLUS tpcc/tpcc @step35createiordr1 > step35.log 2>&1 &
$SQLPLUS tpcc/tpcc @step36createiordr2 > step36.log 2>&1 &
$SQLPLUS tpcc/tpcc @step37createinord > step37.log 2>&1 &
$SQLPLUS tpcc/tpcc @step38createiordl > step38.log 2>&1 &

```

```
wait
```

step29createiware.sql

```

spool step29createiware.log;
set echo on;
drop index iware;

set timing on;

create unique index iware on ware (w_id)
tablespace iware_0
  initrans      3
  parallel      8
  pctfree       1;

spool off;
set echo off;
exit sql.sqlcode;

```

step30createidist.sql

```

spool step30createidist.log;
set echo on;
drop index idist;

set timing on;

create unique index idist on dist (d_w_id, d_id)
  tablespace idist_0
  initrans      3
  parallel      8
  pctfree       1;

spool off;
set echo off;
exit sql.sqlcode;

```

step31createiitem.sql

```

spool step31createiitem.log;
set echo on;
drop index iitem;

set timing on;

create unique index iitem on item (i_id)
  initrans      4
  parallel      2
  pctfree       5
  storage ( freelists 22 freelist groups 43 )
  tablespace    iitem_0;

alter index iitem parallel (degree 8)
spool off;
set echo off;
exit sql.sqlcode;

```

step32createicust1.sql

```
spool step32createicust1.log;
set echo on;
drop index icust1;

set timing on;

create unique index icust1 on cust (c_w_id, c_d_id, c_id)
  initrans      3
  parallel      8
  pctfree       1
  storage ( freelists 22 freelist groups 43 )
  tablespace    icust1_0;
spool off;
set echo off;
exit sql.sqlcode;
```

step33createicust2.sql

```
spool step33createicust2.log;
set echo on;
drop index icust2;

set timing on;

create unique index icust2 on cust (c_last, c_w_id, c_d_id, c_first,
c_id)
  initrans      3
  parallel      8
  pctfree       1
  storage ( freelists 22 freelist groups 43 )
  tablespace    icust2_0;
spool off;
set echo off;
exit sql.sqlcode;
```

step34createistok.sql

```
spool step34createistok.log;
set echo on;
drop index istok;

set timing on;

create unique index istok on stok (s_i_id, s_w_id)
  initrans      3
  parallel      8
  pctfree       1
  storage ( freelists 22 freelist groups 43 )
  tablespace    istok_0;
spool off;
set echo off;
exit sql.sqlcode;
```

step35createiordr1.sql

```
spool step35createiordr1.log;
set echo on;
```

```
drop index iordr1;

set timing on;

create unique index iordr1 on ordr (o_w_id, o_d_id, o_id)
  local
  (
    partition iordr1_0 tablespace iordr1_0,
    partition iordr1_1 tablespace iordr1_1,
    partition iordr1_2 tablespace iordr1_2,
    partition iordr1_3 tablespace iordr1_3
  )
  initrans      3
  parallel      8
  pctfree       1
  storage ( freelists 22 freelist groups 43 );
spool off;
set echo off;
exit sql.sqlcode;
```

step36createiordr2.sql

```
spool step36createiordr2.log;
set echo on;
drop index iordr2;

set timing on;

create unique index iordr2 on ordr (o_w_id, o_d_id, o_c_id, o_id)
  local
  (
    partition iordr2_0 tablespace iordr2_0,
    partition iordr2_1 tablespace iordr2_1,
    partition iordr2_2 tablespace iordr2_2,
    partition iordr2_3 tablespace iordr2_3
  )
  initrans      4
  parallel      8
  pctfree      25
  storage ( freelists 22 freelist groups 43 );
spool off;
set echo off;
exit sql.sqlcode;
```

step37createinord.sql

```
exit sql.sqlcode;
```

step38createiordl.sql

```
exit sql.sqlcode;
```

step41createstoredprocs.sh

```
#!/sh
$SQLPLUS tpcc/tpcc @step41createstoredprocs > junk 2>&1
```

```

if test $? -ne 0
then
    exit 1;
else
    exit 0;
fi

```

step41createstoredprocs.sql

```

spool step41createstoredprocs.log
@initpay
@initnew
spool off
exit sql.sqlcode;

```

initnew.sql

```

-- The initnew package for storing variables used in the
-- New Order anonymous block

```

```

CREATE OR REPLACE PACKAGE initnew
AS
    TYPE intarray IS TABLE OF INTEGER index by binary_integer;
    TYPE distarray IS TABLE OF VARCHAR(24) index by binary_integer;
    nulldate      DATE;
    s_dist        distarray;
    idxlarr       intarray;
    s_remote      intarray;
    PROCEDURE new_init(idxarr intarray);
END initnew;
/
show errors;

```

```

CREATE OR REPLACE PACKAGE BODY initnew AS
    PROCEDURE new_init (idxarr intarray)
    IS
    BEGIN
        -- initialize null date
        nulldate := TO_DATE('01-01-1811', 'MM-DD-YYYY');
        idxlarr := idxarr;
    END new_init;
END initnew;
/
show errors

```

initpay.sql

```

CREATE OR REPLACE PACKAGE initpay
AS
    TYPE rowidarray IS TABLE OF ROWID INDEX BY BINARY_INTEGER;
    row_id          rowidarray;
    cust_rowid      ROWID;
    dist_name       VARCHAR2(11);
    ware_name       VARCHAR2(11);
    c_num           BINARY_INTEGER;
    PROCEDURE pay_init;
END initpay;
/

```

```

show errors;

```

```

CREATE OR REPLACE PACKAGE BODY initpay AS
    PROCEDURE pay_init IS
    BEGIN
        NULL;
    END pay_init;
END initpay;
/
show errors;

```

step47offlinerollsegs.sh

```

#!/sh
$SQLPLUS internal/internal @step47offlinerollsegs > junk 2>&1

if test $? -ne 0
then
    exit 1;
else
    exit 0;
fi

```

step47offlinerollsegs.sql

```

spool step47offlinerollsegs.log
set echo on

alter rollback segment t1 offline;
alter rollback segment t2 offline;
alter rollback segment t3 offline;
alter rollback segment t4 offline;
alter rollback segment t5 offline;
alter rollback segment t6 offline;
alter rollback segment t7 offline;
alter rollback segment t8 offline;
alter rollback segment t9 offline;
alter rollback segment t10 offline;
alter rollback segment t11 offline;
alter rollback segment t12 offline;
alter rollback segment t13 offline;
alter rollback segment t14 offline;
alter rollback segment t15 offline;
alter rollback segment t16 offline;
alter rollback segment t17 offline;
alter rollback segment t18 offline;
alter rollback segment t19 offline;
alter rollback segment t20 offline;
alter rollback segment t21 offline;
alter rollback segment t22 offline;
alter rollback segment t23 offline;
alter rollback segment t24 offline;
alter rollback segment t25 offline;
alter rollback segment t26 offline;
alter rollback segment t27 offline;
alter rollback segment t28 offline;
alter rollback segment t29 offline;
alter rollback segment t30 offline;

spool off

```

```
set echo off
exit sql.sqlcode
```

stepusertemp.sh

```
#!/sh

$SQLPLUS internal/internal @stepusertemp > junk 2>&1

if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi
```

stepusertemp.sql

```
spool stepusertemp.log;

set echo on;

alter user tpcc temporary tablespace temp_0;

set echo off;
spool off;

exit sql.sqlcode;
```

B2. Loader Code

load_dat.sh

```
#!/sh

$TPCCLOAD -M 4800 -w > step20loadware.log 2>&1 &
$TPCCLOAD -M 4800 -d > step21loaddist.log 2>&1 &
$TPCCLOAD -M 4800 -i > step22loaditem.log 2>&1 &

$TPCCLOAD -M 4800 -n -b 1 -e 300 > step24loadnord0.log 2>&1 &
$TPCCLOAD -M 4800 -n -b 301 -e 600 > step24loadnord1.log 2>&1 &
$TPCCLOAD -M 4800 -n -b 601 -e 900 > step24loadnord2.log 2>&1 &
$TPCCLOAD -M 4800 -n -b 901 -e 1200 > step24loadnord3.log 2>&1 &
$TPCCLOAD -M 4800 -n -b 1201 -e 1500 > step24loadnord4.log 2>&1 &
$TPCCLOAD -M 4800 -n -b 1501 -e 1800 > step24loadnord5.log 2>&1 &
$TPCCLOAD -M 4800 -n -b 1801 -e 2100 > step24loadnord6.log 2>&1 &
$TPCCLOAD -M 4800 -n -b 2101 -e 2400 > step24loadnord7.log 2>&1 &
$TPCCLOAD -M 4800 -n -b 2401 -e 2700 > step24loadnord8.log 2>&1 &
$TPCCLOAD -M 4800 -n -b 2701 -e 3000 > step24loadnord9.log 2>&1 &
$TPCCLOAD -M 4800 -n -b 3001 -e 3300 > step24loadnord10.log 2>&1 &
$TPCCLOAD -M 4800 -n -b 3301 -e 3600 > step24loadnord11.log 2>&1 &
$TPCCLOAD -M 4800 -n -b 3601 -e 3900 > step24loadnord12.log 2>&1 &
$TPCCLOAD -M 4800 -n -b 3901 -e 4200 > step24loadnord13.log 2>&1 &
$TPCCLOAD -M 4800 -n -b 4201 -e 4500 > step24loadnord14.log 2>&1 &
$TPCCLOAD -M 4800 -n -b 4501 -e 4800 > step24loadnord15.log 2>&1 &
```

```
wait
```

```
$TPCCLOAD -M 4800 -h -b 1 -e 300 > step23loadhist0.log 2>&1 &
$TPCCLOAD -M 4800 -h -b 301 -e 600 > step23loadhist1.log 2>&1 &
$TPCCLOAD -M 4800 -h -b 601 -e 900 > step23loadhist2.log 2>&1 &
$TPCCLOAD -M 4800 -h -b 901 -e 1200 > step23loadhist3.log 2>&1 &
$TPCCLOAD -M 4800 -h -b 1201 -e 1500 > step23loadhist4.log 2>&1 &
$TPCCLOAD -M 4800 -h -b 1501 -e 1800 > step23loadhist5.log 2>&1 &
$TPCCLOAD -M 4800 -h -b 1801 -e 2100 > step23loadhist6.log 2>&1 &
$TPCCLOAD -M 4800 -h -b 2101 -e 2400 > step23loadhist7.log 2>&1 &
$TPCCLOAD -M 4800 -h -b 2401 -e 2700 > step23loadhist8.log 2>&1 &
$TPCCLOAD -M 4800 -h -b 2701 -e 3000 > step23loadhist9.log 2>&1 &
$TPCCLOAD -M 4800 -h -b 3001 -e 3300 > step23loadhist10.log 2>&1 &
$TPCCLOAD -M 4800 -h -b 3301 -e 3600 > step23loadhist11.log 2>&1 &
$TPCCLOAD -M 4800 -h -b 3601 -e 3900 > step23loadhist12.log 2>&1 &
$TPCCLOAD -M 4800 -h -b 3901 -e 4200 > step23loadhist13.log 2>&1 &
$TPCCLOAD -M 4800 -h -b 4201 -e 4500 > step23loadhist14.log 2>&1 &
$TPCCLOAD -M 4800 -h -b 4501 -e 4800 > step23loadhist15.log 2>&1 &
wait
```

```
$TPCCLOAD -M 4800 -c -b 1 -e 300 > step26loadcust0.log 2>&1 &
$TPCCLOAD -M 4800 -c -b 301 -e 600 > step26loadcust1.log 2>&1 &
$TPCCLOAD -M 4800 -c -b 601 -e 900 > step26loadcust2.log 2>&1 &
$TPCCLOAD -M 4800 -c -b 901 -e 1200 > step26loadcust3.log 2>&1 &
$TPCCLOAD -M 4800 -c -b 1201 -e 1500 > step26loadcust4.log 2>&1 &
$TPCCLOAD -M 4800 -c -b 1501 -e 1800 > step26loadcust5.log 2>&1 &
$TPCCLOAD -M 4800 -c -b 1801 -e 2100 > step26loadcust6.log 2>&1 &
$TPCCLOAD -M 4800 -c -b 2101 -e 2400 > step26loadcust7.log 2>&1 &
$TPCCLOAD -M 4800 -c -b 2401 -e 2700 > step26loadcust8.log 2>&1 &
$TPCCLOAD -M 4800 -c -b 2701 -e 3000 > step26loadcust9.log 2>&1 &
$TPCCLOAD -M 4800 -c -b 3001 -e 3300 > step26loadcust10.log 2>&1 &
$TPCCLOAD -M 4800 -c -b 3301 -e 3600 > step26loadcust11.log 2>&1 &
$TPCCLOAD -M 4800 -c -b 3601 -e 3900 > step26loadcust12.log 2>&1 &
$TPCCLOAD -M 4800 -c -b 3901 -e 4200 > step26loadcust13.log 2>&1 &
$TPCCLOAD -M 4800 -c -b 4201 -e 4500 > step26loadcust14.log 2>&1 &
$TPCCLOAD -M 4800 -c -b 4501 -e 4800 > step26loadcust15.log 2>&1 &
wait
```

```
$TPCCLOAD -M 4800 -o \\.\dummy0.dat -b 1 -e 300 >
step25loadordrord10.log 2>&1 &
$TPCCLOAD -M 4800 -o \\.\dummy1.dat -b 301 -e 600 >
step25loadordrord11.log 2>&1 &
$TPCCLOAD -M 4800 -o \\.\dummy2.dat -b 601 -e 900 >
step25loadordrord12.log 2>&1 &
$TPCCLOAD -M 4800 -o \\.\dummy3.dat -b 901 -e 1200 >
step25loadordrord13.log 2>&1 &
$TPCCLOAD -M 4800 -o \\.\dummy4.dat -b 1201 -e 1500 >
step25loadordrord14.log 2>&1 &
$TPCCLOAD -M 4800 -o \\.\dummy5.dat -b 1501 -e 1800 >
step25loadordrord15.log 2>&1 &
$TPCCLOAD -M 4800 -o \\.\dummy6.dat -b 1801 -e 2100 >
step25loadordrord16.log 2>&1 &
$TPCCLOAD -M 4800 -o \\.\dummy7.dat -b 2101 -e 2400 >
step25loadordrord17.log 2>&1 &
$TPCCLOAD -M 4800 -o \\.\dummy8.dat -b 2401 -e 2700 >
step25loadordrord18.log 2>&1 &
$TPCCLOAD -M 4800 -o \\.\dummy9.dat -b 2701 -e 3000 >
step25loadordrord19.log 2>&1 &
$TPCCLOAD -M 4800 -o \\.\dummy10.dat -b 3001 -e 3300 >
step25loadordrord110.log 2>&1 &
$TPCCLOAD -M 4800 -o \\.\dummy11.dat -b 3301 -e 3600 >
step25loadordrord111.log 2>&1 &
```

```

$TPCCLOAD -M 4800 -o \\.\dummy12.dat -b 3601 -e 3900 >
step25loadordrordl12.log 2>&1 &
$TPCCLOAD -M 4800 -o \\.\dummy13.dat -b 3901 -e 4200 >
step25loadordrordl13.log 2>&1 &
$TPCCLOAD -M 4800 -o \\.\dummy14.dat -b 4201 -e 4500 >
step25loadordrordl14.log 2>&1 &
$TPCCLOAD -M 4800 -o \\.\dummy15.dat -b 4501 -e 4800 >
step25loadordrordl15.log 2>&1 &
wait

$TPCCLOAD -M 4800 -S -j 1 -k 6250 > step27loadstok0.log 2>&1 &
$TPCCLOAD -M 4800 -S -j 6251 -k 12500 > step27loadstok1.log 2>&1 &
$TPCCLOAD -M 4800 -S -j 12501 -k 18750 > step27loadstok2.log 2>&1 &
$TPCCLOAD -M 4800 -S -j 18751 -k 25000 > step27loadstok3.log 2>&1 &
$TPCCLOAD -M 4800 -S -j 25001 -k 31250 > step27loadstok4.log 2>&1 &
$TPCCLOAD -M 4800 -S -j 31251 -k 37500 > step27loadstok5.log 2>&1 &
$TPCCLOAD -M 4800 -S -j 37501 -k 43750 > step27loadstok6.log 2>&1 &
$TPCCLOAD -M 4800 -S -j 43751 -k 50000 > step27loadstok7.log 2>&1 &
$TPCCLOAD -M 4800 -S -j 50001 -k 56250 > step27loadstok8.log 2>&1 &
$TPCCLOAD -M 4800 -S -j 56251 -k 62500 > step27loadstok9.log 2>&1 &
$TPCCLOAD -M 4800 -S -j 62501 -k 68750 > step27loadstok10.log 2>&1 &
$TPCCLOAD -M 4800 -S -j 68751 -k 75000 > step27loadstok11.log 2>&1 &
$TPCCLOAD -M 4800 -S -j 75001 -k 81250 > step27loadstok12.log 2>&1 &
$TPCCLOAD -M 4800 -S -j 81251 -k 87500 > step27loadstok13.log 2>&1 &
$TPCCLOAD -M 4800 -S -j 87501 -k 93750 > step27loadstok14.log 2>&1 &
$TPCCLOAD -M 4800 -S -j 93751 -k 100000 > step27loadstok15.log 2>&1 &
wait

```

tpccload.c

```

#ifdef RCSID
static char *RCSid =
    "$Header: tpccload.c 7030100.1 96/05/13 16:20:36 plai Generic<base>
$ Copyr (c) 1993 Oracle";
#endif /* RCSID */

/*=====
| Copyright (c) 1994 Oracle Corp, Redwood Shores, CA
| OPEN SYSTEMS PERFORMANCE GROUP
| All Rights Reserved
|=====
FILENAME
| tpccload.c
DESCRIPTION
| Load or generate TPC-C database tables.
| Usage: tpccload -M <# of wares> [options]
| options: -A load all tables
|          -w load ware table
|          -d load dist table
|          -c load cust table
|          -i load item table
|          -s load stok table (cluster around s_w id)
|          -S load stok table (cluster around s_i_id)
|          -h load hist table
|          -n load new-order table
|          -o <oline file> load order and order-line
table
|          -b <ware#> beginning ware number
|          -e <ware#> ending ware number
|          -j <item#> beginning item number (with -S)
|          -k <item#> ending item number (with -S)

```

```

| -g generate rows to standard output
|=====*/

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
/* #include <unistd.h> */
#include <time.h>
#include <sys/types.h>
#include "tpcc.h"

#ifdef ORA_NT
#undef boolean
#include <process.h>
#include "dpbcore.h"
# define gettime dpbtimef
# define getcpu dpbcpu
# define lrand48() ((long)rand() <<15 | rand())
#endif /* STDC */
# define PROTO(args) args
#else
# define PROTO(args) ()
#endif
#endif

#define DISTARR 10 /* dist insert array size */
#define CUSTARR 100 /* cust insert array size */
#define STOCARR 100 /* stok insert array size */
#define ITEMARR 100 /* item insert array size */
#define HISTARR 100 /* hist insert array size */
#define ORDEARR 100 /* order insert array size */
#define NEWOARR 100 /* new order insert array size */

#define DISTFAC 10 /* max. disctrict id */
#define CUSTFAC 3000 /* max. cust id */
#define STOCFAC 100000 /* max. stok id */
#define ITEMFAC 100000 /* max. item id */
#define HISTFAC 30000 /* hist / ware */
#define ORDEFAC 3000 /* order / dist */
#define NEWOFAC 900 /* new order / dist */

#define C 0 /* constant in non-uniform dist. eqt. */
#define CNUM1 1 /* first constant in non-uniform dist. eqt. */
#define CNUM2 2 /* second constant in non-uniform dist. eqt. */
#define CNUM3 3 /* third constant in non-uniform dist. eqt. */

#define SEED 2 /* seed for random functions */

#define SQLTXTW "INSERT INTO ware (w_id, w_ytd, w_tax, w_name, w_street_1, w_street_2, w_city, w_state, w_zip) VALUES (:w_id, 30000000, :w_tax, :w_name, :w_street_1, :w_street_2, :w_city, :w_state, :w_zip)"

#define SQLXTD "INSERT INTO dist (d_id, d_w_id, d_ytd, d_tax, d_next_o_id, d_name, d_street_1, d_street_2, d_city, d_state, d_zip) VALUES (:d_id, :d_w_id, 3000000, :d_tax, \ 3001, :d_name, :d_street_1, :d_street_2, :d_city, :d_state, :d_zip) )"

```



```

csrdef *cur;
{
    text msg[2048];
    if (cur->rc) {
        oerhms (lda, cur->rc, msg, 2048);
        fprintf (stderr, "TPC-C load error: %s\n", msg);
    }
}

void quit ()
{
    if (oclose (&curw))
        errrpt (&tpclda, &curw);
    if (oclose (&curd))
        errrpt (&tpclda, &curd);
    if (oclose (&curc))
        errrpt (&tpclda, &curc);
    if (oclose (&curh))
        errrpt (&tpclda, &curh);
    if (oclose (&curs))
        errrpt (&tpclda, &curs);
    if (oclose (&curi))
        errrpt (&tpclda, &curi);
    if (oclose (&curo1))
        errrpt (&tpclda, &curo1);
    if (oclose (&curo2))
        errrpt (&tpclda, &curo2);
    if (oclose (&curo11))
        errrpt (&tpclda, &curo11);
    if (oclose (&curo12))
        errrpt (&tpclda, &curo12);
    if (oclose (&curno))
        errrpt (&tpclda, &curno);
    if (ologof (&tpclda))
        fprintf (stderr, "TPC-C load error: Error in logging off\n");
}

void main (argc, argv)
int argc;
char *argv[];

```

```

{
    char *uid="tpcc/tpcc";
    text sqlbuf[1024];
    int scale=0;
    int i, j;
    int loop;
    int loopcount;
    int cid;
    int dwid;
    int cdid;
    int cwid;
    int sid;
    int swid;
    int olcnt;
    int nrows;
    int row;

    int w_id;
    char w_name[11];
    char w_street_1[21];
    char w_street_2[21];
    char w_city[21];
    char w_state[2];
    char w_zip[9];
    float w_tax;

    int d_id[10];
    int d_w_id[10];
    char d_name[10][11];
    char d_street_1[10][21];
    char d_street_2[10][21];
    char d_city[10][21];
    char d_state[10][2];
    char d_zip[10][9];
    float d_tax[10];

    int c_id[100];
    int c_d_id[100];
    int c_w_id[100];
    char c_first[100][17];
    char c_last[100][17];
    char c_street_1[100][21];
    char c_street_2[100][21];
    char c_city[100][21];
    char c_state[100][2];
    char c_zip[100][9];
    char c_phone[100][16];
    char c_credit[100][2];
    float c_discount[100];
    char c_data[100][501];

    int i_id[100];
    int i_im_id[100];
    int i_price[100];
    char i_name[100][25];
    char i_data[100][51];

    int s_i_id[100];
    int s_w_id[100];
    int s_quantity[100];
    char s_dist_01[100][24];

```

```

char s_dist_02[100][24];
char s_dist_03[100][24];
char s_dist_04[100][24];
char s_dist_05[100][24];
char s_dist_06[100][24];
char s_dist_07[100][24];
char s_dist_08[100][24];
char s_dist_09[100][24];
char s_dist_10[100][24];
char s_data[100][51];

int h_w_id[100];
int h_d_id[100];
int h_c_id[100];
char h_data[100][25];

int o_id[100];
int o_d_id[100];
int o_w_id[100];
int o_c_id[100];
int o_carrier_id[100];
int o_ol_cnt[100];

int ol_o_id[15];
int ol_d_id[15];
int ol_w_id[15];
int ol_number[15];
int ol_i_id[15];
int ol_supply_w_id[15];
int ol_amount[15];
char ol_dist_info[15][24];

int no_o_id[100];
int no_d_id[100];
int no_w_id[100];

char sdate[30];
#ifdef ORA_NT
clock_t begin_time, end_time;
clock_t begin_cpu, end_cpu;

char *arg_ptr, **end_args;
#else
double begin_time, end_time;
double begin_cpu, end_cpu;
double gettime(), getcpu();

extern int getopt();
extern char *optarg;
extern int optind, opterr;

int opt;
#endif

char *argstr="M:AwdcisShno:b:e:j:k:g";
int do_A=0;
int do_w=0;
int do_d=0;
int do_i=0;
int do_c=0;
int do_s=0;
int do_S=0;
int do_h=0;

```

```

int do_o=0;
int do_n=0;
int gen=0;
int bware=1;
int eware=0;
int bitem=1;
int eitem=0;

FILE *olfp=NULL;
char olfname[100];
#ifdef ORA_NT
char fname[100];
FILE *logfile;
#endif /* ORA_NT */

/*-----+
| Parse command line -- look for scale factor.
+-----*/

if (argc == 1) {
    myusage ();
}
#ifdef ORA_NT
end_args = argv + argc;

for (++argv; argv < end_args; )
{
    arg_ptr = *argv++;

    if (*arg_ptr != '-')
    {
        myusage ();
    } else
    {
        switch (arg_ptr[1]) {
            case '?': myusage ();
                    break;
            case 'M': scale = atoi (*argv++);
                    break;
            case 'A': do_A = 1;
                    break;
            case 'w': do_w = 1;
                    break;
            case 'd': do_d = 1;
                    break;
            case 'c': do_c = 1;
                    break;
            case 'i': do_i = 1;
                    break;
            case 's': do_s = 1;
                    break;
            case 'S': do_S = 1;
                    break;
            case 'h': do_h = 1;
                    break;
            case 'n': do_n = 1;
                    break;
            case 'o': do_o = 1;
                    strcpy (olfname, *argv++);
                    break;
            case 'b': bware = atoi (*argv++);
                    break;
            case 'e': eware = atoi (*argv++);

```

```

        break;
    case 'j': bitem = atoi (*argv++);
              break;
    case 'k': eitem = atoi (*argv++);
              break;
    case 'g': gen = 1;
              strcpy (fname, *argv++);
              break;
    case 'l': logfile=fopen(*argv+,"w");
              break;
    default: fprintf (stderr, "THIS SHOULD NEVER HAPPEN!!!\n");
             fprintf (stderr, "(reached default case in getopt
())\n");
             myusage ();
        }
    }

#else

while ((opt = getopt (argc, argv, argstr)) != -1) {
    switch (opt) {
        case '?': myusage ();
                  break;
        case 'M': scale = atoi (optarg);
                  break;
        case 'A': do_A = 1;
                  break;
        case 'w': do_w = 1;
                  break;
        case 'd': do_d = 1;
                  break;
        case 'c': do_c = 1;
                  break;
        case 'i': do_i = 1;
                  break;
        case 's': do_s = 1;
                  break;
        case 'S': do_S = 1;
                  break;
        case 'h': do_h = 1;
                  break;
        case 'n': do_n = 1;
                  break;
        case 'o': do_o = 1;
                  strcpy (olfname, optarg);
                  break;
        case 'b': bware = atoi (optarg);
                  break;
        case 'e': aware = atoi (optarg);
                  break;
        case 'j': bitem = atoi (optarg);
                  break;
        case 'k': eitem = atoi (optarg);
                  break;
        case 'g': gen = 1;
                  break;
        default: fprintf (stderr, "THIS SHOULD NEVER HAPPEN!!!\n");
                 fprintf (stderr, "(reached default case in getopt
())\n");
                 myusage ();
    }
}

```

```

    }
# endif /* ORA_NT */

/*-----*
|          Rudimentary error checking          |
/*-----*/

    if (scale < 1) {
        fprintf (stderr, "Invalid scale factor: '%d'\n", scale);
        myusage ();
    }

    if (!(do_A || do_w || do_d || do_c || do_i || do_s || do_S || do_h
|| do_o ||
        do_n)) {
        fprintf (stderr, "What should I load???\n");
        myusage ();
    }

    if (gen && (do_A || (do_w + do_d + do_c + do_i + do_s + do_S + do_h
+ do_o +
        do_n > 1))) {
        fprintf (stderr, "Can only generate table one at a time\n");
        myusage ();
    }

    if (do_S && (do_A || do_s)) {
        fprintf (stderr, "Cluster stok table around s_w_id or
s_i_id?\n");
        myusage ();
    }

    if (eware <= 0)
        aware = scale;
    if (eitem <= 0)
        eitem = STOCFAC;

    if (do_S) {
        if ((bitem < 1) || (bitem > STOCFAC)) {
            fprintf (stderr, "Invalid beginning item number: '%d'\n",
bitem);
            myusage ();
        }

        if ((eitem < bitem) || (eitem > STOCFAC)) {
            fprintf (stderr, "Invalid ending item number: '%d'\n", eitem);
            myusage ();
        }
    }

    if ((bware < 1) || (bware > scale)) {
        fprintf (stderr, "Invalid beginning ware number: '%d'\n", bware);
        myusage ();
    }

    if ((eware < bware) || (eware > scale)) {
        fprintf (stderr, "Invalid ending ware number: '%d'\n", aware);
        myusage ();
    }

    if (gen && do_o) {
        if ((olfp = fopen (olfname, "w")) == NULL) {

```

```

        fprintf (stderr, "Can't open '%s' for writing order lines\n",
olfname);
        myusage ();
    }
}

/*-----+
| Prepare to insert into database.          |
+-----*/

sysdate (sdate);
if (!gen) {

    /* log on to Oracle */

    if (orlon (&tpclda, (ub1 *) tpchda, (text *) uid, -1, (text *) 0,
-1, 0)) {
        fprintf (stderr, "TPC-C load error: Error in logging on\n");
        errrpt (&tpclda, &tpclda);
        exit (1);
    }

    fprintf (stderr, "\nConnected to Oracle userid '%s'.\n", uid);

    /* turn off auto-commit */

    if (ocof (&tpclda) {
        errrpt (&tpclda, &tpclda);
        ologof (&tpclda);
        exit (1);
    }

    /* open cursors */

    if (oopen (&curw, &tpclda, (text *) 0, -1, -1, (text *) uid, -1))
{
        errrpt (&tpclda, &curw);
        ologof (&tpclda);
        exit (1);
    }

    if (oopen (&curd, &tpclda, (text *) 0, -1, -1, (text *) uid, -1))
{
        errrpt (&tpclda, &curd);
        oclose (&curw);
        ologof (&tpclda);
        exit (1);
    }

    if (oopen (&curc, &tpclda, (text *) 0, -1, -1, (text *) uid, -1))
{
        errrpt (&tpclda, &curc);
        oclose (&curw);
        oclose (&curd);
        ologof (&tpclda);
        exit (1);
    }

    if (oopen (&curh, &tpclda, (text *) 0, -1, -1, (text *) uid, -1))
{
        errrpt (&tpclda, &curh);
        oclose (&curw);
        oclose (&curd);
    }
}

```

```

        oclose (&curc);
        ologof (&tpclda);
        exit (1);
    }

    if (oopen (&curs, &tpclda, (text *) 0, -1, -1, (text *) uid, -1))
{
        errrpt (&tpclda, &curs);
        oclose (&curw);
        oclose (&curd);
        oclose (&curc);
        oclose (&curh);
        ologof (&tpclda);
        exit (1);
    }

    if (oopen (&curi, &tpclda, (text *) 0, -1, -1, (text *) uid, -1))
{
        errrpt (&tpclda, &curi);
        oclose (&curw);
        oclose (&curd);
        oclose (&curc);
        oclose (&curh);
        oclose (&curs);
        ologof (&tpclda);
        exit (1);
    }

    if (oopen (&curo1, &tpclda, (text *) 0, -1, -1, (text *) uid, -
1)) {
        errrpt (&tpclda, &curo1);
        oclose (&curw);
        oclose (&curd);
        oclose (&curc);
        oclose (&curh);
        oclose (&curs);
        oclose (&curi);
        ologof (&tpclda);
        exit (1);
    }

    if (oopen (&curo2, &tpclda, (text *) 0, -1, -1, (text *) uid, -
1)) {
        errrpt (&tpclda, &curo2);
        oclose (&curw);
        oclose (&curd);
        oclose (&curc);
        oclose (&curh);
        oclose (&curs);
        oclose (&curi);
        oclose (&curo1);
        ologof (&tpclda);
        exit (1);
    }

    if (oopen (&curo11, &tpclda, (text *) 0, -1, -1, (text *) uid, -
1)) {
        errrpt (&tpclda, &curo11);
        oclose (&curw);
        oclose (&curd);
        oclose (&curc);
        oclose (&curh);
        oclose (&curs);
    }
}

```

```

        oclose (&curi);
        oclose (&curo1);
        oclose (&curo2);
        ologof (&tpclda);
        exit (1);
    }
1) {
    if (oopen (&curo12, &tpclda, (text *) 0, -1, -1, (text *) uid, -
        errrpt (&tpclda, &curo12);
        oclose (&curw);
        oclose (&curd);
        oclose (&curc);
        oclose (&curh);
        oclose (&curs);
        oclose (&curi);
        oclose (&curo1);
        oclose (&curo2);
        oclose (&curo11);
        ologof (&tpclda);
        exit (1);
    }
1) {
    if (oopen (&curno, &tpclda, (text *) 0, -1, -1, (text *) uid, -
        errrpt (&tpclda, &curno);
        oclose (&curw);
        oclose (&curd);
        oclose (&curc);
        oclose (&curh);
        oclose (&curs);
        oclose (&curi);
        oclose (&curo1);
        oclose (&curo2);
        oclose (&curo11);
        oclose (&curo12);
        ologof (&tpclda);
        exit (1);
    }
    /* parse statements */

    sprintf ((char *) sqlbuf, SQLTXTW);
    if (oparse (&curw, sqlbuf, -1, 0, 1)) {
        errrpt (&tpclda, &curw);
        quit ();
        exit (1);
    }

    sprintf ((char *) sqlbuf, SQLTXTD);
    if (oparse (&curd, sqlbuf, -1, 0, 1)) {
        errrpt (&tpclda, &curd);
        quit ();
        exit (1);
    }

    sprintf ((char *) sqlbuf, SQLTXTC);
    if (oparse (&curc, sqlbuf, -1, 0, 1)) {
        errrpt (&tpclda, &curc);
        quit ();
        exit (1);
    }
}

```

```

    sprintf ((char *) sqlbuf, SQLTXTH);
    if (oparse (&curh, sqlbuf, -1, 0, 1)) {
        errrpt (&tpclda, &curh);
        quit ();
        exit (1);
    }

    sprintf ((char *) sqlbuf, SQLTXTS);
    if (oparse (&curs, sqlbuf, -1, 0, 1)) {
        errrpt (&tpclda, &curs);
        quit ();
        exit (1);
    }

    sprintf ((char *) sqlbuf, SQLTXTI);
    if (oparse (&curi, sqlbuf, -1, 0, 1)) {
        errrpt (&tpclda, &curi);
        quit ();
        exit (1);
    }

    sprintf ((char *) sqlbuf, SQLTXTO1);
    if (oparse (&curo1, sqlbuf, -1, 0, 1)) {
        errrpt (&tpclda, &curo1);
        quit ();
        exit (1);
    }

    sprintf ((char *) sqlbuf, SQLTXTO2);
    if (oparse (&curo2, sqlbuf, -1, 0, 1)) {
        errrpt (&tpclda, &curo2);
        quit ();
        exit (1);
    }

    sprintf ((char *) sqlbuf, SQLTXTOL1);
    if (oparse (&curo11, sqlbuf, -1, 0, 1)) {
        errrpt (&tpclda, &curo11);
        quit ();
        exit (1);
    }

    sprintf ((char *) sqlbuf, SQLTXTOL2);
    if (oparse (&curo12, sqlbuf, -1, 0, 1)) {
        errrpt (&tpclda, &curo12);
        quit ();
        exit (1);
    }

    sprintf ((char *) sqlbuf, SQLTXTNO);
    if (oparse (&curno, sqlbuf, -1, 0, 1)) {
        errrpt (&tpclda, &curno);
        quit ();
        exit (1);
    }

    /* bind variables */

    /* ware */

    if (obndrv (&curw, (text *) ":w_id", -1, (ub1 *) &w_id, sizeof
(w_id),

```

```

        SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errrpt (&tpclda, &curw);
quit ();
exit (1);
}
if (obndrv (&curw, (text *) ":w_name", -1, (ubl *) w_name, 11,
        SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errrpt (&tpclda, &curw);
quit ();
exit (1);
}
21, if (obndrv (&curw, (text *) ":w_street_1", -1, (ubl *) w_street_1,
        SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errrpt (&tpclda, &curw);
quit ();
exit (1);
}
21, if (obndrv (&curw, (text *) ":w_street_2", -1, (ubl *) w_street_2,
        SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errrpt (&tpclda, &curw);
quit ();
exit (1);
}
if (obndrv (&curw, (text *) ":w_city", -1, (ubl *) w_city, 21,
        SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errrpt (&tpclda, &curw);
quit ();
exit (1);
}
if (obndrv (&curw, (text *) ":w_state", -1, (ubl *) w_state, 2,
        SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errrpt (&tpclda, &curw);
quit ();
exit (1);
}
if (obndrv (&curw, (text *) ":w_zip", -1, (ubl *) w_zip, 9,
        SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errrpt (&tpclda, &curw);
quit ();
exit (1);
}
if (obndrv (&curw, (text *) ":w_tax", -1, (ubl *) &w_tax, sizeof
(w_tax),
        SQLT_FLT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errrpt (&tpclda, &curw);
quit ();
exit (1);
}
/* dist */
if (obndrv (&curd, (text *) ":d_id", -1, (ubl *) d_id, sizeof
(int),
        SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {

```

```

errrpt (&tpclda, &curd);
quit ();
exit (1);
}
if (obndrv (&curd, (text *) ":d_w_id", -1, (ubl *) d_w_id, sizeof
(int),
        SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errrpt (&tpclda, &curd);
quit ();
exit (1);
}
if (obndrv (&curd, (text *) ":d_name", -1, (ubl *) d_name, 11,
        SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errrpt (&tpclda, &curd);
quit ();
exit (1);
}
21, if (obndrv (&curd, (text *) ":d_street_1", -1, (ubl *) d_street_1,
        SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errrpt (&tpclda, &curd);
quit ();
exit (1);
}
21, if (obndrv (&curd, (text *) ":d_street_2", -1, (ubl *) d_street_2,
        SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errrpt (&tpclda, &curd);
quit ();
exit (1);
}
if (obndrv (&curd, (text *) ":d_city", -1, (ubl *) d_city, 21,
        SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errrpt (&tpclda, &curd);
quit ();
exit (1);
}
if (obndrv (&curd, (text *) ":d_state", -1, (ubl *) d_state, 2,
        SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errrpt (&tpclda, &curd);
quit ();
exit (1);
}
if (obndrv (&curd, (text *) ":d_zip", -1, (ubl *) d_zip, 9,
        SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errrpt (&tpclda, &curd);
quit ();
exit (1);
}
if (obndrv (&curd, (text *) ":d_tax", -1, (ubl *) d_tax, sizeof
(float),
        SQLT_FLT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errrpt (&tpclda, &curd);
quit ();
exit (1);
}

```

```

}
/* cust */
if (obndrv (&curc, (text *) ":c_id", -1, (ubl *) c_id, sizeof
(int),
        SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curc);
    quit ();
    exit (1);
}
if (obndrv (&curc, (text *) ":c_d_id", -1, (ubl *) c_d_id, sizeof
(int),
        SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curc);
    quit ();
    exit (1);
}
if (obndrv (&curc, (text *) ":c_w_id", -1, (ubl *) c_w_id, sizeof
(int),
        SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curc);
    quit ();
    exit (1);
}
if (obndrv (&curc, (text *) ":c_first", -1, (ubl *) c_first, 17,
        SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curc);
    quit ();
    exit (1);
}
if (obndrv (&curc, (text *) ":c_last", -1, (ubl *) c_last, 17,
        SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curc);
    quit ();
    exit (1);
}
21, if (obndrv (&curc, (text *) ":c_street_1", -1, (ubl *) c_street_1,
        SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curc);
    quit ();
    exit (1);
}
21, if (obndrv (&curc, (text *) ":c_street_2", -1, (ubl *) c_street_2,
        SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curc);
    quit ();
    exit (1);
}
if (obndrv (&curc, (text *) ":c_city", -1, (ubl *) c_city, 21,
        SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curc);
    quit ();
    exit (1);
}

```

```

}
if (obndrv (&curc, (text *) ":c_state", -1, (ubl *) c_state, 2,
        SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curc);
    quit ();
    exit (1);
}
if (obndrv (&curc, (text *) ":c_zip", -1, (ubl *) c_zip, 9,
        SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curc);
    quit ();
    exit (1);
}
if (obndrv (&curc, (text *) ":c_phone", -1, (ubl *) c_phone, 16,
        SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curc);
    quit ();
    exit (1);
}
if (obndrv (&curc, (text *) ":c_credit", -1, (ubl *) c_credit, 2,
        SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curc);
    quit ();
    exit (1);
}
if (obndrv (&curc, (text *) ":c_discount", -1, (ubl *) c_discount,
        sizeof (float), SQLT_FLT, -1, (sb2 *) 0, (text *) 0,
-1,
        -1)) {
    errrpt (&tpclda, &curc);
    quit ();
    exit (1);
}
if (obndrv (&curc, (text *) ":c_data", -1, (ubl *) c_data, 501,
        SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curc);
    quit ();
    exit (1);
}
/* item */
if (obndrv (&curi, (text *) ":i_id", -1, (ubl *) i_id, sizeof
(int),
        SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curi);
    quit ();
    exit (1);
}
if (obndrv (&curi, (text *) ":i_im_id", -1, (ubl *) i_im_id,
sizeof (int),
        SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curi);
    quit ();
    exit (1);
}
}

```

```

if (obndrv (&curi, (text *) ":i_name", -1, (ubl *) i_name, 25,
    SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curi);
    quit ();
    exit (1);
}

if (obndrv (&curi, (text *) ":i_price", -1, (ubl *) i_price,
    sizeof (int), SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1,
    -1)) {
    errrpt (&tpclda, &curi);
    quit ();
    exit (1);
}

if (obndrv (&curi, (text *) ":i_data", -1, (ubl *) i_data, 51,
    SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curi);
    quit ();
    exit (1);
}

/* stok */

(int), if (obndrv (&curs, (text *) ":s_i_id", -1, (ubl *) s_i_id, sizeof
    SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curs);
    quit ();
    exit (1);
}

(int), if (obndrv (&curs, (text *) ":s_w_id", -1, (ubl *) s_w_id, sizeof
    SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curs);
    quit ();
    exit (1);
}

-1)) { if (obndrv (&curs, (text *) ":s_quantity", -1, (ubl *) s_quantity,
    sizeof (int), SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1,
    -1)) {
    errrpt (&tpclda, &curs);
    quit ();
    exit (1);
}

24, if (obndrv (&curs, (text *) ":s_dist_01", -1, (ubl *) s_dist_01,
    SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curs);
    quit ();
    exit (1);
}

24, if (obndrv (&curs, (text *) ":s_dist_02", -1, (ubl *) s_dist_02,
    SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curs);
    quit ();
    exit (1);
}

```

```

}

24, if (obndrv (&curs, (text *) ":s_dist_03", -1, (ubl *) s_dist_03,
    SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curs);
    quit ();
    exit (1);
}

24, if (obndrv (&curs, (text *) ":s_dist_04", -1, (ubl *) s_dist_04,
    SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curs);
    quit ();
    exit (1);
}

24, if (obndrv (&curs, (text *) ":s_dist_05", -1, (ubl *) s_dist_05,
    SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curs);
    quit ();
    exit (1);
}

24, if (obndrv (&curs, (text *) ":s_dist_06", -1, (ubl *) s_dist_06,
    SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curs);
    quit ();
    exit (1);
}

24, if (obndrv (&curs, (text *) ":s_dist_07", -1, (ubl *) s_dist_07,
    SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curs);
    quit ();
    exit (1);
}

24, if (obndrv (&curs, (text *) ":s_dist_08", -1, (ubl *) s_dist_08,
    SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curs);
    quit ();
    exit (1);
}

24, if (obndrv (&curs, (text *) ":s_dist_09", -1, (ubl *) s_dist_09,
    SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curs);
    quit ();
    exit (1);
}

24, if (obndrv (&curs, (text *) ":s_dist_10", -1, (ubl *) s_dist_10,
    SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curs);
    quit ();
}

```

```

    exit (1);
}
if (obndrv (&curs, (text *) ":s_data", -1, (ub1 *) s_data, 51,
           SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curs);
    quit ();
    exit (1);
}
/* hist */
if (obndrv (&curh, (text *) ":h_c_id", -1, (ub1 *) h_c_id, sizeof
(int),
           SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curh);
    quit ();
    exit (1);
}
if (obndrv (&curh, (text *) ":h_c_d_id", -1, (ub1 *) h_d_id,
sizeof (int),
           SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curh);
    quit ();
    exit (1);
}
if (obndrv (&curh, (text *) ":h_c_w_id", -1, (ub1 *) h_w_id,
sizeof (int),
           SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curh);
    quit ();
    exit (1);
}
if (obndrv (&curh, (text *) ":h_d_id", -1, (ub1 *) h_d_id, sizeof
(int),
           SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curh);
    quit ();
    exit (1);
}
if (obndrv (&curh, (text *) ":h_w_id", -1, (ub1 *) h_w_id, sizeof
(int),
           SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curh);
    quit ();
    exit (1);
}
if (obndrv (&curh, (text *) ":h_data", -1, (ub1 *) h_data, 25,
           SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curh);
    quit ();
    exit (1);
}
/* ordl (delivered) */
if (obndrv (&curoll, (text *) ":ol_o_id", -1, (ub1 *) ol_o_id,

```

```

           sizeof (int), SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1,
-1)) {
    errrpt (&tpclda, &curoll);
    quit ();
    exit (1);
}
if (obndrv (&curoll, (text *) ":ol_d_id", -1, (ub1 *) ol_d_id,
           sizeof (int), SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1,
-1)) {
    errrpt (&tpclda, &curoll);
    quit ();
    exit (1);
}
if (obndrv (&curoll, (text *) ":ol_w_id", -1, (ub1 *) ol_w_id,
           sizeof (int), SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1,
-1)) {
    errrpt (&tpclda, &curoll);
    quit ();
    exit (1);
}
if (obndrv (&curoll, (text *) ":ol_number", -1, (ub1 *) ol_number,
           sizeof (int), SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curoll);
    quit ();
    exit (1);
}
if (obndrv (&curoll, (text *) ":ol_i_id", -1, (ub1 *) ol_i_id,
           sizeof (int), SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curoll);
    quit ();
    exit (1);
}
if (obndrv (&curoll, (text *) ":ol_supply_w_id", -1,
           (ub1 *) ol_supply_w_id, sizeof (int), SQLT_INT, -1,
           (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curoll);
    quit ();
    exit (1);
}
if (obndrv (&curoll, (text *) ":ol_dist_info", -1, (ub1 *)
ol_dist_info,
           24, SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curoll);
    quit ();
    exit (1);
}
/* ordl (not delivered) */
if (obndrv (&curoll2, (text *) ":ol_o_id", -1, (ub1 *) ol_o_id,
           sizeof (int), SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1,
-1)) {
    errrpt (&tpclda, &curoll2);
    quit ();
    exit (1);
}

```

```

    if (obndrv (&curo12, (text *) ":ol_d_id", -1, (ub1 *) ol_d_id,
                sizeof (int), SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1,
-1)) {
        errrpt (&tpclda, &curo12);
        quit ();
        exit (1);
    }

    if (obndrv (&curo12, (text *) ":ol_w_id", -1, (ub1 *) ol_w_id,
                sizeof (int), SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1,
-1)) {
        errrpt (&tpclda, &curo12);
        quit ();
        exit (1);
    }

    if (obndrv (&curo12, (text *) ":ol_number", -1, (ub1 *) ol_number,
                sizeof (int), SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
        errrpt (&tpclda, &curo12);
        quit ();
        exit (1);
    }

    if (obndrv (&curo12, (text *) ":ol_i_id", -1, (ub1 *) ol_i_id,
                sizeof (int), SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
        errrpt (&tpclda, &curo12);
        quit ();
        exit (1);
    }

    if (obndrv (&curo12, (text *) ":ol_supply_w_id", -1,
                (ub1 *) ol_supply_w_id, sizeof (int), SQLT_INT, -1,
                (sb2 *) 0, (text *) 0, -1, -1)) {
        errrpt (&tpclda, &curo12);
        quit ();
        exit (1);
    }

    if (obndrv (&curo12, (text *) ":ol_amount", -1, (ub1 *) ol_amount,
                sizeof (int), SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
        errrpt (&tpclda, &curo12);
        quit ();
        exit (1);
    }

    if (obndrv (&curo12, (text *) ":ol_dist_info", -1, (ub1 *)
ol_dist_info,
                24, SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
        errrpt (&tpclda, &curo12);
        quit ();
        exit (1);
    }

    /* ordr (delivered) */

    if (obndrv (&curo1, (text *) ":o_id", -1, (ub1 *) o_id, sizeof
(int),
                SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
        errrpt (&tpclda, &curo1);
        quit ();
        exit (1);
    }
}

```

```

    if (obndrv (&curo1, (text *) ":o_d_id", -1, (ub1 *) o_d_id,
                sizeof (int),
                SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
        errrpt (&tpclda, &curo1);
        quit ();
        exit (1);
    }

    if (obndrv (&curo1, (text *) ":o_w_id", -1, (ub1 *) o_w_id,
                sizeof (int),
                SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
        errrpt (&tpclda, &curo1);
        quit ();
        exit (1);
    }

    if (obndrv (&curo1, (text *) ":o_c_id", -1, (ub1 *) o_c_id,
                sizeof (int),
                SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
        errrpt (&tpclda, &curo1);
        quit ();
        exit (1);
    }

    if (obndrv (&curo1, (text *) ":o_carrier_id", -1, (ub1 *)
o_carrier_id,
                sizeof (int), SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
        errrpt (&tpclda, &curo1);
        quit ();
        exit (1);
    }

    if (obndrv (&curo1, (text *) ":o_ol_cnt", -1, (ub1 *) o_ol_cnt,
                sizeof (int), SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
        errrpt (&tpclda, &curo1);
        quit ();
        exit (1);
    }

    /* ordr (not delivered) */

    if (obndrv (&curo2, (text *) ":o_id", -1, (ub1 *) o_id, sizeof
(int),
                SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
        errrpt (&tpclda, &curo2);
        quit ();
        exit (1);
    }

    if (obndrv (&curo2, (text *) ":o_d_id", -1, (ub1 *) o_d_id,
                sizeof (int),
                SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
        errrpt (&tpclda, &curo2);
        quit ();
        exit (1);
    }

    if (obndrv (&curo2, (text *) ":o_w_id", -1, (ub1 *) o_w_id,
                sizeof (int),
                SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
        errrpt (&tpclda, &curo2);
        quit ();
        exit (1);
    }
}

```

```

    }
    if (obndrv (&curo2, (text *) ":o_c_id", -1, (ubl *) o_c_id,
sizeof (int),
        SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
        errrpt (&tpclda, &curo2);
        quit ();
        exit (1);
    }
    if (obndrv (&curo2, (text *) ":o_ol_cnt", -1, (ubl *) o_ol_cnt,
sizeof (int), SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1,
-1)) {
        errrpt (&tpclda, &curo2);
        quit ();
        exit (1);
    }
    /* new order */
    if (obndrv (&curno, (text *) ":no_o_id", -1, (ubl *) no_o_id,
sizeof (int), SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1,
-1)) {
        errrpt (&tpclda, &curno);
        quit ();
        exit (1);
    }
    if (obndrv (&curno, (text *) ":no_d_id", -1, (ubl *) no_d_id,
sizeof (int), SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1,
-1)) {
        errrpt (&tpclda, &curno);
        quit ();
        exit (1);
    }
    if (obndrv (&curno, (text *) ":no_w_id", -1, (ubl *) no_w_id,
sizeof (int), SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1,
-1)) {
        errrpt (&tpclda, &curno);
        quit ();
        exit (1);
    }
}

/*-----+
| Initialize random number generator |
+-----*/
    srand (SEED);
#ifdef ORA_NT
    srand48 (SEED);
#endif
    initperm ();

/*-----+
| Load the WAREHOUSE table. |
+-----*/

    if (do_A || do_w) {
        nrows = eware - bware + 1;

```

```

        fprintf (stderr, "Loading/generating ware: w%d - w%d (%d rows)\n",
            bware, eware, nrows);

        begin_time = gettime ();
        begin_cpu = getcpu ();

        for (loop = bware; loop <= eware; loop++) {

            w_tax = (float)((lrand48 () % 2001) * 0.0001);
            randstr (w_name, 6, 10);
            randstr (w_street_1, 10, 20);
            randstr (w_street_2, 10, 20);
            randstr (w_city, 10, 20);
            randstr (str2, 2, 2);
            randnum (num9, 9);
            num9[4] = num9[5] = num9[6] = num9[7] = num9[8] = '1';

            if (gen) {
                printf ("%d 30000000 %f %s %s %s %s %s\n", loop, w_tax,
                    w_name, w_street_1, w_street_2, w_city, str2,
num9);
                fflush (stdout);
            }
            else {
                w_id = loop;
                strncpy (w_state, str2, 2);
                strncpy (w_zip, num9, 9);

                if (oexec (&curw)) {
                    errrpt (&tpclda, &curw);
                    orol (&tpclda);
                    fprintf (stderr, "Aborted at ware %d\n", loop);
                    quit ();
                    exit (1);
                }
                else if (ocom (&tpclda)) {
                    errrpt (&tpclda, &tpclda);
                    orol (&tpclda);
                    fprintf (stderr, "Aborted at ware %d\n", loop);
                    quit ();
                    exit (1);
                }
            }
        }

        end_time = gettime ();
        end_cpu = getcpu ();
        fprintf (stderr, "Done. %d rows loaded/generated in %10d sec.
(%10d cpu)\n\n",
            nrows, end_time - begin_time, end_cpu - begin_cpu);
    }

/*-----+
| Load the DISTRICT table. |
+-----*/

    if (do_A || do_d) {
        nrows = (eware - bware + 1) * DISTFAC;

        fprintf (stderr, "Loading/generating dist: w%d - w%d (%d rows)\n",
            bware, eware, nrows);

        begin_time = gettime ();

```

```

begin_cpu = getcpu ();
dwid = bware - 1;
for (row = 0; row < nrows; ) {
    dwid++;
    for (i = 0; i < DISTARR; i++, row++) {
        d_tax[i] = (float)((lrand48 () % 2001) * 0.0001);
        randstr (d_name[i], 6, 10);
        randstr (d_street_1[i], 10, 20);
        randstr (d_street_2[i], 10, 20);
        randstr (d_city[i], 10, 20);
        randstr (str2, 2, 2);
        randnum (num9, 9);
        num9[4] = num9[5] = num9[6] = num9[7] = num9[8] = '1';
        if (gen) {
            /* printf ("%d %d %s %s %s %s %s %s %d 30000.0 3001\n",
                i + 1, dwid, d_name[i], d_street_1[i],
d_street_2[i],
                d_city[i], str2, num9, d_tax[i]); */
            /* Reordered columns */
            printf ("%d %d 3000000 %f 3001 %s %s %s %s %s\n",
                i + 1, dwid, d_tax[i], d_name[i], d_street_1[i],
                d_street_2[i], d_city[i], str2, num9);
        }
        else {
            d_id[i] = i + 1;
            d_w_id[i] = dwid;
            strncpy (d_state[i], str2, 2);
            strncpy (d_zip[i], num9, 9);
        }
    }
    if (gen) {
        fflush (stdout);
    }
    else {
        if (oexn (&curd, DISTARR, 0)) {
            errrpt (&tpclda, &curd);
            orol (&tpclda);
            fprintf (stderr, "Aborted at ware %d, dist 1\n", dwid);
            quit ();
            exit (1);
        }
        else if (ocom (&tpclda)) {
            errrpt (&tpclda, &tpclda);
            orol (&tpclda);
            fprintf (stderr, "Aborted at ware %d, dist 1\n", dwid);
            quit ();
            exit (1);
        }
    }
}
end_time = gettime ();
end_cpu = getcpu ();
fprintf (stderr, "Done.  %d rows loaded/generated in %10d sec.
(%10d cpu)\n\n",
        nrows, end_time - begin_time, end_cpu - begin_cpu);
}

```

```

/*-----+
| Load the CUSTOMER table. |
+-----*/

if (do_A || do_c) {
    nrows = (eware - bware + 1) * CUSTFAC * DISTFAC;
    fprintf (stderr, "Loading/generating cust: w%d - w%d (%d rows)\n",
        bware, eware, nrows);
    begin_time = gettime ();
    begin_cpu = getcpu ();
    cid = 0;
    cdid = 1;
    cwid = bware;
    loopcount = 0;
    for (row = 0; row < nrows; ) {
        for (i = 0; i < CUSTARR; i++, row++) {
            cid++;
            if (cid > CUSTFAC) { /* cycle cust id */
                cid = 1; /* cheap mod */
                cdid++; /* shift dist cycle */
            }
            if (cdid > DISTFAC) {
                cdid = 1; /* shift ware cycle */
                cwid++;
            }
            c_id[i] = cid;
            c_d_id[i] = cdid;
            c_w_id[i] = cwid;
            if (cid <= 1000)
                randlastname (c_last[i], cid - 1);
            else
                randlastname (c_last[i], NURand (255, 0, 999, CNUM1));
            c_credit[i][1] = 'C';
            if (lrand48 () % 10)
                c_credit[i][0] = 'G';
            else
                c_credit[i][0] = 'B';
            c_discount[i] = (float)((lrand48 () % 5001) * 0.0001);
            randstr (c_first[i], 8, 16);
            randstr (c_street_1[i], 10, 20);
            randstr (c_street_2[i], 10, 20);
            randstr (c_city[i], 10, 20);
            randstr (str2, 2, 2);
            randnum (num9, 9);
            num9[4] = num9[5] = num9[6] = num9[7] = num9[8] = '1';
            randnum (num16, 16);
            randstr (c_data[i], 300, 500);
            if (gen) {
                printf ("%d %d %d %s OE %s %s %s %s %s %s %s %s %cC
5000000 %6.4f -1000 1000 1 0 %s\n",
                    cid, cdid, cwid, c_first[i], c_last[i],
                    c_street_1[i], c_street_2[i], c_city[i], str2,
num9,
                    num16, sdate, c_credit[i][0], c_discount[i],
c_data[i]);
            }
            else {

```

```

        strncpy (c_state[i], str2, 2);
        strncpy (c_zip[i], num9, 9);
        strncpy (c_phone[i], num16, 16);
    }
}
if (gen) {
    fflush (stdout);
}
else {
    if (oexn (&curc, CUSTARR, 0)) {
        errrpt (&tpclda, &curc);
        orol (&tpclda);
        fprintf (stderr, "Aborted at w_id %d, d_id %d,
c_id %d\n",
                c_w_id[0], c_d_id[0], c_id[0]);
        quit ();
        exit (1);
    }
    else if (ocom (&tpclda)) {
        errrpt (&tpclda, &tpclda);
        orol (&tpclda);
        fprintf (stderr, "Aborted at w_id %d, d_id %d,
c_id %d\n",
                c_w_id[0], c_d_id[0], c_id[0]);
        quit ();
        exit (1);
    }
}
if ((++loopcount) % 50)
    fprintf (stderr, ".");
else
    fprintf (stderr, " %d rows committed\n ", row);
}
end_time = gettime ();
end_cpu = getcpu ();
fprintf (stderr, "Done.  %d rows loaded/generated in %10d sec.
(%10d cpu)\n\n",
        nrows, end_time - begin_time, end_cpu - begin_cpu);
}
/*-----+
| Load the ITEM table.
+-----*/

if (do_A || do_i) {
    nrows = ITEMFAC;

    fprintf (stderr, "Loading/generating item: (%d rows)\n ",
nrows);

    begin_time = gettime ();
    begin_cpu = getcpu ();

    loopcount = 0;

    for (row = 0; row < nrows; ) {
        for (i = 0; i < ITEMARR; i++, row++) {
            i_im_id[i] = (lrand48 () % 10000) + 1;
            i_price[i] = ((lrand48 () % 9901) + 100);

```

```

        randstr (i_name[i], 14, 24);
        randdatastr (i_data[i], 26, 50);

        if (gen) {
            printf ("%d %d %s %d %s\n", row + 1, i_im_id[i],
i_name[i],
                    i_price[i], i_data[i]);
        }
        else {
            i_id[i] = row + 1;
        }
    }
}
if (gen) {
    fflush (stdout);
}
else {
    if (oexn (&curi, ITEMARR, 0)) {
        errrpt (&tpclda, &curi);
        orol (&tpclda);
        fprintf (stderr, "Aborted at i_id %d\n", i_id[0]);
        quit ();
        exit (1);
    }
    else if (ocom (&tpclda)) {
        errrpt (&tpclda, &tpclda);
        orol (&tpclda);
        fprintf (stderr, "Aborted at i_id %d\n", i_id[0]);
        quit ();
        exit (1);
    }
}
if ((++loopcount) % 50)
    fprintf (stderr, ".");
else
    fprintf (stderr, " %d rows committed\n ", row);
}
end_time = gettime ();
end_cpu = getcpu ();
fprintf (stderr, "Done.  %d rows loaded/generated in %10d sec.
(%10d cpu)\n\n",
        nrows, end_time - begin_time, end_cpu - begin_cpu);
}
/*-----+
| Load the STOCK table.
+-----*/

if (do_A || do_s) {

    nrows = (eware - bware + 1) * STOCFAC;

    fprintf (stderr, "Loading/generating stok: w%d - w%d (%d rows)\n
",
            bware, eware, nrows);

    begin_time = gettime ();
    begin_cpu = getcpu ();

    sid = 0;

```

```

swid = bware;
loopcount = 0;

for (row = 0; row < nrows; ) {
  for (i = 0; i < STOCARR; i++, row++) {
    if (++sid > STOCFAC) { /* cheap mod */
      sid = 1;
      swid++;
    }
    s_quantity[i] = (lrand48 () % 91) + 10;
    randstr (str24[0], 24, 24);
    randstr (str24[1], 24, 24);
    randstr (str24[2], 24, 24);
    randstr (str24[3], 24, 24);
    randstr (str24[4], 24, 24);
    randstr (str24[5], 24, 24);
    randstr (str24[6], 24, 24);
    randstr (str24[7], 24, 24);
    randstr (str24[8], 24, 24);
    randstr (str24[9], 24, 24);
    randdatastr (s_data[i], 26, 50);

    if (gen) {
      printf ("%d %d %d %s 0 0
0 %s\n",
              sid, swid, s_quantity[i], str24[0], str24[1],
              str24[3], str24[4], str24[5], str24[6], str24[7],
              str24[8], str24[9], s_data[i]);
    }
    else {
      s_i_id[i] = sid;
      s_w_id[i] = swid;
      strncpy (s_dist_01[i], str24[0], 24);
      strncpy (s_dist_02[i], str24[1], 24);
      strncpy (s_dist_03[i], str24[2], 24);
      strncpy (s_dist_04[i], str24[3], 24);
      strncpy (s_dist_05[i], str24[4], 24);
      strncpy (s_dist_06[i], str24[5], 24);
      strncpy (s_dist_07[i], str24[6], 24);
      strncpy (s_dist_08[i], str24[7], 24);
      strncpy (s_dist_09[i], str24[8], 24);
      strncpy (s_dist_10[i], str24[9], 24);
    }
  }
}

if (gen) {
  fflush (stdout);
}
else {
  if (oexn (&curs, STOCARR, 0)) {
    errrpt (&tpclda, &curs);
    orol (&tpclda);
    fprintf (stderr, "Aborted at w_id %d, s_i_id %d\n",
s_w_id[0],
            s_i_id[0]);
    quit ();
    exit (1);
  }
  else if (ocom (&tpclda)) {
    errrpt (&tpclda, &tpclda);
    orol (&tpclda);

```

```

    fprintf (stderr, "Aborted at w_id %d, s_i_id %d\n",
s_w_id[0],
            s_i_id[0]);
    quit ();
    exit (1);
  }
}

if ((++loopcount) % 50)
  fprintf (stderr, ".");
else
  fprintf (stderr, " %d rows committed\n ", row);
}

end_time = gettime ();
end_cpu = getcpu ();
fprintf (stderr, "Done. %d rows loaded/generated in %10d sec.
(%10d cpu)\n\n",
        nrows, end_time - begin_time, end_cpu - begin_cpu);
}

/*-----+
| Load the STOCK table (cluster around s_i_id). |
+-----*/

if (do_S) {
  nrows = (eitem - bitem + 1) * (eware - bware + 1);

  fprintf (stderr, "Loading/generating stok: i%d - i%d, w%d - w%d
(%d rows)\n ",
          bitem, eitem, bware, eware, nrows);

  begin_time = gettime ();
  begin_cpu = getcpu ();

  sid = bitem;
  swid = bware - 1;
  loopcount = 0;

  for (row = 0; row < nrows; ) {
    for (i = 0; i < STOCARR; i++, row++) {
      if (++swid > eware) { /* cheap mod */
        swid = bware;
        sid++;
      }
      s_quantity[i] = (lrand48 () % 91) + 10;
      randstr (str24[0], 24, 24);
      randstr (str24[1], 24, 24);
      randstr (str24[2], 24, 24);
      randstr (str24[3], 24, 24);
      randstr (str24[4], 24, 24);
      randstr (str24[5], 24, 24);
      randstr (str24[6], 24, 24);
      randstr (str24[7], 24, 24);
      randstr (str24[8], 24, 24);
      randstr (str24[9], 24, 24);
      randdatastr (s_data[i], 26, 50);

      if (gen) {
        printf ("%d %d %d %s 0 0
0 %s\n",

```

```

        sid, swid, s_quantity[i], str24[0], str24[1],
str24[2],
        str24[3], str24[4], str24[5], str24[6], str24[7],
        str24[8], str24[9], s_data[i]);
    }
    else {
        s_i_id[i] = sid;
        s_w_id[i] = swid;
        strncpy (s_dist_01[i], str24[0], 24);
        strncpy (s_dist_02[i], str24[1], 24);
        strncpy (s_dist_03[i], str24[2], 24);
        strncpy (s_dist_04[i], str24[3], 24);
        strncpy (s_dist_05[i], str24[4], 24);
        strncpy (s_dist_06[i], str24[5], 24);
        strncpy (s_dist_07[i], str24[6], 24);
        strncpy (s_dist_08[i], str24[7], 24);
        strncpy (s_dist_09[i], str24[8], 24);
        strncpy (s_dist_10[i], str24[9], 24);
    }
}
if (gen) {
    fflush (stdout);
}
else {
    if (oexn (&curs, STOCARR, 0)) {
        errrpt (&tpclda, &curs);
        orol (&tpclda);
        fprintf (stderr, "Aborted at w_id %d, s_i_id %d\n",
s_w_id[0],
                s_i_id[0]);
        quit ();
        exit (1);
    }
    else if (ocom (&tpclda)) {
        errrpt (&tpclda, &tpclda);
        orol (&tpclda);
        fprintf (stderr, "Aborted at w_id %d, s_i_id %d\n",
s_w_id[0],
                s_i_id[0]);
        quit ();
        exit (1);
    }
}
if ((++loopcount) % 50)
    fprintf (stderr, ".");
else
    fprintf (stderr, " %d rows committed\n", row);
}
end_time = gettime ();
end_cpu = getcpu ();
fprintf (stderr, "Done.  %d rows loaded/generated in %10d sec.
(%10d cpu)\n\n",
        nrows, end_time - begin_time, end_cpu - begin_cpu);
}
/*-----+
| Load the HISTORY table. |
+-----*/
if (do_A || do_h) {

```

```

nrows = (eware - bware + 1) * HISTFAC;
fprintf (stderr, "Loading/generating hist: w%d - w%d (%d rows)\n
",
        bware, eware, nrows);
begin_time = gettime ();
begin_cpu = getcpu ();
cid = 0;
cdid = 1;
cwid = bware;
loopcount = 0;
for (row = 0; row < nrows; ) {
    for (i = 0; i < HISTARR; i++, row++) {
        cid++;
        if (cid > CUSTFAC) { /* cycle cust id */
            cid = 1; /* cheap mod */
            cdid++; /* shift dist cycle */
            if (cdid > DISTFAC) {
                cdid = 1;
                cwid++; /* shift ware cycle */
            }
        }
        h_c_id[i] = cid;
        h_d_id[i] = cdid;
        h_w_id[i] = cwid;
        randstr (h_data[i], 12, 24);
        if (gen) {
            printf ("%d %d %d %d %d %s 1000 %s\n", cid, cdid, cwid,
cdid,
                    cwid, sdate, h_data[i]);
        }
    }
    if (gen) {
        fflush (stdout);
    }
    else {
        if (oexn (&curh, HISTARR, 0)) {
            errrpt (&tpclda, &curh);
            orol (&tpclda);
            fprintf (stderr, "Aborted at w_id %d, d_id %d,
c_id %d\n",
                    h_w_id[0], h_d_id[0], h_c_id[0]);
            quit ();
            exit (1);
        }
        else if (ocom (&tpclda)) {
            errrpt (&tpclda, &tpclda);
            orol (&tpclda);
            fprintf (stderr, "Aborted at w_id %d, d_id %d,
c_id %d\n",
                    h_w_id[0], h_d_id[0], h_c_id[0]);
            quit ();
            exit (1);
        }
    }
}
if ((++loopcount) % 50)
    fprintf (stderr, ".");
else

```

```

        fprintf (stderr, " %d rows committed\n ", row);
    }

    end_time = gettime ();
    end_cpu = getcpu ();
    fprintf (stderr, "Done.  %d rows loaded/generated in %10d sec.
(%10d cpu)\n\n",
            nrows, end_time - begin_time, end_cpu - begin_cpu);
}

/*-----+
| Load the ORDERS and ORDER-LINE table.          |
+-----*/

if (do_A || do_o) {
    nrows = (eware - bware + 1) * ORDEFAC * DISTFAC;

    fprintf (stderr, "Loading/generating order and order-line: w%d -
w%d (%d ord, ~%d ordl)\n ",
            bware, aware, nrows, nrows * 10);

    begin_time = gettime ();
    begin_cpu = getcpu ();

    cid = 0;
    cdid = 1;
    cwid = bware;
    loopcount = 0;

    for (row = 0; row < nrows; ) {
        for (i = 0; i < ORDEARR; i++, row++) {
            cid++;
            if (cid > ORDEFAC) {          /* cycle cust id */
                cid = 1;                 /* cheap mod */
                cdid++;                 /* shift dist cycle */
            }
            if (cdid > DISTFAC) {
                cdid = 1;
                cwid++;                 /* shift ware cycle */
            }
        }
        o_carrier_id[i] = lrand48 () % 10 + 1;
        o_ol_cnt[i] = olcnt = lrand48 () % 11 + 5;

        if (gen) {
            if (cid < 2101) {
                printf ("%d %d %d %d %s %d %d 1\n", cid, cdid, cwid,
                        randperm3000[cid - 1], sdate, o_carrier_id[i],
                        o_ol_cnt[i]);
            }
            else {
                /* set carrierid to 11 instead of null */
                printf ("%d %d %d %d %s 11 %d 1\n", cid, cdid, cwid,
                        randperm3000[cid - 1], sdate, o_ol_cnt[i]);
            }
        }
        else {
            o_id[i] = cid;
            o_d_id[i] = cdid;
            o_w_id[i] = cwid;
            o_c_id[i] = randperm3000[cid - 1];
        }

        for (j = 0; j < o_ol_cnt[i]; j++) {

```

```

        ol_i_id[j] = sid = lrand48 () % 100000 + 1;
        if (cid < 2101)
            ol_amount[j] = 0;
        else
            ol_amount[j] = (lrand48 () % 999999 + 1) ;
        randstr (str24[j], 24, 24);

        if (gen) {
            if (cid < 2101) {
                fprintf (olfp, "%d %d %d %d %s %d %d 5 %ld %s\n",
                    cid,
                    cdid, cwid, j + 1, sdate, ol_i_id[j],
                    cwid,
                    ol_amount[j], str24[j]);
            }
            else {
                /* Insert a default date instead of null date */
                fprintf (olfp, "%d %d %d %d 01-Jan-1811 %d %d
5 %ld %s\n", cid,
                    cdid, cwid, j + 1, ol_i_id[j], cwid,
                    ol_amount[j], str24[j]);
            }
        }
        else {
            ol_o_id[j] = cid;
            ol_d_id[j] = cdid;
            ol_w_id[j] = cwid;
            ol_number[j] = j + 1;
            ol_supply_w_id[j] = cwid;
            strncpy (ol_dist_info[j], str24[j], 24);
        }
    }

    if (gen) {
        fflush (olfp);
    }
    else {
        if (cid < 2101) {
            if (oexn (&curo11, olcnt, 0)) {
                errrpt (&tpclda, &curo11);
                orol (&tpclda);
                fprintf (stderr, "Aborted at w_id %d, d_id %d,
o_id %d\n",
                    cwid, cdid, cid);
                quit ();
                exit (1);
            }
            else if (ocom (&tpclda)) {
                errrpt (&tpclda, &tpclda);
                orol (&tpclda);
                fprintf (stderr, "Aborted at w_id %d, d_id %d,
o_id %d\n",
                    cwid, cdid, cid);
                quit ();
                exit (1);
            }
        }
        else {
            if (oexn (&curo12, olcnt, 0)) {
                errrpt (&tpclda, &curo12);
                orol (&tpclda);
                fprintf (stderr, "Aborted at w_id %d, d_id %d,
o_id %d\n",

```

```

                cwid, cdid, cid);
            quit ();
            exit (1);
        }
        else if (ocom (&tpclda)) {
            errrpt (&tpclda, &tpclda);
            orol (&tpclda);
            fprintf (stderr, "Aborted at w_id %d, d_id %d,
o_id %d\n",
                cwid, cdid, cid);
            quit ();
            exit (1);
        }
    }
}
}
if (gen) {
    fflush (stdout);
}
else {
    if (cid < 2101) {
        if (oexn (&curo1, ORDEARR, 0)) {
            errrpt (&tpclda, &curo1);
            orol (&tpclda);
            fprintf (stderr, "Aborted at w_id %d, d_id %d,
o_id %d\n ",
                cwid, cdid, cid);
            quit ();
            exit (1);
        }
        else if (ocom (&tpclda)) {
            errrpt (&tpclda, &tpclda);
            orol (&tpclda);
            fprintf (stderr, "Aborted at w_id %d, d_id %d,
o_id %d\n ",
                cwid, cdid, cid);
            quit ();
            exit (1);
        }
    }
    else {
        if (oexn (&curo2, ORDEARR, 0)) {
            errrpt (&tpclda, &curo2);
            orol (&tpclda);
            fprintf (stderr, "Aborted at w_id %d, d_id %d,
o_id %d\n ",
                cwid, cdid, cid);
            quit ();
            exit (1);
        }
        else if (ocom (&tpclda)) {
            errrpt (&tpclda, &tpclda);
            orol (&tpclda);
            fprintf (stderr, "Aborted at w_id %d, d_id %d,
o_id %d\n ",
                cwid, cdid, cid);
            quit ();
            exit (1);
        }
    }
}
}
}

```

```

        if ((++loopcount) % 50)
            fprintf (stderr, ".");
        else
            fprintf (stderr, " %d ordr committed\n ", row);
    }

    end_time = gettime ();
    end_cpu = getcpu ();
    fprintf (stderr, "Done. %d ordr loaded/generated in %10d sec.
(%10d cpu)\n\n",
        nrows, end_time - begin_time, end_cpu - begin_cpu);
}

/*-----+
| Load the NEW-ORDER table. |
+-----*/

if (do_A || do_n) {
    nrows = (eware - bware + 1) * NEWOFAC * DISTFAC;

    fprintf (stderr, "Loading/generating new-order: w%d - w%d (%d
rows)\n ",
        bware, eware, nrows);

    begin_time = gettime ();
    begin_cpu = getcpu ();

    cid = 0;
    cdid = 1;
    cwid = bware;
    loopcount = 0;

    for (row = 0; row < nrows; ) {
        for (i = 0; i < NEWOARR; i++, row++) {
            cid++;
            if (cid > NEWOFAC) {
                cid = 1;
                cdid++;
                if (cdid > DISTFAC) {
                    cdid = 1;
                    cwid++;
                }
            }
        }

        if (gen) {
            printf ("%d %d %d\n", cid + 2100, cdid, cwid);
        }
        else {
            no_o_id[i] = cid + 2100;
            no_d_id[i] = cdid;
            no_w_id[i] = cwid;
        }
    }

    if (gen) {
        fflush (stdout);
    }
    else {
        if (oexn (&curno, NEWOARR, 0)) {
            errrpt (&tpclda, &curno);
            orol (&tpclda);
            fprintf (stderr, "Aborted at w_id %d, d_id %d, o_id %d\n
",

```

```

        cwid, cdid, cid + 2100);
        quit ();
        exit (1);
    }
    else if (ocom (&tpclda)) {
        errrpt (&tpclda, &tpclda);
        orol (&tpclda);
        fprintf (stderr, "Aborted at w_id %d, d_id %d, o_id %d\n",
",
        cwid, cdid, cid + 2100);
        quit ();
        exit (1);
    }
}

if ((++loopcount) % 45)
    fprintf (stderr, ".");
else
    fprintf (stderr, " %d rows committed\n", row);
}

end_time = gettime ();
end_cpu = getcpu ();
fprintf (stderr, "Done.  %d rows loaded/generated in %10d sec.
(%10d cpu)\n\n",
        nrows, end_time - begin_time, end_cpu - begin_cpu);
}

/*-----+
| clean up and exit.                               |
+-----*/

if (olfp)
    fclose (olfp);
if (!gen)
    quit ();
exit (0);
}

void initperm ()
{
    int i;
    int pos;
    int temp;

    /* init randperm3000 */

    for (i = 0; i < 3000; i++)
        randperm3000[i] = i + 1;
    for (i = 3000; i > 0; i--) {
        pos = lrand48 () % i;
        temp = randperm3000[i - 1];
        randperm3000[i - 1] = randperm3000[pos];
        randperm3000[pos] = temp;
    }
}

```

```

void randstr (str, x, y)

char *str;
int x;
int y;

{
    int i, j;
    int len;

    len = (lrand48 () % (y - x + 1)) + x;
    for (i = 0; i < len; i++) {
        j = lrand48 () % 62;
        if (j < 26)
            str[i] = (char) (j + 'a');
        else if (j < 52)
            str[i] = (char) (j - 26 + 'A');
        else
            str[i] = (char) (j - 52 + '0');
    }
    str[len] = '\0';
}

void randdatastr (str, x, y)

char *str;
int x;
int y;

{
    int i, j;
    int len;
    int pos;

    len = (lrand48 () % (y - x + 1)) + x;
    for (i = 0; i < len; i++) {
        j = lrand48 () % 62;
        if (j < 26)
            str[i] = (char) (j + 'a');
        else if (j < 52)
            str[i] = (char) (j - 26 + 'A');
        else
            str[i] = (char) (j - 52 + '0');
    }
    str[len] = '\0';
    if ((lrand48 () % 10) == 0) {
        pos = (lrand48 () % (len - 8));
        str[pos] = 'O';
        str[pos + 1] = 'R';
        str[pos + 2] = 'I';
        str[pos + 3] = 'G';
        str[pos + 4] = 'I';
        str[pos + 5] = 'N';
        str[pos + 6] = 'A';
        str[pos + 7] = 'L';
    }
}

```

```

}

void randnum (str, len)

char *str;
int len;

{
    int i;

    for (i = 0; i < len; i++)
        str[i] = (char) (lrand48 () % 10 + '0');
    str[len] = '\0';
}

void randlastname (str, id)

char *str;
int id;

{
    id = id % 1000;
    strcpy (str, lastname[id / 100]);
    strcat (str, lastname[(id / 10) % 10]);
    strcat (str, lastname[id % 10]);
}

int NURand (A, x, y, cnum)

int A, x, y, cnum;

{
    int a, b;

    a = lrand48 () % (A + 1);
    b = (lrand48 () % (y - x + 1)) + x;
    return (((a | b) + cnum) % (y - x + 1)) + x;
}

void sysdate (sdate)

char *sdate;

{
    time t tp;
    struct tm *tmptr;

    time (&tp);

```

```

tmptr = localtime (&tp);
strftime (sdate, 29, "%d-%b-%Y", tmptr);
}

```

B3. MISC

dml.sql

```

REM=====+
REM          Copyright (c) 1996 Oracle Corp, Redwood Shores, CA      |
REM          OPEN SYSTEMS PERFORMANCE GROUP                          |
REM          All Rights Reserved                                      |
REM=====+
REM FILENAME
REM          dml.sql
REM DESCRIPTION
REM          Disable table locks for TPC-C tables.
REM USAGE
REM          sqlplus tpcc/tpcc dml.sql
REM=====+

connect tpcc/tpcc;
set echo on;

        alter table ware disable table lock;
        alter table dist disable table lock;
        alter table cust disable table lock;
        alter table hist disable table lock;
        alter table item disable table lock;
        alter table stok disable table lock;
        alter table ordr disable table lock;
        alter table nord disable table lock;
        alter table ordl disable table lock;

set echo off;

```

undml.sql

```

REM=====+
REM          Copyright (c) 1996 Oracle Corp, Redwood Shores, CA      |
REM          OPEN SYSTEMS PERFORMANCE GROUP                          |
REM          All Rights Reserved                                      |
REM=====+
REM FILENAME
REM          undml.sql
REM DESCRIPTION
REM          Enable table locks for TPC-C tables.
REM USAGE
REM          sqlplus tpcc/tpcc @undml
REM=====+*/

connect tpcc/tpcc;
spool undml.log;
set echo on;

        alter table ware enable table lock;

```

```

alter table dist enable table lock;
alter table cust enable table lock;
alter table hist enable table lock;
alter table item enable table lock;
alter table stok enable table lock;
alter table ordr enable table lock;
alter table nord enable table lock;
alter table ordl enable table lock;

set echo off;
spool off;

```

views.sql

```

connect tpcc/tpcc;
set echo on;

create or replace view wh_cust
(w_id, w_tax, c_id, c_d_id, c_w_id, c_discount, c_last, c_credit)
as select w.w_id, w.w_tax,
         c.c_id, c.c_d_id, c.c_w_id, c.c_discount, c.c_last,
         c.c_credit
   from cust c, ware w
  where w.w_id = c.c_w_id;

create or replace view wh_dist
(w_id, d_id, d_tax, d_next_o_id, w_tax )
as select w.w_id, d.d_id, d.d_tax, d.d_next_o_id, w.w_tax
   from dist d, ware w
  where w.w_id = d.d_w_id;

create or replace view stock_item
(i_id, s_w_id, i_price, i_name, i_data, s_data, s_quantity,
 s_order_cnt, s_ytd, s_remote_cnt,
 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)
as
select i.i_id, s.w_id, i.i_price, i.i_name, i.i_data, s_data,
       s_quantity,
       s_order_cnt, s_ytd, s_remote_cnt,
       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
   from stok s, item i
  where i.i_id = s.s_i_id;

set echo off;

```

Appendix C : Tunable Parameters

RTE input parameter

The following parameters were used with NEC proprietary RTE.

```
*****
Script used in the test
*****

net time \\\r01 /set

if [ `hostname` = "r01" ]; then
    start cmd /K master -u1300 -s1800 -d90 -c48000 -w4800 -C200 -
S200 -n66 \
    -K21.04,3.50,2.33,2.33,2.33 -T14.03,14.03,11.69,5.84,5.84
fi

sleep 3

case `hostname` in
"r01")
start client7 -Is01 -Mr01 -c1000 -o1
start client7 -Is02 -Mr01 -c1000 -o1001
start client7 -Is03 -Mr01 -c1000 -o2001
start client7 -Is04 -Mr01 -c1000 -o3001
start client7 -Is05 -Mr01 -c1000 -o4001
start client7 -Is06 -Mr01 -c1000 -o5001
start client7 -Is07 -Mr01 -c1000 -o6001
;;
"r011")
start client7 -Is08 -Mr01 -c1000 -o7001
start client7 -Is09 -Mr01 -c1000 -o8001
start client7 -Is10 -Mr01 -c1000 -o9001
start client7 -Is11 -Mr01 -c1000 -o10001
start client7 -Is12 -Mr01 -c1000 -o11001
;;
"r02")
start client7 -Is01 -Mr01 -c1000 -o12001
start client7 -Is02 -Mr01 -c1000 -o13001
start client7 -Is03 -Mr01 -c1000 -o14001
start client7 -Is04 -Mr01 -c1000 -o15001
start client7 -Is05 -Mr01 -c1000 -o16001
start client7 -Is06 -Mr01 -c1000 -o17001
start client7 -Is07 -Mr01 -c1000 -o18001
;;
"r021")
start client7 -Is08 -Mr01 -c1000 -o19001
start client7 -Is09 -Mr01 -c1000 -o20001
start client7 -Is10 -Mr01 -c1000 -o21001
start client7 -Is11 -Mr01 -c1000 -o22001
start client7 -Is12 -Mr01 -c1000 -o23001
;;
"r03")
```

```
start client7 -Is01 -Mr01 -c1000 -o24001
start client7 -Is02 -Mr01 -c1000 -o25001
start client7 -Is03 -Mr01 -c1000 -o26001
start client7 -Is04 -Mr01 -c1000 -o27001
start client7 -Is05 -Mr01 -c1000 -o28001
start client7 -Is06 -Mr01 -c1000 -o29001
start client7 -Is07 -Mr01 -c1000 -o30001
;;
"r031")
start client7 -Is08 -Mr01 -c1000 -o31001
start client7 -Is09 -Mr01 -c1000 -o32001
start client7 -Is10 -Mr01 -c1000 -o33001
start client7 -Is11 -Mr01 -c1000 -o34001
start client7 -Is12 -Mr01 -c1000 -o35001
;;
"r04")
start client7 -Is01 -Mr01 -c1000 -o36001
start client7 -Is02 -Mr01 -c1000 -o37001
start client7 -Is03 -Mr01 -c1000 -o38001
start client7 -Is04 -Mr01 -c1000 -o39001
start client7 -Is05 -Mr01 -c1000 -o40001
start client7 -Is06 -Mr01 -c1000 -o41001
start client7 -Is07 -Mr01 -c1000 -o42001
;;
"r041")
start client7 -Is08 -Mr01 -c1000 -o43001
start client7 -Is09 -Mr01 -c1000 -o44001
start client7 -Is10 -Mr01 -c1000 -o45001
start client7 -Is11 -Mr01 -c1000 -o46001
start client7 -Is12 -Mr01 -c1000 -o47001
;;
esac
```

```
*****
master.exe help message
*****
```

master -- TPC-C master driver Version 2.00.01

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	223
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 Trouguput 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
4480 (NewOrder)
4308 (Payment)
404 (OrderStatus)
404 (Delivery)
404 (StockLevel)

-T Comma Separated List of Think Times (sec)
12.05 (NewOrder)
12.05 (Payment)
10.05 (OrderStatus)
5.05 (Delivery)
5.05 (StockLevel)

-K Comma Separated List of Keying Times (sec)
18.01 (NewOrder)
3.01 (Payment)
2.01 (OrderStatus)
2.01 (Delivery)
2.01 (StockLevel)

-D Web Browser Delay Time (sec)
0.10

```

Note: Command line switches are case sensitive.

```

*****
client7.exe help message
*****

```

client7 -- TPC-C client driver Version 2.00.01

```

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

```

Note: Command line switches are case sensitive.

Server Configuration (Common to all nodes)

Microsoft Windows NT Server version 4.0 Configuration Parameters

The following services were disabled in the Windows NT Control Panel/Service:

```

-Alerter
-Computer Browser
-License Logging Service
-Messenger
-Net logon
-Network DDE

```

```

-NT LM Security Support Provider
-Plug and Play
-Spooler
-TCP/IP Netbios Helper

```

BOOT. INI

The /3gb switch was added to the boot. ini file to cause NT Enterprise Server to allow 3GB of user and 1GB of kernel virtual address space, rather than the usual 2GB of virtual address space.

System Configuration Report

Microsoft Diagnostics Report For \\NODE1

OS Version Report

```

Microsoft (R) Windows NT (TM) Server
Version 4.0 (Build 1381: Service Pack 4) x86 Multiprocessor Free
Registered Owner: nec, nec
Product Number: 70238-415-0002856-97681

```

System Report

```

System: AT/AT COMPATIBLE
Hardware Abstraction Layer: MPS 1.4 - APIC platform
BIOS Date: 03/05/99
BIOS Version: PhoenixBIOS 4.0 Release 6.0.1339

```

Processor list:

```

0: x86 Family 6 Model 7 Stepping 2 GenuineIntel ~500 Mhz
1: x86 Family 6 Model 7 Stepping 2 GenuineIntel ~500 Mhz
2: x86 Family 6 Model 7 Stepping 2 GenuineIntel ~500 Mhz
3: x86 Family 6 Model 7 Stepping 2 GenuineIntel ~500 Mhz
4: x86 Family 6 Model 7 Stepping 2 GenuineIntel ~500 Mhz
5: x86 Family 6 Model 7 Stepping 2 GenuineIntel ~500 Mhz
6: x86 Family 6 Model 7 Stepping 2 GenuineIntel ~500 Mhz
7: x86 Family 6 Model 7 Stepping 2 GenuineIntel ~500 Mhz

```

Video Display Report

```

BIOS Date: 06/25/98
BIOS Version: CL-GD546x Laguna PCI VGA BIOS Version 1.71e c15

```

Adapter:

```

Setting: 1280 x 1024 x 256
72 Hz
Type: cl546xm compatible display adapter
String: Cirrus Logic VisualMedia(TM) Accelerator
Memory: 2 MB
Chip Type: Cirrus Logic 5465
DAC Type: Internal
Driver:
Vendor: Cirrus Logic, Inc.
File(s): cl546xm.sys, cl5465.dll
Version: 4.00.1381.1705-1.705, 4.0.101

```

Drives Report

```

C:\ (Local - NTFS) Total: 2,096,450 KB, Free: 864,951 KB
  Serial Number: 6861 - 4A0F
  Bytes per cluster: 512
  Sectors per cluster: 1
  Filename length: 255
D:\ (Local - NTFS) Total: 6,787,460 KB, Free: 1,820,192 KB
  Serial Number: 84CD - 8F77
  Bytes per cluster: 512
  Sectors per cluster: 8
  Filename length: 255
E:\ (Local - NTFS) Total: 76,856,300 KB, Free: 6,979,884 KB
  Serial Number: 8861 - BB15
  Bytes per cluster: 512
  Sectors per cluster: 8
  Filename length: 255
F:\ (Local - NTFS) Total: 111,014,892 KB, Free: 13,899,100 KB
  Serial Number: 845C - 54C5
  Bytes per cluster: 512
  Sectors per cluster: 8
  Filename length: 255
G:\ (Local - NTFS) Total: 76,856,300 KB, Free: 6,979,988 KB
  Serial Number: C049 - 5FOC
  Bytes per cluster: 512
  Sectors per cluster: 8
  Filename length: 255
H:\ (Local - NTFS) Total: 76,856,300 KB, Free: 6,979,980 KB
  Serial Number: 9C79 - D6F6
  Bytes per cluster: 512
  Sectors per cluster: 8
  Filename length: 255
I:\ (Local - NTFS) Total: 76,856,300 KB, Free: 6,980,016 KB
  Serial Number: E079 - DEAO
  Bytes per cluster: 512
  Sectors per cluster: 8
  Filename length: 255
J:\ (Local - NTFS) Total: 111,014,892 KB, Free: 11,969,816 KB
  Serial Number: 4D6 - 2E2B
  Bytes per cluster: 512
  Sectors per cluster: 8
  Filename length: 255
K:\ (Local - NTFS) Total: 103,711,724 KB, Free: 4,666,868 KB
  Serial Number: 5441 - CCCE
  Bytes per cluster: 512
  Sectors per cluster: 8
  Filename length: 255
L:\ (Local - NTFS) Total: 103,711,724 KB, Free: 4,666,864 KB
  Serial Number: A85B - 5FE8
  Bytes per cluster: 512
  Sectors per cluster: 8
  Filename length: 255
M:\ (Local - NTFS) Total: 103,204,844 KB, Free: 6,089,372 KB
  Serial Number: 7C72 - 9555
  Bytes per cluster: 512
  Sectors per cluster: 8
  Filename length: 255
N:\ (Local - NTFS) Total: 103,711,724 KB, Free: 5,188,088 KB
  Serial Number: 1889 - 1309
  Bytes per cluster: 512
  Sectors per cluster: 8
  Filename length: 255
O:\ (Local - NTFS) Total: 103,204,844 KB, Free: 6,089,372 KB
  Serial Number: 549B - 9B8F
  Bytes per cluster: 512
  Sectors per cluster: 8
  Filename length: 255
P:\ (Local - NTFS) Total: 103,711,724 KB, Free: 6,596,236 KB
  Serial Number: 1CB0 - 4F98
  Bytes per cluster: 512
  Sectors per cluster: 8
  Filename length: 255
Q:\ (Local - NTFS) Total: 103,691,244 KB, Free: 4,646,396 KB

```

```

  Serial Number: 282E - 7FFA
  Bytes per cluster: 512
  Sectors per cluster: 8
  Filename length: 255
R:\ (Local - NTFS) Total: 103,691,244 KB, Free: 4,646,396 KB
  Serial Number: 40CB - C418
  Bytes per cluster: 512
  Sectors per cluster: 8
  Filename length: 255
S:\ (Local - NTFS) Total: 103,691,244 KB, Free: 6,575,756 KB
  Serial Number: B048 - A461
  Bytes per cluster: 512
  Sectors per cluster: 8
  Filename length: 255
T:\ (Local - NTFS) Total: 103,727,084 KB, Free: 5,203,440 KB
  Serial Number: 5C7C - 2E68
  Bytes per cluster: 512
  Sectors per cluster: 8
  Filename length: 255
U:\ (Local - NTFS) Total: 103,727,084 KB, Free: 6,611,596 KB
  Serial Number: 8486 - AD44
  Bytes per cluster: 512
  Sectors per cluster: 8
  Filename length: 255
V:\ (Local - NTFS) Total: 103,727,084 KB, Free: 6,611,596 KB
  Serial Number: 248F - 1FE4
  Bytes per cluster: 512
  Sectors per cluster: 8
  Filename length: 255
Y:\ (Local - NTFS) Total: 102,996,972 KB, Free: 87,897,784 KB
  Serial Number: 54B7 - 8764
  Bytes per cluster: 512
  Sectors per cluster: 8
  Filename length: 255

```

Memory Report

```

-----
Handles: 11,445
Threads: 139
Processes: 27

```

Physical Memory (K)

```

Total: 4,128,176
Available: 1,232,056
File Cache: 17,856

```

Kernel Memory (K)

```

Total: 30,360
Paged: 10,884
Nonpaged: 19,476

```

Commit Charge (K)

```

Total: 2,825,440
Limit: 8,106,156
Peak: 2,988,384

```

Pagefile Space (K)

```

Total: 4,140,032
Total in use: 32,556
Peak: 34,564

```

D:\pagefile.sys

```

Total: 4,140,032
Total in use: 32,556
Peak: 34,564

```

Services Report

```

-----
Alerter                               Stopped (Manual)

```

```

C:\WINNT\System32\services.exe
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Shared Process
Service Dependencies:
  LanmanWorkstation
Computer Browser                               Stopped (Manual)
C:\WINNT\System32\services.exe
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Shared Process
Service Dependencies:
  LanmanWorkstation
  LanmanServer
  LmHosts
ClipBook Server                               Stopped (Manual)
C:\WINNT\System32\clipsrv.exe
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Own Process
Service Dependencies:
  NetDDE
DHCP Client (TDI)                             Stopped (Disabled)
C:\WINNT\System32\services.exe
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Shared Process
Service Dependencies:
  Tcpip
  Afd
  NetBT
EventLog (Event log)                          Running (Automatic)
C:\WINNT\System32\services.exe
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Shared Process
Server                                         Running (Automatic)
C:\WINNT\System32\services.exe
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Shared Process
Group Dependencies:
  TDI
Workstation (NetworkProvider)                 Running (Automatic)
C:\WINNT\System32\services.exe
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Shared Process
Group Dependencies:
  TDI
License Logging Service                       Stopped (Manual)
C:\WINNT\System32\llssrv.exe
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Own Process
TCP/IP NetBIOS Helper                        Stopped (Manual)
C:\WINNT\System32\services.exe
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Shared Process
Group Dependencies:
  NetworkProvider
Messenger                                     Stopped (Manual)
C:\WINNT\System32\services.exe
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Shared Process
Service Dependencies:
  LanmanWorkstation
  NetBios
Navisphere Agent                             Stopped (Manual)

```

```

C:\Program Files\CLARiiON\Navisphere Agent\Naviagent.Exe
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Own Process
Network DDE (NetDDEGroup)                    Stopped (Manual)
C:\WINNT\System32\netdde.exe
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Shared Process
Service Dependencies:
  NetDDESDM
Network DDE DSDM                             Stopped (Manual)
C:\WINNT\System32\netdde.exe
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Shared Process
Net Logon (RemoteValidation)                 Stopped (Manual)
C:\WINNT\System32\lsass.exe
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Shared Process
Service Dependencies:
  LanmanWorkstation
  LmHosts
NT LM Security Support Provider              Stopped (Manual)
C:\WINNT\System32\SERVICES.EXE
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Shared Process
OracleCMService (Cluster Manager)           Running (Manual)
c:\oracle\ref2\osdbin\cmsrvr.exe
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Own Process
Service Dependencies:
  OracleNMService
OracleNMService                              Running (Manual)
c:\oracle\ref2\osdbin\nmsrvr.exe
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Own Process
OracleOraHome81Agent                        Stopped (Manual)
C:\Oracle\Ora81\bin\dbsnmp.exe
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Own Process
OracleOraHome81ClientCache                  Stopped (Manual)
C:\Oracle\Ora81\BIN\ONRSD.EXE
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Own Process
OracleOraHome81CMAdmin                      Stopped (Manual)
C:\Oracle\Ora81\BIN\CMADMIN.EXE
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Own Process
OracleOraHome81CMan                          Stopped (Manual)
C:\Oracle\Ora81\BIN\CMGW.EXE
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Own Process
OracleOraHome81DataGatherer                 Stopped (Manual)
C:\Oracle\Ora81\bin\vppdc.exe
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Own Process
OracleOraHome81TNSListener                  Running (Automatic)
C:\Oracle\Ora81\BIN\TNLSNR
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Own Process

```



```

Changer (Filter)                Stopped (System)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
cirrus (Video)                  Stopped (Disabled)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
cl546x (Video)                  Running (System)
  System32\DRIVERS\cl546xm.sys
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
Cpqarray (SCSI miniport)       Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
cpqfw2e (SCSI miniport)        Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
dac960nt (SCSI miniport)       Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
dce376nt (SCSI miniport)       Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Dell_sda (SCSI miniport)       Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Dell_DGX (Video)               Stopped (Disabled)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
Disk (SCSI Class)              Running (Boot)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
  Group Dependencies:
    SCSI miniport
Diskperf (Filter)              Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
DptScsi (SCSI miniport)        Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
dtc329x (SCSI miniport)        Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
3Com 3C90x Adapter Driver (NDIS) Running (Automatic)
  C:\WINNT\System32\drivers\el90xnd4.sys
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
et4000 (Video)                 Stopped (Disabled)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
Fastfat (Boot file system)     Running (Disabled)
  Error Severity: Normal
  Service Flags: File System Driver, Shared Process
Fd16_700 (SCSI miniport)       Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Fd7000ex (SCSI miniport)       Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Fd8xx (SCSI miniport)          Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
flashpnt (SCSI miniport)       Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Floppy (Primary disk)          Running (System)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
Ftdisk (Filter)                Stopped (Disabled)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
i8042 Keyboard and PS/2 Mouse Port Driver (Keyboard Port) Running (System)

```

```

System32\DRIVERS\i8042prt.sys
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Inport (Pointer Port)          Stopped (Disabled)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
Jazzg300 (Video)               Stopped (Disabled)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
Jazzg364 (Video)               Stopped (Disabled)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
Jzvxl484 (Video)               Stopped (Disabled)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
Keyboard Class Driver (Keyboard Class) Running (System)
  System32\DRIVERS\kbdclass.sys
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
KSecDD (Base)                  Running (System)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
mainte (Extended Base)         Stopped (Manual)
  System32\DRIVERS\mainte.sys
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
mga (Video)                    Stopped (Disabled)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
mga_mil (Video)                Stopped (Disabled)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
mitsumi (SCSI miniport)        Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
mkecr5xx (SCSI miniport)       Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Modem (Extended base)          Stopped (Manual)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
Mouse Class Driver (Pointer Class) Running (System)
  System32\DRIVERS\mouclass.sys
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Msfs (File system)             Running (System)
  Error Severity: Normal
  Service Flags: File System Driver, Shared Process
Mup (Network)                  Running (Manual)
  C:\WINNT\System32\drivers\mup.sys
  Error Severity: Normal
  Service Flags: File System Driver, Shared Process
Ncr53c9x (SCSI miniport)       Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
ncr77c22 (Video)               Stopped (Disabled)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
Ncr700 (SCSI miniport)         Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Ncr710 (SCSI miniport)         Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Microsoft NDIS System Driver (NDIS) Running (System)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
NetBIOS Interface (NetBIOSGroup) Stopped (Manual)
  C:\WINNT\System32\drivers\netbios.sys
  Error Severity: Normal
  Service Flags: File System Driver, Shared Process

```

```

Group Dependencies:
  TDI
WINS Client(TCP/IP) (PNP_TDI) Running (Automatic)
  C:\WINNT\System32\drivers\netbt.sys
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
  Service Dependencies:
    Tcpip
NetDetect Stopped (Manual)
  C:\WINNT\system32\drivers\netdect.sys
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Npfs (File system) Running (System)
  Error Severity: Normal
  Service Flags: File System Driver, Shared Process
Ntfs (File system) Running (Disabled)
  Error Severity: Normal
  Service Flags: File System Driver, Shared Process
Null (Base) Running (System)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Oliscsi (SCSI miniport) Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Parallel (Extended base) Stopped (Automatic)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
  Service Dependencies:
    Parport
  Group Dependencies:
    Parallel arbitrator
Parport (Parallel arbitrator) Stopped (Automatic)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
ParVdm (Extended base) Stopped (Automatic)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
  Service Dependencies:
    Parport
  Group Dependencies:
    Parallel arbitrator
PCIDump (PCI Configuration) Stopped (System)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
Pcmcia (System Bus Extender) Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
PnP ISA Enabler Driver (Base) Stopped (System)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
PSE36 (Primary disk) Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
psidisp (Video) Stopped (Disabled)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
Q110wnt (SCSI miniport) Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
q12100 (SCSI Miniport) Running (Boot)
  C:\WINNT\System32\DRIVERS\q12100.sys
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
qlfilter Running (Automatic)
  System32\DRIVERS\qlfilter.sys
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
qv (Video) Stopped (Disabled)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
Rdr (Network) Running (Manual)

```

```

C:\WINNT\System32\drivers\rdr.sys
  Error Severity: Normal
  Service Flags: File System Driver, Shared Process
s3 (Video) Stopped (Disabled)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
Scsiprnt (Extended base) Stopped (Automatic)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
  Group Dependencies:
    SCSI miniport
Scsiscan (SCSI Class) Running (System)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
  Group Dependencies:
    SCSI miniport
Serial (Extended base) Running (Automatic)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
Sermouse (Pointer Port) Stopped (Disabled)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
Sfloppy (Primary disk) Stopped (System)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
  Group Dependencies:
    SCSI miniport
Simbad (Filter) Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
silcd32 (SCSI miniport) Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Sparrow (SCSI miniport) Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Spock (SCSI miniport) Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Srv (Network) Running (Manual)
  C:\WINNT\System32\drivers\srv.sys
  Error Severity: Normal
  Service Flags: File System Driver, Shared Process
symc810 (SCSI miniport) Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
symc8xx (SCSI miniport) Running (Boot)
  C:\WINNT\system32\drivers\symc8xx.sys
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
T128 (SCSI miniport) Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
T13B (SCSI miniport) Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
TCP/IP Service (PNP TDI) Running (Automatic)
  C:\WINNT\System32\drivers\tcpip.sys
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
tga (Video) Stopped (Disabled)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
tmvl (SCSI miniport) Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Ultral24 (SCSI miniport) Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Ultral4f (SCSI miniport) Stopped (Disabled)
  Error Severity: Normal

```

```

Service Flags: Kernel Driver, Shared Process
Ultra24f (SCSI miniport)          Stopped (Disabled)
  Error Severity: Normal
Service Flags: Kernel Driver, Shared Process
update (Base)                    Stopped (System)
  Error Severity: Ignore
Service Flags: Kernel Driver, Shared Process
v7vram (Video)                   Stopped (Disabled)
  Error Severity: Ignore
Service Flags: Kernel Driver, Shared Process
VgaSave (Video Save)             Running (System)
  C:\WINNT\System32\drivers\vga.sys
  Error Severity: Ignore
Service Flags: Kernel Driver, Shared Process
VgaStart (Video Init)           Stopped (System)
  C:\WINNT\System32\drivers\vga.sys
  Error Severity: Ignore
Service Flags: Kernel Driver, Shared Process
Wd33c93 (SCSI miniport)         Stopped (Disabled)
  Error Severity: Normal
Service Flags: Kernel Driver, Shared Process
wd90c24a (Video)                Stopped (Disabled)
  Error Severity: Ignore
Service Flags: Kernel Driver, Shared Process
wdvga (Video)                   Stopped (Disabled)
  Error Severity: Ignore
Service Flags: Kernel Driver, Shared Process
weitek9 (Video)                 Stopped (Disabled)
  Error Severity: Ignore
Service Flags: Kernel Driver, Shared Process
Xga (Video)                     Stopped (Disabled)
  Error Severity: Ignore
Service Flags: Kernel Driver, Shared Process

```

IRQ and Port Report

Devices	Vector	Level	Affinity
MPS 1.4 - APIC platform	8	8	0x000000ff
MPS 1.4 - APIC platform	0	0	0x000000ff
MPS 1.4 - APIC platform	1	1	0x000000ff
MPS 1.4 - APIC platform	2	2	0x000000ff
MPS 1.4 - APIC platform	3	3	0x000000ff
MPS 1.4 - APIC platform	4	4	0x000000ff
MPS 1.4 - APIC platform	5	5	0x000000ff
MPS 1.4 - APIC platform	6	6	0x000000ff
MPS 1.4 - APIC platform	7	7	0x000000ff
MPS 1.4 - APIC platform	8	8	0x000000ff
MPS 1.4 - APIC platform	9	9	0x000000ff
MPS 1.4 - APIC platform	10	10	0x000000ff
MPS 1.4 - APIC platform	11	11	0x000000ff
MPS 1.4 - APIC platform	12	12	0x000000ff
MPS 1.4 - APIC platform	13	13	0x000000ff
MPS 1.4 - APIC platform	14	14	0x000000ff
MPS 1.4 - APIC platform	15	15	0x000000ff
MPS 1.4 - APIC platform	16	16	0x000000ff
MPS 1.4 - APIC platform	17	17	0x000000ff
MPS 1.4 - APIC platform	18	18	0x000000ff
MPS 1.4 - APIC platform	19	19	0x000000ff
MPS 1.4 - APIC platform	20	20	0x000000ff
MPS 1.4 - APIC platform	21	21	0x000000ff
MPS 1.4 - APIC platform	22	22	0x000000ff
MPS 1.4 - APIC platform	23	23	0x000000ff
MPS 1.4 - APIC platform	24	24	0x000000ff
MPS 1.4 - APIC platform	25	25	0x000000ff
MPS 1.4 - APIC platform	26	26	0x000000ff
MPS 1.4 - APIC platform	27	27	0x000000ff
MPS 1.4 - APIC platform	28	28	0x000000ff
MPS 1.4 - APIC platform	29	29	0x000000ff

```

MPS 1.4 - APIC platform          30 30 0x000000ff
MPS 1.4 - APIC platform          31 31 0x000000ff
MPS 1.4 - APIC platform          32 32 0x000000ff
MPS 1.4 - APIC platform          33 33 0x000000ff
MPS 1.4 - APIC platform          34 34 0x000000ff
MPS 1.4 - APIC platform          35 35 0x000000ff
MPS 1.4 - APIC platform          36 36 0x000000ff
MPS 1.4 - APIC platform          37 37 0x000000ff
MPS 1.4 - APIC platform          38 38 0x000000ff
MPS 1.4 - APIC platform          39 39 0x000000ff
MPS 1.4 - APIC platform          40 40 0x000000ff
MPS 1.4 - APIC platform          41 41 0x000000ff
MPS 1.4 - APIC platform          42 42 0x000000ff
MPS 1.4 - APIC platform          43 43 0x000000ff
MPS 1.4 - APIC platform          44 44 0x000000ff
MPS 1.4 - APIC platform          45 45 0x000000ff
MPS 1.4 - APIC platform          46 46 0x000000ff
MPS 1.4 - APIC platform          47 47 0x000000ff
MPS 1.4 - APIC platform          61 61 0x000000ff
MPS 1.4 - APIC platform          65 65 0x000000ff
MPS 1.4 - APIC platform          80 80 0x000000ff
MPS 1.4 - APIC platform          193 193 0x000000ff
MPS 1.4 - APIC platform          225 225 0x000000ff
MPS 1.4 - APIC platform          253 253 0x000000ff
MPS 1.4 - APIC platform          254 254 0x000000ff
MPS 1.4 - APIC platform          255 255 0x000000ff
i8042prt                          1 1 0xffffffff
i8042prt                          12 12 0xffffffff
Serial                             4 4 0x00000000
Serial                             3 3 0x00000000
altndis                           16 16 0x00000000
El90x                              44 44 0x00000000
Floppy                             6 6 0x00000000
ql2100                             8 8 0x00000000
ql2100                             12 12 0x00000000
ql2100                             8 8 0x00000000
ql2100                             12 12 0x00000000
symc8xx                            48 48 0x00000000
symc8xx                             4 4 0x00000000

```

Devices	Physical Address	Length
MPS 1.4 - APIC platform	0x00000000	0x0000000010
MPS 1.4 - APIC platform	0x00000020	0x0000000002
MPS 1.4 - APIC platform	0x00000040	0x0000000004
MPS 1.4 - APIC platform	0x00000048	0x0000000004
MPS 1.4 - APIC platform	0x00000061	0x0000000001
MPS 1.4 - APIC platform	0x00000070	0x0000000002
MPS 1.4 - APIC platform	0x00000080	0x0000000010
MPS 1.4 - APIC platform	0x00000092	0x0000000001
MPS 1.4 - APIC platform	0x000000a0	0x0000000002
MPS 1.4 - APIC platform	0x000000c0	0x0000000010
MPS 1.4 - APIC platform	0x000000f0	0x0000000010
i8042prt	0x00000060	0x0000000001
i8042prt	0x00000064	0x0000000001
Serial	0x000003f8	0x0000000007
Serial	0x000002f8	0x0000000007
El90x	0x00004800	0x0000000080
Floppy	0x000003f0	0x0000000006
Floppy	0x000003f7	0x0000000001
ql2100	0x00004000	0x0000000110
ql2100	0x00005400	0x0000000110
ql2100	0x00006000	0x0000000110
ql2100	0x00006400	0x0000000110
symc8xx	0x00004400	0x0000000110
symc8xx	0x00005000	0x0000000110
VgaSave	0x000003b0	0x000000000c
VgaSave	0x000003c0	0x0000000020
VgaSave	0x000001ce	0x0000000002

DMA and Memory Report

Devices	Channel	Port
Floppy	2	0

Devices	Physical Address	Length
MPS 1.4 - APIC platform	0xfec00000	0x00000400
MPS 1.4 - APIC platform	0xfec01000	0x00000400
MPS 1.4 - APIC platform	0xfec00000	0x00000400
altndis	0xfe300000	0x00004000
E190x	0xfe012000	0x00000080
ql2100	0xfe010000	0x00001000
ql2100	0xfe305000	0x00001000
ql2100	0xfe600000	0x00001000
ql2100	0xfe601000	0x00001000
symc8xx	0xfe012400	0x00000100
symc8xx	0xfe011000	0x00001000
symc8xx	0xfe306000	0x00000100
symc8xx	0xfe304000	0x00001000
c1546x	0xfe000000	0x0000a000
c1546x	0xfc000000	0x02000000
VgaSave	0x000a0000	0x00020000

Environment Report

System Environment Variables

```
ComSpec=C:\WINNT\system32\cmd.exe
HOME=C:/
NUMBER_OF_PROCESSORS=8
OS=Windows NT
Os2LibPath=C:\WINNT\system32\os2\dll;
Path=C:\Oracle\Ora81\bin;C:\Program
Files\Oracle\jre\1.1.7\bin;d:\mks\mksnt;C:\WINNT\system32;C:\WINNT;.;
PROCESSOR_ARCHITECTURE=x86
PROCESSOR_IDENTIFIER=x86 Family 6 Model 7 Stepping 2, GenuineIntel
PROCESSOR_LEVEL=6
PROCESSOR_REVISION=0702
ROOTDIR=d:/mks
SHELL=d:/mks/mksnt/sh.exe
TMPDIR=C:/TEMP
windir=C:\WINNT
```

Environment Variables for Current User

```
ORACLE_HOME=C:\oracle\ora81
ORACLE_SID=tpcc
SQLPLUS=sqlplus
SQLDBA=svrmgrl
TEMP=C:\TEMP
TMP=C:\TEMP
TPCCLOAD=tpccload.exe
```

Network Report

```
Your Access Level: Admin & Local
Workgroup or Domain: WORKGROUP
Network Version: 4.0
LanRoot: WORKGROUP
Logged On Users: 1
Current User (1): Administrator
```

```
Logon Domain: NODE1
Logon Server: NODE1
```

```
Transport: NetBT_altndis2, 00-60-CF-20-14-C0, VC's: 0, Wan: Wan
Transport: NetBT_E190x1, 00-00-4C-E1-98-70, VC's: 1, Wan: Wan
```

```
Character Wait: 3,600
Collection Time: 250
Maximum Collection Count: 16
Keep Connection: 600
Maximum Commands: 5
Session Time Out: 45
Character Buffer Size: 512
Maximum Threads: 17
Lock Quota: 6,144
Lock Increment: 10
Maximum Locks: 500
Pipe Increment: 10
Maximum Pipes: 500
Cache Time Out: 40
Dormant File Limit: 45
Read Ahead Throughput: 4,294,967,295
Mailslot Buffers: 3
Server Announce Buffers: 20
Illegal Datagrams: 5
Datagram Reset Frequency: 60
Log Election Packets: False
Use Opportunistic Locking: True
Use Unlock Behind: True
Use Close Behind: True
Buffer Pipes: True
Use Lock, Read, Unlock: True
Use NT Caching: True
Use Raw Read: True
Use Raw Write: True
Use Write Raw Data: True
Use Encryption: True
Buffer Deny Write Files: True
Buffer Read Only Files: True
Force Core Creation: True
512 Byte Max Transfer: False
Bytes Received: 463,179
SMB's Received: 478
Paged Read Bytes Requested: 454,656
Non Paged Read Bytes Requested: 72,004
Cache Read Bytes Requested: 72,004
Network Read Bytes Requested: 390,628
Bytes Transmitted: 159,697
SMB's Transmitted: 478
Paged Read Bytes Requested: 0
Non Paged Read Bytes Requested: 161,476
Cache Read Bytes Requested: 0
Network Read Bytes Requested: 161,476
Initially Failed Operations: 0
Failed Completion Operations: 0
Read Operations: 24
Random Read Operations: 3
Read SMB's: 18
Large Read SMB's: 12
Small Read SMB's: 0
Write Operations: 11
Random Write Operations: 0
Write SMB's: 27
Large Write SMB's: 11
Small Write SMB's: 0
Raw Reads Denied: 0
Raw Writes Denied: 0
Network Errors: 0
Sessions: 31
Failed Sessions: 0
Reconnects: 0
```

```

Core Connects: 0
LM 2.0 Connects: 0
LM 2.x Connects: 0
Windows NT Connects: 17
Server Disconnects: 15
Hung Sessions: 0
Use Count: 6
Failed Use Count: 0
Current Commands: 0
Server File Opens: 115
Server Device Opens: 0
Server Jobs Queued: 0
Server Session Opens: 1
Server Sessions Timed Out: 0
Server Sessions Errored Out: 0
Server Password Errors: 0
Server Permission Errors: 0
Server System Errors: 0
Server Bytes Sent: 944,061
Server Bytes Received: 132,046
Server Average Response Time: 0
Server Request Buffers Needed: 0
Server Big Buffers Needed: 0

```

NT/Oracle Regsity Information

No Windows NT registry parameters were modified for this benchmark.

<Oracle>

```

[HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE]
"inst_loc"="C:\Program Files\Oracle\Inventory"

[HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\ALL_HOMES]
"HOME_COUNTER"="1"
"DEFAULT_HOME"="DEFAULT_HOME"
"LAST_HOME"="0"

[HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\ALL_HOMES\ID0]
"NAME"="OraHome81"
"PATH"="C:\Oracle\Ora81"
"NLS_LANG"="NA"

[HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\HOME0]
"ORACLE_GROUP_NAME"="Oracle - OraHome81"
"ORACLE_HOME_NAME"="OraHome81"
"ORACLE_HOME"="C:\Oracle\Ora81"
"NLS_LANG"="AMERICAN_AMERICA.WE8ISO8859P1"
"ORACLE_HOME_KEY"="Software\ORACLE\HOME0"
"MSHELP_TOOLS"="C:\Oracle\Ora81\MSHELP"
"SQLPATH"="C:\Oracle\Ora81\dbs"
"RDBMS_CONTROL"="C:\Oracle\Ora81\DATABASE"
"RDBMS_ARCHIVE"="C:\Oracle\Ora81\DATABASE\ARCHIVE"
"ORACLE_BASE"="C:\Oracle"
"ORA_tpcc_AUTOSTART"=hex(2):54,52,55,45,00
"ORA_tpcc_SHUTDOWN"=hex(2):54,52,55,45,00
"ORA_tpcc_SHUTDOWNNTYPE"=hex(2):69,00
"ORA_tpcc_SHUTDOWN_TIMEOUT"=hex(2):33,30,00
"ORA_TPCC_PWFIL"=hex(2):20,43,3a,5c,6f,72,61,63,6c,65,5c,6f,72,61,38,31,5c,64,\
61,74,61,62,61,73,65,5c,70,77,64,74,70,63,63,2e,6f,72,61,00
"ORA_MAX_ALLOC"="32768"

[HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\OSD]
"PMDDL"="C:\Oracle\Ora81\BIN\PM.DLL"
"IPCDLL"="c:\oracle\ref2\osdbin\IPC.DLL"

```

```

"CMDLL"="c:\oracle\ref2\osdbin\CM.DLL"
"STARTDLL"="c:\oracle\ref2\osdbin\START.DLL"
"IODLL"="c:\oracle\ref2\osdbin\IO.DLL"

[HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\OSD\CM]
"CMDiskFile"=""
"ServiceName"=""
"ClusterName"="Oracle Cluster Manager, version 2.0"
"ErrorLog"="c:\oracle\ref2\osdbin\CMError.log"
"CMsrvrpath"=""
"DefinedNodes"=hex(7):6e,6f,64,65,31,00,6e,6f,64,65,32,00,6e,6f,64,65,33,00,6e,\
6f,64,65,34,00,00
"ConfigNodes"=hex(7):6e,6f,64,65,31,00,6e,6f,64,65,32,00,6e,6f,64,65,33,00,6e,\
6f,64,65,34,00,00
"CMHostname"="node1"
"HeartBeat"=dword:00015f90
"AutoJoin"=dword:00000000
"PollInterval"=dword:000003e8
"MissCount"=dword:00000020
"ClusterId"=dword:00000000
"CMServiceport"=dword:0001869e

[HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\OSD\IPC]
"IpchostName"="node1"

[HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\OSD\PM]

[HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\OSDBACKUP]
"IPCDLL"="c:\oracle\ref2\osdbin\tcp_ipc.dll"
"CMDLL"="c:\oracle\ref2\osdbin\CM.DLL"
"STARTDLL"="c:\oracle\ref2\osdbin\START.DLL"
"IODLL"="c:\oracle\ref2\osdbin\IO.DLL"
"PMDDL"="C:\Oracle\Ora81\BIN\PM.DLL"

[HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\OSDBACKUP\CM]
"CMDiskFile"=""
"ServiceName"=""
"ClusterName"="Oracle Cluster Manager, version 2.0"
"ErrorLog"="c:\oracle\ref2\osdbin\CMError.log"
"CMsrvrpath"=""
"DefinedNodes"=hex(7):6e,6f,64,65,31,00,6e,6f,64,65,32,00,6e,6f,64,65,33,00,6e,\
6f,64,65,34,00,00
"ConfigNodes"=hex(7):6e,6f,64,65,31,00,6e,6f,64,65,32,00,6e,6f,64,65,33,00,6e,\
6f,64,65,34,00,00
"CMHostname"="node1"
"HeartBeat"=dword:00015f90
"AutoJoin"=dword:00000000
"PollInterval"=dword:000003e8
"MissCount"=dword:00000020
"ClusterId"=dword:00000000
"CMServiceport"=dword:0001869e

[HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\OSDBACKUP\IPC]
"IpchostName"="node1"

[HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\OSDBACKUP\PM]

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\EventLog\Application\Oracle.tpcc]
"EventMessageFile"=hex(2):43,3a,5c,4f,72,61,63,6c,65,5c,4f,72,61,38,31,5c,42,\
49,4e,5c,4f,52,41,45,56,52,55,53,38,2e,44,4c,4c,00
"TypesSupported"=dword:00000007

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\EventLog\Application\OracleAgent]
"EventMessageFile"="C:\Oracle\Ora81\AGENTBIN\snmmsg.dll"
"TypesSupported"="7"

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\OracleCMService]
"Type"=dword:00000010
"Start"=dword:00000003
"ErrorControl"=dword:00000001
"ImagePath"=hex(2):63,3a,5c,6f,72,61,63,6c,65,5c,72,65,66,32,5c,6f,73,64,62,69,\

```

```
6e,5c,63,6d,73,72,76,72,2e,65,78,65,00
"DisplayName"="OracleCMService"
"Group"="Cluster Manager"
"DependOnService"=hex(7):4f,72,61,63,6c,65,4e,4d,53,65,72,76,69,63,65,00,00
"DependOnGroup"=hex(7):00
"ObjectName"="LocalSystem"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\OracleCMService\Security]
"Security"=hex:01,00,14,80,c0,00,00,00,cc,00,00,00,14,00,00,00,34,00,00,00,02,\
00,20,00,01,00,00,00,02,80,18,00,ff,01,0f,00,01,01,00,00,00,00,01,00,00,\
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,00,18,00,8d,01,02,00,01,01,00,\
00,00,00,00,01,00,00,00,00,74,00,73,00,00,00,1c,00,fd,01,02,00,01,02,00,00,\
00,00,00,05,20,00,00,00,23,02,00,00,00,76,00,63,00,00,00,1c,00,ff,01,0f,00,01,\
02,00,00,00,00,05,20,00,00,00,20,02,00,00,76,00,63,00,00,00,1c,00,ff,01,\
0f,00,01,02,00,00,00,00,05,20,00,00,00,25,02,00,00,76,00,63,00,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,05,12,00,00,00,25,02,00,00,01,01,00,00,\
00,00,00,05,12,00,00,00,01,01,00,00,00,00,05,12,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\OracleCMService\Enum]
"0"="Root\LEGACY_ORACLECMSERVICE\0000"
"Count"=dword:00000001
"NextInstance"=dword:00000001
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\OracleNMService]
"Type"=dword:00000010
"Start"=dword:00000003
"ErrorControl"=dword:00000001
"ImagePath"=hex(2):63,3a,5c,6f,72,61,63,6c,65,5c,72,65,66,32,5c,6f,73,64,62,69,\
6e,5c,6e,6d,73,72,76,72,2e,65,78,65,00
"DisplayName"="OracleNMService"
"ObjectName"="LocalSystem"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\OracleNMService\Security]
"Security"=hex:01,00,14,80,c0,00,00,00,cc,00,00,00,14,00,00,00,34,00,00,00,02,\
00,20,00,01,00,00,00,02,80,18,00,ff,01,0f,00,01,01,00,00,00,00,01,00,00,\
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,00,18,00,8d,01,02,00,01,01,00,\
00,00,00,00,01,00,00,00,00,00,00,00,00,00,1c,00,fd,01,02,00,01,02,00,00,\
00,00,00,05,20,00,00,00,23,02,00,00,00,00,00,00,00,00,1c,00,ff,01,0f,00,01,\
02,00,00,00,00,05,20,00,00,00,20,02,00,00,00,00,00,00,00,00,1c,00,ff,01,\
0f,00,01,02,00,00,00,00,05,20,00,00,00,25,02,00,00,00,00,00,00,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,05,12,00,00,00,25,02,00,00,01,01,00,00,\
00,00,00,05,12,00,00,00,01,01,00,00,00,00,05,12,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\OracleNMService\Enum]
"0"="Root\LEGACY_ORACLENMSERVICE\0000"
"Count"=dword:00000001
"NextInstance"=dword:00000001
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\OracleOraHome81Agent]
"Type"=dword:00000010
"Start"=dword:00000003
"ErrorControl"=dword:00000001
"ImagePath"=hex(2):43,3a,5c,6f,72,61,63,6c,65,5c,4f,72,61,63,31,5c,62,69,6e,5c,\
64,62,73,6e,6d,70,2e,65,78,65,00
"DisplayName"="OracleOraHome81Agent"
"ObjectName"="LocalSystem"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\OracleOraHome81Agent\Security]
"Security"=hex:01,00,14,80,c0,00,00,00,cc,00,00,00,14,00,00,00,34,00,00,00,02,\
00,20,00,01,00,00,00,02,80,18,00,ff,01,0f,00,01,01,00,00,00,00,01,00,00,\
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,00,18,00,8d,01,02,00,01,01,00,\
00,00,00,00,01,00,00,00,00,20,02,00,00,00,00,1c,00,fd,01,02,00,01,02,00,00,\
00,00,00,05,20,00,00,00,23,02,00,00,00,00,00,00,00,00,1c,00,ff,01,0f,00,01,\
02,00,00,00,00,05,20,00,00,00,20,02,00,00,00,00,00,00,00,00,1c,00,ff,01,\
0f,00,01,02,00,00,00,00,05,20,00,00,00,25,02,00,00,00,00,00,00,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,05,12,00,00,00,25,02,00,00,01,01,00,00,\
00,00,00,05,12,00,00,00,01,01,00,00,00,00,05,12,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\OracleOraHome81Agent\Enum]
"0"="Root\LEGACY_ORACLEORAHOME81AGENT\0000"
"Count"=dword:00000001
```

```
"NextInstance"=dword:00000001
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\OracleOraHome81ClientCache]
"Type"=dword:00000010
"Start"=dword:00000003
"ErrorControl"=dword:00000001
"ImagePath"=hex(2):43,3a,5c,6f,72,61,63,6c,65,5c,4f,72,61,63,31,5c,42,49,4e,5c,\
4f,4e,52,53,44,2e,45,58,45,00
"DisplayName"="OracleOraHome81ClientCache"
"ObjectName"="LocalSystem"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\OracleOraHome81ClientCache\Security]
"Security"=hex:01,00,14,80,c0,00,00,00,cc,00,00,00,14,00,00,00,34,00,00,00,02,\
00,20,00,01,00,00,00,02,80,18,00,ff,01,0f,00,01,01,00,00,00,00,01,00,00,\
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,00,18,00,8d,01,02,00,01,01,00,\
00,00,00,00,01,00,00,00,00,20,02,00,00,00,00,1c,00,fd,01,02,00,01,02,00,00,\
00,00,00,05,20,00,00,00,23,02,00,00,00,00,00,00,00,00,1c,00,ff,01,0f,00,01,\
02,00,00,00,00,05,20,00,00,00,20,02,00,00,00,00,00,00,00,00,1c,00,ff,01,\
0f,00,01,02,00,00,00,00,05,20,00,00,00,25,02,00,00,00,00,00,00,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,05,12,00,00,00,25,02,00,00,01,01,00,00,\
00,00,00,05,12,00,00,00,01,01,00,00,00,00,05,12,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\OracleOraHome81CAdmin]
"Type"=dword:00000010
"Start"=dword:00000003
"ErrorControl"=dword:00000001
"ImagePath"=hex(2):43,3a,5c,6f,72,61,63,6c,65,5c,4f,72,61,63,31,5c,42,49,4e,5c,\
43,4d,41,44,4d,49,4e,2e,45,58,45,00
"DisplayName"="OracleOraHome81CAdmin"
"ObjectName"="LocalSystem"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\OracleOraHome81CAdmin\Security]
"Security"=hex:01,00,14,80,c0,00,00,00,cc,00,00,00,14,00,00,00,34,00,00,00,02,\
00,20,00,01,00,00,00,02,80,18,00,ff,01,0f,00,01,01,00,00,00,00,01,00,00,\
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,00,18,00,8d,01,02,00,01,01,00,\
00,00,00,00,01,00,00,00,00,20,02,00,00,00,00,1c,00,fd,01,02,00,01,02,00,00,\
00,00,00,05,20,00,00,00,23,02,00,00,00,00,00,00,00,00,1c,00,ff,01,0f,00,01,\
02,00,00,00,00,05,20,00,00,00,20,02,00,00,00,00,00,00,00,00,1c,00,ff,01,\
0f,00,01,02,00,00,00,00,05,20,00,00,00,25,02,00,00,00,00,00,00,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,05,12,00,00,00,25,02,00,00,01,01,00,00,\
00,00,00,05,12,00,00,00,01,01,00,00,00,00,05,12,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\OracleOraHome81CAdmin\Enum]
"0"="Root\LEGACY_ORACLEORAHOME81CADMIN\0000"
"Count"=dword:00000001
"NextInstance"=dword:00000001
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\OracleOraHome81CMan]
"Type"=dword:00000010
"Start"=dword:00000003
"ErrorControl"=dword:00000001
"ImagePath"=hex(2):43,3a,5c,6f,72,61,63,6c,65,5c,4f,72,61,63,31,5c,42,49,4e,5c,\
43,4d,47,57,2e,45,58,45,00
"DisplayName"="OracleOraHome81CMan"
"ObjectName"="LocalSystem"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\OracleOraHome81CMan\Security]
"Security"=hex:01,00,14,80,c0,00,00,00,cc,00,00,00,14,00,00,00,34,00,00,00,02,\
00,20,00,01,00,00,00,02,80,18,00,ff,01,0f,00,01,01,00,00,00,00,01,00,00,\
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,00,18,00,8d,01,02,00,01,01,00,\
00,00,00,00,01,00,00,00,00,20,02,00,00,00,00,1c,00,fd,01,02,00,01,02,00,00,\
00,00,00,05,20,00,00,00,23,02,00,00,00,00,00,00,00,00,1c,00,ff,01,0f,00,01,\
02,00,00,00,00,05,20,00,00,00,20,02,00,00,00,00,00,00,00,00,1c,00,ff,01,\
0f,00,01,02,00,00,00,00,05,20,00,00,00,25,02,00,00,00,00,00,00,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,05,12,00,00,00,25,02,00,00,01,01,00,00,\
00,00,00,05,12,00,00,00,01,01,00,00,00,00,05,12,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\OracleOraHome81DataGatherer]
"Type"=dword:00000010
"Start"=dword:00000003
"ErrorControl"=dword:00000001
```

```

"ImagePath"=hex(2):43,3a,5c,4f,72,61,63,6c,65,5c,4f,72,61,38,31,5c,62,69,6e,5c,\
76,70,70,64,63,2e,65,78,65,00
"DisplayName"="OracleOraHome81DataGatherer"
"ObjectName"="LocalSystem"

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\OracleOraHome81DataGatherer\Security]
"Security"=hex:01,00,14,80,c0,00,00,00,cc,00,00,00,14,00,00,00,34,00,00,02,\
00,20,00,01,00,00,00,02,80,18,00,ff,01,0f,00,01,01,00,00,00,00,01,00,00,\
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,00,18,00,8d,01,02,00,01,01,00,\
00,00,00,01,00,00,00,00,6e,00,61,00,00,00,1c,00,fd,01,02,00,01,02,00,00,\
00,00,00,05,20,00,00,00,23,02,00,00,67,00,65,00,00,00,1c,00,ff,01,0f,00,01,\
02,00,00,00,00,05,20,00,00,00,20,02,00,00,67,00,65,00,00,00,1c,00,ff,01,\
0f,00,01,02,00,00,00,00,05,20,00,00,00,25,02,00,00,67,00,65,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,05,12,00,00,00,25,02,00,00,01,01,00,00,\
00,00,00,05,12,00,00,00,01,01,00,00,00,00,05,12,00,00,00

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\OracleOraHome81TNSListener]
"Type"=dword:00000010
"Start"=dword:00000002
"ErrorControl"=dword:00000001
"ImagePath"=hex(2):43,3a,5c,4f,72,61,63,6c,65,5c,4f,72,61,38,31,5c,42,49,4e,5c,\
54,4e,53,4c,53,4e,52,20,00
"DisplayName"="OracleOraHome81TNSListener"
"ObjectName"="LocalSystem"

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\OracleOraHome81TNSListener\Security]
"Security"=hex:01,00,14,80,c0,00,00,00,cc,00,00,00,14,00,00,00,34,00,00,02,\
00,20,00,01,00,00,00,02,80,18,00,ff,01,0f,00,01,01,00,00,00,00,01,00,00,\
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,00,18,00,8d,01,02,00,01,01,00,\
00,00,00,01,00,00,00,00,8b,d5,11,7c,00,00,1c,00,fd,01,02,00,01,02,00,00,\
00,00,00,05,20,00,00,00,23,02,00,00,31,f3,d2,11,00,00,1c,00,ff,01,0f,00,01,\
02,00,00,00,00,00,05,20,00,00,00,20,02,00,00,31,f3,d2,11,00,00,1c,00,ff,01,\
0f,00,01,02,00,00,00,00,05,20,00,00,00,25,02,00,00,31,f3,d2,11,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,05,12,00,00,00,25,02,00,00,01,01,00,00,\
00,00,00,05,12,00,00,00,01,01,00,00,00,00,05,12,00,00,00

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\OracleOraHome81TNSListener\Enum]
"0"="Root\LEGACY_ORACLEORAHOME81TNSLISTENER\0000"
"Count"=dword:00000001
"NextInstance"=dword:00000001

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\OracleServicepcc]
"Type"=dword:00000110
"Start"=dword:00000003
"ErrorControl"=dword:00000001
"ImagePath"=hex(2):63,3a,5c,6f,72,61,63,6c,65,5c,6f,72,61,38,31,5c,62,69,6e,5c,\
4f,52,41,43,4c,45,2e,45,58,45,20,74,70,63,63,00
"DisplayName"="OracleServicepcc"
"ObjectName"="LocalSystem"

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\OracleServicepcc\Security]
"Security"=hex:01,00,14,80,c0,00,00,00,cc,00,00,00,14,00,00,00,34,00,00,02,\
00,20,00,01,00,00,00,02,80,18,00,ff,01,0f,00,01,01,00,00,00,00,01,00,00,\
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,00,18,00,8d,01,02,00,01,01,00,\
00,00,00,01,00,00,00,00,20,02,00,00,00,00,1c,00,fd,01,02,00,01,02,00,00,\
00,00,00,05,20,00,00,00,23,02,00,00,00,00,00,00,00,00,00,1c,00,ff,01,0f,00,01,\
02,00,00,00,00,05,20,00,00,00,20,02,00,00,00,00,00,00,00,00,1c,00,ff,01,\
0f,00,01,02,00,00,00,00,05,20,00,00,00,25,02,00,00,00,00,00,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,05,12,00,00,00,25,02,00,00,01,01,00,00,\
00,00,00,05,12,00,00,00,01,01,00,00,00,00,05,12,00,00,00

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\OracleServicepcc\Enum]
"0"="Root\LEGACY_ORACLESERVICEPCC\0000"
"Count"=dword:00000001
"NextInstance"=dword:00000001

```

Client Configuration (common to all clients)

NT Client configuration information

Microsoft Diagnostics Report For \\CL01

OS Version Report

```

Microsoft (R) Windows NT (TM) Server
Version 4.0 (Build 1381: Service Pack 4) x86 Multiprocessor Free
Registered Owner: nec, nec
Product Number: 50382-415-0024195-92625

```

System Report

```

System: AT/AT COMPATIBLE
Hardware Abstraction Layer: MPS 1.4 - APIC platform
BIOS Date: 03/06/99
BIOS Version: PhoenixBIOS 4.0 Release 6.0.0327

```

Processor list:

```

0: x86 Family 6 Model 7 Stepping 3 GenuineIntel ~498 Mhz
1: x86 Family 6 Model 7 Stepping 3 GenuineIntel ~498 Mhz

```

Video Display Report

```

BIOS Date: 06/25/98
BIOS Version: CL-GD546x Laguna PCI VGA BIOS Version 1.71e c15

```

Adapter:

```

Setting: 1024 x 768 x 65536
70 Hz
Type: cl546xm compatible display adapter
String: Cirrus Logic VisualMedia(TM) Accelerator
Memory: 2 MB
Chip Type: Cirrus Logic 5465
DAC Type: Internal
Driver:
Vendor: Cirrus Logic, Inc.
File(s): cl546xm.sys, cl5465.dll
Version: 4.00.1381.1705-1.705, 4.0.101

```

Drives Report

```

C:\ (Local - NTFS) Total: 4,192,933 KB, Free: 3,574,689 KB
Serial Number: 3015 - 6BFE
Bytes per cluster: 512
Sectors per cluster: 1
Filename length: 255
D:\ (Local - NTFS) Total: 4,690,976 KB, Free: 3,093,224 KB
Serial Number: A80B - 6C76
Bytes per cluster: 512
Sectors per cluster: 8
Filename length: 255

```

Memory Report

```

Handles: 1,084
Threads: 96
Processes: 15

```

Physical Memory (K)

Total: 1,047,984
Available: 919,744
File Cache: 10,780

Kernel Memory (K)
Total: 17,708
Paged: 6,924
Nonpaged: 10,784

Commit Charge (K)
Total: 35,008
Limit: 2,518,240
Peak: 617,736

Pagefile Space (K)
Total: 1,584,128
Total in use: 17,060
Peak: 288,824

D:\pagefile.sys
Total: 1,584,128
Total in use: 17,060
Peak: 288,824

Services Report

```
-----  
Alerter                               Stopped (Manual)  
  C:\WINNT\System32\services.exe  
  Service Account Name: LocalSystem  
  Error Severity: Normal  
  Service Flags: Shared Process  
  Service Dependencies:  
    LanmanWorkstation  
Computer Browser                       Stopped (Manual)  
  C:\WINNT\System32\services.exe  
  Service Account Name: LocalSystem  
  Error Severity: Normal  
  Service Flags: Shared Process  
  Service Dependencies:  
    LanmanWorkstation  
    LanmanServer  
    LmHosts  
ClipBook Server                         Stopped (Manual)  
  C:\WINNT\system32\clipsrv.exe  
  Service Account Name: LocalSystem  
  Error Severity: Normal  
  Service Flags: Own Process  
  Service Dependencies:  
    NetDDE  
DHCP Client (TDI)                       Stopped (Disabled)  
  C:\WINNT\System32\services.exe  
  Service Account Name: LocalSystem  
  Error Severity: Normal  
  Service Flags: Shared Process  
  Service Dependencies:  
    Tcpip  
    Afd  
    NetBT  
EventLog (Event log)                   Running (Automatic)  
  C:\WINNT\system32\services.exe  
  Service Account Name: LocalSystem  
  Error Severity: Normal  
  Service Flags: Shared Process  
Server                                   Running (Automatic)  
  C:\WINNT\System32\services.exe  
  Service Account Name: LocalSystem  
  Error Severity: Normal  
  Service Flags: Shared Process  
  Group Dependencies:
```

```
TDI  
Workstation (NetworkProvider)           Running (Automatic)  
  C:\WINNT\System32\services.exe  
  Service Account Name: LocalSystem  
  Error Severity: Normal  
  Service Flags: Shared Process  
  Group Dependencies:  
TDI  
License Logging Service                 Stopped (Manual)  
  C:\WINNT\System32\llssrv.exe  
  Service Account Name: LocalSystem  
  Error Severity: Normal  
  Service Flags: Own Process  
TCP/IP NetBIOS Helper                   Running (Automatic)  
  C:\WINNT\System32\services.exe  
  Service Account Name: LocalSystem  
  Error Severity: Normal  
  Service Flags: Shared Process  
  Group Dependencies:  
    NetworkProvider  
Messenger                               Stopped (Manual)  
  C:\WINNT\System32\services.exe  
  Service Account Name: LocalSystem  
  Error Severity: Normal  
  Service Flags: Shared Process  
  Service Dependencies:  
    LanmanWorkstation  
    NetBios  
Network DDE (NetDDEGroup)               Stopped (Manual)  
  C:\WINNT\system32\netdde.exe  
  Service Account Name: LocalSystem  
  Error Severity: Normal  
  Service Flags: Shared Process  
  Service Dependencies:  
    NetDDEDSDM  
Network DDE DSDM                       Stopped (Manual)  
  C:\WINNT\system32\netdde.exe  
  Service Account Name: LocalSystem  
  Error Severity: Normal  
  Service Flags: Shared Process  
Net Logon (RemoteValidation)            Stopped (Manual)  
  C:\WINNT\System32\lsass.exe  
  Service Account Name: LocalSystem  
  Error Severity: Normal  
  Service Flags: Shared Process  
  Service Dependencies:  
    LanmanWorkstation  
    LmHosts  
NT LM Security Support Provider          Running (Manual)  
  C:\WINNT\System32\SERVICES.EXE  
  Service Account Name: LocalSystem  
  Error Severity: Normal  
  Service Flags: Shared Process  
OracleOraHome81ClientCache              Stopped (Manual)  
  C:\Oracle\Ora81\BIN\ONRSD.EXE  
  Service Account Name: LocalSystem  
  Error Severity: Normal  
  Service Flags: Own Process  
Plug and Play (PlugPlay)                Running (Automatic)  
  C:\WINNT\system32\services.exe  
  Service Account Name: LocalSystem  
  Error Severity: Normal  
  Service Flags: Shared Process  
Protected Storage                       Stopped (Manual)  
  c:\winnt\system32\pstores.exe  
  Service Account Name: LocalSystem  
  Error Severity: Normal  
  Service Flags: Own Process, Interactive  
  Service Dependencies:  
    RpcSs  
Directory Replicator                     Stopped (Manual)
```

```

C:\WINNT\System32\lmrepl.exe
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Own Process
Service Dependencies:
  LanmanWorkstation
  LanmanServer
Remote Procedure Call (RPC) Locator      Stopped (Manual)
C:\WINNT\System32\LOCATOR.EXE
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Own Process
Service Dependencies:
  LanmanWorkstation
  Rdr
Remote Procedure Call (RPC) Service      Running (Automatic)
C:\WINNT\system32\RpcSs.exe
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Own Process
Schedule                                 Stopped (Manual)
C:\WINNT\System32\AtSvc.Exe
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Own Process
SNMP                                       Stopped (Manual)
C:\WINNT\System32\snmp.exe
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Own Process
Service Dependencies:
  Tcpip
  EventLog
SNMP Trap Service                         Stopped (Manual)
C:\WINNT\System32\snmptrap.exe
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Own Process
Service Dependencies:
  Tcpip
  EventLog
Spooler (SpoolerGroup)                   Stopped (Manual)
C:\WINNT\system32\spoolss.exe
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Own Process, Interactive
Telephony Service                         Stopped (Manual)
C:\WINNT\system32\tapisrv.exe
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Own Process
TUXEDO IPC Helper                         Running (Automatic)
C:\TUXEDO\bin\tuxipc.exe
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Own Process
Tlisten (Port: 3050)                      Stopped (Manual)
C:\TUXEDO\bin\tlisten.exe
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Own Process
UPS                                       Stopped (Manual)
C:\WINNT\System32\ups.exe
Service Account Name: LocalSystem
Error Severity: Normal
Service Flags: Own Process
World Wide Web Publishing Service         Stopped (Automatic)
C:\WINNT\System32\inetsrv\inetinfo.exe
Service Account Name: LocalSystem
Error Severity: Ignore
Service Flags: Shared Process

```

```

Service Dependencies:
  RPCSS
  NTLMSPP

```

Drivers Report

```

-----
Abiosdsk (Primary disk)                   Stopped (Disabled)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
AFD Networking Support Environment (TDI)    Running (Automatic)
C:\WINNT\System32\drivers\afd.sys
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Ahal54x (SCSI miniport)                   Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Ahal74x (SCSI miniport)                   Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
aic78xx (SCSI miniport)                   Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Always (SCSI miniport)                    Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
amiOnt (SCSI miniport)                     Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
amsint (SCSI miniport)                     Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Arrow (SCSI miniport)                      Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
atapi (SCSI miniport)                      Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Atdisk (Primary disk)                      Stopped (Disabled)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
ati (Video)                                Stopped (Disabled)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
Beep (Base)                                 Running (System)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
BusLogic (SCSI miniport)                   Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Busmouse (Pointer Port)                    Stopped (Disabled)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
Cdaudio (Filter)                           Stopped (System)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
Cdfs (File system)                          Running (Disabled)
  Error Severity: Normal
  Service Flags: File System Driver, Shared Process
  Group Dependencies:
    SCSI CDROM Class
Cdrom (SCSI CDROM Class)                   Running (System)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
  Group Dependencies:
    SCSI miniport
Changer (Filter)                            Stopped (System)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
cirrus (Video)                              Stopped (Disabled)

```

```

Error Severity: Ignore
Service Flags: Kernel Driver, Shared Process
c1546x (Video) Running (System)
System32\DRIVERS\c1546x.sys
Error Severity: Ignore
Service Flags: Kernel Driver, Shared Process
Cpqarray (SCSI miniport) Stopped (Disabled)
Error Severity: Normal
Service Flags: Kernel Driver, Shared Process
cpqfw2e (SCSI miniport) Stopped (Disabled)
Error Severity: Normal
Service Flags: Kernel Driver, Shared Process
dac960nt (SCSI miniport) Stopped (Disabled)
Error Severity: Normal
Service Flags: Kernel Driver, Shared Process
dce376nt (SCSI miniport) Stopped (Disabled)
Error Severity: Normal
Service Flags: Kernel Driver, Shared Process
Delldsa (SCSI miniport) Stopped (Disabled)
Error Severity: Normal
Service Flags: Kernel Driver, Shared Process
Dell_DGX (Video) Stopped (Disabled)
Error Severity: Ignore
Service Flags: Kernel Driver, Shared Process
Disk (SCSI Class) Running (Boot)
Error Severity: Ignore
Service Flags: Kernel Driver, Shared Process
Group Dependencies:
SCSI miniport
Diskperf (Filter) Stopped (Disabled)
Error Severity: Normal
Service Flags: Kernel Driver, Shared Process
DptScsi (SCSI miniport) Stopped (Disabled)
Error Severity: Normal
Service Flags: Kernel Driver, Shared Process
dtt329x (SCSI miniport) Stopped (Disabled)
Error Severity: Normal
Service Flags: Kernel Driver, Shared Process
3Com 3C90x Adapter Driver (NDIS) Running (Automatic)
C:\WINNT\System32\drivers\el90xnd4.sys
Error Severity: Normal
Service Flags: Kernel Driver, Shared Process
Adaptec EMPCI Adapter Driver (NDIS) Running (Automatic)
C:\WINNT\System32\drivers\EMPCI.sys
Error Severity: Normal
Service Flags: Kernel Driver, Shared Process
et4000 (Video) Stopped (Disabled)
Error Severity: Ignore
Service Flags: Kernel Driver, Shared Process
Fastfat (Boot file system) Stopped (Disabled)
Error Severity: Normal
Service Flags: File System Driver, Shared Process
Fd16_700 (SCSI miniport) Stopped (Disabled)
Error Severity: Normal
Service Flags: Kernel Driver, Shared Process
Fd7000ex (SCSI miniport) Stopped (Disabled)
Error Severity: Normal
Service Flags: Kernel Driver, Shared Process
Fd8xx (SCSI miniport) Stopped (Disabled)
Error Severity: Normal
Service Flags: Kernel Driver, Shared Process
flashpnt (SCSI miniport) Stopped (Disabled)
Error Severity: Normal
Service Flags: Kernel Driver, Shared Process
Floppy (Primary disk) Running (System)
Error Severity: Ignore
Service Flags: Kernel Driver, Shared Process
Ftdisk (Filter) Stopped (Disabled)
Error Severity: Ignore
Service Flags: Kernel Driver, Shared Process
i8042 Keyboard and PS/2 Mouse Port Driver (Keyboard Port) Running (System)

```

```

System32\DRIVERS\i8042prt.sys
Error Severity: Normal
Service Flags: Kernel Driver, Shared Process
Inport (Pointer Port) Stopped (Disabled)
Error Severity: Ignore
Service Flags: Kernel Driver, Shared Process
Jazzg300 (Video) Stopped (Disabled)
Error Severity: Ignore
Service Flags: Kernel Driver, Shared Process
Jazzg364 (Video) Stopped (Disabled)
Error Severity: Ignore
Service Flags: Kernel Driver, Shared Process
Jzvx1484 (Video) Stopped (Disabled)
Error Severity: Ignore
Service Flags: Kernel Driver, Shared Process
Keyboard Class Driver (Keyboard Class) Running (System)
System32\DRIVERS\kbdclass.sys
Error Severity: Normal
Service Flags: Kernel Driver, Shared Process
KSecDD (Base) Running (System)
Error Severity: Normal
Service Flags: Kernel Driver, Shared Process
mga (Video) Stopped (Disabled)
Error Severity: Ignore
Service Flags: Kernel Driver, Shared Process
mga_mil (Video) Stopped (Disabled)
Error Severity: Ignore
Service Flags: Kernel Driver, Shared Process
mitsumi (SCSI miniport) Stopped (Disabled)
Error Severity: Normal
Service Flags: Kernel Driver, Shared Process
mkecr5xx (SCSI miniport) Stopped (Disabled)
Error Severity: Normal
Service Flags: Kernel Driver, Shared Process
Modem (Extended base) Stopped (Manual)
Error Severity: Ignore
Service Flags: Kernel Driver, Shared Process
Mouse Class Driver (Pointer Class) Running (System)
System32\DRIVERS\mouclass.sys
Error Severity: Normal
Service Flags: Kernel Driver, Shared Process
Msfs (File system) Running (System)
Error Severity: Normal
Service Flags: File System Driver, Shared Process
Mup (Network) Running (Manual)
C:\WINNT\System32\drivers\mup.sys
Error Severity: Normal
Service Flags: File System Driver, Shared Process
Ncr53c9x (SCSI miniport) Stopped (Disabled)
Error Severity: Normal
Service Flags: Kernel Driver, Shared Process
ncr77c22 (Video) Stopped (Disabled)
Error Severity: Ignore
Service Flags: Kernel Driver, Shared Process
Ncr700 (SCSI miniport) Stopped (Disabled)
Error Severity: Normal
Service Flags: Kernel Driver, Shared Process
Ncr710 (SCSI miniport) Stopped (Disabled)
Error Severity: Normal
Service Flags: Kernel Driver, Shared Process
Microsoft NDIS System Driver (NDIS) Running (System)
Error Severity: Normal
Service Flags: Kernel Driver, Shared Process
NetBIOS Interface (NetBIOSGroup) Stopped (Manual)
C:\WINNT\System32\drivers\netbios.sys
Error Severity: Normal
Service Flags: File System Driver, Shared Process
Group Dependencies:
TDI
WINS Client (TCP/IP) (PNP_TDI) Running (Automatic)
C:\WINNT\System32\drivers\netbt.sys

```

```

Error Severity: Normal
Service Flags: Kernel Driver, Shared Process
Service Dependencies:
  Tcpip
NetDetect                               Stopped (Manual)
  C:\WINNT\system32\drivers\netdtect.sys
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Npfs (File system)                       Running (System)
  Error Severity: Normal
  Service Flags: File System Driver, Shared Process
Ntfs (File system)                       Running (Disabled)
  Error Severity: Normal
  Service Flags: File System Driver, Shared Process
Null (Base)                              Running (System)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Oliscsi (SCSI miniport)                  Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Parallel (Extended base)                 Running (Automatic)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
Service Dependencies:
  Parport
Group Dependencies:
  Parallel arbitrator
Parport (Parallel arbitrator)            Running (Automatic)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
ParVdm (Extended base)                   Running (Automatic)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
Service Dependencies:
  Parport
Group Dependencies:
  Parallel arbitrator
PCIDump (PCI Configuration)              Stopped (System)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
Pcmcia (System Bus Extender)             Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
PnP ISA Enabler Driver (Base)            Stopped (System)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
psidisp (Video)                          Stopped (Disabled)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
Q110wnt (SCSI miniport)                  Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
qv (Video)                                Stopped (Disabled)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
Rdr (Network)                            Running (Manual)
  C:\WINNT\System32\drivers\rdr.sys
  Error Severity: Normal
  Service Flags: File System Driver, Shared Process
s3 (Video)                                Stopped (Disabled)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
Scsiprnt (Extended base)                 Stopped (Automatic)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
Group Dependencies:
  SCSI miniport
Scsiscan (SCSI Class)                    Running (System)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
Group Dependencies:

```

```

SCSI miniport
Serial (Extended base)                   Running (Automatic)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
Sermouse (Pointer Port)                  Stopped (Disabled)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
Sfloppy (Primary disk)                   Stopped (System)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
Group Dependencies:
  SCSI miniport
Simbad (Filter)                          Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
slcd32 (SCSI miniport)                   Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Sparrow (SCSI miniport)                  Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Spock (SCSI miniport)                    Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Srv (Network)                            Running (Manual)
  C:\WINNT\System32\drivers\srv.sys
  Error Severity: Normal
  Service Flags: File System Driver, Shared Process
symc810 (SCSI miniport)                  Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
symc8xx (SCSI miniport)                  Running (Boot)
  C:\WINNT\system32\drivers\symc8xx.sys
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
T128 (SCSI miniport)                     Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
T13B (SCSI miniport)                     Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
TCP/IP Service (PNP_TDI)                  Running (Automatic)
  C:\WINNT\System32\drivers\tcpip.sys
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
tga (Video)                               Stopped (Disabled)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
tmvl (SCSI miniport)                     Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Ultra124 (SCSI miniport)                  Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Ultra14f (SCSI miniport)                  Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
Ultra24f (SCSI miniport)                  Stopped (Disabled)
  Error Severity: Normal
  Service Flags: Kernel Driver, Shared Process
update (Base)                             Stopped (System)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
v7vram (Video)                            Stopped (Disabled)
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
VgaSave (Video Save)                      Running (System)
  C:\WINNT\System32\drivers\vga.sys
  Error Severity: Ignore
  Service Flags: Kernel Driver, Shared Process
VgaStart (Video Init)                     Stopped (System)

```

```

C:\WINNT\System32\drivers\vga.sys
Error Severity: Ignore
Service Flags: Kernel Driver, Shared Process
Wd33c93 (SCSI miniport) Stopped (Disabled)
Error Severity: Normal
Service Flags: Kernel Driver, Shared Process
wd90c24a (Video) Stopped (Disabled)
Error Severity: Ignore
Service Flags: Kernel Driver, Shared Process
wdvga (Video) Stopped (Disabled)
Error Severity: Ignore
Service Flags: Kernel Driver, Shared Process
weitek9 (Video) Stopped (Disabled)
Error Severity: Ignore
Service Flags: Kernel Driver, Shared Process
Xga (Video) Stopped (Disabled)
Error Severity: Ignore
Service Flags: Kernel Driver, Shared Process

```

IRQ and Port Report

Devices	Vector	Level	Affinity
MPS 1.4 - APIC platform	8	8	0x00000003
MPS 1.4 - APIC platform	0	0	0x00000003
MPS 1.4 - APIC platform	1	1	0x00000003
MPS 1.4 - APIC platform	2	2	0x00000003
MPS 1.4 - APIC platform	3	3	0x00000003
MPS 1.4 - APIC platform	4	4	0x00000003
MPS 1.4 - APIC platform	5	5	0x00000003
MPS 1.4 - APIC platform	6	6	0x00000003
MPS 1.4 - APIC platform	7	7	0x00000003
MPS 1.4 - APIC platform	8	8	0x00000003
MPS 1.4 - APIC platform	9	9	0x00000003
MPS 1.4 - APIC platform	10	10	0x00000003
MPS 1.4 - APIC platform	11	11	0x00000003
MPS 1.4 - APIC platform	12	12	0x00000003
MPS 1.4 - APIC platform	13	13	0x00000003
MPS 1.4 - APIC platform	14	14	0x00000003
MPS 1.4 - APIC platform	15	15	0x00000003
MPS 1.4 - APIC platform	16	16	0x00000003
MPS 1.4 - APIC platform	17	17	0x00000003
MPS 1.4 - APIC platform	18	18	0x00000003
MPS 1.4 - APIC platform	19	19	0x00000003
MPS 1.4 - APIC platform	20	20	0x00000003
MPS 1.4 - APIC platform	21	21	0x00000003
MPS 1.4 - APIC platform	22	22	0x00000003
MPS 1.4 - APIC platform	23	23	0x00000003
MPS 1.4 - APIC platform	24	24	0x00000003
MPS 1.4 - APIC platform	25	25	0x00000003
MPS 1.4 - APIC platform	26	26	0x00000003
MPS 1.4 - APIC platform	27	27	0x00000003
MPS 1.4 - APIC platform	28	28	0x00000003
MPS 1.4 - APIC platform	29	29	0x00000003
MPS 1.4 - APIC platform	30	30	0x00000003
MPS 1.4 - APIC platform	31	31	0x00000003
MPS 1.4 - APIC platform	32	32	0x00000003
MPS 1.4 - APIC platform	33	33	0x00000003
MPS 1.4 - APIC platform	34	34	0x00000003
MPS 1.4 - APIC platform	35	35	0x00000003
MPS 1.4 - APIC platform	36	36	0x00000003
MPS 1.4 - APIC platform	37	37	0x00000003
MPS 1.4 - APIC platform	38	38	0x00000003
MPS 1.4 - APIC platform	39	39	0x00000003
MPS 1.4 - APIC platform	40	40	0x00000003
MPS 1.4 - APIC platform	41	41	0x00000003
MPS 1.4 - APIC platform	42	42	0x00000003
MPS 1.4 - APIC platform	43	43	0x00000003
MPS 1.4 - APIC platform	44	44	0x00000003

MPS 1.4 - APIC platform	45	45	0x00000003
MPS 1.4 - APIC platform	46	46	0x00000003
MPS 1.4 - APIC platform	47	47	0x00000003
MPS 1.4 - APIC platform	61	61	0x00000003
MPS 1.4 - APIC platform	65	65	0x00000003
MPS 1.4 - APIC platform	80	80	0x00000003
MPS 1.4 - APIC platform	193	193	0x00000003
MPS 1.4 - APIC platform	225	225	0x00000003
MPS 1.4 - APIC platform	253	253	0x00000003
MPS 1.4 - APIC platform	254	254	0x00000003
MPS 1.4 - APIC platform	255	255	0x00000003
i8042prt	1	1	0xffffffff
i8042prt	12	12	0xffffffff
Serial	4	4	0x00000000
Serial	3	3	0x00000000
El90x	44	44	0x00000000
EMPCI	16	16	0x41603857
EMPCI	16	16	0x80f8c020
EMPCI	16	16	0x80f8c020
EMPCI	16	16	0x000019a3
EMPCI	16	16	0x00000000
EMPCI	16	16	0x00000000
EMPCI	16	16	0x00000000
EMPCI	16	16	0x00000000
Floppy	6	6	0x00000000
symc8xx	48	48	0x00000000
symc8xx	48	48	0x00000000

Devices	Physical Address	Length
MPS 1.4 - APIC platform	0x00000000	0x0000000010
MPS 1.4 - APIC platform	0x00000020	0x0000000002
MPS 1.4 - APIC platform	0x00000040	0x0000000004
MPS 1.4 - APIC platform	0x00000048	0x0000000004
MPS 1.4 - APIC platform	0x00000061	0x0000000001
MPS 1.4 - APIC platform	0x00000070	0x0000000002
MPS 1.4 - APIC platform	0x00000080	0x0000000010
MPS 1.4 - APIC platform	0x00000092	0x0000000001
MPS 1.4 - APIC platform	0x000000a0	0x0000000002
MPS 1.4 - APIC platform	0x000000c0	0x0000000010
MPS 1.4 - APIC platform	0x000000f0	0x0000000010
i8042prt	0x00000060	0x0000000001
i8042prt	0x00000064	0x0000000001
Parport	0x00000378	0x0000000003
Serial	0x000003f8	0x0000000007
Serial	0x000002f8	0x0000000007
El90x	0x00004400	0x0000000080
EMPCI	0x00005000	0x0000000080
EMPCI	0x00005080	0x0000000080
EMPCI	0x00005400	0x0000000080
EMPCI	0x00007000	0x0000000080
EMPCI	0x00007080	0x0000000080
EMPCI	0x00007400	0x0000000080
EMPCI	0x00007480	0x0000000080
EMPCI	0x00005480	0x0000000080
Floppy	0x000003f0	0x0000000006
Floppy	0x000003f7	0x0000000001
symc8xx	0x00004000	0x0000000100
symc8xx	0x00006000	0x0000000100
VgaSave	0x000003b0	0x000000000c
VgaSave	0x000003c0	0x0000000020
VgaSave	0x000001ce	0x0000000002

DMA and Memory Report

Devices	Channel	Port
Floppy	2	0

Devices	Physical Address	Length
MPS 1.4 - APIC platform	0xfec00000	0x00000400
MPS 1.4 - APIC platform	0xfec01000	0x00000400
MPS 1.4 - APIC platform	0xfec02000	0x00000400
E190x	0xfe011000	0x00000080
symc8xx	0xfe011400	0x00000100
symc8xx	0xfe010000	0x00000100
symc8xx	0xfe501000	0x00000100
symc8xx	0xfe500000	0x00000100
c1546x	0xfe000000	0x0000a000
c1546x	0xfc000000	0x02000000
VgaSave	0x000a0000	0x00020000

Environment Report

System Environment Variables

```

APPDIR=C:\inetpub\wwwroot
ComSpec=C:\WINNT\system32\cmd.exe
HOME=C:/
NODE_MAX_W=1200
NUMBER_OF_PROCESSORS=2
OS=Windows_NT
Os2LibPath=C:\WINNT\system32\os2\dll;
Path=C:\Oracle\Ora81\bin;C:\Program
Files\Oracle\jre\1.1.7\bin;C:\mks\mksnt;C:\WINNT\system32;C:\WINNT;C:\TUXEDO\bin;
PROCESSOR_ARCHITECTURE=x86
PROCESSOR_IDENTIFIER=x86 Family 6 Model 7 Stepping 3, GenuineIntel
PROCESSOR_LEVEL=6
PROCESSOR_REVISION=0703
ROOTDIR=C:/mks
SHELL=C:/mks/mksnt/sh.exe
TMCONTEXTS=1
TMPDIR=C:/TEMP
TUXCONFIG=C:\inetpub\wwwroot\tuxconfig
TUXDIR=C:\TUXEDO
windir=C:\WINNT

```

Environment Variables for Current User

```

local=tpcc
ORACLE_HOME=C:\oracle\ora81
SQLPLUS=sqlplus
SQLDBA=svrmgrl
TEMP=C:\TEMP
TMP=C:\TEMP

```

Network Report

```

Your Access Level: Admin & Local
Workgroup or Domain: WORKGROUP
Network Version: 4.0
LanRoot: WORKGROUP
Logged On Users: 1
Current User (1): Administrator
Logon Domain: CL01
Logon Server: CL01

```

```

Transport: NetBT_E190x1, 00-00-4C-D1-84-14, VC's: 0, Wan: Wan
Transport: NetBT_EMPC13, 00-00-92-A7-84-5C, VC's: 0, Wan: Wan
Transport: NetBT_EMPC14, 00-00-92-A7-84-5D, VC's: 0, Wan: Wan
Transport: NetBT_EMPC12, 00-00-92-A7-84-5F, VC's: 0, Wan: Wan
Transport: NetBT_EMPC15, 00-00-92-A7-84-5E, VC's: 0, Wan: Wan

```

```

Transport: NetBT_EMPC16, 00-00-92-A7-84-44, VC's: 0, Wan: Wan
Transport: NetBT_EMPC17, 00-00-92-A7-84-45, VC's: 0, Wan: Wan
Transport: NetBT_EMPC18, 00-00-92-A7-84-46, VC's: 0, Wan: Wan
Transport: NetBT_EMPC19, 00-00-92-A7-84-47, VC's: 0, Wan: Wan

```

```

Character Wait: 3,600
Collection Time: 250
Maximum Collection Count: 16
Keep Connection: 600
Maximum Commands: 5
Session Time Out: 45
Character Buffer Size: 512
Maximum Threads: 17
Lock Quota: 6,144
Lock Increment: 10
Maximum Locks: 500
Pipe Increment: 10
Maximum Pipes: 500
Cache Time Out: 40
Dormant File Limit: 45
Read Ahead Throughput: 4,294,967,295
Mailslot Buffers: 3
Server Announce Buffers: 20
Illegal Datagrams: 5
Datagram Reset Frequency: 60
Log Election Packets: False
Use Opportunistic Locking: True
Use Unlock Behind: True
Use Close Behind: True
Buffer Pipes: True
Use Lock, Read, Unlock: True
Use NT Caching: True
Use Raw Read: True
Use Raw Write: True
Use Write Raw Data: True
Use Encryption: True
Buffer Deny Write Files: True
Buffer Read Only Files: True
Force Core Creation: True
512 Byte Max Transfer: False
Bytes Received: 22,225,172
SMB's Received: 9,499
Paged Read Bytes Requested: 19,546,112
Non Paged Read Bytes Requested: 2,885,632
Cache Read Bytes Requested: 161,280
Network Read Bytes Requested: 21,645,672
Bytes Transmitted: 2,109,888
SMB's Transmitted: 9,491
Paged Read Bytes Requested: 0
Non Paged Read Bytes Requested: 21,611,924
Cache Read Bytes Requested: 0
Network Read Bytes Requested: 21,611,924
Initially Failed Operations: 0
Failed Completion Operations: 0
Read Operations: 8,579
Random Read Operations: 739
Read SMB's: 8,264
Large Read SMB's: 369
Small Read SMB's: 5,328
Write Operations: 352
Random Write Operations: 0
Write SMB's: 352
Large Write SMB's: 352
Small Write SMB's: 0
Raw Reads Denied: 0
Raw Writes Denied: 0
Network Errors: 0
Sessions: 18
Failed Sessions: 0
Reconnects: 0
Core Connects: 0

```

```
LM 2.0 Connects: 0
LM 2.x Connects: 0
Windows NT Connects: 9
Server Disconnects: 0
Hung Sessions: 0
Use Count: 0
Failed Use Count: 0
Current Commands: 0
Server File Opens: 46
Server Device Opens: 0
Server Jobs Queued: 0
Server Session Opens: 0
Server Sessions Timed Out: 2
Server Sessions Errored Out: 2
Server Password Errors: 0
Server Permission Errors: 0
Server System Errors: 0
Server Bytes Sent: 3,027,029
Server Bytes Received: 333,222
Server Average Response Time: 0
Server Request Buffers Needed: 0
Server Big Buffers Needed: 0
```

Client Resisry infromation

<Inet Information Server Registry >

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Parameters]
"BandwidthLevel"=dword:ffffff
"ListenBackLog"=dword:00000200
"PoolThreadLimit"=dword:00000200
"ThreadTimeout"=dword:00001c20
"MemoryCacheSize"=dword:00000000

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Parameters\Filter]
"FilterType"=dword:00000000
"NumGrantSites"=dword:00000000
"NumDenySites"=dword:00000000

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Parameters\MimeMap]
"text/html,htm,,h"=""
"image/gif,gif,,g"=""
"image/jpeg,jpg,,:"=""
"text/plain,txt,,0"=""
"text/html,html,,h"=""
"image/jpeg,jpeg,,:"=""
"image/jpeg,jpe,,:"=""
"image/bmp,bmp,,:"=""
"application/octet-stream,*,,5"=""
"application/pdf,pdf,,5"=""
"application/octet-stream,bin,,5"=""
"application/oda,oda,,5"=""
"application/zip,zip,,9"=""
"application/rtf,rtf,,5"=""
"application/postscript,ps,,5"=""
"application/postscript,ai,,5"=""
"application/postscript,eps,,5"=""
"application/mac-binhex40,hqx,,4"=""
"application/msword,doc,,5"=""
"application/msword,dot,,5"=""
"application/winhelp,hlp,,5"=""
"video/mpeg,mpeg,,:"=""
"video/mpeg,mpg,,;"=""
"video/mpeg,mpe,,;"=""
"video/x-msvideo,avi,,<"=""
"video/quicktime,qt,,;"=""
"video/quicktime,mov,,;"=""
```

```
"video/x-sgi-movie,movie,,<"=""
"x-world/x-vrml,wrl,,5"=""
"x-world/x-vrml,xaf,,5"=""
"x-world/x-vrml,xof,,5"=""
"x-world/x-vrml,flr,,5"=""
"x-world/x-vrml,wrz,,5"=""
"application/x-director,dcr,,5"=""
"application/x-director,dir,,5"=""
"application/x-director,dxr,,5"=""
"image/cis-cod,cod,,5"=""
"image/x-cmx,cmx,,5"=""
"application/envoy,evy,,5"=""
"application/x-msaccess,mdb,,5"=""
"application/x-mscardfile,crd,,5"=""
"application/x-msclip,clip,,5"=""
"application/octet-stream,exe,,5"=""
"application/x-msexcel,xla,,5"=""
"application/x-msexcel,xlc,,5"=""
"application/x-msexcel,xlm,,5"=""
"application/x-msexcel,xls,,5"=""
"application/x-msexcel,xlt,,5"=""
"application/x-msexcel,xlw,,5"=""
"application/x-msmediaview,m13,,5"=""
"application/x-msmediaview,m14,,5"=""
"application/x-msmoney,mny,,5"=""
"application/x-mspowerpoint,ppt,,5"=""
"application/x-msproject,mpp,,5"=""
"application/x-mspublisher,pub,,5"=""
"application/x-msterminal,trm,,5"=""
"application/x-msworks,wks,,5"=""
"application/x-mswrite,wri,,5"=""
"application/x-msmetafile,wmf,,5"=""
"application/x-csh,csh,,5"=""
"application/x-dvi,dvi,,5"=""
"application/x-hdf,hdf,,5"=""
"application/x-latex,latex,,5"=""
"application/x-netcdf,nc,,5"=""
"application/x-netcdf,cdf,,5"=""
"application/x-sh,sh,,5"=""
"application/x-tcl,tcl,,5"=""
"application/x-tex,tex,,5"=""
"application/x-texinfo,texinfo,,5"=""
"application/x-texinfo,txi,,5"=""
"application/x-troff,t,,5"=""
"application/x-troff,tr,,5"=""
"application/x-troff,roff,,5"=""
"application/x-troff-man,man,,5"=""
"application/x-troff-me,me,,5"=""
"application/x-troff-ms,ms,,5"=""
"application/x-wais-source,src,,7"=""
"application/x-bcpio,bcpio,,5"=""
"application/x-cpio,cpio,,5"=""
"application/x-gtar,gtar,,9"=""
"application/x-shar,shar,,5"=""
"application/x-sv4cpio,sv4cpio,,5"=""
"application/x-sv4crc,sv4crc,,5"=""
"application/x-tar,tar,,5"=""
"application/x-ustar,ustar,,5"=""
"audio/basic,au,,<"=""
"audio/basic,snd,,<"=""
"audio/x-aiff,aiff,,<"=""
"audio/x-aiff,aiff,,<"=""
"audio/x-aiff,aifc,,<"=""
"audio/x-wav,wav,,<"=""
"audio/x-pn-realaudio,ram,,<"=""
"image/ief,ief,,:"=""
"image/tiff,tiff,,:"=""
"image/tiff,tif,,:"=""
"image/x-cmu-raster,ras,,:"=""
"image/x-portable-anymap,pnm,,:"=""
"image/x-portable-bitmap,pbm,,:"=""
```

```
"image/x-portable-graymap,pgm,,:=""
"image/x-portable-pixmap,ppm,,:=""
"image/x-rgb,rgb,,:=""
"image/x-xbitmap,xbm,,:=""
"image/x-xpixmap,xpm,,:=""
"image/x-xwindowdump,xwd,,:=""
"text/html,stm,h=""
"text/plain,bas,,0=""
"text/plain,c,,0=""
"text/plain,h,,0=""
"text/richtext,rtx,,0=""
"text/tab-separated-values,tsv,,0=""
"text/x-setext,etx,,0=""
"application/x-perfmon,pmc,,5=""
"application/x-perfmon,pma,,5=""
"application/x-perfmon,pmr,,5=""
"application/x-perfmon,pml,,5=""
"application/x-perfmon,pmw,,5=""
```

<WWW Service Registry>

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Parameters]
"MajorVersion"=dword:00000002
"MinorVersion"=dword:00000000
"AdminName"="Administrator"
"AdminEmail"="Admin@corp.com"
"MaxConnections"=dword:000186a0
"LogType"=dword:00000000
"LogFileDirectory"=hex(2):25,53,79,73,74,65,6d,52,6f,6f,74,25,5c,53,79,73,74,\
65,6d,33,32,5c,4c,6f,67,46,69,6c,65,73,00
"LogFileTruncateSize"=dword:01388000
"LogFilePeriod"=dword:00000001
"LogFileFormat"=dword:00000000
"LogSqlDataSource"="HTTPLOG"
"LogSqlTableName"="Internetlog"
"LogSqlUserName"="InternetAdmin"
"LogSqlPassword"="sqllog"
"Authorization"=dword:00000001
"AnonymousUserName"="IUSR_CL01"
"Default Load File"="Default.htm"
"Dir Browse Control"=dword:4000001e
"CheckForWAISDB"=dword:00000000
"CacheExtensions"=dword:00000001
"GlobalExpire"=dword:ffffffff
"ServerSideIncludesEnabled"=dword:00000001
"ServerSideIncludesExtension"=".stm"
"DebugFlags"=dword:00000008
"ScriptTimeout"=dword:00000384
"ConnectionTimeout"=dword:00001c20
"InstallPath"="C:\WINNT\System32\inetrv"
"SecurePort"=dword:000001bb
"Filter DLLs"="C:\WINNT\System32\inetrv\ssfilt.dll"
"AccessDeniedMessage"="Error: Access is Denied."
"NTAuthenticationProviders"="NTLM"
"ServerComment"=""
"AcceptExOutstanding"=dword:00001000

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Parameters\Script Map]
".idc"="C:\WINNT\System32\inetrv\httpodbc.dll"

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Parameters\Virtual Roots]
"/,="C:\InetPub\wwwroot,,5"
"/Scripts,="C:\InetPub\scripts,,4"
"/iisadmin,="C:\WINNT\System32\inetrv\iisadmin,,1"
```

<Web Client APP (tpcc.dll) >

```
[HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\tpcc]
```

```
"LOG"="OFF"
"PATH"="c:\\temp\\"
"MaxConnections"="12010"
"MaximumWarehouses"="4800"
"DEBUG"="OFF"
```

Tuxedo Configuration

```
### TUXEDO UBBCONFIG FILE ###
*RESOURCES
IPCKEY 133133

MAXACCESSERS 1200
MAXSERVERS 95
MAXSERVICES 1200
MODEL SHM
MASTER CL01
LDBAL Y
SCANUNIT 15
BLOCKTIME 60
BBLQUERY 60

*MACHINES
DEFAULT:

"CL01" LMID= CL01
TUXDIR="C:\tuxedo"
APPDIR="C:\inetPub\wwwroot"
TUXCONFIG="c:\inetpub\wwwroot\tuxconfig"
ULOGPFX="C:\InetPub\wwwroot\ULOG"
TYPE="WinNT"
UID= 0
GID= 0

*GROUPS
GROUPGEN
LMID=CL01 GRPNO=1 OPENINFO=NONE

*SERVERS
DEFAULT:

GENERIC SRVGRP=GROUPGEN
SRVID=100
MIN=45 MAX=95
REPLYQ=Y
RQADDR=OPSTUXSERVER
CLOPT="-s OPSTUXSERVER:OPSTUXSERVER"

*SERVICES
#OPSTUXSERVER

### TUXEDO IPC RESOURCE REGISTRY ###

[HKEY_LOCAL_MACHINE\SOFTWARE\BEA Systems\TUXEDO\6.4\Environment]
"TUXDIR"="C:\\TUXEDO"
"NLSPATH"="C:\\TUXEDO\\locale\\C"
"ULOGDIR"="C:\\TUXEDO"
```

```
"ULOGPFX"="C:\\ULOG"  
"ULOGOUT"=dword:00000002  
"TUXIPC_MSG_BYTES"=dword:00010000  
"TUXIPC_MSG_HDRS"=dword:00001fc0  
"TUXIPC_MSG_QUEUES"=dword:000005dc  
"TUXIPC_MSG_QUEUE_BYTES"=dword:00010000  
"TUXIPC_MSG_SEG_BYTES"=dword:00000040  
"TUXIPC_MSG_SEGS"=dword:00007fff  
"TUXIPC_PROC"=dword:000005dc  
"TUXIPC_SEM"=dword:00000800  
"TUXIPC_SEM_IDS"=dword:00000800  
"TUXIPC_SEM_UNDO"=dword:00000800  
"TUXIPC_SHM_PROCS"=dword:00030d40  
"TUXIPC_SHM_SEGS"=dword:000186a0
```

```
[HKEY_LOCAL_MACHINE\\SOFTWARE\\BEA  
Systems\\TUXEDO\\6.4\\Environment\\Services]
```

```
[HKEY_LOCAL_MACHINE\\SOFTWARE\\BEA  
Systems\\TUXEDO\\6.4\\Environment\\Services\\3050]
```

Oracle 8i parameters

Startup Parameters

<Init_1.ora>

```
instance_number=1
thread=1
ifile=init_common.ora
```

```
rollback_segments=(t_0_1 ,t_0_2 ,t_0_3 ,t_0_4 ,t_0_5 ,t_0_6 ,t_0_7 ,t_0
_8 ,t_0_9 ,t_0_10 ,
t_0_11 ,t_0_12 ,t_0_13 ,t_0_14 ,t_0_15 ,t_0_16 ,t_0_17 ,t_0_18 ,t_0_19 ,
t_0_20 ,
t_0_21 ,t_0_22 ,t_0_23 ,t_0_24 ,t_0_25 ,t_0_26 ,t_0_27 ,t_0_28 ,t_0_29 ,
t_0_30 ,
t_0_31 ,t_0_32 ,t_0_33 ,t_0_34 ,t_0_35 ,t_0_36 ,t_0_37 ,t_0_38 ,t_0_39 ,
t_0_40 ,
t_0_41 ,t_0_42 ,t_0_43 ,t_0_44 ,t_0_45 ,t_0_46 ,t_0_47 ,t_0_48 ,t_0_49 ,
t_0_50 ,
t_0_51 ,t_0_52 ,t_0_53 ,t_0_54 ,t_0_55 ,t_0_56 ,t_0_57 ,t_0_58 ,t_0_59 ,
t_0_60 ,
t_0_61 ,t_0_62 ,t_0_63 ,t_0_64 ,t_0_65 ,t_0_66 ,t_0_67 ,t_0_68 ,t_0_69 ,
t_0_70 ,
t_0_71 ,t_0_72 ,t_0_73 ,t_0_74 ,t_0_75 ,t_0_76 ,t_0_77 ,t_0_78 ,t_0_79 ,
t_0_80 ,
t_0_81 ,t_0_82 ,t_0_83 ,t_0_84 ,t_0_85 ,t_0_86 ,t_0_87 ,t_0_88 ,t_0_89 ,
t_0_90 ,
t_0_91 ,t_0_92 ,t_0_93 ,t_0_94 ,t_0_95 ,t_0_96 ,t_0_97 ,t_0_98 ,t_0_99 ,
t_0_100 ,
t_0_101 ,t_0_102 ,t_0_103 ,t_0_104 ,t_0_105 ,t_0_106 ,t_0_107 ,t_0_108 ,
t_0_109 ,t_0_110 ,
t_0_111 ,t_0_112 ,t_0_113 ,t_0_114 ,t_0_115 ,t_0_116 ,t_0_117 ,t_0_118 ,
t_0_119 ,t_0_120 ,
t_0_121 ,t_0_122 ,t_0_123 ,t_0_124 ,t_0_125 ,t_0_126 ,t_0_127 ,t_0_128 ,
t_0_129 ,t_0_130 ,
t_0_131 ,t_0_132 ,t_0_133 ,t_0_134 ,t_0_135 ,t_0_136 ,t_0_137 ,t_0_138 ,
t_0_139 ,t_0_140 ,
t_0_141 ,t_0_142 ,t_0_143 ,t_0_144 ,t_0_145 ,t_0_146 ,t_0_147 ,t_0_148 ,
t_0_149 ,t_0_150 ,
t_0_151 ,t_0_152 ,t_0_153 ,t_0_154 ,t_0_155 ,t_0_156 ,t_0_157 ,t_0_158 ,
t_0_159 ,t_0_160 ,
t_0_161 ,t_0_162 ,t_0_163 ,t_0_164 ,t_0_165 ,t_0_166 ,t_0_167 ,t_0_168 ,
t_0_169 ,t_0_170 ,
t_0_171 ,t_0_172 ,t_0_173 ,t_0_174 ,t_0_175 ,t_0_176 ,t_0_177 ,t_0_178 ,
t_0_179 ,t_0_180 ,
t_0_181 ,t_0_182 ,t_0_183 ,t_0_184 ,t_0_185 ,t_0_186 ,t_0_187 ,t_0_188 ,
t_0_189 ,t_0_190 ,
t_0_191 ,t_0_192 ,t_0_193 ,t_0_194 ,t_0_195 ,t_0_196 ,t_0_197 ,t_0_198 ,
t_0_199 ,t_0_200)
```

<Init_2.ora>

```
instance_number=2
thread=2
ifile=init_common.ora
```

```
rollback_segments=(t_1_1 ,t_1_2 ,t_1_3 ,t_1_4 ,t_1_5 ,t_1_6 ,t_1_7 ,t_1
_8 ,t_1_9 ,t_1_10 ,
t_1_11 ,t_1_12 ,t_1_13 ,t_1_14 ,t_1_15 ,t_1_16 ,t_1_17 ,t_1_18 ,t_1_19 ,
t_1_20 ,
t_1_21 ,t_1_22 ,t_1_23 ,t_1_24 ,t_1_25 ,t_1_26 ,t_1_27 ,t_1_28 ,t_1_29 ,
t_1_30 ,
t_1_31 ,t_1_32 ,t_1_33 ,t_1_34 ,t_1_35 ,t_1_36 ,t_1_37 ,t_1_38 ,t_1_39 ,
t_1_40 ,
t_1_41 ,t_1_42 ,t_1_43 ,t_1_44 ,t_1_45 ,t_1_46 ,t_1_47 ,t_1_48 ,t_1_49 ,
t_1_50 ,
t_1_51 ,t_1_52 ,t_1_53 ,t_1_54 ,t_1_55 ,t_1_56 ,t_1_57 ,t_1_58 ,t_1_59 ,
t_1_60 ,
t_1_61 ,t_1_62 ,t_1_63 ,t_1_64 ,t_1_65 ,t_1_66 ,t_1_67 ,t_1_68 ,t_1_69 ,
t_1_70 ,
t_1_71 ,t_1_72 ,t_1_73 ,t_1_74 ,t_1_75 ,t_1_76 ,t_1_77 ,t_1_78 ,t_1_79 ,
t_1_80 ,
t_1_81 ,t_1_82 ,t_1_83 ,t_1_84 ,t_1_85 ,t_1_86 ,t_1_87 ,t_1_88 ,t_1_89 ,
t_1_90 ,
t_1_91 ,t_1_92 ,t_1_93 ,t_1_94 ,t_1_95 ,t_1_96 ,t_1_97 ,t_1_98 ,t_1_99 ,
t_1_100 ,
t_1_101 ,t_1_102 ,t_1_103 ,t_1_104 ,t_1_105 ,t_1_106 ,t_1_107 ,t_1_108 ,
t_1_109 ,t_1_110 ,
t_1_111 ,t_1_112 ,t_1_113 ,t_1_114 ,t_1_115 ,t_1_116 ,t_1_117 ,t_1_118 ,
t_1_119 ,t_1_120 ,
t_1_121 ,t_1_122 ,t_1_123 ,t_1_124 ,t_1_125 ,t_1_126 ,t_1_127 ,t_1_128 ,
t_1_129 ,t_1_130 ,
t_1_131 ,t_1_132 ,t_1_133 ,t_1_134 ,t_1_135 ,t_1_136 ,t_1_137 ,t_1_138 ,
t_1_139 ,t_1_140 ,
t_1_141 ,t_1_142 ,t_1_143 ,t_1_144 ,t_1_145 ,t_1_146 ,t_1_147 ,t_1_148 ,
t_1_149 ,t_1_150 ,
t_1_151 ,t_1_152 ,t_1_153 ,t_1_154 ,t_1_155 ,t_1_156 ,t_1_157 ,t_1_158 ,
t_1_159 ,t_1_160 ,
t_1_161 ,t_1_162 ,t_1_163 ,t_1_164 ,t_1_165 ,t_1_166 ,t_1_167 ,t_1_168 ,
t_1_169 ,t_1_170 ,
t_1_171 ,t_1_172 ,t_1_173 ,t_1_174 ,t_1_175 ,t_1_176 ,t_1_177 ,t_1_178 ,
t_1_179 ,t_1_180 ,
t_1_181 ,t_1_182 ,t_1_183 ,t_1_184 ,t_1_185 ,t_1_186 ,t_1_187 ,t_1_188 ,
t_1_189 ,t_1_190 ,
t_1_191 ,t_1_192 ,t_1_193 ,t_1_194 ,t_1_195 ,t_1_196 ,t_1_197 ,t_1_198 ,
t_1_199 ,t_1_200)
```

<Init_3.ora>

```
instance_number=3
thread=3
ifile=init_common.ora
```

```
rollback_segments=(t_2_1 ,t_2_2 ,t_2_3 ,t_2_4 ,t_2_5 ,t_2_6 ,t_2_7 ,t_2
_8 ,t_2_9 ,t_2_10 ,
t_2_11 ,t_2_12 ,t_2_13 ,t_2_14 ,t_2_15 ,t_2_16 ,t_2_17 ,t_2_18 ,t_2_19 ,
t_2_20 ,
```

```

t_2_21 ,t_2_22 ,t_2_23 ,t_2_24 ,t_2_25 ,t_2_26 ,t_2_27 ,t_2_28 ,t_2_29 ,
t_2_30 ,
t_2_31 ,t_2_32 ,t_2_33 ,t_2_34 ,t_2_35 ,t_2_36 ,t_2_37 ,t_2_38 ,t_2_39 ,
t_2_40 ,
t_2_41 ,t_2_42 ,t_2_43 ,t_2_44 ,t_2_45 ,t_2_46 ,t_2_47 ,t_2_48 ,t_2_49 ,
t_2_50 ,
t_2_51 ,t_2_52 ,t_2_53 ,t_2_54 ,t_2_55 ,t_2_56 ,t_2_57 ,t_2_58 ,t_2_59 ,
t_2_60 ,
t_2_61 ,t_2_62 ,t_2_63 ,t_2_64 ,t_2_65 ,t_2_66 ,t_2_67 ,t_2_68 ,t_2_69 ,
t_2_70 ,
t_2_71 ,t_2_72 ,t_2_73 ,t_2_74 ,t_2_75 ,t_2_76 ,t_2_77 ,t_2_78 ,t_2_79 ,
t_2_80 ,
t_2_81 ,t_2_82 ,t_2_83 ,t_2_84 ,t_2_85 ,t_2_86 ,t_2_87 ,t_2_88 ,t_2_89 ,
t_2_90 ,
t_2_91 ,t_2_92 ,t_2_93 ,t_2_94 ,t_2_95 ,t_2_96 ,t_2_97 ,t_2_98 ,t_2_99 ,
t_2_100 ,
t_2_101 ,t_2_102 ,t_2_103 ,t_2_104 ,t_2_105 ,t_2_106 ,t_2_107 ,t_2_108 ,
t_2_109 ,t_2_110 ,
t_2_111 ,t_2_112 ,t_2_113 ,t_2_114 ,t_2_115 ,t_2_116 ,t_2_117 ,t_2_118 ,
t_2_119 ,t_2_120 ,
t_2_121 ,t_2_122 ,t_2_123 ,t_2_124 ,t_2_125 ,t_2_126 ,t_2_127 ,t_2_128 ,
t_2_129 ,t_2_130 ,
t_2_131 ,t_2_132 ,t_2_133 ,t_2_134 ,t_2_135 ,t_2_136 ,t_2_137 ,t_2_138 ,
t_2_139 ,t_2_140 ,
t_2_141 ,t_2_142 ,t_2_143 ,t_2_144 ,t_2_145 ,t_2_146 ,t_2_147 ,t_2_148 ,
t_2_149 ,t_2_150 ,
t_2_151 ,t_2_152 ,t_2_153 ,t_2_154 ,t_2_155 ,t_2_156 ,t_2_157 ,t_2_158 ,
t_2_159 ,t_2_160 ,
t_2_161 ,t_2_162 ,t_2_163 ,t_2_164 ,t_2_165 ,t_2_166 ,t_2_167 ,t_2_168 ,
t_2_169 ,t_2_170 ,
t_2_171 ,t_2_172 ,t_2_173 ,t_2_174 ,t_2_175 ,t_2_176 ,t_2_177 ,t_2_178 ,
t_2_179 ,t_2_180 ,
t_2_181 ,t_2_182 ,t_2_183 ,t_2_184 ,t_2_185 ,t_2_186 ,t_2_187 ,t_2_188 ,
t_2_189 ,t_2_190 ,
t_2_191 ,t_2_192 ,t_2_193 ,t_2_194 ,t_2_195 ,t_2_196 ,t_2_197 ,t_2_198 ,
t_2_199 ,t_2_200)

```

<Init_4.ora>

```

instance_number=4
thread=4
ifile=init_common.ora

```

```

rollback_segments=(t_3_1 ,t_3_2 ,t_3_3 ,t_3_4 ,t_3_5 ,t_3_6 ,t_3_7 ,t_3_
8 ,t_3_9 ,t_3_10 ,
t_3_11 ,t_3_12 ,t_3_13 ,t_3_14 ,t_3_15 ,t_3_16 ,t_3_17 ,t_3_18 ,t_3_19 ,
t_3_20 ,
t_3_21 ,t_3_22 ,t_3_23 ,t_3_24 ,t_3_25 ,t_3_26 ,t_3_27 ,t_3_28 ,t_3_29 ,
t_3_30 ,
t_3_31 ,t_3_32 ,t_3_33 ,t_3_34 ,t_3_35 ,t_3_36 ,t_3_37 ,t_3_38 ,t_3_39 ,
t_3_40 ,
t_3_41 ,t_3_42 ,t_3_43 ,t_3_44 ,t_3_45 ,t_3_46 ,t_3_47 ,t_3_48 ,t_3_49 ,
t_3_50 ,
t_3_51 ,t_3_52 ,t_3_53 ,t_3_54 ,t_3_55 ,t_3_56 ,t_3_57 ,t_3_58 ,t_3_59 ,
t_3_60 ,
t_3_61 ,t_3_62 ,t_3_63 ,t_3_64 ,t_3_65 ,t_3_66 ,t_3_67 ,t_3_68 ,t_3_69 ,
t_3_70 ,

```

```

t_3_71 ,t_3_72 ,t_3_73 ,t_3_74 ,t_3_75 ,t_3_76 ,t_3_77 ,t_3_78 ,t_3_79 ,
t_3_80 ,
t_3_81 ,t_3_82 ,t_3_83 ,t_3_84 ,t_3_85 ,t_3_86 ,t_3_87 ,t_3_88 ,t_3_89 ,
t_3_90 ,
t_3_91 ,t_3_92 ,t_3_93 ,t_3_94 ,t_3_95 ,t_3_96 ,t_3_97 ,t_3_98 ,t_3_99 ,
t_3_100 ,
t_3_101 ,t_3_102 ,t_3_103 ,t_3_104 ,t_3_105 ,t_3_106 ,t_3_107 ,t_3_108 ,
t_3_109 ,t_3_110 ,
t_3_111 ,t_3_112 ,t_3_113 ,t_3_114 ,t_3_115 ,t_3_116 ,t_3_117 ,t_3_118 ,
t_3_119 ,t_3_120 ,
t_3_121 ,t_3_122 ,t_3_123 ,t_3_124 ,t_3_125 ,t_3_126 ,t_3_127 ,t_3_128 ,
t_3_129 ,t_3_130 ,
t_3_131 ,t_3_132 ,t_3_133 ,t_3_134 ,t_3_135 ,t_3_136 ,t_3_137 ,t_3_138 ,
t_3_139 ,t_3_140 ,
t_3_141 ,t_3_142 ,t_3_143 ,t_3_144 ,t_3_145 ,t_3_146 ,t_3_147 ,t_3_148 ,
t_3_149 ,t_3_150 ,
t_3_151 ,t_3_152 ,t_3_153 ,t_3_154 ,t_3_155 ,t_3_156 ,t_3_157 ,t_3_158 ,
t_3_159 ,t_3_160 ,
t_3_161 ,t_3_162 ,t_3_163 ,t_3_164 ,t_3_165 ,t_3_166 ,t_3_167 ,t_3_168 ,
t_3_169 ,t_3_170 ,
t_3_171 ,t_3_172 ,t_3_173 ,t_3_174 ,t_3_175 ,t_3_176 ,t_3_177 ,t_3_178 ,
t_3_179 ,t_3_180 ,
t_3_181 ,t_3_182 ,t_3_183 ,t_3_184 ,t_3_185 ,t_3_186 ,t_3_187 ,t_3_188 ,
t_3_189 ,t_3_190 ,
t_3_191 ,t_3_192 ,t_3_193 ,t_3_194 ,t_3_195 ,t_3_196 ,t_3_197 ,t_3_198 ,
t_3_199 ,t_3_200)

```

<Init_common.ora>

```

control_files = ("\\.\control_001", "\\.\control_002")
use_indirect_data_buffers = FALSE
db_block_buffers = 990000
buffer_pool_recycle = (buffers:19800, lru_latches:4)
buffer_pool_keep = (buffers:792000, lru_latches:6)
db_block_max_dirty_target = 0
db_block_lru_latches = 16

```

```

db_file_multiblock_read_count = 32
db_file_direct_io_count = 64
parallel_server = TRUE
lm_locks = 980000
lm_ress = 570000
lm_procs = 600
gc_releasable_locks = 15000
gc_files_to_locks = "1=1500:\
2,28,33,39-44,616-627,629-639,641-650,652-\
654=4EACH:628,640,651=240EACH:\
15,655-674,676-685,687-697,699-701=5EACH:675,686,698=1100EACH:\
3,140-145,149-189=1EACH:\
29-32,484-527=1EACH:\
16-19,528-571=1EACH:\
23-26,572-615=1EACH:\
6,12-14,396-439=2EACH:\
54-76=79200:78-100=79200:102-124=79200:4,45-53,126-\
139=79200:77,101,125=18750EACH:\
219-241=33350:243-265=33350:267-289=33350:35,210-218,291-\
304=34800:242,266,290=5040EACH:\

```

```

702,704-706=1EACH:\
703,707-709=1EACH:\
27,190-192=1EACH:\
36,207-209=1EACH:\
7-10,440-483=1EACH:\
11,20-22,305-348=3EACH:\
38,196-206=1EACH:\
5,146-148=1EACH:\
37,193-195=1EACH:\
34,349-395=1EACH:\
710-757=1EACH:\
758=1"
gc_rollback_locks="0-830=20EACH"
timed_statistics = FALSE
db_files           = 810
open_cursors       = 500
processes          = 300
max_rollback_segments = 210
transactions_per_rollback_segment = 1
dml_locks          = 500
shared_pool_size   = 55000000
log_buffer         = 10485760
log_archive_start  = FALSE
log_checkpoint_interval = 1000000000
log_checkpoint_timeout = 0
log_checkpoints_to_alert = TRUE
db_name            = tpcc
db_block_size      = 2048
pre_page_sga       = TRUE
replication_dependency_tracking = FALSE
transaction_auditing = FALSE
parallel_min_servers = 0
parallel_max_servers = 0
recovery_parallelism = 20
db_block_checking = FALSE
_db_block_cache_protect = FALSE
remote_login_passwordfile= shared
cursor_space_for_time = TRUE
hash_join_enabled = FALSE
fast_start_io_target = 0
_offline_rollback_segments = (t1, t2, t3, t4, t5, t6, t7, t8, t9, t10,
t11, t12, t13, t14, t15, t16, t17, t18, t19, t20, t21, t22, t23, t24,
t25, t26, t27, t28, t29, t30)
_db_handles_cached = 5
_db_block_cache_protect = FALSE
_db_block_hash_latches = 20000
_db_file_noncontig_mblock_read_count = 1
_db_aging_hot_criteria = 2
_db_aging_stay_count = 0
_db_writer_max_writes = 210
_db_writer_chunk_writes = 210
db_writer_processes = 2
_lm_direct_sends = all
distributed_transactions = 0
mts_max_servers = 0
gc_defer_time = 0

```


Enclosure 4 Disk 9 Enabled
Enclosure 5 Disk 0 Enabled
Enclosure 5 Disk 1 Enabled
Enclosure 5 Disk 2 Enabled
Enclosure 5 Disk 3 Enabled
Enclosure 5 Disk 4 Enabled
Enclosure 5 Disk 5 Enabled
Read cache: DISABLED
Write cache: DISABLED
RAID Type: RAID0
Element Size: 64
Lun Capacity: 116753

LOGICAL UNIT NUMBER 4
Enclosure 5 Disk 6 Enabled
Enclosure 5 Disk 7 Enabled
Enclosure 5 Disk 8 Enabled
Enclosure 5 Disk 9 Enabled
Enclosure 6 Disk 0 Enabled
Enclosure 6 Disk 1 Enabled
Enclosure 6 Disk 2 Enabled
Enclosure 6 Disk 3 Enabled
Enclosure 6 Disk 4 Enabled
Enclosure 6 Disk 5 Enabled
Enclosure 6 Disk 6 Enabled
Enclosure 6 Disk 7 Enabled
Enclosure 6 Disk 8 Enabled
Enclosure 6 Disk 9 Enabled

Read cache: DISABLED
Write cache: DISABLED
RAID Type: RAID0
Element Size: 64
Lun Capacity: 116753

LOGICAL UNIT NUMBER 5
Enclosure 7 Disk 0 Enabled
Enclosure 7 Disk 1 Enabled
Enclosure 7 Disk 2 Enabled
Enclosure 7 Disk 3 Enabled
Enclosure 7 Disk 4 Enabled
Enclosure 7 Disk 5 Enabled
Enclosure 7 Disk 6 Enabled
Enclosure 7 Disk 7 Enabled
Enclosure 7 Disk 8 Enabled
Enclosure 7 Disk 9 Enabled

Read cache: DISABLED
Write cache: DISABLED
RAID Type: RAID5
Element Size: 64
Lun Capacity: 75055
ECHO is on.

==== SCSI3:0:3:0 ====

LOGICAL UNIT NUMBER 0
Enclosure 0 Disk 0 Enabled
Enclosure 0 Disk 1 Enabled
Enclosure 0 Disk 2 Enabled
Enclosure 0 Disk 3 Enabled
Enclosure 0 Disk 4 Enabled
Enclosure 0 Disk 5 Enabled
Enclosure 0 Disk 6 Enabled
Enclosure 0 Disk 7 Enabled
Enclosure 0 Disk 8 Enabled
Enclosure 0 Disk 9 Enabled
Enclosure 1 Disk 0 Enabled
Enclosure 1 Disk 1 Enabled
Enclosure 1 Disk 2 Enabled
Enclosure 1 Disk 3 Enabled

Read cache: DISABLED
Write cache: DISABLED
RAID Type: RAID0

Element Size: 64
Lun Capacity: 116753

LOGICAL UNIT NUMBER 1
Enclosure 1 Disk 4 Enabled
Enclosure 1 Disk 5 Enabled
Enclosure 1 Disk 6 Enabled
Enclosure 1 Disk 7 Enabled
Enclosure 1 Disk 8 Enabled
Enclosure 1 Disk 9 Enabled
Enclosure 2 Disk 0 Enabled
Enclosure 2 Disk 1 Enabled
Enclosure 2 Disk 2 Enabled
Enclosure 2 Disk 3 Enabled
Enclosure 2 Disk 4 Enabled
Enclosure 2 Disk 5 Enabled
Enclosure 2 Disk 6 Enabled
Enclosure 2 Disk 7 Enabled

Read cache: DISABLED
Write cache: DISABLED
RAID Type: RAID0
Element Size: 64
Lun Capacity: 116753

LOGICAL UNIT NUMBER 2
Enclosure 2 Disk 8 Enabled
Enclosure 2 Disk 9 Enabled
Enclosure 3 Disk 0 Enabled
Enclosure 3 Disk 1 Enabled
Enclosure 3 Disk 2 Enabled
Enclosure 3 Disk 3 Enabled
Enclosure 3 Disk 4 Enabled
Enclosure 3 Disk 5 Enabled
Enclosure 3 Disk 6 Enabled
Enclosure 3 Disk 7 Enabled
Enclosure 3 Disk 8 Enabled
Enclosure 3 Disk 9 Enabled
Enclosure 4 Disk 0 Enabled
Enclosure 4 Disk 1 Enabled

Read cache: DISABLED
Write cache: DISABLED
RAID Type: RAID0
Element Size: 64
Lun Capacity: 116753

LOGICAL UNIT NUMBER 3
Enclosure 4 Disk 2 Enabled
Enclosure 4 Disk 3 Enabled
Enclosure 4 Disk 4 Enabled
Enclosure 4 Disk 5 Enabled
Enclosure 4 Disk 6 Enabled
Enclosure 4 Disk 7 Enabled
Enclosure 4 Disk 8 Enabled
Enclosure 4 Disk 9 Enabled
Enclosure 5 Disk 0 Enabled
Enclosure 5 Disk 1 Enabled
Enclosure 5 Disk 2 Enabled
Enclosure 5 Disk 3 Enabled
Enclosure 5 Disk 4 Enabled
Enclosure 5 Disk 5 Enabled

Read cache: DISABLED
Write cache: DISABLED
RAID Type: RAID0
Element Size: 64
Lun Capacity: 116753

LOGICAL UNIT NUMBER 4
Enclosure 5 Disk 6 Enabled
Enclosure 5 Disk 7 Enabled
Enclosure 5 Disk 8 Enabled
Enclosure 5 Disk 9 Enabled

Enclosure 6 Disk 0 Enabled
Enclosure 6 Disk 1 Enabled
Enclosure 6 Disk 2 Enabled
Enclosure 6 Disk 3 Enabled
Enclosure 6 Disk 4 Enabled
Enclosure 6 Disk 5 Enabled
Enclosure 6 Disk 6 Enabled
Enclosure 6 Disk 7 Enabled
Enclosure 6 Disk 8 Enabled
Enclosure 6 Disk 9 Enabled

Read cache: DISABLED
Write cache: DISABLED
RAID Type: RAID0
Element Size: 64
Lun Capacity: 116753

LOGICAL UNIT NUMBER 5
Enclosure 7 Disk 0 Enabled
Enclosure 7 Disk 2 Enabled
Enclosure 7 Disk 4 Enabled
Enclosure 7 Disk 1 Enabled
Enclosure 7 Disk 3 Enabled
Enclosure 7 Disk 5 Enabled

Read cache: DISABLED
Write cache: ENABLED
RAID Type: RAID1/0
Element Size: 64
Lun Capacity: 84681
ECHO is on.

==== SCSI4:0:1:0 ====

LOGICAL UNIT NUMBER 0
Enclosure 0 Disk 0 Enabled
Enclosure 0 Disk 1 Enabled
Enclosure 0 Disk 2 Enabled
Enclosure 0 Disk 3 Enabled
Enclosure 0 Disk 4 Enabled
Enclosure 0 Disk 5 Enabled
Enclosure 0 Disk 6 Enabled
Enclosure 0 Disk 7 Enabled
Enclosure 0 Disk 8 Enabled
Enclosure 0 Disk 9 Enabled
Enclosure 1 Disk 0 Enabled
Enclosure 1 Disk 1 Enabled
Enclosure 1 Disk 2 Enabled
Enclosure 1 Disk 3 Enabled

Read cache: DISABLED
Write cache: DISABLED
RAID Type: RAID0
Element Size: 64
Lun Capacity: 116753

LOGICAL UNIT NUMBER 1
Enclosure 1 Disk 4 Enabled
Enclosure 1 Disk 5 Enabled
Enclosure 1 Disk 6 Enabled
Enclosure 1 Disk 7 Enabled
Enclosure 1 Disk 8 Enabled
Enclosure 1 Disk 9 Enabled
Enclosure 2 Disk 0 Enabled
Enclosure 2 Disk 1 Enabled
Enclosure 2 Disk 2 Enabled
Enclosure 2 Disk 3 Enabled
Enclosure 2 Disk 4 Enabled
Enclosure 2 Disk 5 Enabled
Enclosure 2 Disk 6 Enabled
Enclosure 2 Disk 7 Enabled

Read cache: DISABLED
Write cache: DISABLED
RAID Type: RAID0
Element Size: 64

Lun Capacity: 116753

LOGICAL UNIT NUMBER 2

Enclosure 2 Disk 8 Enabled
Enclosure 2 Disk 9 Enabled
Enclosure 3 Disk 0 Enabled
Enclosure 3 Disk 1 Enabled
Enclosure 3 Disk 2 Enabled
Enclosure 3 Disk 3 Enabled
Enclosure 3 Disk 4 Enabled
Enclosure 3 Disk 5 Enabled
Enclosure 3 Disk 6 Enabled
Enclosure 3 Disk 7 Enabled
Enclosure 3 Disk 8 Enabled
Enclosure 3 Disk 9 Enabled
Enclosure 4 Disk 0 Enabled
Enclosure 4 Disk 1 Enabled
Read cache: DISABLED
Write cache: DISABLED
RAID Type: RAID0
Element Size: 64
Lun Capacity: 116753

LOGICAL UNIT NUMBER 3

Enclosure 4 Disk 2 Enabled
Enclosure 4 Disk 3 Enabled
Enclosure 4 Disk 4 Enabled
Enclosure 4 Disk 5 Enabled
Enclosure 4 Disk 6 Enabled
Enclosure 4 Disk 7 Enabled
Enclosure 4 Disk 8 Enabled
Enclosure 4 Disk 9 Enabled
Enclosure 5 Disk 0 Enabled
Enclosure 5 Disk 1 Enabled
Enclosure 5 Disk 2 Enabled
Enclosure 5 Disk 3 Enabled
Enclosure 5 Disk 4 Enabled
Enclosure 5 Disk 5 Enabled
Read cache: DISABLED
Write cache: DISABLED
RAID Type: RAID0
Element Size: 64
Lun Capacity: 116753

LOGICAL UNIT NUMBER 4

Enclosure 5 Disk 6 Enabled
Enclosure 5 Disk 7 Enabled
Enclosure 5 Disk 8 Enabled
Enclosure 5 Disk 9 Enabled
Enclosure 6 Disk 0 Enabled
Enclosure 6 Disk 1 Enabled
Enclosure 6 Disk 2 Enabled
Enclosure 6 Disk 3 Enabled
Enclosure 6 Disk 4 Enabled
Enclosure 6 Disk 5 Enabled
Enclosure 6 Disk 6 Enabled
Enclosure 6 Disk 7 Enabled
Enclosure 6 Disk 8 Enabled
Enclosure 6 Disk 9 Enabled
Read cache: DISABLED
Write cache: DISABLED
RAID Type: RAID0
Element Size: 64
Lun Capacity: 116753

LOGICAL UNIT NUMBER 5

Enclosure 7 Disk 0 Enabled
Enclosure 7 Disk 1 Enabled
Enclosure 7 Disk 2 Enabled
Enclosure 7 Disk 3 Enabled
Enclosure 7 Disk 4 Enabled

Enclosure 7 Disk 5 Enabled
Enclosure 7 Disk 6 Enabled
Enclosure 7 Disk 7 Enabled
Enclosure 7 Disk 8 Enabled
Enclosure 7 Disk 9 Enabled
Read cache: DISABLED
Write cache: DISABLED
RAID Type: RAID5
Element Size: 64
Lun Capacity: 75055
ECHO is on.

==== SCSI4:0:3:0 ====

LOGICAL UNIT NUMBER 0

Enclosure 0 Disk 0 Enabled
Enclosure 0 Disk 1 Enabled
Enclosure 0 Disk 2 Enabled
Enclosure 0 Disk 3 Enabled
Enclosure 0 Disk 4 Enabled
Enclosure 0 Disk 5 Enabled
Enclosure 0 Disk 6 Enabled
Enclosure 0 Disk 7 Enabled
Enclosure 0 Disk 8 Enabled
Enclosure 0 Disk 9 Enabled
Enclosure 1 Disk 0 Enabled
Enclosure 1 Disk 1 Enabled
Enclosure 1 Disk 2 Enabled
Enclosure 1 Disk 3 Enabled
Read cache: DISABLED
Write cache: DISABLED
RAID Type: RAID0
Element Size: 64
Lun Capacity: 116753

LOGICAL UNIT NUMBER 1

Enclosure 1 Disk 4 Enabled
Enclosure 1 Disk 5 Enabled
Enclosure 1 Disk 6 Enabled
Enclosure 1 Disk 7 Enabled
Enclosure 1 Disk 8 Enabled
Enclosure 1 Disk 9 Enabled
Enclosure 2 Disk 0 Enabled
Enclosure 2 Disk 1 Enabled
Enclosure 2 Disk 2 Enabled
Enclosure 2 Disk 3 Enabled
Enclosure 2 Disk 4 Enabled
Enclosure 2 Disk 5 Enabled
Enclosure 2 Disk 6 Enabled
Enclosure 2 Disk 7 Enabled
Read cache: DISABLED
Write cache: DISABLED
RAID Type: RAID0
Element Size: 64
Lun Capacity: 116753

LOGICAL UNIT NUMBER 2

Enclosure 2 Disk 8 Enabled
Enclosure 2 Disk 9 Enabled
Enclosure 3 Disk 0 Enabled
Enclosure 3 Disk 1 Enabled
Enclosure 3 Disk 2 Enabled
Enclosure 3 Disk 3 Enabled
Enclosure 3 Disk 4 Enabled
Enclosure 3 Disk 5 Enabled
Enclosure 3 Disk 6 Enabled
Enclosure 3 Disk 7 Enabled
Enclosure 3 Disk 8 Enabled
Enclosure 3 Disk 9 Enabled
Enclosure 4 Disk 0 Enabled
Enclosure 4 Disk 1 Enabled
Read cache: DISABLED

Write cache: DISABLED
RAID Type: RAID0
Element Size: 64
Lun Capacity: 116753

LOGICAL UNIT NUMBER 3

Enclosure 4 Disk 2 Enabled
Enclosure 4 Disk 3 Enabled
Enclosure 4 Disk 4 Enabled
Enclosure 4 Disk 5 Enabled
Enclosure 4 Disk 6 Enabled
Enclosure 4 Disk 7 Enabled
Enclosure 4 Disk 8 Enabled
Enclosure 4 Disk 9 Enabled
Enclosure 5 Disk 0 Enabled
Enclosure 5 Disk 1 Enabled
Enclosure 5 Disk 2 Enabled
Enclosure 5 Disk 3 Enabled
Enclosure 5 Disk 4 Enabled
Enclosure 5 Disk 5 Enabled
Read cache: DISABLED
Write cache: DISABLED
RAID Type: RAID0
Element Size: 64
Lun Capacity: 116753

LOGICAL UNIT NUMBER 4

Enclosure 5 Disk 6 Enabled
Enclosure 5 Disk 7 Enabled
Enclosure 5 Disk 8 Enabled
Enclosure 5 Disk 9 Enabled
Enclosure 6 Disk 0 Enabled
Enclosure 6 Disk 1 Enabled
Enclosure 6 Disk 2 Enabled
Enclosure 6 Disk 3 Enabled
Enclosure 6 Disk 4 Enabled
Enclosure 6 Disk 5 Enabled
Enclosure 6 Disk 6 Enabled
Enclosure 6 Disk 7 Enabled
Enclosure 6 Disk 8 Enabled
Enclosure 6 Disk 9 Enabled
Read cache: DISABLED
Write cache: DISABLED
RAID Type: RAID0
Element Size: 64
Lun Capacity: 116753

LOGICAL UNIT NUMBER 5

Enclosure 7 Disk 0 Enabled
Enclosure 7 Disk 2 Enabled
Enclosure 7 Disk 4 Enabled
Enclosure 7 Disk 1 Enabled
Enclosure 7 Disk 3 Enabled
Enclosure 7 Disk 5 Enabled
Read cache: DISABLED
Write cache: ENABLED
RAID Type: RAID1/0
Element Size: 64
Lun Capacity: 84681
ECHO is on.

==== SCSI4:0:5:0 ====

LOGICAL UNIT NUMBER 0

Enclosure 0 Disk 0 Enabled
Enclosure 0 Disk 1 Enabled
Enclosure 0 Disk 2 Enabled
Enclosure 0 Disk 3 Enabled
Enclosure 0 Disk 4 Enabled
Enclosure 0 Disk 5 Enabled
Enclosure 0 Disk 6 Enabled
Enclosure 0 Disk 7 Enabled

Enclosure 0 Disk 8 Enabled
Enclosure 0 Disk 9 Enabled
Enclosure 1 Disk 0 Enabled
Enclosure 1 Disk 1 Enabled
Enclosure 1 Disk 2 Enabled
Enclosure 1 Disk 3 Enabled
Read cache: DISABLED
Write cache: DISABLED
RAID Type: RAID0
Element Size: 64
Lun Capacity: 116753

LOGICAL UNIT NUMBER 1

Enclosure 1 Disk 4 Enabled
Enclosure 1 Disk 5 Enabled
Enclosure 1 Disk 6 Enabled
Enclosure 1 Disk 7 Enabled
Enclosure 1 Disk 8 Enabled
Enclosure 1 Disk 9 Enabled
Enclosure 2 Disk 0 Enabled
Enclosure 2 Disk 1 Enabled
Enclosure 2 Disk 2 Enabled
Enclosure 2 Disk 3 Enabled
Enclosure 2 Disk 4 Enabled
Enclosure 2 Disk 5 Enabled
Enclosure 2 Disk 6 Enabled
Enclosure 2 Disk 7 Enabled
Read cache: DISABLED
Write cache: DISABLED
RAID Type: RAID0
Element Size: 64
Lun Capacity: 116753

LOGICAL UNIT NUMBER 2

Enclosure 2 Disk 8 Enabled
Enclosure 2 Disk 9 Enabled
Enclosure 3 Disk 0 Enabled
Enclosure 3 Disk 1 Enabled
Enclosure 3 Disk 2 Enabled
Enclosure 3 Disk 3 Enabled
Enclosure 3 Disk 4 Enabled
Enclosure 3 Disk 5 Enabled
Enclosure 3 Disk 6 Enabled
Enclosure 3 Disk 7 Enabled
Enclosure 3 Disk 8 Enabled
Enclosure 3 Disk 9 Enabled
Enclosure 4 Disk 0 Enabled
Enclosure 4 Disk 1 Enabled
Read cache: DISABLED
Write cache: DISABLED
RAID Type: RAID0
Element Size: 64
Lun Capacity: 116753

LOGICAL UNIT NUMBER 3

Enclosure 4 Disk 2 Enabled
Enclosure 4 Disk 3 Enabled
Enclosure 4 Disk 4 Enabled
Enclosure 4 Disk 5 Enabled
Enclosure 4 Disk 6 Enabled
Enclosure 4 Disk 7 Enabled
Enclosure 4 Disk 8 Enabled
Enclosure 4 Disk 9 Enabled
Enclosure 5 Disk 0 Enabled
Enclosure 5 Disk 1 Enabled
Enclosure 5 Disk 2 Enabled
Enclosure 5 Disk 3 Enabled
Enclosure 5 Disk 4 Enabled
Enclosure 5 Disk 5 Enabled
Read cache: DISABLED
Write cache: DISABLED

RAID Type: RAID0
Element Size: 64
Lun Capacity: 116753

LOGICAL UNIT NUMBER 4

Enclosure 5 Disk 6 Enabled
Enclosure 5 Disk 7 Enabled
Enclosure 5 Disk 8 Enabled
Enclosure 5 Disk 9 Enabled
Enclosure 6 Disk 0 Enabled
Enclosure 6 Disk 1 Enabled
Enclosure 6 Disk 2 Enabled
Enclosure 6 Disk 3 Enabled
Enclosure 6 Disk 4 Enabled
Enclosure 6 Disk 5 Enabled
Enclosure 6 Disk 6 Enabled
Enclosure 6 Disk 7 Enabled
Enclosure 6 Disk 8 Enabled
Enclosure 6 Disk 9 Enabled
Read cache: DISABLED
Write cache: DISABLED
RAID Type: RAID5
Element Size: 64
Lun Capacity: 108413

LOGICAL UNIT NUMBER 5

Enclosure 7 Disk 0 Enabled
Enclosure 7 Disk 1 Enabled
Read cache: DISABLED
Write cache: ENABLED
RAID Type: RAID1
Element Size: N/A
Lun Capacity: 8339

LOGICAL UNIT NUMBER 6

Enclosure 7 Disk 2 Enabled
Enclosure 7 Disk 3 Enabled
Read cache: DISABLED
Write cache: ENABLED
RAID Type: RAID1
Element Size: N/A
Lun Capacity: 8339

LOGICAL UNIT NUMBER 7

Enclosure 7 Disk 4 Enabled
Enclosure 7 Disk 5 Enabled
Enclosure 7 Disk 6 Enabled
Enclosure 7 Disk 7 Enabled
Enclosure 7 Disk 8 Enabled
Enclosure 7 Disk 9 Enabled
Read cache: DISABLED
Write cache: DISABLED
RAID Type: RAID0
Element Size: 64
Lun Capacity: 50037
ECHO is on.

==== SCSI5:0:1:0 =====

LOGICAL UNIT NUMBER 0

Enclosure 0 Disk 0 Enabled
Enclosure 0 Disk 1 Enabled
Enclosure 0 Disk 2 Enabled
Enclosure 0 Disk 3 Enabled
Enclosure 0 Disk 4 Enabled
Enclosure 0 Disk 5 Enabled
Enclosure 0 Disk 6 Enabled
Enclosure 0 Disk 7 Enabled
Enclosure 0 Disk 8 Enabled
Enclosure 0 Disk 9 Enabled
Enclosure 1 Disk 0 Enabled
Enclosure 1 Disk 1 Enabled

Enclosure 1 Disk 2 Enabled
Enclosure 1 Disk 3 Enabled
Read cache: DISABLED
Write cache: DISABLED
RAID Type: RAID0
Element Size: 64
Lun Capacity: 116753

LOGICAL UNIT NUMBER 1

Enclosure 1 Disk 4 Enabled
Enclosure 1 Disk 5 Enabled
Enclosure 1 Disk 6 Enabled
Enclosure 1 Disk 7 Enabled
Enclosure 1 Disk 8 Enabled
Enclosure 1 Disk 9 Enabled
Enclosure 2 Disk 0 Enabled
Enclosure 2 Disk 1 Enabled
Enclosure 2 Disk 2 Enabled
Enclosure 2 Disk 3 Enabled
Enclosure 2 Disk 4 Enabled
Enclosure 2 Disk 5 Enabled
Enclosure 2 Disk 6 Enabled
Enclosure 2 Disk 7 Enabled
Read cache: DISABLED
Write cache: DISABLED
RAID Type: RAID0
Element Size: 64
Lun Capacity: 116753

LOGICAL UNIT NUMBER 2

Enclosure 2 Disk 8 Enabled
Enclosure 2 Disk 9 Enabled
Enclosure 3 Disk 0 Enabled
Enclosure 3 Disk 1 Enabled
Enclosure 3 Disk 2 Enabled
Enclosure 3 Disk 3 Enabled
Enclosure 3 Disk 4 Enabled
Enclosure 3 Disk 5 Enabled
Enclosure 3 Disk 6 Enabled
Enclosure 3 Disk 7 Enabled
Enclosure 3 Disk 8 Enabled
Enclosure 3 Disk 9 Enabled
Enclosure 4 Disk 0 Enabled
Enclosure 4 Disk 1 Enabled
Read cache: DISABLED
Write cache: DISABLED
RAID Type: RAID0
Element Size: 64
Lun Capacity: 116753

LOGICAL UNIT NUMBER 3

Enclosure 4 Disk 2 Enabled
Enclosure 4 Disk 3 Enabled
Enclosure 4 Disk 4 Enabled
Enclosure 4 Disk 5 Enabled
Enclosure 4 Disk 6 Enabled
Enclosure 4 Disk 7 Enabled
Enclosure 4 Disk 8 Enabled
Enclosure 4 Disk 9 Enabled
Enclosure 5 Disk 0 Enabled
Enclosure 5 Disk 1 Enabled
Enclosure 5 Disk 2 Enabled
Enclosure 5 Disk 3 Enabled
Enclosure 5 Disk 4 Enabled
Enclosure 5 Disk 5 Enabled
Read cache: DISABLED
Write cache: DISABLED
RAID Type: RAID0
Element Size: 64
Lun Capacity: 116753

LOGICAL UNIT NUMBER 4
 Enclosure 5 Disk 6 Enabled
 Enclosure 5 Disk 7 Enabled
 Enclosure 5 Disk 8 Enabled
 Enclosure 5 Disk 9 Enabled
 Enclosure 6 Disk 0 Enabled
 Enclosure 6 Disk 1 Enabled
 Enclosure 6 Disk 2 Enabled
 Enclosure 6 Disk 3 Enabled
 Enclosure 6 Disk 4 Enabled
 Enclosure 6 Disk 5 Enabled
 Enclosure 6 Disk 6 Enabled
 Enclosure 6 Disk 7 Enabled
 Enclosure 6 Disk 8 Enabled
 Enclosure 6 Disk 9 Enabled
 Read cache: DISABLED
 Write cache: DISABLED
 RAID Type: RAID0
 Element Size: 64
 Lun Capacity: 116753

LOGICAL UNIT NUMBER 5
 Enclosure 7 Disk 0 Enabled
 Enclosure 7 Disk 1 Enabled
 Enclosure 7 Disk 2 Enabled
 Enclosure 7 Disk 3 Enabled
 Enclosure 7 Disk 4 Enabled
 Enclosure 7 Disk 5 Enabled
 Enclosure 7 Disk 6 Enabled
 Enclosure 7 Disk 7 Enabled
 Enclosure 7 Disk 8 Enabled
 Enclosure 7 Disk 9 Enabled
 Read cache: DISABLED
 Write cache: DISABLED
 RAID Type: RAID5
 Element Size: 64
 Lun Capacity: 75055
 ECHO is on.

==== SCSI5:0:3:0 =====

LOGICAL UNIT NUMBER 0
 Enclosure 0 Disk 0 Enabled
 Enclosure 0 Disk 1 Enabled
 Enclosure 0 Disk 2 Enabled
 Enclosure 0 Disk 3 Enabled
 Enclosure 0 Disk 4 Enabled
 Enclosure 0 Disk 5 Enabled
 Enclosure 0 Disk 6 Enabled
 Enclosure 0 Disk 7 Enabled
 Enclosure 0 Disk 8 Enabled
 Enclosure 0 Disk 9 Enabled
 Enclosure 1 Disk 0 Enabled
 Enclosure 1 Disk 1 Enabled
 Enclosure 1 Disk 2 Enabled
 Enclosure 1 Disk 3 Enabled
 Read cache: DISABLED
 Write cache: DISABLED
 RAID Type: RAID0
 Element Size: 64
 Lun Capacity: 116753

LOGICAL UNIT NUMBER 1
 Enclosure 1 Disk 4 Enabled
 Enclosure 1 Disk 5 Enabled
 Enclosure 1 Disk 6 Enabled
 Enclosure 1 Disk 7 Enabled
 Enclosure 1 Disk 8 Enabled
 Enclosure 1 Disk 9 Enabled
 Enclosure 2 Disk 0 Enabled
 Enclosure 2 Disk 1 Enabled
 Enclosure 2 Disk 2 Enabled

Enclosure 2 Disk 3 Enabled
 Enclosure 2 Disk 4 Enabled
 Enclosure 2 Disk 5 Enabled
 Enclosure 2 Disk 6 Enabled
 Enclosure 2 Disk 7 Enabled
 Read cache: DISABLED
 Write cache: DISABLED
 RAID Type: RAID0
 Element Size: 64
 Lun Capacity: 116753

LOGICAL UNIT NUMBER 2
 Enclosure 2 Disk 8 Enabled
 Enclosure 2 Disk 9 Enabled
 Enclosure 3 Disk 0 Enabled
 Enclosure 3 Disk 1 Enabled
 Enclosure 3 Disk 2 Enabled
 Enclosure 3 Disk 3 Enabled
 Enclosure 3 Disk 4 Enabled
 Enclosure 3 Disk 5 Enabled
 Enclosure 3 Disk 6 Enabled
 Enclosure 3 Disk 7 Enabled
 Enclosure 3 Disk 8 Enabled
 Enclosure 3 Disk 9 Enabled
 Enclosure 4 Disk 0 Enabled
 Enclosure 4 Disk 1 Enabled
 Read cache: DISABLED
 Write cache: DISABLED
 RAID Type: RAID0
 Element Size: 64
 Lun Capacity: 116753

LOGICAL UNIT NUMBER 3
 Enclosure 4 Disk 2 Enabled
 Enclosure 4 Disk 3 Enabled
 Enclosure 4 Disk 4 Enabled
 Enclosure 4 Disk 5 Enabled
 Enclosure 4 Disk 6 Enabled
 Enclosure 4 Disk 7 Enabled
 Enclosure 4 Disk 8 Enabled
 Enclosure 4 Disk 9 Enabled
 Enclosure 5 Disk 0 Enabled
 Enclosure 5 Disk 1 Enabled
 Enclosure 5 Disk 2 Enabled
 Enclosure 5 Disk 3 Enabled
 Enclosure 5 Disk 4 Enabled
 Enclosure 5 Disk 5 Enabled
 Read cache: DISABLED
 Write cache: DISABLED
 RAID Type: RAID0
 Element Size: 64
 Lun Capacity: 116753

LOGICAL UNIT NUMBER 4
 Enclosure 5 Disk 6 Enabled
 Enclosure 5 Disk 7 Enabled
 Enclosure 5 Disk 8 Enabled
 Enclosure 5 Disk 9 Enabled
 Enclosure 6 Disk 0 Enabled
 Enclosure 6 Disk 1 Enabled
 Enclosure 6 Disk 2 Enabled
 Enclosure 6 Disk 3 Enabled
 Enclosure 6 Disk 4 Enabled
 Enclosure 6 Disk 5 Enabled
 Enclosure 6 Disk 6 Enabled
 Enclosure 6 Disk 7 Enabled
 Enclosure 6 Disk 8 Enabled
 Enclosure 6 Disk 9 Enabled
 Read cache: DISABLED
 Write cache: DISABLED
 RAID Type: RAID0

Element Size: 64
 Lun Capacity: 116753

LOGICAL UNIT NUMBER 5
 Enclosure 7 Disk 0 Enabled
 Enclosure 7 Disk 2 Enabled
 Enclosure 7 Disk 4 Enabled
 Enclosure 7 Disk 1 Enabled
 Enclosure 7 Disk 3 Enabled
 Enclosure 7 Disk 5 Enabled
 Read cache: DISABLED
 Write cache: ENABLED
 RAID Type: RAID1/0
 Element Size: 64
 Lun Capacity: 84681

Appendix D : Space Calculation

180 Day Space

Note : Numbers are in Blocks(2KBytes) unless otherwise specified									
Warehouses	4800	tpmC	50208.43	tpmC/W	10.46	Total Space			
Table	Rows	Data	Index	5% Space	8H Space	OVERSIZE			
Warehouse	4,800	5,182	2,150	367		7,699			
District	48,000	48,300	4,096	2,620		55,016			
Item	100,000	6,768	2,496	463		9,727			
New-order	43,200,000	368,640	0	18,432		387,072			
History	144,000,000	4,321,280	0		723,216	5,044,496			
Orders	144,000,000	2,997,248	4,482,560		1,251,831	8,731,639			
Customer	144,000,000	72,585,600	5,555,200	3,907,040		82,047,840			
Order-line	1,440,025,184	56,885,760	0		9,520,482	66,406,242			
Stock	480,000,000	96,254,784	5,079,040	5,066,691		106,400,515			
Totals		233,473,562	15,125,542	8,995,613	11,495,529	269,090,246			
SEGMENT	BLOCKS	USED	REQUIRED w 5%	NEEDED with 8H	Over Head	OVERSIZE			
CUSTCLUSTER	87,490,560	72,585,600	76,214,880	76,214,880	762,149	11,275,680			
ICUST1	2,048,000	1,704,960	1,790,208	1,790,208	17,902	257,792			
ICUST2	5,898,240	3,850,240	4,042,752	4,042,752	40,428	1,855,488			
DISTCLUSTER	245,760	48,300	50,715	50,715	507	195,045			
IDIST	61,440	4,096	4,301	4,301	43	57,139			
HIST	13,025,280	4,321,280	4,321,280	5,044,496	50,445	7,980,784			
ITEMCLUSTER	245,760	6,768	7,106	7,106	71	238,654			
IITEM	20,480	2,496	2,621	2,621	26	17,859			
INORD	1,228,800	368,640	387,072	387,072	3,871	841,728			
INORDL	114,032,640	56,885,760	56,885,760	66,406,242	664,062	47,626,398			
ORDR	9,093,120	2,997,368	2,997,368	3,499,012	34,990	5,594,108			
ORDR1	5,406,720	1,689,600	1,689,600	1,972,374	19,724	3,434,346			
ORDR2	8,847,360	2,792,960	2,792,960	3,260,394	32,604	5,586,966			
STOKCLUSTER	115,507,200	96,254,784	101,067,523	101,067,523	1,010,675	14,439,677			
ISTOK	6,266,880	5,079,040	5,332,992	5,332,992	53,330	933,888			
WARECLUSTER	245,760	4,935	5,182	5,182	52	240,578			
IWARE	40,960	2,048	2,150	2,150	22	38,810			
SYSTEM	258,560	258,560	258,560	258,560	2,586	0			
ROLLBACK	32,740	32740	32,740	32,740	32740	0			
Totals	369,996,260	248,890,175	257,885,770	269,381,320	2,726,226	100,614,940			
Dynamic space	64,204,288	Sum of Data for Order, Order-Line and History (excluding free extents)							
Static space	196,116,655	Data + Index + 5% Space + Overhead - Dynamic space							
Free space	9,060,377	Total Seg. Size - Dynamic Space - Static Space - Not Needed							
Daily growth	10,745,322	(Dynamic space/W * 62.5)* tpmC							
Daily spread	(7,057,605)	Free space - 1.5 * Daily growth (Zero if negative)							
180 day	2,130,274,554	Static space + 180 (daily growth + daily spread)							
180 day (GB)	4063.18	Excludes OS, Paging and RDBMS Logs							
Log used (Byte)	34811230228	log file used during entire run							
Total N-O Txn	2551581	Total count of N-O transactions during entire run							
Log per N-O txn	13643.00	Number of Bytes per New-Order transaction							
8 Hour Log (GB)	306.22	Excludes mirrored							
OS (GB)	8.23	4 node each							
180 day space	Space Needed (GB)	Disk Size (GB)	Disks Priced	Priced Size (GB)					
	4,096.10	8.23	676	5,564.46					
Logs (with mirror)	612.44	16.63	40	665.08					

Appendix E : Price Quotation

Oracle

Monday 28 June 1999

Mr. Tsao Kemal
 NEC Corporation
 1-10, NISSHINCHU BLDG,
 TOKYO 113-8501, JAPAN

Dear Mr. Kemal

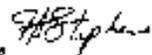
In response to your inquiry, here is the U.S. pricing and availability information for the specified configuration:

Description	Qty	Unit Price	Extended Price	Syr Maint. Price
Oracle RDBMS and SQL*Net	1	\$408,860	\$408,860	\$408,860
Oracle RDBMS with Oracle Parallel Server (OPS) and Partitioning options.				

Part number Oracle 8i level 8.1.6.0,
 Support level Business,
 Availability 30 November, 1999.

Yours sincerely,

Frank Stephens, Oracle
 Manager, NT Benchmark
 652 613 1490




22 June, 1999

Mr. Robin Gomi
 NEC Systems, Inc.
 Boxborough MA

Dear Mr. Gomi

Per your request I am enclosing the pricing information regarding TUXEDO 6.4 that you requested. This pricing applies to Tuxedo 6.1, 6.2, 6.3, 6.4 and 6.5. Please note that Tuxedo 6.5 is our most recent version of Tuxedo but that all 6.x releases are generally available.

Core functionality services pricing is appropriate for your activities. As per the table below NEC NT server systems are classified as either a Tier 1, Tier 2 or Tier 3 server depending on the CPU Capacity of the system. The NEC Express5800 HV8600 is a Tier 3 system. The NEC Express5800 HX4600 is a tier 2 system. The quote is valid for 90 days from the date of this offer.

Tuxedo Core Functionality Services (CFS) Program Product Pricing and Description

TUX-CFS provides a basic level of middleware support for distributed computing, and is best used by organizations with substantial resources and knowledge for advanced distributed computing implementations.

TUX-CFS prices are server only and are based on the overall performance characteristics of the server and uses the same five tier computer classification as TUXEDO 6.x. Prices range from \$3,000 for Tier 1 to \$250,000 for Tier 5. Under this pricing option EVERY system running TUX-CFS at the user's site must have a TUXEDO license installed and pay the appropriate per server license fees.

BEA Tux/CFS Unlimited User License Fees Per Server

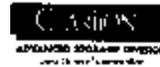
Unlimited User License fees per server	Number of Users	Dollar Amount	Maintenance (5 x 8) per year	Maintenance (7 x 24) per year
Tier 1 - PC Servers with 1 or 2 CPUs, entry level RISC Uniprocessor workstations and servers (Class 1 and Class 2)	Unlimited	\$3,000.00	\$480.00	\$580.00
Tier 2 - PC Servers with 3 or 4 CPUs, Midrange RISC Uniprocessor servers and workstations (Class 3)	Unlimited	\$12,000.00	\$1,800.00	\$2,640.00
Tier 3 - Midrange Multiprocessors up to 8 CPUs per system capacity (Class 4 and 5)	Unlimited	\$30,000.00	\$4,500.00	\$8,600.00
Tier 4 - Large (more than 8, less than 32 CPUs) and Mainframe	Unlimited	\$100,000.00	\$15,000.00	\$22,000.00

Systems (Class 6)				
Tier 5 - Massively Parallel Systems > 32 processors	Unlimited	\$250,000.00	\$37,500.00	\$55,000.00

Intel based server tier classifications:

Platform	Operating System	Tier 4 Class 1	Tier 1 Class 2	Tier 2 Class 3	Tier 3 Class 4	Tier 2 Class 5
Intel Pentium Pentium Pro PCs	Interactive R3.2 ESIX SVR 4.0 SCO UNIX 3.2.2 and 3.2.4 SCO DDT 2.x,3 x Solaris x86 2.X UnixWare, Windows NT 3.51.0	All 385/486 PCs are Class 1	ALL Pentium and Pentium Pro PCs with 1 or 2 CPUs capacity are Tier 1	ALL Pentium and Pentium Pro PCs with 3 or 4 CPUs capacity are Tier 2		ALL Pentium and Pentium Pro PCs with 5,6,7, or 8 CPUs Capacity are Tier 3

Very Truly Yours,


Lewis D. Brenlano,
Director, Market Planning

June 24, 1999

Yoshiko Tsutsi
General Manager
Workstations and Servers Division
1-7-1, Nishitokyo, Fuchu
Tokyo 127-8502, Japan

Dear Tom-Liam,

Below, please find the quotation for CASION Desk Array equipment requested by NEC for the TPC-C benchmark. This quotation includes hardware equipment and 5 year on-site maintenance. Please don't include on-site travel with any additional charges.

Storage

Name	Vendor	P/N	List Price	Qty	Subtotal	Quote	
DG Clarion PC5700	DG	C7851G-F78	\$8,500.00	10	\$85,000.00	\$39,000.00	
HA Model, C5701 10 drive	DG	C5701-H-A	\$80,800.00	12	\$728,000.00	\$435,600.00	
UPE							
HA Model, C5851 10 drive	DG	C5851D-A	\$8,200.00	64	\$664,200.00	\$318,720.00	
DAE							
Cabinet Mounting Kit for DAE	UG	C76803-A	\$100.00	80	\$8,000.00	\$3,720.00	
DPE							
8 GB 10Krpm HDD Option	DG	C8810FLG-A	\$1,850.00	740	\$1,368,000.00	\$821,400.00	
18GB 10Krpm HDD Option	DE	C1710FG-A	\$2,850.00	44	\$125,400.00	\$75,240.00	
10MKB Cache Option	DG	C775123-A	\$14,000.00	12	\$168,000.00	\$100,800.00	
Quad Rackmount SPS Option	DG	C7714G-A	\$4,200.00	22	\$92,400.00	\$50,840.00	
Single 9-port copper FC Hub	DG	C9404D-V1	\$1,400.00	6	\$8,400.00	\$15,840.00	
10m Cable	DG	154950203	\$200.00	22	\$4,400.00	\$2,640.00	
					Total:	\$3,072,000.00	\$1,843,200.00

The Quote above represents a discount of 40% off published list price.

		Premium Care Price	Premium Care Price	Standard Care Model	Standard Care Price
C5701R-A	HA, C5701 DDP Rackmount	B7CLDFPE-1	\$1,900	B7CLDFPE-2	\$1,200
C5851D-A	HA, C5851 DAE Rackmount	B7CLDFAE-1	\$750	B7CLDFAE-2	\$434
C7714G-A	Dual Rackmount SPS	N/A			
C8810FLG-A	8 GB HDD	N/A			

CT71DF-G	520 Dsk. DR				
	13.5in 1000	N/A			
	120 Rev. 10E				
CT7512G-A	20MB	N/A			
	Cache on 128				
	MB/D4				
CS30LD-V1	Single 9-Pin	ENCLOSURE	8740	BYCLDVAE 2	8190
	FC Hub				
154900013	1in. Cable	N/A			
CT761G-P1E	30U Rack	N/A			

Priority Care Service is 24/24 (6-4 hour response)
Standard Care Service is 24/8 (8-4 hour response)

Best regards and good luck.

Thomas J. Kelly
DEM Program Mgr. AQ
CLARSON

FROM PACKARD BELL INC

06/17/1999 09:16

P. 2

Unique
Technologies

Unique Technologies, Inc.
380 Middlesex Turnpike
Sudbury, MA 01803

phone 781.229.0047
fax 781.229.0031
 toll free 800.556.0025

www.unique-technologies.com

QUOTATION

June 17, 1999

Robin Goetz
NBC Computer Systems Division
1414 Massachusetts Avenue
Boxborough, MA 01719

Inside Sales: **Timmy Grazio**

Fax# 978-635-6304
Fax# 978-635-6288

and for your use - should you change items in quote

The Internet Computer Headquarters™ 1.800.COM & 1.800.HAWAII.COM

BOTTOM LINE TELECOMMUNICATIONS

BLT

581 Broadview Terrace
Sturbridge, CT 06156-4907
(800) 953-0111 Voice/FAX
sd@blt.com/jk@blt.com/Internet
http://www.bottom-line.com

QUOTE

Order #: 12401134
Invoice Date: 6/15/99
Account #: 002-000-00002
Card #:
Exp. Date:
Approval #:

Bill To:
QUOTE
QUOTE
QUOTE
Quote Number: 00000000

Ship To:
NBC COMPUTER SYSTEMS DIVISION
ROBIN GOETZ
SERVER WORK CENTER
FAX: 978-635-6304
EMAIL: R.ROBIN@NBC.COM

YOUR P.O.#	SALES REP:	SHIPPING VIA:	SHIP DATE:	TERMS:	TAX ID#:
	David J. Dunn	UPS Ground 3500 SLIPPING (CONT. CONT.) AL USA ONLY		Net 30, Prepay. or Wire Trade	

ITEM ID	DESCRIPTION	QTY.	UNIT PRICE	EXTENDED
8100	CATALYST 2512 NO. 14100 1200 14100 14100 14100 WHSE. 64 IN STOCK	1.00	\$1,245.00	\$1,245.00

Payment is expected when the line is first specified. We will assess a service charge of 1.6% per month, retroactive to the due date, on any late payment amounts. If you have any questions, please refer to the order number on our website at: www.bl.com

SUB TOTAL:	\$1,245.00
SHIPPING:	\$0.00
HANDLING:	\$2.00
TAX:	\$0.00
TOTAL:	\$1,245.00
REBURDEN:	\$0.00
BALANCE DUE:	\$1,245.00

Shipping is ONLY for payment method shown. Credit card pricing is as shown on the order. @ \$125.00, plus handling.

Thank you for your business!

Page 1 of 1

For your request, Unique Technologies is pleased to submit the following quotation:

MFR	PART	LEAD	LEAD-TIME
QLOGIC	QLA2100	1-99	STOCK
		\$774.00	

Robin, thank you for the opportunity to quote the above Qlogic cards. Please let me know if I can help you with anything else.

Best regards,
Timmy Grazio

Commercial Communications, Inc.

June 22, 1999

Mr. Robin Gomi
NEC
1414 Mass. Ave.
Boxborough, Ma. 01719

Dear Mr. Gomi,

Thank you for allowing Commercial Communications Inc. to quote on the Communications Components.

The cost of a Compaq JP1016C Hub is: \$16.75 each

This quote is based on a quantity of 3300
As requested

The above pricing does not include Shipping or Tax

- 1.) Pricing is based on information supplied.
- 2.) Above quote is good for 30 days.

Please feel free to call me if you have any questions.
Sincerely,

Phil Bateman
psmb



Submitted To:
Mr. Robin Gomi
NEC Computer Systems Division
1414 Massachusetts Ave
Boxborough, MA 01719
gomi.gom1@nec.com (617)78-695-6304

Quotation

Number: NECCSD-001
Date: 10/26/99

Alteon Networks is pleased to provide this price quotation for your convenience. Please feel free to ask for details on any of our products either directly to our US office or to your local Sales Manager.

Please address any questions to:
Craig Lum 770-867-6008
clum@alteon.com

LINE	QTY	DESCRIPTION	MODEL NO.	PRC	UNIT PRICE DISC PRICE	AMOUNT
1	4	ACERIC PCI	710011		\$885.00 \$885.00	\$3,540.00
2	1	Alteon 180	700104		\$14,896.00 \$14,896.00	\$14,896.00
3					\$0.00	\$0.00
4					\$0.00	\$0.00
5					\$0.00	\$0.00
TOTALS						\$18,436.00

Special Terms and Conditions

1. Equipment will be shipped upon receipt of a valid purchase order and approved credit.
2. Standard lead time is 10 business days from receipt of order.

Validity: 30 days from date issued unless otherwise noted above.
Terms: 100 Net/Net in US/CA. Net 30 on approved credit.
Shipping: Shipping charges apply unless otherwise noted and will be billed on shipment.
Payment: US dollars.

Submitted by: Craig Lum
Alteon Networks, Inc.