



TPC BenchmarkTMC

Full Disclosure Report

*Fujitsu
PRIMEQUEST 580A c/s
W/ 99 Front-Ends*

running

*Oracle Database 10g Release 2
Enterprise Edition with Partitioning*

December 4, 2008

First Edition - December 2008

The benchmark results contained in this document were submitted for compliance with version 5.10 of the TPC Benchmark C Standard Specification. The result of that action is to place these benchmark results into the sixty day “under review” status as of December 2008.

Fujitsu believes that the information in this document is accurate as of the publication date. The information in this document is subject to change without notice. Fujitsu 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, Fujitsu 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. Fujitsu does not warrant or represent that a user can or will achieve similar performance expressed in transactions per minute (tpmC) or normalized price/performance (\$/tpmC). No warranty of system performance or price/performance is expressed or implied in this report.

Copyright (C) 2008 Fujitsu Limited. 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, December 2008.

Fujitsu and PRIMEQUEST are trademarks or registered trademarks of Fujitsu Limited.

PRIMERGY is a registered trademark of Fujitsu-Siemens Computers GmbH.

ORACLE, SQL*DBA, SQL*Loader, SQL*net, SQL*Plus, Oracle10g, Pro*c and PL/SQL are trademarks of Oracle Corporation.


Intel, Pentium, XEON and Itanium2 are trademarks or registered trademarks of Intel Corporation.

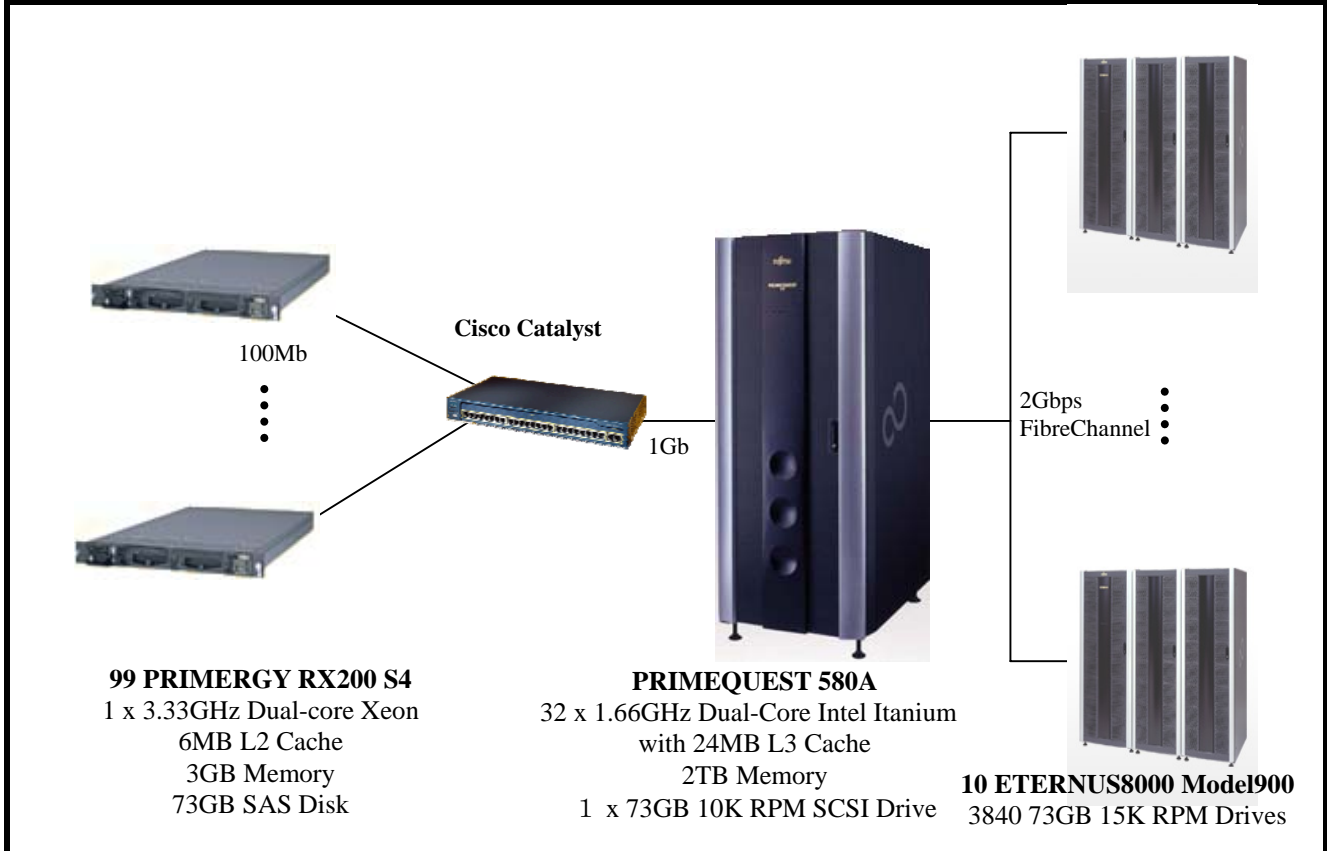
Linux is a registered trademarks of Linus Torvalds.

Red Hat is a registered trademarks of Red Hat, Inc.

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

All other brand or product names mentioned herein are trademarks or registered trademarks of their respective owners.

	PRIMEQUEST 580A c/s w/99 Front-Ends			TPC-C Rev 5.10 TPC pricing 1.3.0
				Report Date: December 4, 2008
Total System Cost	TPC-C Throughput	Price/Performance	Availability Date	
\$8,950,005 USD	2,382,032 tpmC	\$3.76 USD/tpmC	December 4, 2008	
Database Server Processors/Cores/Threads	Database Manager	Operating system	Other Software	Number of users
32/64/128 Intel Itanium 1.66GHz	Oracle Database 10g Release 2 Enterprise Edition with Partitioning	Red Hat Enterprise Linux 4 AS	Tuxedo 8.1	1,936,000



System Component	Qty	Server:	Qty	Each of 99 Clients:
Processors/Cores/Threads	32/64/128	1.66GHz Dual-Core Intel Itanium 9150M	1/2/2	3.33GHz Intel Dual-core Xeon
Cache Memory		24MB L3 Cache		6MB L2 Cache
Memory	64	32GB (4 x 8GB DDR2-400)	3	1GB(2 x 512MB PC2-5300F) 2GB(2 x 1024MB PC2-5300F)
Disk Controllers	40	4G bps FibreChannel (used at 2Gbps)	1	Serial Attached SCSI (SAS)
Disk Drives	1 3840	73GB 10K rpm 73GB 15K rpm	1	73 GB 15K rpm
Total Storage		280,393 GB		7,227 GB



**PRIMEQUEST 580A c/s
w/99 Front-Ends**

TPC-C Rev 5.10
TPC Pricing 1.3.0
Report Date: December 4, 2008

Description	Part Number	Third Party Brand Pricing	Unit Price	Qty	Extended Price	3Year Maint. Price
Server Hardware						
PRIMEQUEST 580A Base Unit	MC5COP311U		1	150,000.00	150,000.00	456,768.00
System Board	MC-97SB21		1	18,000.00	18,000.00	
CPU Module(Dual core Itanium 9150M/1.66GHz/24MB L3/6	MC-01HA11		1	32,560.00	32,560.00	
32GB Memory Module (4x8GB DDR2-400)	MC-02A611		1	77,440.00	77,440.00	
I/O Unit	MC-97UX11		1	15,000.00	15,000.00	
BMC Module	MC-97BM11		1	1,720.00	1,720.00	
Disk Drive Unit (3.5inch, 73GB, 10,000rpm, Ultra320)	MC-03D321		1	680.00	680.00	
Gigabit Switch Board (w/ 8 external 1000Base-T ports)	MC-87GE11		1	11,850.00	11,850.00	
Additional Power Supply	MC-57PS21U		1	7,600.00	7,600.00	
External I/O Cabinet	MC-87RK11U		1	20,850.00	20,850.00	
PCI-Box	MC-07PB21U		1	20,700.00	20,700.00	
PCI Unit	MC-07PU31		1	4,250.00	4,250.00	
PCI Cassette	MC-07PC11		1	370.00	370.00	
PCI Unit Cable (5m)	MC-07CA11		1	890.00	890.00	
FibreChannel Card (4Gbps, PCI-X, dual port)	MC-08FC41		1	4,270.00	4,270.00	
FibreChannel Cable (15m, LC-LC)	CBL-MLLB15		1	330.00	330.00	
17" Flat Panel Display	96-343680-000		1	520.00	520.00	
USB Keyboard W/Trackball	MC-07KB51		1	600.00	600.00	
Server Hardware Subtotals					6,694,330.00	456,768.00
Storage						
ETERNUS8000 Model900 Base Unit w/ 2 Controllers, 4 Drive Enclosures	E890S20AU		1	180,000.00	180,000.00	1,237,908.00
Additional Expansion Rack	E800CR1U		1	12,000.00	12,000.00	
Additional Controllers (2 sets)	E800CJ01U		1	40,000.00	40,000.00	
Cache Memory (2GBx4)	E800CM41U		1	14,000.00	14,000.00	
FibreChannel Host Interfaces (4Gbps, dual port, 2 sets)	E800CH14U		1	12,800.00	12,800.00	
Drive Enclosures for Base Unit (4 sets)	E800CE11U		1	39,400.00	39,400.00	
Drive Enclosures for Expansion Rack (4 sets)	E800CE21U		1	31,000.00	31,000.00	
Disk Drive Unit (73GB, 15,000rpm)	E800CA3U		1	650.00	650.00	
Storage Subtotals					7,754,400.00	1,237,908.00
Server Software						
Red Hat Enterprise Linux 4 AS (for Intel Itanium)	PMK0348US		1	7,497.00	7,497.00	5,130.00
Oracle Database 10g Enterprise Edition, Unlimited Users, Per Processor, 3 years			2	23,750.00	47,500.00	
Partitioning, Unlimited Users, Per Processor, 3 years			2	5,750.00	11,500.00	
Oracle Database Server Support Package for 3 years			2	2,300.00	4,600.00	6,900.00
Server Software Subtotals					951,497.00	12,030.00
Client Hardware						
PRIMERGY RX200 S4	S26361-K1167-V101		1	1,174.84	1,174.84	93,555.00
Xeon DP X5260 3.33 GHz 6MB 1333MHz	S26361-F3881-E333		1	1,422.00	1,422.00	
1GB Base Memory (2 x 512MB PC2-5300F)	S26361-F3263-E521		1	109.36	109.36	
2GB Memory Module (2 x 1024MB PC-5300F)	S26361-F3263-E522		1	149.42	149.42	
Drive Bay (2 x 3.5" HDD slots)	S26361-F3317-E400		1	45.00	45.00	
Hard Disk Drive (3.5inch, 73GB, 15,000rpm Hot Plug, SAS ;	S26361-F3204-E573		1	227.00	227.00	
RAID 0/1 SAS based on LSI MegaRAID 4Port	S26361-F3257-E4		1	127.00	127.00	
Internal CD-RW / DVD Unit	S26361-F3268-E1		1	68.40	68.40	
USB FDD	S26361-F2382-L20		1	42.34	42.34	
PRIMECENTER Rack 19inch Rack (24U)	S26361-K826-V102		1	2,011.01	2,011.01	
Dummy panel, plastics, 2U + assembly	S26361-F2735-E131		1	16.00	16.00	
RackMountKit 1-2U servers	S26361-F2735-E110		1	67.00	67.00	
Cable Powercord (USA) 1.8m, grey	T26139-Y1742-E10		1	11.00	11.00	
LCDRC23 17" TFT	S26361-K1023-V200		1	1,716.00	1,716.00	
KVM Switch (8ports, 1U)	S26361-F2293-E801		1	708.00	708.00	
Console Switch adapter KVM S2 PS/2-VGA	S26361-F2293-L201		1	110.00	110.00	
KVM Cable (1.8m)	S26361-F2293-L20		1	10.80	10.80	
Client Hardware Subtotals					378,145.57	93,555.00



**PRIMEQUEST 580A c/s
w/99 Front-Ends**

TPC-C Rev 5.10
TPC Pricing 1.3.0
Report Date: December 4, 2008

Description	Part Number	Third Party Brand Pricing	Unit Price	Qty	Extended Price	3Year Maint. Price
Client Software						
Red Hat Enterprise Linux 4 ES (for x86)	PMK0343US	1	672.00	99	66,528.00	0.00
Tuxedo CFS-R Tier 1		2	1,800.00	99	178,200.00	
Oracle Premium Support for 3 years		2	39,204.00	3		117,612.00
Client Software Subtotals					244,728.00	117,612.00
User Connectivity						
Cisco Catalyst 2950T-24 Switch		3	839.00	5	4,195.00	
Cisco SMARTnet 24x7x4 Maintenance		4		5		1,799.85
User Connectivity Subtotals					4,195.00	1,799.85
Total					16,027,295.57	1,919,672.85
Oracle Mandatory E-Business Discount		2			-311,678.00	
Large Configuration Discount and Support Prepayment		1			-8,025,026.05	-660,260.10
Total					7,690,592.00	1,259,413.00
Three-Year Cost of Ownership						\$8,950,005

Pricing Sources: 1 = Fujitsu , 2 = Oracle , 3 = getTnew.com , 4 = TechOnWeb.com
 Audited by: Francois Raab, InfoSizing, Inc. (www.sizing.com)
 Oracle pricing contact: MaryBeth Pierantoni, mary.beth.pierantoni@oracle.com, 916-315-5081
 *32=0.50 x 64. Explanation: For the purposes of counting the number of processors which require licensing, an Intel multicore chip with "n" cores shall be determined by multiplying "n" cores by a factor of 0.50.

Three-Year Cost of Ownership USD	\$8,950,005
tpmC	2,382,032
\$ USD / tpmC	\$3.76

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

**Numerical Quantities Summary for
PRIMEQUEST 580A c/s w/ 99 Front-Ends**

Oracle Database 10g Release 2 Enterprise Edition with Partitioning

MQTH, Computed Maximum Qualified Throughput **2,382,032 tpmC**

Response Times (in seconds)	Average	90th %	Maximum
New-Order	0.835	1.597	20.834
Payment	0.821	1.586	20.815
Order-Status	0.828	1.593	20.737
Delivery (interactive portion)	0.103	0.104	0.328
Delivery (deferred portion)	0.726	1.491	20.472
Stock-Level	0.811	1.576	20.788
Menu	0.103	0.104	0.706

Transaction Mix, in percent of total transaction

New-Order	44.94%
Payment	43.01%
Order-Status	4.02%
Delivery(interactive)	4.02%
Stock-Level	4.01%

Emulation Delay (in seconds)

	Response Time	Menu
New-Order	0.1	0.1
Payment	0.1	0.1
Order-Status	0.1	0.1
Delivery (interactive)	0.1	0.1
Stock-Level	0.1	0.1

Keying/Think Times (in seconds)

	Keying Time			Think Time		
	Min	Avg	Max	Min	Avg	Max
New-Order	18.002	18.012	18.035	0.000	12.015	120.202
Payment	3.002	3.012	3.036	0.000	12.016	120.201
Order-Status	2.003	2.012	2.030	0.000	10.013	100.193
Delivery (interactive)	2.003	2.012	2.035	0.000	5.017	50.200
Stock-Level	2.003	2.012	2.031	0.000	5.017	50.199

Test Duration

Ramp-up time	171 Min. 5 Sec.
Measurement interval	120 minutes
Transactions during measurement interval(all types)	636,339,216

Checkpointing

Number of checkpoints	4
Checkpoint interval (seconds)	1,777

Table Of Contents

PREFACE

GENERAL ITEMS	4
0.1 APPLICATION CODE AND DEFINITION STATEMENTS	4
0.2 TEST SPONSOR.....	4
0.3 PARAMETER SETTINGS.....	4
0.4 CONFIGURATION DIAGRAMS.....	5
CLAUSE 1: LOGICAL DATA BASE DESIGN RELATED ITEMS	7
1.1 TABLE DEFINITIONS	7
1.2 PHYSICAL ORGANIZATION OF DATABASE	7
1.3 INSERT AND DELETE OPERATIONS	7
1.4 PARTITIONING.....	7
1.5 REPLICATION, DUPLICATION OR ADDITIONS	8
CLAUSE 2: TRANSACTION AND TERMINAL PROFILES RELATED ITEMS	9
2.1 RANDOM NUMBER GENERATION	9
2.2 INPUT/OUTPUT SCREEN LAYOUT.....	9
2.3 PRICED TERMINAL FEATURE VERIFICATION	9
2.4 PRESENTATION MANAGER OR INTELLIGENT TERMINAL	9
2.5 TRANSACTION PROFILES	10
2.6 QUEUING MECHANISM.....	10
CLAUSE 3: TRANSACTION AND SYSTEM PROPERTIES RELATED ITEMS	11
3.1 TRANSACTION SYSTEM PROPERTIES (ACID).....	11
3.2 ATOMICITY.....	11
3.2.1 Completed Transactions.....	11
3.2.2 Aborted Transactions	12
3.3 CONSISTENCY	12
3.4 ISOLATION	12
3.5 DURABILITY.....	13
3.5.1 Loss of Log Disk	13
3.5.2 Loss of Data Disk	13
3.5.3 Instantaneous Interruption and Loss of Memory.....	14
CLAUSE 4: SCALING AND DATA BASE POPULATION RELATED ITEMS	15
4.1 INITIAL CARDINALITY OF TABLES.....	15
4.2 DATABASE LAYOUT	16
4.3 TYPE OF DATABASE.....	43
4.4 DATABASE MAPPING	43
4.5 60 DAY SPACE.....	43
CLAUSE 5: PERFORMANCE METRICS AND RESPONSE TIME RELATED ITEMS	44
5.1 THROUGHPUT.....	44
5.2 RESPONSE TIMES.....	44

5.3 KEYING AND THINK TIMES 45

5.4 RESPONSE TIME FREQUENCY DISTRIBUTION CURVES AND OTHER GRAPHS 45

5.5 STEADY STATE DETERMINATION 50

5.6 WORK PERFORMED DURING STEADY STATE 50

5.7 REPRODUCIBILITY 51

5.8 MEASUREMENT PERIOD DURATION 51

5.9 REGULATION OF TRANSACTION MIX 51

5.10 TRANSACTION STATISTICS 52

5.11 CHECKPOINT COUNT AND LOCATION..... 52

CLAUSE 6: SUT, DRIVER, AND COMMUNICATION DEFINITION RELATED ITEMS 53

6.1 RTE DESCRIPTIONS 53

6.2 LOSS OF TERMINAL CONNECTIONS..... 53

6.3 EMULATED COMPONENTS..... 53

6.4 FUNCTIONAL DIAGRAMS..... 53

6.5 NETWORKS..... 54

6.6 OPERATOR INTERVENTION 54

CLAUSE 7: PRICING RELATED ITEMS 55

7.1 HARDWARE AND SOFTWARE COMPONENTS 55

7.2 AVAILABILITY 55

7.3 THROUGHPUT AND PRICE PERFORMANCE..... 55

7.4 COUNTRY SPECIFIC PRICING 56

7.5 USAGE PRICING 56

7.6 SYSTEM PRICING 56

CLAUSE 9: AUDIT RELATED ITEMS 57

9.1 AUDITOR’S REPORT 57

9.2 AVAILABILITY OF THE FULL DISCLOSURE REPORT..... 57

APPENDIX A: CLIENT SOURCE CODE 58

APPENDIX B: SERVER SOURCE CODE..... 102

APPENDIX C: RTE SCRIPTS..... 147

APPENDIX D: SYSTEM TUNABLES..... 179

APPENDIX E: DATABASE CREATION CODE 266

APPENDIX F: 60 DAY SPACE CALCULATION 299

APPENDIX G: NUMERICAL QUANTITIES SUMMARY PER CLIENT 300

APPENDIX H: PRICE QUOTES 310

APPENDIX I: AUDITOR’S ATTESTATION LETTER..... 312

Abstract

Overview

This report documents the methodology and results of the TPC Benchmark C test conducted by Fujitsu Ltd. on the Fujitsu PRIMEQUEST 580A w/99 Front-Ends. The operating system and the DBMS used on the server were Red Hat Enterprise Linux 4 AS for Itanium Processor Family and Oracle Database 10g Release 2 Enterprise Edition with Partitioning.

The operating system on the clients was Red Hat Enterprise Linux 4 ES for x86.

Those clients ran Apache HTTP Server and Tuxedo 8.1 CFS-R.

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

TPC Benchmark C Metrics

The standard TPC Benchmark C metrics, tpmC (transactions per minute), price per tpmC (three year capital cost per measured tpmC), and the availability date are reported as:

2,382,032 tpmC
\$3.76 USD/tpmC
December 4, 2008

Standard and Executive Summary Statements

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

Auditor

The benchmark configuration, environment and methodology, along with the pricing model used to calculate the cost per tpmC, were audited by Francois Raab of InfoSizing to verify compliance with the relevant TPC specifications.

Preface

The TPC Benchmark C was developed by the Transaction Processing Performance Council (TPC). The TPC was founded to define transaction processing benchmarks and to disseminate objective, verifiable performance data to the industry. This full disclosure report is based on the TPC Benchmark C Standard Specifications Version 5.10.

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 throughput” measuring the number of orders processed per minute. Multiple transactions are used to simulate the business activity of processing an order, and each transaction is subject to a

response time constraint. The performance metric for this benchmark is expressed in transactions-per-minute-C (tpmC). To be compliant with the TPC-C standard, all references to tpmC results must include the tpmC rate, the associated price-per-tpmC, and the availability date of the priced configuration.

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

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

General Items

0.1 Application Code and Definition Statements

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

Appendix A and B contain all source codes implemented in this benchmark.

0.2 Test Sponsor

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

Fujitsu and Oracle Corp. were joint sponsors of this TPC Benchmark C.

0.3 Parameter Settings

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

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

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

Appendix D contains the parameters for the database, the operating system, and the

configuration for the transaction monitor.

0.4 Configuration Diagrams

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

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

The System Under Test (SUT), a PRIMEQUEST 580A c/s w/99 Front-Ends, is depicted in the following diagrams.

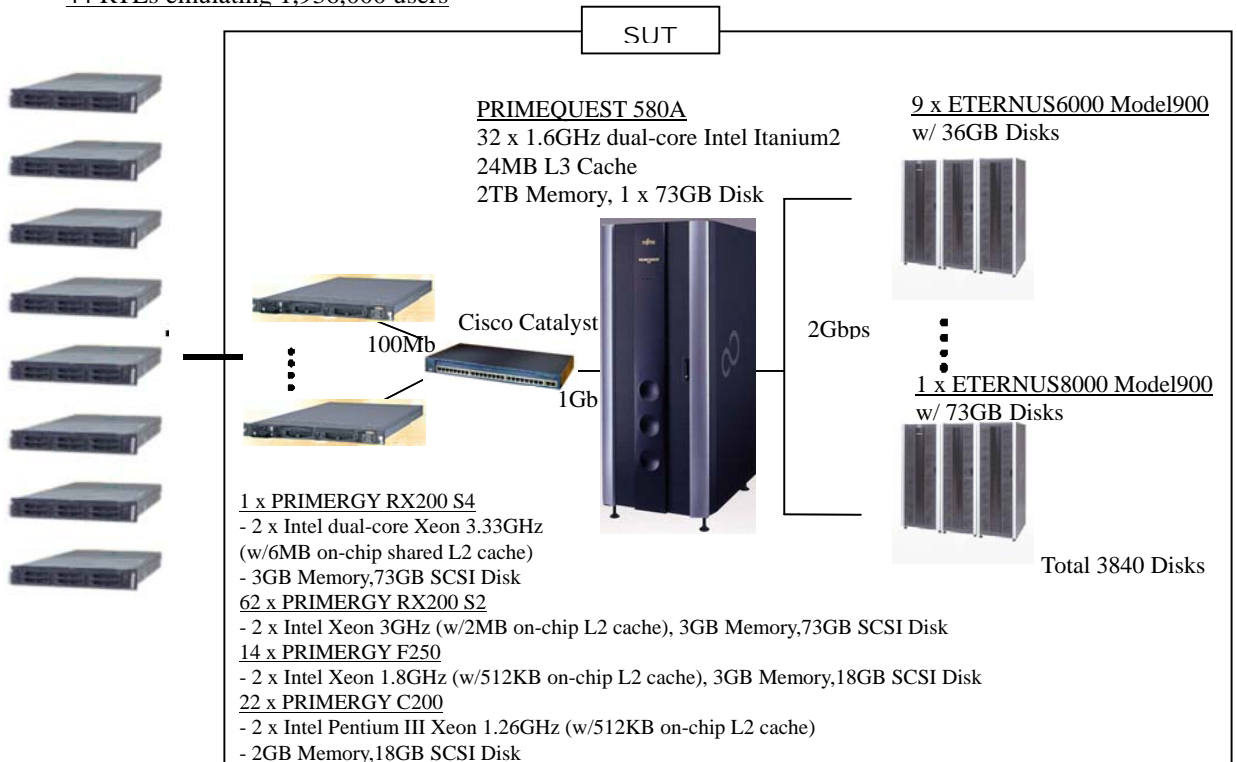
The configuration diagrams for both the tested and priced systems are included on the following pages.

There were differences between the priced and measured configurations. The differences are:

- A RTE was used in the tested configuration.
- The clients other than the PRIMERGY RX200 S4 were replaced by the PRIMERGY RX200 S4 in the priced configuration.
- The ETERNUS6000 Storage subsystems were replaced by the ETERNUS8000 Storage subsystems in the priced configuration.

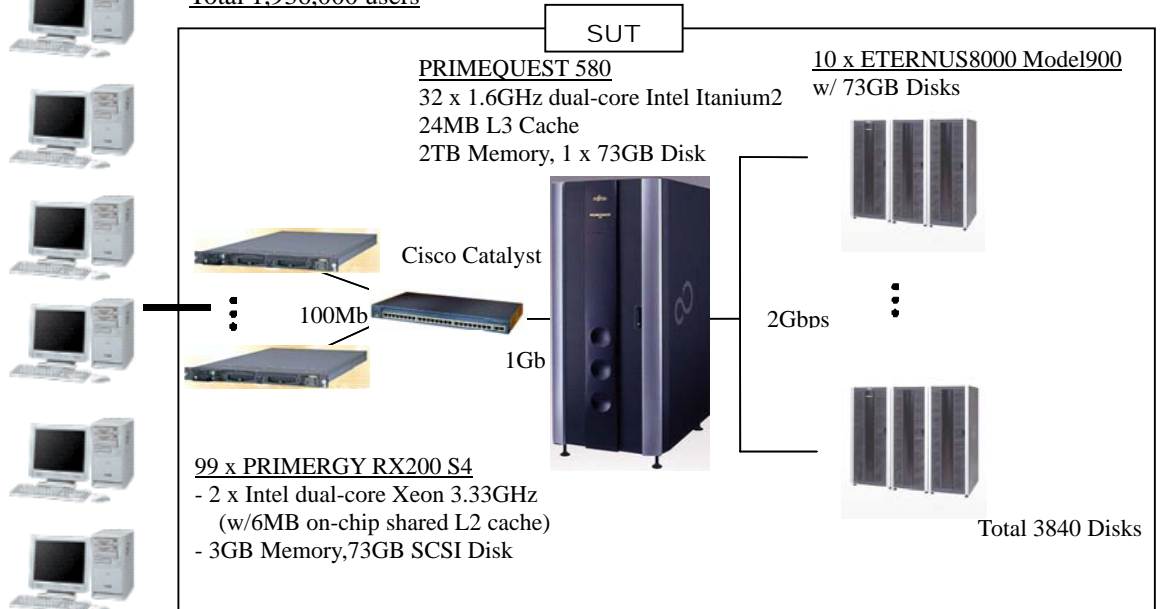
PRIMEQUEST 580A Tested Configuration

44 RTEs emulating 1,936,000 users



PRIMEQUEST 580A Priced Configuration

Total 1,936,000 users



Clause 1 Related Items

1.1 Table Definitions

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

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

1.2 Physical Organization of Database

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

Physical space was allocated to Oracle Database 10g Release2 Enterprise Edition with Partitioning on the server disks according to the details provided in section 4.2. The size of the space segments on each disk was calculated to provide even distribution of data across the disk drives.

1.3 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 maximum key value for these new rows.

All insert and delete functions were verified and fully operational during the entire benchmark.

1.4 Partitioning

While there are a few restrictions placed upon horizontal or vertical partitioning of

tables and rows in the TPC-C benchmark, any such partitioning must be disclosed. Replication of tables, if used, must be disclosed. Additional and/or duplicated attributes in any table must be disclosed along with a statement on the impact on performance.

Horizontal partitioning was used for one (hist) of the tables and one (iordr2) of the indices. The detail of this partitioning can be understood by examining the table and index definition statements in Appendix E. Vertical partitioning and additional or duplicated attributes were not used in this implementation.

1.5 Replication, Duplication or Additions

Replication of tables, if used, must be disclosed(see Clause 1.4.6). Additional and/or duplicated attributes in any table must be disclosed along with a statement on the impact on performance(see Clause 1.4.7).

No replications, duplications or additional attributes were used in this benchmark.

Clause 2 Related Items

2.1 Random Number Generation

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

The seeds for each user were generated using the terminal id and the unix time of measurement start, which was given by the RTE master process. The terminal id is unique number across all RTE emulated users. Since the seeds were incremented by the same start value, they were also unique across all users.

2.2 Input/Output Screen Layout

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

All screen layouts followed the specification exactly.

2.3 Priced Terminal Feature Verification

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

The terminal attributes were verified by the auditor manually exercising each specification during the onsite audit portion of this benchmark.

2.4 Presentation Manager or Intelligent Terminal

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

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

2.5 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 backs as a results of an unused item number must be disclosed.

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

The percentage of Delivery transactions that were skipped as a result of an insufficient number of rows in the NEW-ORDER table must be disclosed. The mix (i.e., percentages) of transaction types seen by the SUT must be disclosed.

Table 2.1 lists the numerical quantities that Clauses 8.1.3.5 to 8.1.3.11 require.

Table 2.1 Transaction Statistics

Statistic		Value
New Order	Home warehouse order lines	99.00%
	Remote warehouse order lines	1.00%
	Rolled back transactions	1.00%
	Average items per order	10.00
Payment	Home warehouse	85.00%
	Remote warehouse	15.00%
	Accessed by last name	60.00%
Order Status	Accessed by last name	60.00%
Delivery	Skipped transactions	None
Transaction Mix	New Order	44.94%
	Payment	43.01%
	Order status	4.02%
	Delivery	4.02%
	Stock level	4.01%

2.6 Queuing Mechanism

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

Delivery transactions were submitted to servers using the same mechanism that other transactions used, Tuxedo API. The only difference was that `tpacall()` was used instead of `tpcall()` to call the server process asynchronously, i.e., control would return to the client thread immediately and the deferred delivery part would complete asynchronously in the server process.

Clause 3 Related Items

3.1 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 SUT must support during the execution of the benchmark. Those properties are Atomicity, Consistency, Isolation and Durability (ACID).

This section defines each of those properties, describes the steps taken to ensure that they were present during the test and describes a series of tests done to demonstrate compliance with the specification.

3.2 Atomicity

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.

3.2.1 Completed Transactions

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

A row was randomly selected from the warehouse, district and customer tables, and the balances noted. A payment transaction was started with the same warehouse, district and customer identifiers and a known amount. The payment transaction was

committed and the rows were verified to contain correctly updated balances.

3.2.2 Aborted Transactions

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

A row was randomly selected from the warehouse, district and customer tables, and the balances noted. A payment transaction was started with the same warehouse, district and customer identifiers and a known amount. The payment transaction was rolled back and the rows were verified to contain the original balances.

3.3 Consistency

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

The benchmark specification requires explicit demonstration of the following four consistency conditions;

- The sum of the district balances in a warehouse is equal to the warehouse balance;
- for each district, the next order id minus one is equal to the maximum order id in the ORDER table and equal to the maximum new order id in the NEW-ORDER table;
- for each district, the maximum order id minus minimum order id in the ORDER table plus one equals the number of rows in the NEW-ORDER table for that district;
- for each district, the sum of the order line counts in the ORDER table equals the number of rows in the ORDER-LINE table for that district.

These consistency conditions were tested using a shell script to issue queries to the database. The results of the queries verified that the database was consistent for all four tests.

A performance run was completed including a full 120 minutes of steady state and checkpoints.

The shell script was executed again. The result of the same queries verified that the database remained consistent after the run.

3.4 Isolation

Sufficient conditions must be enabled at either the system or application level to ensure the required isolation defined above (clause 3.4.1) is obtained.

The benchmark specification defines nine required tests to be performed to demonstrate that the required levels of transaction isolation are met. These tests, described in Clauses 3.4.2.1 - 3.4.2.9, were all performed and verified as required.

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

For Isolation test seven, case D was followed.

3.5 Durability

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

3.5.1 Loss of Log Disk

To demonstrate recovery from a permanent failure of durable media containing the Oracle recovery log data. The following steps were executed using 193,600 warehouses of the database:

1. The database was backed up to extra disks.
2. The total number of orders was determined by the sum of D_NEXT_O_ID of all rows in the DISTRICT table giving the beginning count.
3. The RTE was started with 1,936,000 users.
4. The test was allowed to run for a minimum of 5 minutes after the system reached the steady state.
5. One of the log disks was removed from the cabinet to cause a log disk failure. Since the log was configured as RAID0+1, the transactions continued to run without interruption.
6. After the run for another minimum of 5 minutes, the RTE was finished successfully.
7. Step 2 was repeated and the difference between the first and second counts was noted.
8. The success file was used to determine the number of NEW_ORDERS successfully returned to the RTE.
9. The counts in step 7 and 8 were compared, verifying that all committed transactions were successfully recovered.
10. Data from the success file was used to query the database to demonstrate that successful transactions had corresponding rows in the ORDER table and that rolled back transactions did not.

This demonstration was executed under the configuration with the Oracle recovery log laid on one ETERNUS8000 storage subsystem and with all Oracle TPC-C tables on 9 ETERNUS6000 storage subsystems. A disk on the ETERNUS8000 storage was removed for the failure.

3.5.2 Loss of Data Disk

To demonstrate recovery from a permanent failure of durable media containing the Oracle TPC-C tables, the following steps were executed using 193,600 warehouses of the database:

1. The database was backed up to extra disks.
2. The total number of orders was determined by the sum of D_NEXT_O_ID of all rows in the DISTRICT table giving the beginning count.
3. The RTE was started with 1,936,000 users.
4. The test was allowed to run for a minimum of 5 minutes after the system reached the steady state.
5. A disk array failure was caused by removing a disk from the disk array cabinet.
6. The RTE was shut down.
7. Oracle was shutdown abort.
8. New disks were returned into the disk cabinet to recover the RAID system.
9. Data from the backup disks was restored.
10. Oracle was restarted and the media recovery utility started.
11. Step 2 was repeated and the difference between the first and second counts was noted.
12. The success file was used to determine the number of NEW_ORDERS successfully returned to the RTE.

13. The counts in step 11 and 12 were compared, verifying that all committed transactions were successfully recovered.
14. Data from the success file was used to query the database to demonstrate that successful transactions had corresponding rows in the ORDER table and that rolled back transactions did not.

This demonstration was executed under the configuration with the Oracle recovery log laid on one ETERNUS6000 storage subsystem and with the Oracle TPC-C tables on 8 ETERNUS6000 and 1 ETERNUS8000 storage subsystems. A disk on the ETERNUS8000 storage was removed for the failure.

3.5.3 Instantaneous Interruption and Loss of Memory

Because loss of power erases the contents of memory, the instantaneous interruption and the loss of memory tests were combined into a single test.

This test was executed on a fully scaled database of 193,600 warehouses under a full load of 1,936,000 users. The following steps were executed:

1. The total number of orders was determined by the sum of D_NEXT_O_ID of all rows in the DISTRICT table giving the beginning count.
2. The RTE was started with 1,936,000 users.
3. The test was allowed to run for a minimum of 5 minutes after the system reached the steady state.
4. The primary power to the server was shutdown.
5. Power was restored and the system performed an automatic recovery.
6. Oracle was restarted and performed an automatic recovery.
7. Step 1 was repeated and the difference between the first and second counts was noted.
8. The success file was used to determine the number of NEW-ORDERS successfully returned to the RTE.
9. The counts in step 7 and 8 were compared, verifying that all committed transactions had been successfully recovered.
10. Data from the success file was used to query the database to demonstrate successful transactions had corresponding rows in the ORDER table, and rolled back transactions did not.

This demonstration was executed under the configuration with Oracle recovery log laid on one ETERNUS6000 storage subsystem and with the Oracle TPC-C tables on 8 ETERNUS6000 and 1 ETERNUS8000 storage subsystem.

Clause 4 Related Items

4.1 Initial Cardinality of Tables

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

The TPC-C database was initially configured with 208,320 warehouses.

Table 4.1 Number of Rows for Server

Table	Occurrences
Warehouse	208,320
District	2,083,200
Customer	6,249,600,000
History	6,249,600,000
Order	6,249,600,000
New Order	1,874,880,000
Order Line	62,498,465,280
Stock	20,832,000,000
Item	100,000

4.2 Database Layout

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

The following description depicts the data base configuration of the system tested.

1) Data files

We used 9 storage subsystems for data files. Eight storages had 4 controllers and 8 FibreChannel ports (4 @ dual-port) to connect to the Database server. It had 400 disk drives, which were configured into 25 RAID0 volumes having 16 disks in each group.

One storage had 2 controllers and 4 FibreChannel ports (2 @ dual-port) to connect to the Database server. It had 256 disk drives, which were configured into 16 RAID0 volumes having 16 disks in each group.

2 logical volumes were allocated on each RAID0 volume for this benchmark test. One of them was configured with 10 partitions (Type A) and the other was configured with 6 partitions (Type B) to lay the TPC-C data files.

Type A:

- 4 partitions for stok
- 5 partitions for cust
- 1 partition for ordr

Type B:

- 1 partition for icust2
- 1 partition for iordr2
- 1 partition for hist and nord
- 1 partition for temp
- 1 partition for icust1 or istok1
- 1 partition for other tables and indexes

The software RAID tool was used to create the large volumes. The data files of nord, ordr and hist were on each large volume configured with partitions for them. Total 432 logical volumes (2 logical volumes for each RAID volumes) were used for the data files.

2) Log files

The database logs were configured with 16 RAID0+1 volumes. Each RAID0+1 volume consisted of 24 disks (12 disks + 12 disks mirrored). A log file was configured with 16 LUs (1 LU from each RAID0+1 volume) using Linux mdadm software RAID utility to spread accesses across all 16 volumes.

3) Partition type

The following list shows how partitioned the devices (logical volumes) were:

[Type A : the size of each device is 204,800MB]

sdac sdae sdag sdal sdan
sdap sdaw sday sdba sdbf
sdbh sdbj sdfg sdfi sdfk
sdfp sdfr sdfu sdhj sdhl
sdhn sdhu sdhw sdhy sdit
sdiv sdix sdjc sdje sdjg
sdjl sdjn sdjp sdju sdjw
sdjy sdk skd sdkf sdkh
sdkm skko skkq skkv skkx
sdkz sdle sdlg sdli sdln
sdlp sdlr sdlw sdly sdm
sdma sdmf sdmh sdmj sdmo
sdmq sdms sdmx sdmz sdnb
sdng sdni sdnk sdnr sdnt
sdnv sdo sdoa sdoc sdoe
sdoj sdol sdon sdos sdou
sdow sdpb sdpc sdpe sdpk
sdpm sdpo sdpv sdpx sdpy
sdqe sdqg sdqi sdqn sdqp
sdqr sdqw sdqy sdra sdrh
sdrj sdrk sdrq sdrs sdru
sdrz sdsb sdsd sdsi sdsj
sdsr sdsu sdst sdsv sdt
sdta sdte sdte sdte sdte
sdtn sdtp sdtr sdtt sdtx
sdtz sdub sdud sduf sduh
sduj sdul sdun sdup sdur
sdut sdv sdv sdv sdv sdv
sdvb sdvd sdvf sdvh sdvj
sdvl sdvq sdvs sdvu sdvz
sdwb sdwd sdwi sdwk sdwm
sdwt sdwv sdwx sdxc sdxc
sdxe sdxe sdxl sdxn sdxp
sdxu sdw sdw sdyd sdyf
sdyh sdym sdyo sdyq sdyv
sdyx sdyz sdze sdzg sdzi
sdzp sdzr sdzt sdzy sdaaa
sdaac sdaah sdaaj sdaal sdaaq
sdaas sdaau sdfy sdga sdgc
sdge sdgg sdgi sdgk sdgm
sdid sdif sdih sdij sdil
sdin sdip sdir sdar sdhp
sdnm sdpq sdre sdwo sdzk
sdtv

[Type B : the size of each device is 153,600MB]

sdad sdaf sdah sdam sdao
sdaq sdax sdaz sdbb sdbg
sdbi sdbk sdfh sdfj sdfi
sdfq sdfs sdfu sdhk sdhm
sdho sdhv sdhx sdhz sdiu
sdiw sdiy sdjd sdjf sdjh
sdjm sdjo sdjq sdjv sdjx
sdjz sdke sdkg sdki sdkn
sdkp sdkr sdkw sdky sdl
sdla sdlf sdlh sdj sdlo
sdlq sdls sdlx sdz sdmb
sdmg sdmi sdmk sdmp sdmr
sdmt sdmy sdn sdna sdnc
sdnh sdnj sdnl sdns sdnu
sdnw sdob sdod sdof sdok
sdom sdoo sdot sdov sdox
sdp sdpc sdpe sdpg sdpl
sdpn sdpp sdpw sdpy sdqa
sdqf sdqh sdqj sdqo sdqq
sdqs sdqx sdqz sdrb sdri
sdrk sdrm sdr rrr sdrv
sdsa sdsc sdse sdsj sdsi
sdsn sdss sdsu sdsw sdtb
sdt d sdtf sdtk sdtm sdto
sdtq sdts sdtu sdy sdu
sdua sduc sdue sdug sdui
sduk sdum sduo sduq sdus
sduu sduw sduy sdva sdvc
sdve sdvg sdvi sdvk sdvm
sdvr sdvt sdvv sdw sdwa
sdwc sdwe sdwj sdwl sdwn
sdwu sdww sdwy sdx sdx
sdxf
sdxx sdz sdy sdye sdyg
sdyi sdyn sdyp sdyr sdyw
sdyz sdza sdzf sdzh sdzj
sdzq sdzs sdzu sdzz sdaab
sdaad sdaai sdaak sdaam sdaar
sdaat sdaav sdfz sdgb sdgd
sdgf sdgh sdgj sdgl sdgn
sdie sdig sdii sdik sdim
sdio sdiq sdis sdas sdhq
sdnn sdpr sdrd sdwp sdzl
sdtw

4) Relation between data files and devices

The following list shows the relation between data files, raw device files and device names:

datafile name	raw number	device name	datafile name	raw number	device name
cust_0_0	raw769	sdk6	ior2_0_47	raw2167	sdkr1
cust_0_1	raw770	sdk7	ior2_0_48	raw2168	sdkw1
cust_0_10	raw779	sdx6	ior2_0_49	raw2169	sdky1
cust_0_100	raw869	sdkz6	ior2_0_5	raw2125	sdyl1
cust_0_101	raw870	sdkz7	ior2_0_50	raw2170	sdla1
cust_0_102	raw871	sdle6	ior2_0_51	raw2171	sdlf1
cust_0_103	raw872	sdle7	ior2_0_52	raw2172	sdlh1
cust_0_104	raw873	sdlg6	ior2_0_53	raw2173	sdlj1
cust_0_105	raw874	sdlg7	ior2_0_54	raw2174	sdlo1
cust_0_106	raw875	sqli6	ior2_0_55	raw2175	sdli1
cust_0_107	raw876	sqli7	ior2_0_56	raw2176	sdlm1
cust_0_108	raw877	sdln6	ior2_0_57	raw2177	sdlx1
cust_0_109	raw878	sdln7	ior2_0_58	raw2178	sdly1
cust_0_11	raw780	sdx7	ior2_0_59	raw2179	sdmb1
cust_0_110	raw879	sdlp6	ior2_0_6	raw2126	sdad1
cust_0_111	raw880	sdlp7	ior2_0_60	raw2180	sdmg1
cust_0_112	raw881	sdlr6	ior2_0_61	raw2181	sdmi1
cust_0_113	raw882	sdlr7	ior2_0_62	raw2182	sdmk1
cust_0_114	raw883	sdlw6	ior2_0_63	raw2183	sdmp1
cust_0_115	raw884	sdlw7	ior2_0_64	raw2184	sdmr1
cust_0_116	raw885	sdly6	ior2_0_65	raw2185	sdmt1
cust_0_117	raw886	sdly7	ior2_0_66	raw2186	sdmy1
cust_0_118	raw887	sdma6	ior2_0_67	raw2187	sdna1
cust_0_119	raw888	sdma7	ior2_0_68	raw2188	sdnc1
cust_0_12	raw781	sdac6	ior2_0_69	raw2189	sdnh1
cust_0_120	raw889	sdmf6	ior2_0_7	raw2127	sdaf1
cust_0_121	raw890	sdmf7	ior2_0_70	raw2190	sdnj1
cust_0_122	raw891	sdmh6	ior2_0_71	raw2191	sdnl1
cust_0_123	raw892	sdmh7	ior2_0_72	raw2192	sdns1
cust_0_124	raw893	sdmj6	ior2_0_73	raw2193	sdnu1
cust_0_125	raw894	sdmj7	ior2_0_74	raw2194	sdnw1
cust_0_126	raw895	sdmo6	ior2_0_75	raw2195	sdob1
cust_0_127	raw896	sdmo7	ior2_0_76	raw2196	sdod1
cust_0_128	raw897	sdmq6	ior2_0_77	raw2197	sdof1
cust_0_129	raw898	sdmq7	ior2_0_78	raw2198	sdok1
cust_0_13	raw782	sdac7	ior2_0_79	raw2199	sdom1
cust_0_130	raw899	sdms6	ior2_0_8	raw2128	sdah1
cust_0_131	raw900	sdms7	ior2_0_80	raw2200	sdoo1
cust_0_132	raw901	sdmx6	ior2_0_81	raw2201	sdot1
cust_0_133	raw902	sdmx7	ior2_0_82	raw2202	sdov1
cust_0_134	raw903	sdmz6	ior2_0_83	raw2203	sdox1
cust_0_135	raw904	sdmz7	ior2_0_84	raw2204	sdpc1
cust_0_136	raw905	sdnb6	ior2_0_85	raw2205	sdpe1
cust_0_137	raw906	sdnb7	ior2_0_86	raw2206	sdpg1
cust_0_138	raw907	sdng6	ior2_0_87	raw2207	sdpl1
cust_0_139	raw908	sdng7	ior2_0_88	raw2208	sdpn1
cust_0_14	raw783	sdae6	ior2_0_89	raw2209	sdpp1
cust_0_140	raw909	sdni6	ior2_0_9	raw2129	sdam1
cust_0_141	raw910	sdni7	ior2_0_90	raw2210	sdpw1
cust_0_142	raw911	sdnk6	ior2_0_91	raw2211	sdpy1
cust_0_143	raw912	sdnk7	ior2_0_92	raw2212	sdqa1
cust_0_144	raw913	sdnr6	ior2_0_93	raw2213	sdqf1
cust_0_145	raw914	sdnr7	ior2_0_94	raw2214	sdqh1
cust_0_146	raw915	sdnt6	ior2_0_95	raw2215	sdqj1
cust_0_147	raw916	sdnt7	ior2_0_96	raw2216	sdqo1
cust_0_148	raw917	sdnv6	ior2_0_97	raw2217	sdqq1
cust_0_149	raw918	sdnv7	ior2_0_98	raw2218	sdqs1
cust_0_15	raw784	sdae7	ior2_0_99	raw2219	sdqx1
cust_0_150	raw919	sdoa6	istok_0_0	raw2552	sdkw5
cust_0_151	raw920	sdoa7	istok_0_1	raw2553	sdky5
cust_0_152	raw921	sdoc6	istok_0_10	raw2562	sdly5
cust_0_153	raw922	sdoc7	istok_0_100	raw2652	sdyp5

datafile name	raw number	device name	datafile name	raw number	device name
cust_0_154	raw923	sdoe6	istok_0_101	raw2653	sdyr5
cust_0_155	raw924	sdoe7	istok_0_102	raw2654	sdyw5
cust_0_156	raw925	sdoj6	istok_0_103	raw2655	sdyy5
cust_0_157	raw926	sdoj7	istok_0_104	raw2656	sdza5
cust_0_158	raw927	sdol6	istok_0_105	raw2657	sdzf5
cust_0_159	raw928	sdol7	istok_0_106	raw2658	sdzh5
cust_0_16	raw785	sdag6	istok_0_107	raw2659	sdzj5
cust_0_160	raw929	sdon6	istok_0_108	raw2660	sdzq5
cust_0_161	raw930	sdon7	istok_0_109	raw2661	sdzs5
cust_0_162	raw931	sdos6	istok_0_11	raw2563	sdmb5
cust_0_163	raw932	sdos7	istok_0_110	raw2662	sdzu5
cust_0_164	raw933	sdou6	istok_0_111	raw2663	sdzz5
cust_0_165	raw934	sdou7	istok_0_112	raw2664	sdaab5
cust_0_166	raw935	sdow6	istok_0_113	raw2665	sdaad5
cust_0_167	raw936	sdow7	istok_0_114	raw2666	sdaai5
cust_0_168	raw937	sdpb6	istok_0_115	raw2667	sdaak5
cust_0_169	raw938	sdpb7	istok_0_116	raw2668	sdaam5
cust_0_17	raw786	sdag7	istok_0_117	raw2669	sdaar5
cust_0_170	raw939	sdpd6	istok_0_118	raw2670	sdaat5
cust_0_171	raw940	sdpd7	istok_0_119	raw2671	sdaav5
cust_0_172	raw941	sdpf6	istok_0_12	raw2564	sdmg5
cust_0_173	raw942	sdpf7	istok_0_120	raw2672	sdtk5
cust_0_174	raw943	sdpk6	istok_0_121	raw2673	sdtm5
cust_0_175	raw944	sdpk7	istok_0_122	raw2674	sdto5
cust_0_176	raw945	sdpm6	istok_0_123	raw2675	sdtq5
cust_0_177	raw946	sdpm7	istok_0_124	raw2676	sdt5
cust_0_178	raw947	sdpo6	istok_0_125	raw2677	sdtu5
cust_0_179	raw948	sdpo7	istok_0_126	raw2678	sdt5
cust_0_18	raw787	sdal6	istok_0_127	raw2679	sdua5
cust_0_180	raw949	sdpv6	istok_0_128	raw2680	sduc5
cust_0_181	raw950	sdpv7	istok_0_129	raw2681	sdue5
cust_0_182	raw951	sdpx6	istok_0_13	raw2565	sdmi5
cust_0_183	raw952	sdpx7	istok_0_130	raw2682	sdug5
cust_0_184	raw953	sdpz6	istok_0_131	raw2683	sdui5
cust_0_185	raw954	sdpz7	istok_0_132	raw2684	sduk5
cust_0_186	raw955	sdqe6	istok_0_133	raw2685	sdum5
cust_0_187	raw956	sdqe7	istok_0_134	raw2686	sduo5
cust_0_188	raw957	sdqg6	istok_0_135	raw2687	sduq5
cust_0_189	raw958	sdqg7	istok_0_136	raw2688	sdus5
cust_0_19	raw788	sdal7	istok_0_137	raw2689	sduu5
cust_0_190	raw959	sdqi6	istok_0_138	raw2690	sduw5
cust_0_191	raw960	sdqi7	istok_0_139	raw2691	sduy5
cust_0_192	raw961	sdqn6	istok_0_14	raw2566	sdmk5
cust_0_193	raw962	sdqn7	istok_0_140	raw2692	sdva5
cust_0_194	raw963	sdqp6	istok_0_141	raw2693	sdvc5
cust_0_195	raw964	sdqp7	istok_0_142	raw2694	sdve5
cust_0_196	raw965	sdqr6	istok_0_143	raw2695	sdvg5
cust_0_197	raw966	sdqr7	istok_0_15	raw2567	sdmp5
cust_0_198	raw967	sdqw6	istok_0_16	raw2568	sdmr5
cust_0_199	raw968	sdqw7	istok_0_17	raw2569	sdmt5
cust_0_2	raw771	sdm6	istok_0_18	raw2570	sdmy5
cust_0_20	raw789	sdan6	istok_0_19	raw2571	sdna5
cust_0_200	raw969	sdqy6	istok_0_2	raw2554	sdla5
cust_0_201	raw970	sdqy7	istok_0_20	raw2572	sdnc5
cust_0_202	raw971	sdra6	istok_0_21	raw2573	sdnh5
cust_0_203	raw972	sdra7	istok_0_22	raw2574	sdnj5
cust_0_204	raw973	sdrh6	istok_0_23	raw2575	sdn5
cust_0_205	raw974	sdrh7	istok_0_24	raw2576	sdns5
cust_0_206	raw975	sdri6	istok_0_25	raw2577	sdnu5
cust_0_207	raw976	sdri7	istok_0_26	raw2578	sdnw5
cust_0_208	raw977	sdr6	istok_0_27	raw2579	sdob5
cust_0_209	raw978	sdr7	istok_0_28	raw2580	sdod5

datafile name	raw number	device name	datafile name	raw number	device name
cust_0_21	raw790	sdan7	istok_0_29	raw2581	sdof5
cust_0_210	raw979	sdrq6	istok_0_3	raw2585	sdlf5
cust_0_211	raw980	sdrq7	istok_0_30	raw2582	sdok5
cust_0_212	raw981	sdrs6	istok_0_31	raw2583	sdom5
cust_0_213	raw982	sdrs7	istok_0_32	raw2584	sdoos5
cust_0_214	raw983	sdru6	istok_0_33	raw2585	sdot5
cust_0_215	raw984	sdru7	istok_0_34	raw2586	sdov5
cust_0_216	raw985	sdrz6	istok_0_35	raw2587	sdox5
cust_0_217	raw986	sdrz7	istok_0_36	raw2588	sdpc5
cust_0_218	raw987	sdsb6	istok_0_37	raw2589	sdpe5
cust_0_219	raw988	sdsb7	istok_0_38	raw2590	sdpg5
cust_0_22	raw791	sdap6	istok_0_39	raw2591	sdpl5
cust_0_220	raw989	sdsd6	istok_0_4	raw2556	sdlh5
cust_0_221	raw990	sdsd7	istok_0_40	raw2592	sdpn5
cust_0_222	raw991	sdsi6	istok_0_41	raw2593	sdpp5
cust_0_223	raw992	sdsi7	istok_0_42	raw2594	sdpw5
cust_0_224	raw993	sdsks6	istok_0_43	raw2595	sdpy5
cust_0_225	raw994	sdsks7	istok_0_44	raw2596	sdqa5
cust_0_226	raw995	sdsms6	istok_0_45	raw2597	sdqf5
cust_0_227	raw996	sdsms7	istok_0_46	raw2598	sdqh5
cust_0_228	raw997	sdsr6	istok_0_47	raw2599	sdqj5
cust_0_229	raw998	sdsr7	istok_0_48	raw2600	sdqo5
cust_0_23	raw792	sdap7	istok_0_49	raw2601	sdqq5
cust_0_230	raw999	sdst6	istok_0_5	raw2557	sdlj5
cust_0_231	raw1000	sdst7	istok_0_50	raw2602	sdqs5
cust_0_232	raw1001	sdsvs6	istok_0_51	raw2603	sdqx5
cust_0_233	raw1002	sdsvs7	istok_0_52	raw2604	sdqz5
cust_0_234	raw1003	sdta6	istok_0_53	raw2605	sdrb5
cust_0_235	raw1004	sdta7	istok_0_54	raw2606	sdri5
cust_0_236	raw1005	sdtc6	istok_0_55	raw2607	sdrk5
cust_0_237	raw1006	sdtc7	istok_0_56	raw2608	sdrm5
cust_0_238	raw1007	sdte6	istok_0_57	raw2609	sdrri5
cust_0_239	raw1008	sdte7	istok_0_58	raw2610	sdrti5
cust_0_24	raw793	sdaw6	istok_0_59	raw2611	sdrv5
cust_0_240	raw1009	sdtd6	istok_0_6	raw2558	sdlo5
cust_0_241	raw1010	sdtd7	istok_0_60	raw2612	sdsa5
cust_0_242	raw1011	sdtl6	istok_0_61	raw2613	sdsc5
cust_0_243	raw1012	sdtl7	istok_0_62	raw2614	sdse5
cust_0_244	raw1013	sdtm6	istok_0_63	raw2615	sdsh5
cust_0_245	raw1014	sdtm7	istok_0_64	raw2616	sdsli5
cust_0_246	raw1015	sdtm6	istok_0_65	raw2617	sdsni5
cust_0_247	raw1016	sdtm7	istok_0_66	raw2618	sdsso5
cust_0_248	raw1017	sdrtr6	istok_0_67	raw2619	sdsu5
cust_0_249	raw1018	sdrtr7	istok_0_68	raw2620	sdsww5
cust_0_25	raw794	sdaw7	istok_0_69	raw2621	sdtb5
cust_0_250	raw1019	sdttd6	istok_0_7	raw2559	sdlq5
cust_0_251	raw1020	sdttd7	istok_0_70	raw2622	sdttd5
cust_0_252	raw1021	sdttx6	istok_0_71	raw2623	sdttd5
cust_0_253	raw1022	sdttx7	istok_0_72	raw2624	sdvii5
cust_0_254	raw1023	sdtz6	istok_0_73	raw2625	sdvk5
cust_0_255	raw1024	sdtz7	istok_0_74	raw2626	sdvmm5
cust_0_256	raw1025	sdub6	istok_0_75	raw2627	sdvri5
cust_0_257	raw1026	sdub7	istok_0_76	raw2628	sdvtt5
cust_0_258	raw1027	sdud6	istok_0_77	raw2629	sdvvi5
cust_0_259	raw1028	sdud7	istok_0_78	raw2630	sdwa5
cust_0_26	raw795	sday6	istok_0_79	raw2631	sdwc5
cust_0_260	raw1029	sduf6	istok_0_8	raw2560	sdlis5
cust_0_261	raw1030	sduf7	istok_0_80	raw2632	sdwe5
cust_0_262	raw1031	sduh6	istok_0_81	raw2633	sdwj5
cust_0_263	raw1032	sduh7	istok_0_82	raw2634	sdwl5
cust_0_264	raw1033	sduj6	istok_0_83	raw2635	sdwn5

datafile name	raw number	device name	datafile name	raw number	device name
cust_0_265	raw1034	sdul7	istok_0_84	raw2636	sdwu5
cust_0_266	raw1035	sdul6	istok_0_85	raw2637	sdww5
cust_0_267	raw1036	sdul7	istok_0_86	raw2638	sdwy5
cust_0_268	raw1037	sdun6	istok_0_87	raw2639	sdx5
cust_0_269	raw1038	sdun7	istok_0_88	raw2640	sdx5
cust_0_27	raw796	sday7	istok_0_89	raw2641	sdx5
cust_0_270	raw1039	sdup6	istok_0_9	raw2561	sdlx5
cust_0_271	raw1040	sdup7	istok_0_90	raw2642	sdxm5
cust_0_272	raw1041	sdur6	istok_0_91	raw2643	sdxo5
cust_0_273	raw1042	sdur7	istok_0_92	raw2644	sdxq5
cust_0_274	raw1043	sdut6	istok_0_93	raw2645	sdxv5
cust_0_275	raw1044	sdut7	istok_0_94	raw2646	sdx5
cust_0_276	raw1045	sduv6	istok_0_95	raw2647	sdxz5
cust_0_277	raw1046	sduv7	istok_0_96	raw2648	sdye5
cust_0_278	raw1047	sdux6	istok_0_97	raw2649	sdyg5
cust_0_279	raw1048	sdux7	istok_0_98	raw2650	sdyi5
cust_0_28	raw797	sdba6	istok_0_99	raw2651	sdyn5
cust_0_280	raw1049	sduz6	item_0_0	raw2697	sdn6
cust_0_281	raw1050	sduz7	ware_0_0	raw2698	sdp6
cust_0_282	raw1051	sdvb6	log_1_1	raw2701	sdv6
cust_0_283	raw1052	sdvb7	log_1_2	raw2702	sdad6
cust_0_284	raw1053	sdvd6	nord_0_0	raw3252	dm-76
cust_0_285	raw1054	sdvd7	nord_0_1	raw3253	dm-77
cust_0_286	raw1055	sdvf6	nord_0_10	raw3262	dm-86
cust_0_287	raw1056	sdvf7	nord_0_11	raw3263	dm-87
cust_0_288	raw1057	sdvh6	nord_0_14	raw3266	dm-90
cust_0_289	raw1058	sdvh7	nord_0_15	raw3267	dm-91
cust_0_29	raw798	sdba7	nord_0_16	raw3268	dm-92
cust_0_290	raw1059	sdvj6	nord_0_17	raw3269	dm-93
cust_0_291	raw1060	sdvj7	nord_0_19	raw3271	dm-95
cust_0_292	raw1061	sdvl6	nord_0_2	raw3254	dm-78
cust_0_293	raw1062	sdvl7	nord_0_21	raw3197	dm-21
cust_0_294	raw1063	sdvq6	nord_0_22	raw3198	dm-22
cust_0_295	raw1064	sdvq7	nord_0_23	raw3199	dm-23
cust_0_296	raw1065	sdvs6	nord_0_25	raw3201	dm-25
cust_0_297	raw1066	sdvs7	nord_0_26	raw3202	dm-26
cust_0_298	raw1067	sdvu6	nord_0_27	raw3203	dm-27
cust_0_299	raw1068	sdvu7	nord_0_3	raw3255	dm-79
cust_0_3	raw772	sdm7	nord_0_31	raw3207	dm-31
cust_0_30	raw799	sdbf6	nord_0_35	raw3211	dm-35
cust_0_300	raw1069	sdvz6	nord_0_37	raw3213	dm-37
cust_0_301	raw1070	sdvz7	nord_0_39	raw3215	dm-39
cust_0_302	raw1071	sdwb6	nord_0_4	raw3256	dm-80
cust_0_303	raw1072	sdwb7	nord_0_49	raw3225	dm-49
cust_0_304	raw1073	sdwd6	nord_0_5	raw3257	dm-81
cust_0_305	raw1074	sdwd7	nord_0_50	raw3226	dm-50
cust_0_306	raw1075	sdwi6	nord_0_52	raw3228	dm-52
cust_0_307	raw1076	sdwi7	nord_0_53	raw3229	dm-53
cust_0_308	raw1077	sdwk6	nord_0_56	raw3232	dm-56
cust_0_309	raw1078	sdwk7	nord_0_57	raw3233	dm-57
cust_0_31	raw800	sdbf7	nord_0_58	raw3234	dm-58
cust_0_310	raw1079	sdwm6	nord_0_62	raw3238	dm-62
cust_0_311	raw1080	sdwm7	nord_0_64	raw3240	dm-64
cust_0_312	raw1081	sdwt6	nord_0_65	raw3241	dm-65
cust_0_313	raw1082	sdwt7	nord_0_66	raw3242	dm-66
cust_0_314	raw1083	sdwv6	nord_0_67	raw3243	dm-67
cust_0_315	raw1084	sdwv7	nord_0_69	raw3245	dm-69
cust_0_316	raw1085	sdwx6	nord_0_71	raw3247	dm-71
cust_0_317	raw1086	sdwx7	nord_0_72	raw3248	dm-72
cust_0_318	raw1087	sdx6	nord_0_73	raw3249	dm-73

datafile name	raw number	device name	datafile name	raw number	device name
cust_0_319	raw1088	sdxc7	nord_0_82	raw3258	dm-82
cust_0_32	raw801	sdbb6	nord_0_83	raw3259	dm-83
cust_0_320	raw1089	sdxe6	nord_0_84	raw3260	dm-84
cust_0_321	raw1090	sdxe7	nord_0_85	raw3261	dm-85
cust_0_322	raw1091	sdxc6	nord_0_88	raw3264	dm-88
cust_0_323	raw1092	sdxc7	nord_0_89	raw3265	dm-89
cust_0_324	raw1093	sdxl6	nord_0_94	raw3270	dm-94
cust_0_325	raw1094	sdxl7	ordr_0_0	raw1736	dm-288
cust_0_326	raw1095	sdxn6	ordr_0_1	raw1737	dm-289
cust_0_327	raw1096	sdxn7	ordr_0_10	raw1746	dm-298
cust_0_328	raw1097	sdxp6	ordr_0_100	raw1836	dm-100
cust_0_329	raw1098	sdxp7	ordr_0_101	raw1837	dm-101
cust_0_33	raw802	sdbb7	ordr_0_102	raw1838	dm-102
cust_0_330	raw1099	sdxu6	ordr_0_103	raw1839	dm-103
cust_0_331	raw1100	sdxu7	ordr_0_104	raw1840	dm-104
cust_0_332	raw1101	sdxc6	ordr_0_105	raw1841	dm-105
cust_0_333	raw1102	sdxc7	ordr_0_106	raw1842	dm-106
cust_0_334	raw1103	sdxy6	ordr_0_107	raw1843	dm-107
cust_0_335	raw1104	sdxy7	ordr_0_108	raw1844	dm-108
cust_0_336	raw1105	sdyc6	ordr_0_109	raw1845	dm-109
cust_0_337	raw1106	sdyc7	ordr_0_11	raw1747	dm-299
cust_0_338	raw1107	sdyc6	ordr_0_110	raw1846	dm-110
cust_0_339	raw1108	sdyc7	ordr_0_111	raw1847	dm-111
cust_0_34	raw803	sdbj6	ordr_0_112	raw1848	dm-112
cust_0_340	raw1109	sdyc6	ordr_0_113	raw1849	dm-113
cust_0_341	raw1110	sdyc7	ordr_0_114	raw1850	dm-114
cust_0_342	raw1111	sdym6	ordr_0_115	raw1851	dm-115
cust_0_343	raw1112	sdym7	ordr_0_116	raw1852	dm-116
cust_0_344	raw1113	sdyc6	ordr_0_117	raw1853	dm-117
cust_0_345	raw1114	sdyc7	ordr_0_118	raw1854	dm-118
cust_0_346	raw1115	sdyc6	ordr_0_119	raw1855	dm-119
cust_0_347	raw1116	sdyc7	ordr_0_12	raw1748	dm-300
cust_0_348	raw1117	sdyc6	ordr_0_120	raw1856	dm-120
cust_0_349	raw1118	sdyc7	ordr_0_121	raw1857	dm-121
cust_0_35	raw804	sdbj7	ordr_0_122	raw1858	dm-122
cust_0_350	raw1119	sdyc6	ordr_0_123	raw1859	dm-123
cust_0_351	raw1120	sdyc7	ordr_0_124	raw1860	dm-124
cust_0_352	raw1121	sdyc6	ordr_0_125	raw1861	dm-125
cust_0_353	raw1122	sdyc7	ordr_0_126	raw1862	dm-126
cust_0_354	raw1123	sdzc6	ordr_0_127	raw1863	dm-127
cust_0_355	raw1124	sdzc7	ordr_0_128	raw1864	dm-128
cust_0_356	raw1125	sdzg6	ordr_0_129	raw1865	dm-129
cust_0_357	raw1126	sdzg7	ordr_0_13	raw1749	dm-301
cust_0_358	raw1127	sdzi6	ordr_0_130	raw1866	dm-130
cust_0_359	raw1128	sdzi7	ordr_0_131	raw1867	dm-131
cust_0_36	raw805	sdfg6	ordr_0_132	raw1868	dm-132
cust_0_360	raw1129	sdzp6	ordr_0_133	raw1869	dm-133
cust_0_361	raw1130	sdzp7	ordr_0_134	raw1870	dm-134
cust_0_362	raw1131	sdzr6	ordr_0_135	raw1871	dm-135
cust_0_363	raw1132	sdzr7	ordr_0_136	raw1872	dm-136
cust_0_364	raw1133	sdzt6	ordr_0_137	raw1873	dm-137
cust_0_365	raw1134	sdzt7	ordr_0_138	raw1874	dm-138
cust_0_366	raw1135	sdzy6	ordr_0_139	raw1875	dm-139
cust_0_367	raw1136	sdzy7	ordr_0_14	raw1750	dm-302
cust_0_368	raw1137	sdaaa6	ordr_0_140	raw1876	dm-140
cust_0_369	raw1138	sdaaa7	ordr_0_141	raw1877	dm-141
cust_0_37	raw806	sdfg7	ordr_0_142	raw1878	dm-142
cust_0_370	raw1139	sdaac6	ordr_0_143	raw1879	dm-143
cust_0_371	raw1140	sdaac7	ordr_0_144	raw1880	dm-144

datafile name	raw number	device name	datafile name	raw number	device name
cust_0_372	raw1141	sdaah6	ordr_0_145	raw1881	dm-145
cust_0_373	raw1142	sdaah7	ordr_0_146	raw1882	dm-146
cust_0_374	raw1143	sdaaj6	ordr_0_147	raw1883	dm-147
cust_0_375	raw1144	sdaaj7	ordr_0_148	raw1884	dm-148
cust_0_376	raw1145	sdaal6	ordr_0_149	raw1885	dm-149
cust_0_377	raw1146	sdaal7	ordr_0_15	raw1751	dm-303
cust_0_378	raw1147	sdaaq6	ordr_0_150	raw1886	dm-150
cust_0_379	raw1148	sdaaq7	ordr_0_151	raw1887	dm-151
cust_0_38	raw807	sdfi6	ordr_0_152	raw1888	dm-152
cust_0_380	raw1149	sdaas6	ordr_0_153	raw1889	dm-153
cust_0_381	raw1150	sdaas7	ordr_0_154	raw1890	dm-154
cust_0_382	raw1151	sdaau6	ordr_0_155	raw1891	dm-155
cust_0_383	raw1152	sdaau7	ordr_0_156	raw1892	dm-156
cust_0_384	raw1153	sdk8	ordr_0_157	raw1893	dm-157
cust_0_385	raw1154	sdk9	ordr_0_158	raw1894	dm-158
cust_0_386	raw1155	sdar6	ordr_0_159	raw1895	dm-159
cust_0_387	raw1156	sdm8	ordr_0_16	raw1752	dm-304
cust_0_388	raw1157	sdm9	ordr_0_160	raw1896	dm-160
cust_0_389	raw1158	sdar7	ordr_0_161	raw1897	dm-161
cust_0_39	raw808	sdfi7	ordr_0_162	raw1898	dm-162
cust_0_390	raw1159	sdo8	ordr_0_163	raw1899	dm-163
cust_0_391	raw1160	sdo9	ordr_0_164	raw1900	dm-164
cust_0_392	raw1161	sdar8	ordr_0_165	raw1901	dm-165
cust_0_393	raw1162	sdt8	ordr_0_166	raw1902	dm-166
cust_0_394	raw1163	sdt9	ordr_0_167	raw1903	dm-167
cust_0_395	raw1164	sdar9	ordr_0_168	raw1904	dm-168
cust_0_396	raw1165	sdv8	ordr_0_169	raw1905	dm-169
cust_0_397	raw1166	sdv9	ordr_0_17	raw1753	dm-305
cust_0_398	raw1167	sdfy6	ordr_0_170	raw1906	dm-170
cust_0_399	raw1168	sdx8	ordr_0_171	raw1907	dm-171
cust_0_4	raw773	sdo6	ordr_0_172	raw1908	dm-172
cust_0_40	raw809	sdfk6	ordr_0_173	raw1909	dm-173
cust_0_400	raw1169	sdx9	ordr_0_174	raw1910	dm-174
cust_0_401	raw1170	sdfy7	ordr_0_175	raw1911	dm-175
cust_0_402	raw1171	sdac8	ordr_0_176	raw1912	dm-176
cust_0_403	raw1172	sdac9	ordr_0_177	raw1913	dm-177
cust_0_404	raw1173	sdfy8	ordr_0_178	raw1914	dm-178
cust_0_405	raw1174	sdae8	ordr_0_179	raw1915	dm-179
cust_0_406	raw1175	sdae9	ordr_0_18	raw1754	dm-306
cust_0_407	raw1176	sdfy9	ordr_0_180	raw1916	dm-180
cust_0_408	raw1177	sdag8	ordr_0_181	raw1917	dm-181
cust_0_409	raw1178	sdag9	ordr_0_182	raw1918	dm-182
cust_0_41	raw810	sdfk7	ordr_0_183	raw1919	dm-183
cust_0_410	raw1179	sdga6	ordr_0_184	raw1920	dm-184
cust_0_411	raw1180	sdal8	ordr_0_185	raw1921	dm-185
cust_0_412	raw1181	sdal9	ordr_0_186	raw1922	dm-186
cust_0_413	raw1182	sdga7	ordr_0_187	raw1923	dm-187
cust_0_414	raw1183	sdan8	ordr_0_188	raw1924	dm-188
cust_0_415	raw1184	sdan9	ordr_0_189	raw1925	dm-189
cust_0_416	raw1185	sdga8	ordr_0_19	raw1755	dm-307
cust_0_417	raw1186	sdap8	ordr_0_190	raw1926	dm-190
cust_0_418	raw1187	sdap9	ordr_0_191	raw1927	dm-191
cust_0_419	raw1188	sdga9	ordr_0_2	raw1738	dm-290
cust_0_42	raw811	sdfp6	ordr_0_20	raw1756	dm-308
cust_0_420	raw1189	sdaw8	ordr_0_21	raw1757	dm-309
cust_0_421	raw1190	sdaw9	ordr_0_22	raw1758	dm-310
cust_0_422	raw1191	sdgc6	ordr_0_23	raw1759	dm-311
cust_0_423	raw1192	sday8	ordr_0_24	raw1760	dm-312
cust_0_424	raw1193	sday9	ordr_0_25	raw1761	dm-313
cust_0_425	raw1194	sdgc7	ordr_0_26	raw1762	dm-314

datafile name	raw number	device name	datafile name	raw number	device name
cust_0_426	raw1195	sdba8	ordr_0_27	raw1763	dm-315
cust_0_427	raw1196	sdba9	ordr_0_28	raw1764	dm-316
cust_0_428	raw1197	sdgc8	ordr_0_29	raw1765	dm-317
cust_0_429	raw1198	sdbf8	ordr_0_3	raw1739	dm-291
cust_0_43	raw812	sdfp7	ordr_0_30	raw1766	dm-318
cust_0_430	raw1199	sdbf9	ordr_0_31	raw1767	dm-319
cust_0_431	raw1200	sdgc9	ordr_0_32	raw1768	dm-320
cust_0_432	raw1201	sdbh8	ordr_0_33	raw1769	dm-321
cust_0_433	raw1202	sdbh9	ordr_0_34	raw1770	dm-322
cust_0_434	raw1203	sdgc6	ordr_0_35	raw1771	dm-323
cust_0_435	raw1204	sdbj8	ordr_0_36	raw1772	dm-324
cust_0_436	raw1205	sdbj9	ordr_0_37	raw1773	dm-325
cust_0_437	raw1206	sdge7	ordr_0_38	raw1774	dm-326
cust_0_438	raw1207	sdfg8	ordr_0_39	raw1775	dm-327
cust_0_439	raw1208	sdfg9	ordr_0_4	raw1740	dm-292
cust_0_44	raw813	sdfr6	ordr_0_40	raw1776	dm-328
cust_0_440	raw1209	sdge8	ordr_0_41	raw1777	dm-329
cust_0_441	raw1210	sdfi8	ordr_0_42	raw1778	dm-330
cust_0_442	raw1211	sdfi9	ordr_0_43	raw1779	dm-331
cust_0_443	raw1212	sdge9	ordr_0_44	raw1780	dm-332
cust_0_444	raw1213	sdff8	ordr_0_45	raw1781	dm-333
cust_0_445	raw1214	sdff9	ordr_0_46	raw1782	dm-334
cust_0_446	raw1215	sdgg6	ordr_0_47	raw1783	dm-335
cust_0_447	raw1216	sdfp8	ordr_0_48	raw1784	dm-336
cust_0_448	raw1217	sdfp9	ordr_0_49	raw1785	dm-337
cust_0_449	raw1218	sdgg7	ordr_0_5	raw1741	dm-293
cust_0_45	raw814	sdfr7	ordr_0_50	raw1786	dm-338
cust_0_450	raw1219	sdfr8	ordr_0_51	raw1787	dm-339
cust_0_451	raw1220	sdfr9	ordr_0_52	raw1788	dm-340
cust_0_452	raw1221	sdgg8	ordr_0_53	raw1789	dm-341
cust_0_453	raw1222	sdft8	ordr_0_54	raw1790	dm-342
cust_0_454	raw1223	sdft9	ordr_0_55	raw1791	dm-343
cust_0_455	raw1224	sdgg9	ordr_0_56	raw1792	dm-344
cust_0_456	raw1225	sdhj8	ordr_0_57	raw1793	dm-345
cust_0_457	raw1226	sdhj9	ordr_0_58	raw1794	dm-346
cust_0_458	raw1227	sdgi6	ordr_0_59	raw1795	dm-347
cust_0_459	raw1228	sdhl8	ordr_0_6	raw1742	dm-294
cust_0_46	raw815	sdft6	ordr_0_60	raw1796	dm-348
cust_0_460	raw1229	sdhl9	ordr_0_61	raw1797	dm-349
cust_0_461	raw1230	sdgi7	ordr_0_62	raw1798	dm-350
cust_0_462	raw1231	sdhn8	ordr_0_63	raw1799	dm-351
cust_0_463	raw1232	sdhn9	ordr_0_64	raw1800	dm-352
cust_0_464	raw1233	sdgi8	ordr_0_65	raw1801	dm-353
cust_0_465	raw1234	sdhu8	ordr_0_66	raw1802	dm-354
cust_0_466	raw1235	sdhu9	ordr_0_67	raw1803	dm-355
cust_0_467	raw1236	sdgi9	ordr_0_68	raw1804	dm-356
cust_0_468	raw1237	sdhw8	ordr_0_69	raw1805	dm-357
cust_0_469	raw1238	sdhw9	ordr_0_7	raw1743	dm-295
cust_0_47	raw816	sdft7	ordr_0_70	raw1806	dm-358
cust_0_470	raw1239	sdgk6	ordr_0_71	raw1807	dm-359
cust_0_471	raw1240	sdhy8	ordr_0_72	raw1808	dm-360
cust_0_472	raw1241	sdhy9	ordr_0_73	raw1809	dm-361
cust_0_473	raw1242	sdgk7	ordr_0_74	raw1810	dm-362
cust_0_474	raw1243	sdit8	ordr_0_75	raw1811	dm-363
cust_0_475	raw1244	sdit9	ordr_0_76	raw1812	dm-364
cust_0_476	raw1245	sdgk8	ordr_0_77	raw1813	dm-365
cust_0_477	raw1246	sdiv8	ordr_0_78	raw1814	dm-366
cust_0_478	raw1247	sdiv9	ordr_0_79	raw1815	dm-367
cust_0_479	raw1248	sdgk9	ordr_0_8	raw1744	dm-296

datafile name	raw number	device name	datafile name	raw number	device name
cust_0_48	raw817	sdhj6	ordr_0_80	raw1816	dm-368
cust_0_480	raw1249	sdix8	ordr_0_81	raw1817	dm-369
cust_0_481	raw1250	sdix9	ordr_0_82	raw1818	dm-370
cust_0_482	raw1251	sdgm6	ordr_0_83	raw1819	dm-371
cust_0_483	raw1252	sdje8	ordr_0_84	raw1820	dm-372
cust_0_484	raw1253	sdje9	ordr_0_85	raw1821	dm-373
cust_0_485	raw1254	sdgm7	ordr_0_86	raw1822	dm-374
cust_0_486	raw1255	sdje8	ordr_0_87	raw1823	dm-375
cust_0_487	raw1256	sdje9	ordr_0_88	raw1824	dm-376
cust_0_488	raw1257	sdgm8	ordr_0_89	raw1825	dm-377
cust_0_489	raw1258	sdjg8	ordr_0_9	raw1745	dm-297
cust_0_49	raw818	sdhj7	ordr_0_90	raw1826	dm-378
cust_0_490	raw1259	sdjg9	ordr_0_91	raw1827	dm-379
cust_0_491	raw1260	sdgm9	ordr_0_92	raw1828	dm-380
cust_0_492	raw1261	sdjl8	ordr_0_93	raw1829	dm-381
cust_0_493	raw1262	sdjl9	ordr_0_94	raw1830	dm-382
cust_0_494	raw1263	sdhp6	ordr_0_95	raw1831	dm-383
cust_0_495	raw1264	sdjn8	ordr_0_96	raw1832	dm-96
cust_0_496	raw1265	sdjn9	ordr_0_97	raw1833	dm-97
cust_0_497	raw1266	sdhp7	ordr_0_98	raw1834	dm-98
cust_0_498	raw1267	sdjp8	ordr_0_99	raw1835	dm-99
cust_0_499	raw1268	sdjp9	roll1	raw2889	sdn7
cust_0_5	raw774	sdo7	sp_0	raw2890	sdp7
cust_0_50	raw819	sdhl6	stok_0_0	raw1	sdk1
cust_0_500	raw1269	sdhp8	stok_0_1	raw2	sdk2
cust_0_501	raw1270	sdju8	stok_0_10	raw11	sdo3
cust_0_502	raw1271	sdju9	stok_0_100	raw101	sdhl1
cust_0_503	raw1272	sdhp9	stok_0_101	raw102	sdhl2
cust_0_504	raw1273	sdjw8	stok_0_102	raw103	sdhl3
cust_0_505	raw1274	sdjw9	stok_0_103	raw104	sdgm2
cust_0_506	raw1275	sdiid6	stok_0_104	raw105	sdqw5
cust_0_507	raw1276	sdjy8	stok_0_105	raw106	sdux5
cust_0_508	raw1277	sdjy9	stok_0_106	raw107	sdxc5
cust_0_509	raw1278	sdiid7	stok_0_107	raw108	sdgm3
cust_0_51	raw820	sdhl7	stok_0_108	raw109	sdhu1
cust_0_510	raw1279	sdkd8	stok_0_109	raw110	sdhu2
cust_0_511	raw1280	sdkd9	stok_0_11	raw12	sdar3
cust_0_512	raw1281	sdiid8	stok_0_110	raw111	sdhu3
cust_0_513	raw1282	sdkf8	stok_0_111	raw112	sdhp1
cust_0_514	raw1283	sdkf9	stok_0_112	raw113	sdhw1
cust_0_515	raw1284	sdiid9	stok_0_113	raw114	sdhw2
cust_0_516	raw1285	sdkh8	stok_0_114	raw115	sdhw3
cust_0_517	raw1286	sdkh9	stok_0_115	raw116	sdhp2
cust_0_518	raw1287	sdif6	stok_0_116	raw117	sdhy1
cust_0_519	raw1288	sdkm8	stok_0_117	raw118	sdhy2
cust_0_52	raw821	sdhn6	stok_0_118	raw119	sdhy3
cust_0_520	raw1289	sdkm9	stok_0_119	raw120	sdhp3
cust_0_521	raw1290	sdif7	stok_0_12	raw13	sdt1
cust_0_522	raw1291	sdko8	stok_0_120	raw121	sdt1
cust_0_523	raw1292	sdko9	stok_0_121	raw122	sdt2
cust_0_524	raw1293	sdif8	stok_0_122	raw123	sdt3
cust_0_525	raw1294	sdkq8	stok_0_123	raw124	sdiid1
cust_0_526	raw1295	sdkq9	stok_0_124	raw125	sdiv1
cust_0_527	raw1296	sdif9	stok_0_125	raw126	sdiv2
cust_0_528	raw1297	sdkv8	stok_0_126	raw127	sdiv3
cust_0_529	raw1298	sdkv9	stok_0_127	raw128	sdiid2
cust_0_53	raw822	sdhn7	stok_0_128	raw129	sdix1
cust_0_530	raw1299	sdih6	stok_0_129	raw130	sdix2
cust_0_531	raw1300	sdkx8	stok_0_13	raw14	sdt2
cust_0_532	raw1301	sdkx9	stok_0_130	raw131	sdix3

datafile name	raw number	device name	datafile name	raw number	device name
cust_0_533	raw1302	sdih7	stok_0_131	raw132	sdid3
cust_0_534	raw1303	sdkz8	stok_0_132	raw133	sdjc1
cust_0_535	raw1304	sdkz9	stok_0_133	raw134	sdjc2
cust_0_536	raw1305	sdih8	stok_0_134	raw135	sdjc3
cust_0_537	raw1306	sdle8	stok_0_135	raw136	sdif1
cust_0_538	raw1307	sdle9	stok_0_136	raw137	sdje1
cust_0_539	raw1308	sdih9	stok_0_137	raw138	sdje2
cust_0_54	raw823	sdhu6	stok_0_138	raw139	sdje3
cust_0_540	raw1309	sdlg8	stok_0_139	raw140	sdif2
cust_0_541	raw1310	sdlg9	stok_0_14	raw15	sdt3
cust_0_542	raw1311	sdij6	stok_0_140	raw141	sdjg1
cust_0_543	raw1312	sdli8	stok_0_141	raw142	sdjg2
cust_0_544	raw1313	sdli9	stok_0_142	raw143	sdjg3
cust_0_545	raw1314	sdij7	stok_0_143	raw144	sdif3
cust_0_546	raw1315	sdln8	stok_0_144	raw145	sdjl1
cust_0_547	raw1316	sdln9	stok_0_145	raw146	sdjl2
cust_0_548	raw1317	sdij8	stok_0_146	raw147	sdjl3
cust_0_549	raw1318	sdlp8	stok_0_147	raw148	sdih1
cust_0_55	raw824	sdhu7	stok_0_148	raw149	sdjn1
cust_0_550	raw1319	sdlp9	stok_0_149	raw150	sdjn2
cust_0_551	raw1320	sdij9	stok_0_15	raw16	sdfy1
cust_0_552	raw1321	sdlr8	stok_0_150	raw151	sdjn3
cust_0_553	raw1322	sdlr9	stok_0_151	raw152	sdih2
cust_0_554	raw1323	sdil6	stok_0_152	raw153	sdjp1
cust_0_555	raw1324	sdlw8	stok_0_153	raw154	sdjp2
cust_0_556	raw1325	sdlw9	stok_0_154	raw155	sdjp3
cust_0_557	raw1326	sdil7	stok_0_155	raw156	sdih3
cust_0_558	raw1327	sdly8	stok_0_156	raw157	sdju1
cust_0_559	raw1328	sdly9	stok_0_157	raw158	sdju2
cust_0_56	raw825	sdhw6	stok_0_158	raw159	sdju3
cust_0_560	raw1329	sdil8	stok_0_159	raw160	sdij1
cust_0_561	raw1330	sdma8	stok_0_16	raw17	sdv1
cust_0_562	raw1331	sdma9	stok_0_160	raw161	sdjw1
cust_0_563	raw1332	sdil9	stok_0_161	raw162	sdjw2
cust_0_564	raw1333	sdmf8	stok_0_162	raw163	sdjw3
cust_0_565	raw1334	sdmf9	stok_0_163	raw164	sdij2
cust_0_566	raw1335	sdin6	stok_0_164	raw165	sdjy1
cust_0_567	raw1336	sdmh8	stok_0_165	raw166	sdjy2
cust_0_568	raw1337	sdmh9	stok_0_166	raw167	sdjy3
cust_0_569	raw1338	sdin7	stok_0_167	raw168	sdij3
cust_0_57	raw826	sdhw7	stok_0_168	raw169	sdkd1
cust_0_570	raw1339	sdmj8	stok_0_169	raw170	sdkd2
cust_0_571	raw1340	sdmj9	stok_0_17	raw18	sdv2
cust_0_572	raw1341	sdin8	stok_0_170	raw171	sdkd3
cust_0_573	raw1342	sdmo8	stok_0_171	raw172	sdil1
cust_0_574	raw1343	sdmo9	stok_0_172	raw173	sdkf1
cust_0_575	raw1344	sdin9	stok_0_173	raw174	sdkf2
cust_0_576	raw1345	sdmq8	stok_0_174	raw175	sdkf3
cust_0_577	raw1346	sdmq9	stok_0_175	raw176	sdil2
cust_0_578	raw1347	sdip6	stok_0_176	raw177	sdkh1
cust_0_579	raw1348	sdms8	stok_0_177	raw178	sdkh2
cust_0_58	raw827	sdhy6	stok_0_178	raw179	sdkh3
cust_0_580	raw1349	sdms9	stok_0_179	raw180	sdil3
cust_0_581	raw1350	sdip7	stok_0_18	raw19	sdv3
cust_0_582	raw1351	sdmx8	stok_0_180	raw181	sdkm1
cust_0_583	raw1352	sdmx9	stok_0_181	raw182	sdkm2
cust_0_584	raw1353	sdip8	stok_0_182	raw183	sdkm3
cust_0_585	raw1354	sdmz8	stok_0_183	raw184	sdin1
cust_0_586	raw1355	sdmz9	stok_0_184	raw185	sdko1
cust_0_587	raw1356	sdip9	stok_0_185	raw186	sdko2

datafile name	raw number	device name	datafile name	raw number	device name
cust_0_588	raw1357	sdbn8	stok_0_186	raw187	sdko3
cust_0_589	raw1358	sdbn9	stok_0_187	raw188	sdin2
cust_0_59	raw828	sdhy7	stok_0_188	raw189	sdkq1
cust_0_590	raw1359	sdir6	stok_0_189	raw190	sdkq2
cust_0_591	raw1360	sding8	stok_0_19	raw20	sdfy2
cust_0_592	raw1361	sding9	stok_0_190	raw191	sdkq3
cust_0_593	raw1362	sdir7	stok_0_191	raw192	sdin3
cust_0_594	raw1363	sdni8	stok_0_192	raw193	sdkv1
cust_0_595	raw1364	sdni9	stok_0_193	raw194	sdkv2
cust_0_596	raw1365	sdir8	stok_0_194	raw195	sdkv3
cust_0_597	raw1366	sdkn8	stok_0_195	raw196	sdip1
cust_0_598	raw1367	sdkn9	stok_0_196	raw197	sdkx1
cust_0_599	raw1368	sdir9	stok_0_197	raw198	sdkx2
cust_0_6	raw775	sdt6	stok_0_198	raw199	sdkx3
cust_0_60	raw829	sdit6	stok_0_199	raw200	sdip2
cust_0_600	raw1369	sdnr8	stok_0_2	raw3	sdk3
cust_0_601	raw1370	sdnr9	stok_0_20	raw21	sdx1
cust_0_602	raw1371	sdnm6	stok_0_200	raw201	sdkz1
cust_0_603	raw1372	sdnr8	stok_0_201	raw202	sdkz2
cust_0_604	raw1373	sdnr9	stok_0_202	raw203	sdkz3
cust_0_605	raw1374	sdir7	stok_0_203	raw204	sdip3
cust_0_606	raw1375	sdir8	stok_0_204	raw205	sdle1
cust_0_607	raw1376	sdir9	stok_0_205	raw206	sdle2
cust_0_608	raw1377	sdir6	stok_0_206	raw207	sdle3
cust_0_609	raw1378	sdir7	stok_0_207	raw208	sdir1
cust_0_61	raw830	sdir8	stok_0_208	raw209	sdlg1
cust_0_610	raw1379	sdir9	stok_0_209	raw210	sdlg2
cust_0_611	raw1380	sdir6	stok_0_21	raw22	sdx2
cust_0_612	raw1381	sdir7	stok_0_210	raw211	sdlg3
cust_0_613	raw1382	sdir8	stok_0_211	raw212	sdir2
cust_0_614	raw1383	sdir9	stok_0_212	raw213	sdl11
cust_0_615	raw1384	sdir6	stok_0_213	raw214	sdl12
cust_0_616	raw1385	sdir7	stok_0_214	raw215	sdl13
cust_0_617	raw1386	sdir8	stok_0_215	raw216	sdir3
cust_0_618	raw1387	sdir9	stok_0_216	raw217	sdl11
cust_0_619	raw1388	sdir6	stok_0_217	raw218	sdl12
cust_0_62	raw831	sdir7	stok_0_218	raw219	sdl13
cust_0_620	raw1389	sdir8	stok_0_219	raw220	sdl11
cust_0_621	raw1390	sdir9	stok_0_22	raw23	sdx3
cust_0_622	raw1391	sdir6	stok_0_220	raw221	sdlp1
cust_0_623	raw1392	sdir7	stok_0_221	raw222	sdlp2
cust_0_624	raw1393	sdir8	stok_0_222	raw223	sdlp3
cust_0_625	raw1394	sdir9	stok_0_223	raw224	sdl11
cust_0_626	raw1395	sdir6	stok_0_224	raw225	sdlr1
cust_0_627	raw1396	sdir7	stok_0_225	raw226	sdlr2
cust_0_628	raw1397	sdir8	stok_0_226	raw227	sdlr3
cust_0_629	raw1398	sdir9	stok_0_227	raw228	sdl11
cust_0_63	raw832	sdir6	stok_0_228	raw229	sdlw1
cust_0_630	raw1399	sdir7	stok_0_229	raw230	sdlw2
cust_0_631	raw1400	sdir8	stok_0_23	raw24	sdly3
cust_0_632	raw1401	sdir9	stok_0_230	raw231	sdlw3
cust_0_633	raw1402	sdir6	stok_0_231	raw232	sdpq1
cust_0_634	raw1403	sdir7	stok_0_232	raw233	sdly1
cust_0_635	raw1404	sdir8	stok_0_233	raw234	sdly2
cust_0_636	raw1405	sdir9	stok_0_234	raw235	sdly3
cust_0_637	raw1406	sdir6	stok_0_235	raw236	sdpq2
cust_0_638	raw1407	sdir7	stok_0_236	raw237	sdma1
cust_0_639	raw1408	sdir8	stok_0_237	raw238	sdma2
cust_0_64	raw833	sdir9	stok_0_238	raw239	sdma3
cust_0_640	raw1409	sdir6	stok_0_239	raw240	sdpq3

datafile name	raw number	device name	datafile name	raw number	device name
cust_0_641	raw1410	sdtv7	stok_0_24	raw25	sdac1
cust_0_642	raw1411	sdpf8	stok_0_240	raw241	sdmf1
cust_0_643	raw1412	sdpf9	stok_0_241	raw242	sdmf2
cust_0_644	raw1413	sdtv8	stok_0_242	raw243	sdmf3
cust_0_645	raw1414	sdpk8	stok_0_243	raw244	sdrc1
cust_0_646	raw1415	sdpk9	stok_0_244	raw245	sdmh1
cust_0_647	raw1416	sdtv9	stok_0_245	raw246	sdmh2
cust_0_648	raw1417	sdpm8	stok_0_246	raw247	sdmh3
cust_0_649	raw1418	sdpm9	stok_0_247	raw248	sdrc2
cust_0_65	raw834	sdix7	stok_0_248	raw249	sdmj1
cust_0_650	raw1419	sdwo6	stok_0_249	raw250	sdmj2
cust_0_651	raw1420	sdpo8	stok_0_25	raw26	sdac2
cust_0_652	raw1421	sdpo9	stok_0_250	raw251	sdmj3
cust_0_653	raw1422	sdwo7	stok_0_251	raw252	sdrc3
cust_0_654	raw1423	sdpv8	stok_0_252	raw253	sdmo1
cust_0_655	raw1424	sdpv9	stok_0_253	raw254	sdmo2
cust_0_656	raw1425	sdwo8	stok_0_254	raw255	sdmo3
cust_0_657	raw1426	sdpx8	stok_0_255	raw256	sdtv1
cust_0_658	raw1427	sdpx9	stok_0_256	raw257	sdmq1
cust_0_659	raw1428	sdwo9	stok_0_257	raw258	sdmq2
cust_0_66	raw835	sdjc6	stok_0_258	raw259	sdmq3
cust_0_660	raw1429	sdpz8	stok_0_259	raw260	sdtv2
cust_0_661	raw1430	sdpz9	stok_0_26	raw27	sdac3
cust_0_662	raw1431	sdzk6	stok_0_260	raw261	sdms1
cust_0_663	raw1432	sdqe8	stok_0_261	raw262	sdms2
cust_0_664	raw1433	sdqe9	stok_0_262	raw263	sdms3
cust_0_665	raw1434	sdzk7	stok_0_263	raw264	sdtv3
cust_0_666	raw1435	sdqg8	stok_0_264	raw265	sdmx1
cust_0_667	raw1436	sdqg9	stok_0_265	raw266	sdmx2
cust_0_668	raw1437	sdzk8	stok_0_266	raw267	sdmx3
cust_0_669	raw1438	sdqi8	stok_0_267	raw268	sdwo1
cust_0_67	raw836	sdjc7	stok_0_268	raw269	sdmz1
cust_0_670	raw1439	sdqi9	stok_0_269	raw270	sdmz2
cust_0_671	raw1440	sdzk9	stok_0_27	raw28	sdga1
cust_0_672	raw1441	sdqn8	stok_0_270	raw271	sdmz3
cust_0_673	raw1442	sdqn9	stok_0_271	raw272	sdwo2
cust_0_674	raw1443	sdk10	stok_0_272	raw273	sdnb1
cust_0_675	raw1444	sdqp8	stok_0_273	raw274	sdnb2
cust_0_676	raw1445	sdqp9	stok_0_274	raw275	sdnb3
cust_0_677	raw1446	sdo10	stok_0_275	raw276	sdwo3
cust_0_678	raw1447	sdqr8	stok_0_276	raw277	sdng1
cust_0_679	raw1448	sdqr9	stok_0_277	raw278	sdng2
cust_0_68	raw837	sdje6	stok_0_278	raw279	sdng3
cust_0_680	raw1449	sdv10	stok_0_279	raw280	sdzk1
cust_0_681	raw1450	sdqw8	stok_0_28	raw29	sdae1
cust_0_682	raw1451	sdqw9	stok_0_280	raw281	sdni1
cust_0_683	raw1452	sdac10	stok_0_281	raw282	sdni2
cust_0_684	raw1453	sdqy8	stok_0_282	raw283	sdni3
cust_0_685	raw1454	sdqy9	stok_0_283	raw284	sdzk2
cust_0_686	raw1455	sdag10	stok_0_284	raw285	sdnk1
cust_0_687	raw1456	sdra8	stok_0_285	raw286	sdnk2
cust_0_688	raw1457	sdra9	stok_0_286	raw287	sdnk3
cust_0_689	raw1458	sdan10	stok_0_287	raw288	sdzk3
cust_0_69	raw838	sdje7	stok_0_288	raw289	sdnr1
cust_0_690	raw1459	sdrh8	stok_0_289	raw290	sdnr2
cust_0_691	raw1460	sdrh9	stok_0_29	raw30	sdae2
cust_0_692	raw1461	sdaw10	stok_0_290	raw291	sdnr3
cust_0_693	raw1462	sdri8	stok_0_291	raw292	sdk5
cust_0_694	raw1463	sdri9	stok_0_292	raw293	sdnt1
cust_0_695	raw1464	sdba10	stok_0_293	raw294	sdnt2

datafile name	raw number	device name	datafile name	raw number	device name
cust_0_696	raw1465	sdr18	stok_0_294	raw295	sdnt3
cust_0_697	raw1466	sdr19	stok_0_295	raw296	sdo5
cust_0_698	raw1467	sdbh10	stok_0_296	raw297	sdnv1
cust_0_699	raw1468	sdrq8	stok_0_297	raw298	sdnv2
cust_0_7	raw776	sdt7	stok_0_298	raw299	sdnv3
cust_0_70	raw839	sdjg6	stok_0_299	raw300	sdv5
cust_0_700	raw1469	sdrq9	stok_0_3	raw4	sdar1
cust_0_701	raw1470	sdfg10	stok_0_30	raw31	sdae3
cust_0_702	raw1471	sdrs8	stok_0_300	raw301	sdoa1
cust_0_703	raw1472	sdrs9	stok_0_301	raw302	sdoa2
cust_0_704	raw1473	sdfk10	stok_0_302	raw303	sdoa3
cust_0_705	raw1474	sdru8	stok_0_303	raw304	sdac5
cust_0_706	raw1475	sdru9	stok_0_304	raw305	sdoc1
cust_0_707	raw1476	sdf10	stok_0_305	raw306	sdoc2
cust_0_708	raw1477	sdrz8	stok_0_306	raw307	sdoc3
cust_0_709	raw1478	sdrz9	stok_0_307	raw308	sdag5
cust_0_71	raw840	sdjg7	stok_0_308	raw309	sdoe1
cust_0_710	raw1479	sdhj10	stok_0_309	raw310	sdoe2
cust_0_711	raw1480	sdsb8	stok_0_31	raw32	sdga2
cust_0_712	raw1481	sdsb9	stok_0_310	raw311	sdoe3
cust_0_713	raw1482	sdhn10	stok_0_311	raw312	sdan5
cust_0_714	raw1483	sdsd8	stok_0_312	raw313	sdoj1
cust_0_715	raw1484	sdsd9	stok_0_313	raw314	sdoj2
cust_0_716	raw1485	sdhw10	stok_0_314	raw315	sdoj3
cust_0_717	raw1486	sdsi8	stok_0_315	raw316	sdaw5
cust_0_718	raw1487	sdsi9	stok_0_316	raw317	sdol1
cust_0_719	raw1488	sdt10	stok_0_317	raw318	sdol2
cust_0_72	raw841	sdj16	stok_0_318	raw319	sdol3
cust_0_720	raw1489	sds8	stok_0_319	raw320	sdba5
cust_0_721	raw1490	sds89	stok_0_32	raw33	sdag1
cust_0_722	raw1491	sdix10	stok_0_320	raw321	sdon1
cust_0_723	raw1492	sds88	stok_0_321	raw322	sdon2
cust_0_724	raw1493	sds89	stok_0_322	raw323	sdon3
cust_0_725	raw1494	sdje10	stok_0_323	raw324	sdbh5
cust_0_726	raw1495	sdsr8	stok_0_324	raw325	sdos1
cust_0_727	raw1496	sdsr9	stok_0_325	raw326	sdos2
cust_0_728	raw1497	sdj110	stok_0_326	raw327	sdos3
cust_0_729	raw1498	sds8	stok_0_327	raw328	sdfg5
cust_0_73	raw842	sdj17	stok_0_328	raw329	sdou1
cust_0_730	raw1499	sds89	stok_0_329	raw330	sdou2
cust_0_731	raw1500	sdjp10	stok_0_33	raw34	sdag2
cust_0_732	raw1501	sds88	stok_0_330	raw331	sdou3
cust_0_733	raw1502	sds89	stok_0_331	raw332	sdfk5
cust_0_734	raw1503	sdjw10	stok_0_332	raw333	sdow1
cust_0_735	raw1504	sdta8	stok_0_333	raw334	sdow2
cust_0_736	raw1505	sdta9	stok_0_334	raw335	sdow3
cust_0_737	raw1506	sdkd10	stok_0_335	raw336	sdf8
cust_0_738	raw1507	sdtc8	stok_0_336	raw337	sdpb1
cust_0_739	raw1508	sdtc9	stok_0_337	raw338	sdpb2
cust_0_74	raw843	sdjn6	stok_0_338	raw339	sdpb3
cust_0_740	raw1509	sdkh10	stok_0_339	raw340	sdol5
cust_0_741	raw1510	sdte8	stok_0_34	raw35	sdag3
cust_0_742	raw1511	sdte9	stok_0_340	raw341	sdpd1
cust_0_743	raw1512	sdko10	stok_0_341	raw342	sdpd2
cust_0_744	raw1513	sdtj8	stok_0_342	raw343	sdpd3
cust_0_745	raw1514	sdtj9	stok_0_343	raw344	sdzi5
cust_0_746	raw1515	sdkv10	stok_0_344	raw345	sdpf1
cust_0_747	raw1516	sdtl8	stok_0_345	raw346	sdpf2
cust_0_748	raw1517	sdtl9	stok_0_346	raw347	sdpf3
cust_0_749	raw1518	sdkz10	stok_0_347	raw348	sdhw5

datafile name	raw number	device name	datafile name	raw number	device name
cust_0_75	raw844	sdjn7	stok_0_348	raw349	sdpk1
cust_0_750	raw1519	sdtn8	stok_0_349	raw350	sdpk2
cust_0_751	raw1520	sdtn9	stok_0_35	raw36	sdga3
cust_0_752	raw1521	sdlg10	stok_0_350	raw351	sdpk3
cust_0_753	raw1522	sdtp8	stok_0_351	raw352	sdit5
cust_0_754	raw1523	sdtp9	stok_0_352	raw353	sdpm1
cust_0_755	raw1524	sdln10	stok_0_353	raw354	sdpm2
cust_0_756	raw1525	sdtr8	stok_0_354	raw355	sdpm3
cust_0_757	raw1526	sdtr9	stok_0_355	raw356	sdix5
cust_0_758	raw1527	sdlr10	stok_0_356	raw357	sdpo1
cust_0_759	raw1528	sdtt8	stok_0_357	raw358	sdpo2
cust_0_76	raw845	sdjp6	stok_0_358	raw359	sdpo3
cust_0_760	raw1529	sdtt9	stok_0_359	raw360	sdje5
cust_0_761	raw1530	sdly10	stok_0_36	raw37	sdal1
cust_0_762	raw1531	sdtx8	stok_0_360	raw361	sdpv1
cust_0_763	raw1532	sdtx9	stok_0_361	raw362	sdpv2
cust_0_764	raw1533	sdmf10	stok_0_362	raw363	sdpv3
cust_0_765	raw1534	sdtz8	stok_0_363	raw364	sdj5
cust_0_766	raw1535	sdtz9	stok_0_364	raw365	sdpx1
cust_0_767	raw1536	sdmj10	stok_0_365	raw366	sdpx2
cust_0_768	raw1537	sdub8	stok_0_366	raw367	sdpx3
cust_0_769	raw1538	sdub9	stok_0_367	raw368	sdjp5
cust_0_77	raw846	sdjp7	stok_0_368	raw369	sdpz1
cust_0_770	raw1539	sdmq10	stok_0_369	raw370	sdpz2
cust_0_771	raw1540	sdud8	stok_0_37	raw38	sdal2
cust_0_772	raw1541	sdud9	stok_0_370	raw371	sdpz3
cust_0_773	raw1542	sdmx10	stok_0_371	raw372	sdjw5
cust_0_774	raw1543	sduf8	stok_0_372	raw373	sdqe1
cust_0_775	raw1544	sduf9	stok_0_373	raw374	sdqe2
cust_0_776	raw1545	sdnb10	stok_0_374	raw375	sdqe3
cust_0_777	raw1546	sduh8	stok_0_375	raw376	sdkd5
cust_0_778	raw1547	sduh9	stok_0_376	raw377	sdqg1
cust_0_779	raw1548	sdni10	stok_0_377	raw378	sdqg2
cust_0_78	raw847	sdju6	stok_0_378	raw379	sdqg3
cust_0_780	raw1549	sdju8	stok_0_379	raw380	sdkh5
cust_0_781	raw1550	sdju9	stok_0_38	raw39	sdal3
cust_0_782	raw1551	sdnr10	stok_0_380	raw381	sdqi1
cust_0_783	raw1552	sdul8	stok_0_381	raw382	sdqi2
cust_0_784	raw1553	sdul9	stok_0_382	raw383	sdqi3
cust_0_785	raw1554	sdnv10	stok_0_383	raw384	sdko5
cust_0_786	raw1555	sdun8	stok_0_384	raw385	sdqn1
cust_0_787	raw1556	sdun9	stok_0_385	raw386	sdqn2
cust_0_788	raw1557	sdoc10	stok_0_386	raw387	sdqn3
cust_0_789	raw1558	sdup8	stok_0_387	raw388	sdkv5
cust_0_79	raw848	sdju7	stok_0_388	raw389	sdqp1
cust_0_790	raw1559	sdup9	stok_0_389	raw390	sdqp2
cust_0_791	raw1560	sdjo10	stok_0_39	raw40	sdgc1
cust_0_792	raw1561	sdur8	stok_0_390	raw391	sdqp3
cust_0_793	raw1562	sdur9	stok_0_391	raw392	sdkz5
cust_0_794	raw1563	sdon10	stok_0_392	raw393	sdqr1
cust_0_795	raw1564	sdut8	stok_0_393	raw394	sdqr2
cust_0_796	raw1565	sdut9	stok_0_394	raw395	sdqr3
cust_0_797	raw1566	sdou10	stok_0_395	raw396	sdlg5
cust_0_798	raw1567	sduv8	stok_0_396	raw397	sdqw1
cust_0_799	raw1568	sduv9	stok_0_397	raw398	sdqw2
cust_0_8	raw777	sdv6	stok_0_398	raw399	sdqw3
cust_0_80	raw849	sdjw6	stok_0_399	raw400	sdln5
cust_0_800	raw1569	sdpb10	stok_0_4	raw5	sdm1
cust_0_801	raw1570	sdux8	stok_0_40	raw41	sdan1

datafile name	raw number	device name	datafile name	raw number	device name
cust_0_802	raw1571	sdux9	stok_0_400	raw401	sdqy1
cust_0_803	raw1572	sdpf10	stok_0_401	raw402	sdqy2
cust_0_804	raw1573	sduz8	stok_0_402	raw403	sdqy3
cust_0_805	raw1574	sduz9	stok_0_403	raw404	sdlr5
cust_0_806	raw1575	sdpm10	stok_0_404	raw405	sdra1
cust_0_807	raw1576	sdvb8	stok_0_405	raw406	sdra2
cust_0_808	raw1577	sdvb9	stok_0_406	raw407	sdra3
cust_0_809	raw1578	sdpv10	stok_0_407	raw408	sdly5
cust_0_81	raw850	sdjw7	stok_0_408	raw409	sdrh1
cust_0_810	raw1579	sdvd8	stok_0_409	raw410	sdrh2
cust_0_811	raw1580	sdvd9	stok_0_41	raw42	sdan2
cust_0_812	raw1581	sdpz10	stok_0_410	raw411	sdrh3
cust_0_813	raw1582	sdvf8	stok_0_411	raw412	sdmf5
cust_0_814	raw1583	sdvf9	stok_0_412	raw413	sdrj1
cust_0_815	raw1584	sdqg10	stok_0_413	raw414	sdrj2
cust_0_816	raw1585	sdvh8	stok_0_414	raw415	sdrj3
cust_0_817	raw1586	sdvh9	stok_0_415	raw416	sdmj5
cust_0_818	raw1587	sdqn10	stok_0_416	raw417	sdr11
cust_0_819	raw1588	sdvj8	stok_0_417	raw418	sdr12
cust_0_82	raw851	sdjy6	stok_0_418	raw419	sdr13
cust_0_820	raw1589	sdvj9	stok_0_419	raw420	sdmq5
cust_0_821	raw1590	sdqr10	stok_0_42	raw43	sdan3
cust_0_822	raw1591	sdvl8	stok_0_420	raw421	sdrq1
cust_0_823	raw1592	sdvl9	stok_0_421	raw422	sdrq2
cust_0_824	raw1593	sdqy10	stok_0_422	raw423	sdrq3
cust_0_825	raw1594	sdvq8	stok_0_423	raw424	sdmx5
cust_0_826	raw1595	sdvq9	stok_0_424	raw425	sdrs1
cust_0_827	raw1596	sdrh10	stok_0_425	raw426	sdrs2
cust_0_828	raw1597	sdvs8	stok_0_426	raw427	sdrs3
cust_0_829	raw1598	sdvs9	stok_0_427	raw428	sdrnb5
cust_0_83	raw852	sdjy7	stok_0_428	raw429	sdru1
cust_0_830	raw1599	sdr110	stok_0_429	raw430	sdru2
cust_0_831	raw1600	sdvu8	stok_0_43	raw44	sdgc2
cust_0_832	raw1601	sdvu9	stok_0_430	raw431	sdru3
cust_0_833	raw1602	sdrs10	stok_0_431	raw432	sdrni5
cust_0_834	raw1603	sdvz8	stok_0_432	raw433	sdrz1
cust_0_835	raw1604	sdvz9	stok_0_433	raw434	sdrz2
cust_0_836	raw1605	sdrz10	stok_0_434	raw435	sdrz3
cust_0_837	raw1606	sdwb8	stok_0_435	raw436	sdrn5
cust_0_838	raw1607	sdwb9	stok_0_436	raw437	sdsb1
cust_0_839	raw1608	sdsd10	stok_0_437	raw438	sdsb2
cust_0_84	raw853	sdkd6	stok_0_438	raw439	sdsb3
cust_0_840	raw1609	sdwd8	stok_0_439	raw440	sdrnv5
cust_0_841	raw1610	sdwd9	stok_0_44	raw45	sdap1
cust_0_842	raw1611	sdsd10	stok_0_440	raw441	sdsd1
cust_0_843	raw1612	sdwi8	stok_0_441	raw442	sdsd2
cust_0_844	raw1613	sdwi9	stok_0_442	raw443	sdsd3
cust_0_845	raw1614	sdsr10	stok_0_443	raw444	sdoc5
cust_0_846	raw1615	sdwk8	stok_0_444	raw445	sdsi1
cust_0_847	raw1616	sdwk9	stok_0_445	raw446	sdsi2
cust_0_848	raw1617	sdsv10	stok_0_446	raw447	sdsi3
cust_0_849	raw1618	sdwm8	stok_0_447	raw448	sdoj5
cust_0_85	raw854	sdkd7	stok_0_448	raw449	sdsd1
cust_0_850	raw1619	sdwm9	stok_0_449	raw450	sdsd2
cust_0_851	raw1620	sdtc10	stok_0_45	raw46	sdap2
cust_0_852	raw1621	sdwt8	stok_0_450	raw451	sdsd3
cust_0_853	raw1622	sdwt9	stok_0_451	raw452	sdon5
cust_0_854	raw1623	sdtd10	stok_0_452	raw453	sdsd1
cust_0_855	raw1624	sdwv8	stok_0_453	raw454	sdsd2
cust_0_856	raw1625	sdwv9	stok_0_454	raw455	sdsd3

datafile name	raw number	device name	datafile name	raw number	device name
cust_0_857	raw1626	sdtn10	stok_0_455	raw456	sdou5
cust_0_858	raw1627	sdwx8	stok_0_456	raw457	sdsr1
cust_0_859	raw1628	sdwx9	stok_0_457	raw458	sdsr2
cust_0_86	raw855	sdkf6	stok_0_458	raw459	sdsr3
cust_0_860	raw1629	sdtr10	stok_0_459	raw460	sdpb5
cust_0_861	raw1630	sdxc8	stok_0_46	raw47	sdap3
cust_0_862	raw1631	sdxc9	stok_0_460	raw461	sdst1
cust_0_863	raw1632	sdtx10	stok_0_461	raw462	sdst2
cust_0_864	raw1633	sdxe8	stok_0_462	raw463	sdst3
cust_0_865	raw1634	sdxe9	stok_0_463	raw464	sdpf5
cust_0_866	raw1635	sduub10	stok_0_464	raw465	sdsv1
cust_0_867	raw1636	sdxcg8	stok_0_465	raw466	sdsv2
cust_0_868	raw1637	sdxcg9	stok_0_466	raw467	sdsv3
cust_0_869	raw1638	sduf10	stok_0_467	raw468	sdpm5
cust_0_87	raw856	sdkf7	stok_0_468	raw469	sdta1
cust_0_870	raw1639	sdxl8	stok_0_469	raw470	sdta2
cust_0_871	raw1640	sdxl9	stok_0_47	raw48	sdgc3
cust_0_872	raw1641	sduj10	stok_0_470	raw471	sdta3
cust_0_873	raw1642	sdxn8	stok_0_471	raw472	sdpv5
cust_0_874	raw1643	sdxn9	stok_0_472	raw473	sdte1
cust_0_875	raw1644	sduun10	stok_0_473	raw474	sdte2
cust_0_876	raw1645	sdxp8	stok_0_474	raw475	sdte3
cust_0_877	raw1646	sdxp9	stok_0_475	raw476	sdpz5
cust_0_878	raw1647	sdur10	stok_0_476	raw477	sdte1
cust_0_879	raw1648	sdxu8	stok_0_477	raw478	sdte2
cust_0_88	raw857	sdkh6	stok_0_478	raw479	sdte3
cust_0_880	raw1649	sdxu9	stok_0_479	raw480	sdqg5
cust_0_881	raw1650	sduv10	stok_0_48	raw49	sdaw1
cust_0_882	raw1651	sdxw8	stok_0_480	raw481	sdjt1
cust_0_883	raw1652	sdxw9	stok_0_481	raw482	sdjt2
cust_0_884	raw1653	sduz10	stok_0_482	raw483	sdjt3
cust_0_885	raw1654	sdxxy8	stok_0_483	raw484	sdqn5
cust_0_886	raw1655	sdxxy9	stok_0_484	raw485	sdtl1
cust_0_887	raw1656	sduvd10	stok_0_485	raw486	sdtl2
cust_0_888	raw1657	sdyd8	stok_0_486	raw487	sdtl3
cust_0_889	raw1658	sdyd9	stok_0_487	raw488	sdqr5
cust_0_89	raw858	sdkh7	stok_0_488	raw489	sdtn1
cust_0_890	raw1659	sdvh10	stok_0_489	raw490	sdtn2
cust_0_891	raw1660	sdyf8	stok_0_49	raw50	sdaw2
cust_0_892	raw1661	sdyf9	stok_0_490	raw491	sdtn3
cust_0_893	raw1662	sduvl10	stok_0_491	raw492	sdqy5
cust_0_894	raw1663	sdyh8	stok_0_492	raw493	sdtpl
cust_0_895	raw1664	sdyh9	stok_0_493	raw494	sdtpl2
cust_0_896	raw1665	sdvs10	stok_0_494	raw495	sdtpl3
cust_0_897	raw1666	sdym8	stok_0_495	raw496	sdrh5
cust_0_898	raw1667	sdym9	stok_0_496	raw497	sdtr1
cust_0_899	raw1668	sdvz10	stok_0_497	raw498	sdtr2
cust_0_9	raw778	sdv7	stok_0_498	raw499	sdtr3
cust_0_90	raw859	sdkm6	stok_0_499	raw500	sdr15
cust_0_900	raw1669	sdyo8	stok_0_5	raw6	sdm2
cust_0_901	raw1670	sdyo9	stok_0_50	raw51	sdaw3
cust_0_902	raw1671	sdwd10	stok_0_500	raw501	sdt11
cust_0_903	raw1672	sdyq8	stok_0_501	raw502	sdt12
cust_0_904	raw1673	sdyq9	stok_0_502	raw503	sdt13
cust_0_905	raw1674	sdwk10	stok_0_503	raw504	sdrs5
cust_0_906	raw1675	sdyv8	stok_0_504	raw505	sdtx1
cust_0_907	raw1676	sdyv9	stok_0_505	raw506	sdtx2
cust_0_908	raw1677	sdwt10	stok_0_506	raw507	sdtx3
cust_0_909	raw1678	sdyx8	stok_0_507	raw508	sdrz5

datafile name	raw number	device name	datafile name	raw number	device name
cust_0_91	raw860	sdkm7	stok_0_508	raw509	sdtz1
cust_0_910	raw1679	sdyx9	stok_0_509	raw510	sdtz2
cust_0_911	raw1680	sdwx10	stok_0_51	raw52	sdge1
cust_0_912	raw1681	sdyz8	stok_0_510	raw511	sdtz3
cust_0_913	raw1682	sdyz9	stok_0_511	raw512	sdsd5
cust_0_914	raw1683	sdxe10	stok_0_512	raw513	sdub1
cust_0_915	raw1684	sdze8	stok_0_513	raw514	sdub2
cust_0_916	raw1685	sdze9	stok_0_514	raw515	sdub3
cust_0_917	raw1686	sdxl10	stok_0_515	raw516	sdsd5
cust_0_918	raw1687	sdzg8	stok_0_516	raw517	sdud1
cust_0_919	raw1688	sdzg9	stok_0_517	raw518	sdud2
cust_0_92	raw861	sdko6	stok_0_518	raw519	sdud3
cust_0_920	raw1689	sdxp10	stok_0_519	raw520	sdsr5
cust_0_921	raw1690	sdzi8	stok_0_52	raw53	sday1
cust_0_922	raw1691	sdzi9	stok_0_520	raw521	sduf1
cust_0_923	raw1692	sdxw10	stok_0_521	raw522	sduf2
cust_0_924	raw1693	sdzp8	stok_0_522	raw523	sduf3
cust_0_925	raw1694	sdzp9	stok_0_523	raw524	sdsd5
cust_0_926	raw1695	sdzd10	stok_0_524	raw525	sduh1
cust_0_927	raw1696	sdzr8	stok_0_525	raw526	sduh2
cust_0_928	raw1697	sdzr9	stok_0_526	raw527	sduh3
cust_0_929	raw1698	sdyh10	stok_0_527	raw528	sdtc5
cust_0_93	raw862	sdko7	stok_0_528	raw529	sduj1
cust_0_930	raw1699	sdzt8	stok_0_529	raw530	sduj2
cust_0_931	raw1700	sdzt9	stok_0_53	raw54	sday2
cust_0_932	raw1701	sdyo10	stok_0_530	raw531	sduj3
cust_0_933	raw1702	sdzy8	stok_0_531	raw532	sdtj5
cust_0_934	raw1703	sdzy9	stok_0_532	raw533	sdul1
cust_0_935	raw1704	sdyv10	stok_0_533	raw534	sdul2
cust_0_936	raw1705	sdaaa8	stok_0_534	raw535	sdul3
cust_0_937	raw1706	sdaaa9	stok_0_535	raw536	sdtm5
cust_0_938	raw1707	sdyz10	stok_0_536	raw537	sdun1
cust_0_939	raw1708	sdaac8	stok_0_537	raw538	sdun2
cust_0_94	raw863	sdkq6	stok_0_538	raw539	sdun3
cust_0_940	raw1709	sdaac9	stok_0_539	raw540	sdr5
cust_0_941	raw1710	sdzg10	stok_0_54	raw55	sday3
cust_0_942	raw1711	sdaah8	stok_0_540	raw541	sdup1
cust_0_943	raw1712	sdaah9	stok_0_541	raw542	sdup2
cust_0_944	raw1713	sdzp10	stok_0_542	raw543	sdup3
cust_0_945	raw1714	sdaaj8	stok_0_543	raw544	sdtx5
cust_0_946	raw1715	sdaaj9	stok_0_544	raw545	sdur1
cust_0_947	raw1716	sdzt10	stok_0_545	raw546	sdur2
cust_0_948	raw1717	sdaal8	stok_0_546	raw547	sdur3
cust_0_949	raw1718	sdaal9	stok_0_547	raw548	sdub5
cust_0_95	raw864	sdkq7	stok_0_548	raw549	sdut1
cust_0_950	raw1719	sdaaa10	stok_0_549	raw550	sdut2
cust_0_951	raw1720	sdaaq8	stok_0_55	raw56	sdge2
cust_0_952	raw1721	sdaaq9	stok_0_550	raw551	sdut3
cust_0_953	raw1722	sdaah10	stok_0_551	raw552	sduf5
cust_0_954	raw1723	sdaas8	stok_0_552	raw553	sduv1
cust_0_955	raw1724	sdaas9	stok_0_553	raw554	sduv2
cust_0_956	raw1725	sdaal10	stok_0_554	raw555	sduv3
cust_0_957	raw1726	sdaau8	stok_0_555	raw556	sduj5
cust_0_958	raw1727	sdaau9	stok_0_556	raw557	sdux1
cust_0_959	raw1728	sdaas10	stok_0_557	raw558	sdux2
cust_0_96	raw865	sdkv6	stok_0_558	raw559	sdux3
cust_0_97	raw866	sdkv7	stok_0_559	raw560	sdun5
cust_0_98	raw867	sdkx6	stok_0_56	raw57	sdba1
cust_0_99	raw868	sdkx7	stok_0_560	raw561	sduz1

datafile name	raw number	device name	datafile name	raw number	device name
dist_0_0	raw2888	sd17	stok_0_561	raw562	sduz2
hist_0_0	raw3080	dm-192	stok_0_562	raw563	sduz3
hist_0_1	raw3081	dm-193	stok_0_563	raw564	sdur5
hist_0_10	raw3090	dm-202	stok_0_564	raw565	sdvb1
hist_0_100	raw3180	dm-4	stok_0_565	raw566	sdvb2
hist_0_101	raw3181	dm-5	stok_0_566	raw567	sdvb3
hist_0_102	raw3182	dm-6	stok_0_567	raw568	sduv5
hist_0_103	raw3183	dm-7	stok_0_568	raw569	sdvd1
hist_0_104	raw3184	dm-8	stok_0_569	raw570	sdvd2
hist_0_105	raw3185	dm-9	stok_0_57	raw58	sdba2
hist_0_106	raw3186	dm-10	stok_0_570	raw571	sdvd3
hist_0_107	raw3187	dm-11	stok_0_571	raw572	sduz5
hist_0_108	raw3188	dm-12	stok_0_572	raw573	sdvf1
hist_0_109	raw3189	dm-13	stok_0_573	raw574	sdvf2
hist_0_11	raw3091	dm-203	stok_0_574	raw575	sdvf3
hist_0_110	raw3190	dm-14	stok_0_575	raw576	sdvd5
hist_0_111	raw3191	dm-15	stok_0_576	raw577	sdvh1
hist_0_112	raw3192	dm-16	stok_0_577	raw578	sdvh2
hist_0_113	raw3193	dm-17	stok_0_578	raw579	sdvh3
hist_0_114	raw3194	dm-18	stok_0_579	raw580	sdvh5
hist_0_115	raw3195	dm-19	stok_0_58	raw59	sdba3
hist_0_116	raw3196	dm-20	stok_0_580	raw581	sdvj1
hist_0_12	raw3092	dm-204	stok_0_581	raw582	sdvj2
hist_0_120	raw3200	dm-24	stok_0_582	raw583	sdvj3
hist_0_124	raw3204	dm-28	stok_0_583	raw584	sdvl5
hist_0_125	raw3205	dm-29	stok_0_584	raw585	sdvl1
hist_0_126	raw3206	dm-30	stok_0_585	raw586	sdvl2
hist_0_128	raw3208	dm-32	stok_0_586	raw587	sdvl3
hist_0_129	raw3209	dm-33	stok_0_587	raw588	sdvs5
hist_0_13	raw3093	dm-205	stok_0_588	raw589	sdvq1
hist_0_130	raw3210	dm-34	stok_0_589	raw590	sdvq2
hist_0_132	raw3212	dm-36	stok_0_59	raw60	sdge3
hist_0_134	raw3214	dm-38	stok_0_590	raw591	sdvq3
hist_0_136	raw3216	dm-40	stok_0_591	raw592	sdvz5
hist_0_137	raw3217	dm-41	stok_0_592	raw593	sdvs1
hist_0_138	raw3218	dm-42	stok_0_593	raw594	sdvs2
hist_0_139	raw3219	dm-43	stok_0_594	raw595	sdvs3
hist_0_14	raw3094	dm-206	stok_0_595	raw596	sdwd5
hist_0_140	raw3220	dm-44	stok_0_596	raw597	sdvu1
hist_0_141	raw3221	dm-45	stok_0_597	raw598	sdvu2
hist_0_142	raw3222	dm-46	stok_0_598	raw599	sdvu3
hist_0_143	raw3223	dm-47	stok_0_599	raw600	sdwk5
hist_0_144	raw3224	dm-48	stok_0_6	raw7	sdm3
hist_0_147	raw3227	dm-51	stok_0_60	raw61	sdbf1
hist_0_15	raw3095	dm-207	stok_0_600	raw601	sdvz1
hist_0_150	raw3230	dm-54	stok_0_601	raw602	sdvz2
hist_0_151	raw3231	dm-55	stok_0_602	raw603	sdvz3
hist_0_155	raw3235	dm-59	stok_0_603	raw604	sdwt5
hist_0_156	raw3236	dm-60	stok_0_604	raw605	sdwb1
hist_0_157	raw3237	dm-61	stok_0_605	raw606	sdwb2
hist_0_159	raw3239	dm-63	stok_0_606	raw607	sdwb3
hist_0_16	raw3096	dm-208	stok_0_607	raw608	sdwx5
hist_0_164	raw3244	dm-68	stok_0_608	raw609	sdwd1
hist_0_166	raw3246	dm-70	stok_0_609	raw610	sdwd2
hist_0_17	raw3097	dm-209	stok_0_61	raw62	sdbf2
hist_0_170	raw3250	dm-74	stok_0_610	raw611	sdwd3
hist_0_171	raw3251	dm-75	stok_0_611	raw612	sdxe5
hist_0_18	raw3098	dm-210	stok_0_612	raw613	sdwi1
hist_0_19	raw3099	dm-211	stok_0_613	raw614	sdwi2
hist_0_2	raw3082	dm-194	stok_0_614	raw615	sdwi3

datafile name	raw number	device name	datafile name	raw number	device name
hist_0_20	raw3100	dm-212	stok_0_615	raw616	sdxl5
hist_0_21	raw3101	dm-213	stok_0_616	raw617	sdwk1
hist_0_22	raw3102	dm-214	stok_0_617	raw618	sdwk2
hist_0_23	raw3103	dm-215	stok_0_618	raw619	sdwk3
hist_0_24	raw3104	dm-216	stok_0_619	raw620	sdxp5
hist_0_25	raw3105	dm-217	stok_0_62	raw63	sdbf3
hist_0_26	raw3106	dm-218	stok_0_620	raw621	sdwm1
hist_0_27	raw3107	dm-219	stok_0_621	raw622	sdwm2
hist_0_28	raw3108	dm-220	stok_0_622	raw623	sdwm3
hist_0_29	raw3109	dm-221	stok_0_623	raw624	sdwx5
hist_0_3	raw3083	dm-195	stok_0_624	raw625	sdwt1
hist_0_30	raw3110	dm-222	stok_0_625	raw626	sdwt2
hist_0_31	raw3111	dm-223	stok_0_626	raw627	sdwt3
hist_0_32	raw3112	dm-224	stok_0_627	raw628	sdyd5
hist_0_33	raw3113	dm-225	stok_0_628	raw629	sdwv1
hist_0_34	raw3114	dm-226	stok_0_629	raw630	sdwv2
hist_0_35	raw3115	dm-227	stok_0_63	raw64	sdgg1
hist_0_36	raw3116	dm-228	stok_0_630	raw631	sdwv3
hist_0_37	raw3117	dm-229	stok_0_631	raw632	sdyh5
hist_0_38	raw3118	dm-230	stok_0_632	raw633	sdwx1
hist_0_39	raw3119	dm-231	stok_0_633	raw634	sdwx2
hist_0_4	raw3084	dm-196	stok_0_634	raw635	sdwx3
hist_0_40	raw3120	dm-232	stok_0_635	raw636	sdyo5
hist_0_41	raw3121	dm-233	stok_0_636	raw637	sdxc1
hist_0_42	raw3122	dm-234	stok_0_637	raw638	sdxc2
hist_0_43	raw3123	dm-235	stok_0_638	raw639	sdxc3
hist_0_44	raw3124	dm-236	stok_0_639	raw640	sdyv5
hist_0_45	raw3125	dm-237	stok_0_64	raw65	sdbh1
hist_0_46	raw3126	dm-238	stok_0_640	raw641	sdxe1
hist_0_47	raw3127	dm-239	stok_0_641	raw642	sdxe2
hist_0_48	raw3128	dm-240	stok_0_642	raw643	sdxe3
hist_0_49	raw3129	dm-241	stok_0_643	raw644	sdyz5
hist_0_5	raw3085	dm-197	stok_0_644	raw645	sdxc1
hist_0_50	raw3130	dm-242	stok_0_645	raw646	sdxc2
hist_0_51	raw3131	dm-243	stok_0_646	raw647	sdxc3
hist_0_52	raw3132	dm-244	stok_0_647	raw648	sdzg5
hist_0_53	raw3133	dm-245	stok_0_648	raw649	sdxl1
hist_0_54	raw3134	dm-246	stok_0_649	raw650	sdxl2
hist_0_55	raw3135	dm-247	stok_0_65	raw66	sdbh2
hist_0_56	raw3136	dm-248	stok_0_650	raw651	sdxl3
hist_0_57	raw3137	dm-249	stok_0_651	raw652	sdzp5
hist_0_58	raw3138	dm-250	stok_0_652	raw653	sdxn1
hist_0_59	raw3139	dm-251	stok_0_653	raw654	sdxn2
hist_0_6	raw3086	dm-198	stok_0_654	raw655	sdxn3
hist_0_60	raw3140	dm-252	stok_0_655	raw656	sdzt5
hist_0_61	raw3141	dm-253	stok_0_656	raw657	sdxp1
hist_0_62	raw3142	dm-254	stok_0_657	raw658	sdxp2
hist_0_63	raw3143	dm-255	stok_0_658	raw659	sdxp3
hist_0_64	raw3144	dm-256	stok_0_659	raw660	sdaaa5
hist_0_65	raw3145	dm-257	stok_0_66	raw67	sdbh3
hist_0_66	raw3146	dm-258	stok_0_660	raw661	sdxu1
hist_0_67	raw3147	dm-259	stok_0_661	raw662	sdxu2
hist_0_68	raw3148	dm-260	stok_0_662	raw663	sdxu3
hist_0_69	raw3149	dm-261	stok_0_663	raw664	sdaah5
hist_0_7	raw3087	dm-199	stok_0_664	raw665	sdwx1
hist_0_70	raw3150	dm-262	stok_0_665	raw666	sdwx2
hist_0_71	raw3151	dm-263	stok_0_666	raw667	sdwx3
hist_0_72	raw3152	dm-264	stok_0_667	raw668	sdaal5
hist_0_73	raw3153	dm-265	stok_0_668	raw669	sdxy1
hist_0_74	raw3154	dm-266	stok_0_669	raw670	sdxy2

datafile name	raw number	device name	datafile name	raw number	device name
hist_0_75	raw3155	dm-267	stok_0_67	raw68	sdgg2
hist_0_76	raw3156	dm-268	stok_0_670	raw671	sdxy3
hist_0_77	raw3157	dm-269	stok_0_671	raw672	sdaas5
hist_0_78	raw3158	dm-270	stok_0_672	raw673	sdyd1
hist_0_79	raw3159	dm-271	stok_0_673	raw674	sdyd2
hist_0_8	raw3088	dm-200	stok_0_674	raw675	sdyd3
hist_0_80	raw3160	dm-272	stok_0_675	raw676	sdar5
hist_0_81	raw3161	dm-273	stok_0_676	raw677	sdyf1
hist_0_82	raw3162	dm-274	stok_0_677	raw678	sdyf2
hist_0_83	raw3163	dm-275	stok_0_678	raw679	sdyf3
hist_0_84	raw3164	dm-276	stok_0_679	raw680	sdga5
hist_0_85	raw3165	dm-277	stok_0_68	raw69	sdbj1
hist_0_86	raw3166	dm-278	stok_0_680	raw681	sdyh1
hist_0_87	raw3167	dm-279	stok_0_681	raw682	sdyh2
hist_0_88	raw3168	dm-280	stok_0_682	raw683	sdyh3
hist_0_89	raw3169	dm-281	stok_0_683	raw684	sdge5
hist_0_9	raw3089	dm-201	stok_0_684	raw685	sdym1
hist_0_90	raw3170	dm-282	stok_0_685	raw686	sdym2
hist_0_91	raw3171	dm-283	stok_0_686	raw687	sdym3
hist_0_92	raw3172	dm-284	stok_0_687	raw688	sdgi5
hist_0_93	raw3173	dm-285	stok_0_688	raw689	sdyo1
hist_0_94	raw3174	dm-286	stok_0_689	raw690	sdyo2
hist_0_95	raw3175	dm-287	stok_0_69	raw70	sdbj2
hist_0_96	raw3176	dm-0	stok_0_690	raw691	sdyo3
hist_0_97	raw3177	dm-1	stok_0_691	raw692	sdgm5
hist_0_98	raw3178	dm-2	stok_0_692	raw693	sdyq1
hist_0_99	raw3179	dm-3	stok_0_693	raw694	sdyq2
icust1_0_0	raw2504	sd15	stok_0_694	raw695	sdyq3
icust1_0_1	raw2505	sdn5	stok_0_695	raw696	sdid5
icust1_0_10	raw2514	sdao5	stok_0_696	raw697	sdyv1
icust1_0_11	raw2515	sdaq5	stok_0_697	raw698	sdyv2
icust1_0_12	raw2516	sdax5	stok_0_698	raw699	sdyv3
icust1_0_13	raw2517	sdaz5	stok_0_699	raw700	sdih5
icust1_0_14	raw2518	sdbb5	stok_0_7	raw8	sdar2
icust1_0_15	raw2519	sdbg5	stok_0_70	raw71	sdbj3
icust1_0_16	raw2520	sdbi5	stok_0_700	raw701	sdyx1
icust1_0_17	raw2521	sdbk5	stok_0_701	raw702	sdyx2
icust1_0_18	raw2522	sdfh5	stok_0_702	raw703	sdyx3
icust1_0_19	raw2523	sdfj5	stok_0_703	raw704	sdil5
icust1_0_2	raw2506	sdp5	stok_0_704	raw705	sdyz1
icust1_0_20	raw2524	sdf15	stok_0_705	raw706	sdyz2
icust1_0_21	raw2525	sdfq5	stok_0_706	raw707	sdyz3
icust1_0_22	raw2526	sdfs5	stok_0_707	raw708	sdip5
icust1_0_23	raw2527	sdfu5	stok_0_708	raw709	sdze1
icust1_0_24	raw2528	sdhk5	stok_0_709	raw710	sdze2
icust1_0_25	raw2529	sdhm5	stok_0_71	raw72	sdgg3
icust1_0_26	raw2530	sdho5	stok_0_710	raw711	sdze3
icust1_0_27	raw2531	sdhv5	stok_0_711	raw712	sdnm5
icust1_0_28	raw2532	sdhx5	stok_0_712	raw713	sdzg1
icust1_0_29	raw2533	sdhz5	stok_0_713	raw714	sdzg2
icust1_0_3	raw2507	sdu5	stok_0_714	raw715	sdzg3
icust1_0_30	raw2534	sdiu5	stok_0_715	raw716	sdrc5
icust1_0_31	raw2535	sdiw5	stok_0_716	raw717	sdzi1
icust1_0_32	raw2536	sdii5	stok_0_717	raw718	sdzi2
icust1_0_33	raw2537	sdjd5	stok_0_718	raw719	sdzi3
icust1_0_34	raw2538	sdjf5	stok_0_719	raw720	sdwo5
icust1_0_35	raw2539	sdjh5	stok_0_72	raw73	sdfg1
icust1_0_36	raw2540	sdjm5	stok_0_720	raw721	sdzp1
icust1_0_37	raw2541	sdjo5	stok_0_721	raw722	sdzp2

datafile name	raw number	device name	datafile name	raw number	device name
icust1_0_38	raw2542	sdjq5	stok_0_722	raw723	sdzp3
icust1_0_39	raw2543	sdjv5	stok_0_723	raw724	sdf5
icust1_0_4	raw2508	sdw5	stok_0_724	raw725	sdzr1
icust1_0_40	raw2544	sdjx5	stok_0_725	raw726	sdzr2
icust1_0_41	raw2545	sdjz5	stok_0_726	raw727	sdzr3
icust1_0_42	raw2546	sdke5	stok_0_727	raw728	sdzr5
icust1_0_43	raw2547	sdkg5	stok_0_728	raw729	sdzt1
icust1_0_44	raw2548	sdki5	stok_0_729	raw730	sdzt2
icust1_0_45	raw2549	sdkn5	stok_0_73	raw74	sdfg2
icust1_0_46	raw2550	sdkp5	stok_0_730	raw731	sdzt3
icust1_0_47	raw2551	sdkr5	stok_0_731	raw732	sdkq5
icust1_0_5	raw2509	sdj5	stok_0_732	raw733	sdzy1
icust1_0_6	raw2510	sdad5	stok_0_733	raw734	sdzy2
icust1_0_7	raw2511	sdaf5	stok_0_734	raw735	sdzy3
icust1_0_8	raw2512	sdah5	stok_0_735	raw736	sdzy5
icust1_0_9	raw2513	sdam5	stok_0_736	raw737	sdaaa1
icust2_0_0	raw2312	sdll2	stok_0_737	raw738	sdaaa2
icust2_0_1	raw2313	sdn2	stok_0_738	raw739	sdaaa3
icust2_0_10	raw2322	sdao2	stok_0_739	raw740	sdnk5
icust2_0_100	raw2412	sdqz2	stok_0_74	raw75	sdfg3
icust2_0_101	raw2413	sdrb2	stok_0_740	raw741	sdaac1
icust2_0_102	raw2414	sdri2	stok_0_741	raw742	sdaac2
icust2_0_103	raw2415	sdrk2	stok_0_742	raw743	sdaac3
icust2_0_104	raw2416	sdrm2	stok_0_743	raw744	sdaac5
icust2_0_105	raw2417	sdr2	stok_0_744	raw745	sdaah1
icust2_0_106	raw2418	sdr2	stok_0_745	raw746	sdaah2
icust2_0_107	raw2419	sdrv2	stok_0_746	raw747	sdaah3
icust2_0_108	raw2420	sdsa2	stok_0_747	raw748	sdqi5
icust2_0_109	raw2421	sdse2	stok_0_748	raw749	sdaaj1
icust2_0_11	raw2323	sdaq2	stok_0_749	raw750	sdaaj2
icust2_0_110	raw2422	sdse2	stok_0_75	raw76	sdgi1
icust2_0_111	raw2423	sdsl2	stok_0_750	raw751	sdaaj3
icust2_0_112	raw2424	sdsl2	stok_0_751	raw752	sdaaj5
icust2_0_113	raw2425	sdsn2	stok_0_752	raw753	sdaa1
icust2_0_114	raw2426	sdss2	stok_0_753	raw754	sdaa2
icust2_0_115	raw2427	sdsu2	stok_0_754	raw755	sdaa3
icust2_0_116	raw2428	sdsw2	stok_0_755	raw756	sdte5
icust2_0_117	raw2429	sdtb2	stok_0_756	raw757	sdaaq1
icust2_0_118	raw2430	sdt2	stok_0_757	raw758	sdaaq2
icust2_0_119	raw2431	sdtf2	stok_0_758	raw759	sdaaq3
icust2_0_12	raw2324	sdax2	stok_0_759	raw760	sdaaq5
icust2_0_120	raw2432	sdvi2	stok_0_76	raw77	sdfi1
icust2_0_121	raw2433	sdvk2	stok_0_760	raw761	sdaas1
icust2_0_122	raw2434	sdvm2	stok_0_761	raw762	sdaas2
icust2_0_123	raw2435	sdvr2	stok_0_762	raw763	sdaas3
icust2_0_124	raw2436	sdvt2	stok_0_763	raw764	sdvf5
icust2_0_125	raw2437	sdvv2	stok_0_764	raw765	sdaau1
icust2_0_126	raw2438	sdwa2	stok_0_765	raw766	sdaau2
icust2_0_127	raw2439	sdwe2	stok_0_766	raw767	sdaau3
icust2_0_128	raw2440	sdwe2	stok_0_767	raw768	sdaau5
icust2_0_129	raw2441	sdwj2	stok_0_768	raw2913	sdhm7
icust2_0_13	raw2325	sdaz2	stok_0_77	raw78	sdfi2
icust2_0_130	raw2442	sdwl2	stok_0_78	raw79	sdfi3
icust2_0_131	raw2443	sdwn2	stok_0_79	raw80	sdgi2
icust2_0_132	raw2444	sdwu2	stok_0_8	raw9	sdo1
icust2_0_133	raw2445	sdww2	stok_0_80	raw81	sdfl1
icust2_0_134	raw2446	sdwy2	stok_0_81	raw82	sdfl2
icust2_0_135	raw2447	sdxd2	stok_0_82	raw83	sdfl3
icust2_0_136	raw2448	sdxf2	stok_0_83	raw84	sdgi3
icust2_0_137	raw2449	sdxx2	stok_0_84	raw85	sdfl1

datafile name	raw number	device name	datafile name	raw number	device name
icust2_0_138	raw2450	sdxm2	stok_0_85	raw86	sdfp2
icust2_0_139	raw2451	sdxo2	stok_0_86	raw87	sdfp3
icust2_0_14	raw2326	sdbb2	stok_0_87	raw88	sdgk1
icust2_0_140	raw2452	sdxq2	stok_0_88	raw89	sdfrr1
icust2_0_141	raw2453	sdxv2	stok_0_89	raw90	sdfrr2
icust2_0_142	raw2454	sdxz2	stok_0_9	raw10	sdo2
icust2_0_143	raw2455	sdxz2	stok_0_90	raw91	sdfrr3
icust2_0_144	raw2456	sdye2	stok_0_91	raw92	sdgk2
icust2_0_145	raw2457	sdyg2	stok_0_92	raw93	sdftr1
icust2_0_146	raw2458	sdyi2	stok_0_93	raw94	sdftr2
icust2_0_147	raw2459	sdyn2	stok_0_94	raw95	sdftr3
icust2_0_148	raw2460	sdyp2	stok_0_95	raw96	sdgk3
icust2_0_149	raw2461	sdyr2	stok_0_96	raw97	sdfi5
icust2_0_15	raw2327	sdbg2	stok_0_97	raw98	sdju5
icust2_0_150	raw2462	sdyy2	stok_0_98	raw99	sdma5
icust2_0_151	raw2463	sdyy2	stok_0_99	raw100	sdgm1
icust2_0_152	raw2464	sdza2	system_1	raw2891	sdu7
icust2_0_153	raw2465	sdzf2	system_10	raw3008	sdvi7
icust2_0_154	raw2466	sdzh2	system_11	raw3009	sdvk7
icust2_0_155	raw2467	sdzj2	system_12	raw3010	sdvm7
icust2_0_156	raw2468	sdzq2	system_13	raw3011	sdvr7
icust2_0_157	raw2469	sdzs2	system_14	raw3012	sdvt7
icust2_0_158	raw2470	sdzu2	system_15	raw3013	sdvv7
icust2_0_159	raw2471	sdzz2	system_16	raw3014	sdwa7
icust2_0_16	raw2328	sdbi2	system_17	raw3015	sdwc7
icust2_0_160	raw2472	sdaab2	system_18	raw3016	sdwe7
icust2_0_161	raw2473	sdaad2	system_19	raw3017	sdwj7
icust2_0_162	raw2474	sdaai2	system_2	raw3000	sds17
icust2_0_163	raw2475	sdaak2	system_20	raw3018	sdw17
icust2_0_164	raw2476	sdaam2	system_21	raw3019	sdwn7
icust2_0_165	raw2477	sdaar2	system_22	raw3020	sdwu7
icust2_0_166	raw2478	sdaat2	system_23	raw3021	sdww7
icust2_0_167	raw2479	sdaav2	system_24	raw3022	sdwy7
icust2_0_168	raw2480	sdtk2	system_25	raw3023	sdx7
icust2_0_169	raw2481	sdtm2	system_26	raw3024	sdx7
icust2_0_17	raw2329	sdbk2	system_27	raw3025	sdxh7
icust2_0_170	raw2482	sdto2	system_28	raw3026	sdxm7
icust2_0_171	raw2483	sdtq2	system_29	raw3027	sdxo7
icust2_0_172	raw2484	sdt2	system_3	raw3001	sdsn7
icust2_0_173	raw2485	sdtu2	system_30	raw3028	sdxq7
icust2_0_174	raw2486	sdy2	system_31	raw3029	sdxv7
icust2_0_175	raw2487	sdua2	system_32	raw3030	sdx7
icust2_0_176	raw2488	sduc2	system_33	raw3031	sdxz7
icust2_0_177	raw2489	sdue2	system_34	raw3032	sdye7
icust2_0_178	raw2490	sdug2	system_35	raw3033	sdyg7
icust2_0_179	raw2491	sdui2	system_36	raw3034	sdyi7
icust2_0_18	raw2330	sdfh2	system_37	raw3035	sdyn7
icust2_0_180	raw2492	sduk2	system_38	raw3036	sdyp7
icust2_0_181	raw2493	sdu2	system_39	raw3037	sdyr7
icust2_0_182	raw2494	sduo2	system_4	raw3002	sdss7
icust2_0_183	raw2495	sduq2	system_40	raw3038	sdyw7
icust2_0_184	raw2496	sdus2	system_41	raw3039	sdyy7
icust2_0_185	raw2497	sduu2	system_42	raw3040	sdza7
icust2_0_186	raw2498	sduw2	system_43	raw3041	sdzf7
icust2_0_187	raw2499	sduy2	system_44	raw3042	sdzh7
icust2_0_188	raw2500	sdva2	system_45	raw3043	sdzj7
icust2_0_189	raw2501	sdvc2	system_46	raw3044	sdzq7
icust2_0_19	raw2331	sdvf2	system_47	raw3045	sdzs7
icust2_0_190	raw2502	sdve2	system_48	raw3046	sdzu7
icust2_0_191	raw2503	sdvg2	system_49	raw3047	sdzz7

datafile name	raw number	device name	datafile name	raw number	device name
icust2_0_2	raw2314	sdp2	system_5	raw3003	sdsu7
icust2_0_20	raw2332	sdf12	system_50	raw3048	sdaab7
icust2_0_21	raw2333	sdfq2	system_51	raw3049	sdaad7
icust2_0_22	raw2334	sdfs2	system_52	raw3050	sdaai7
icust2_0_23	raw2335	sdfu2	system_53	raw3051	sdaak7
icust2_0_24	raw2336	sdhk2	system_54	raw3052	sdaam7
icust2_0_25	raw2337	sdhm2	system_55	raw3053	sdaar7
icust2_0_26	raw2338	sdho2	system_56	raw3054	sdaat7
icust2_0_27	raw2339	sdhv2	system_57	raw3055	sdaav7
icust2_0_28	raw2340	sdhx2	system_58	raw3056	sdtk7
icust2_0_29	raw2341	sdhz2	system_59	raw3057	sdtm7
icust2_0_3	raw2315	sdu2	system_6	raw3004	sds7
icust2_0_30	raw2342	sdiu2	system_60	raw3058	sdto7
icust2_0_31	raw2343	sdiw2	system_61	raw3059	sdtq7
icust2_0_32	raw2344	sdiy2	system_62	raw3060	sdt7
icust2_0_33	raw2345	sdjd2	system_63	raw3061	sdtu7
icust2_0_34	raw2346	sdjf2	system_64	raw3062	sdy7
icust2_0_35	raw2347	sdjh2	system_65	raw3063	sdua7
icust2_0_36	raw2348	sdjm2	system_66	raw3064	sduc7
icust2_0_37	raw2349	sdjo2	system_67	raw3065	sdue7
icust2_0_38	raw2350	sdjq2	system_68	raw3066	sdu7
icust2_0_39	raw2351	sdjv2	system_69	raw3067	sdui7
icust2_0_4	raw2316	sdw2	system_7	raw3005	sdtb7
icust2_0_40	raw2352	sdjx2	system_70	raw3068	sduk7
icust2_0_41	raw2353	sdjz2	system_71	raw3069	sdum7
icust2_0_42	raw2354	sdke2	system_72	raw3070	sduo7
icust2_0_43	raw2355	sdkg2	system_73	raw3071	sduq7
icust2_0_44	raw2356	sdki2	system_74	raw3072	sdu7
icust2_0_45	raw2357	sdkn2	system_8	raw3006	sdt7
icust2_0_46	raw2358	sdkp2	system_9	raw3007	sdtf7
icust2_0_47	raw2359	sdkr2	temp_0_0	raw1928	sd12
icust2_0_48	raw2360	sdkw2	temp_0_1	raw1929	sdm12
icust2_0_49	raw2361	sdky2	temp_0_10	raw1938	sdan12
icust2_0_5	raw2317	sd2	temp_0_100	raw2028	sdqy12
icust2_0_50	raw2362	sdla2	temp_0_101	raw2029	sdra12
icust2_0_51	raw2363	sdlf2	temp_0_102	raw2030	sdrh12
icust2_0_52	raw2364	sdlh2	temp_0_103	raw2031	sdrj12
icust2_0_53	raw2365	sdlj2	temp_0_104	raw2032	sdr12
icust2_0_54	raw2366	sdlo2	temp_0_105	raw2033	sdrq12
icust2_0_55	raw2367	sdllq2	temp_0_106	raw2034	sdrs12
icust2_0_56	raw2368	sdls2	temp_0_107	raw2035	sdru12
icust2_0_57	raw2369	sdlx2	temp_0_108	raw2036	sdrz12
icust2_0_58	raw2370	sdlz2	temp_0_109	raw2037	sdsb12
icust2_0_59	raw2371	sdm2	temp_0_11	raw1939	sdap12
icust2_0_6	raw2318	sdad2	temp_0_110	raw2038	sdsd12
icust2_0_60	raw2372	sdmg2	temp_0_111	raw2039	sdsi12
icust2_0_61	raw2373	sdmi2	temp_0_112	raw2040	sds12
icust2_0_62	raw2374	sdmk2	temp_0_113	raw2041	sds12
icust2_0_63	raw2375	sdmp2	temp_0_114	raw2042	sdsr12
icust2_0_64	raw2376	sdmr2	temp_0_115	raw2043	sdst12
icust2_0_65	raw2377	sdmt2	temp_0_116	raw2044	sds12
icust2_0_66	raw2378	sdmy2	temp_0_117	raw2045	sdta12
icust2_0_67	raw2379	sdna2	temp_0_118	raw2046	sdtc12
icust2_0_68	raw2380	sdnc2	temp_0_119	raw2047	sdt12
icust2_0_69	raw2381	sdnh2	temp_0_12	raw1940	sdaw12
icust2_0_7	raw2319	sdaf2	temp_0_120	raw2048	sdtj12
icust2_0_70	raw2382	sdnj2	temp_0_121	raw2049	sdt12
icust2_0_71	raw2383	sdnl2	temp_0_122	raw2050	sdt12
icust2_0_72	raw2384	sdns2	temp_0_123	raw2051	sdt12
icust2_0_73	raw2385	sdnu2	temp_0_124	raw2052	sdt12

datafile name	raw number	device name	datafile name	raw number	device name
icust2_0_74	raw2386	sdnw2	temp_0_125	raw2053	sdt12
icust2_0_75	raw2387	sdo2	temp_0_126	raw2054	sdtx12
icust2_0_76	raw2388	sdod2	temp_0_127	raw2055	sdtz12
icust2_0_77	raw2389	sdof2	temp_0_128	raw2056	sdub12
icust2_0_78	raw2390	sdok2	temp_0_129	raw2057	sdud12
icust2_0_79	raw2391	sdom2	temp_0_13	raw1941	sday12
icust2_0_8	raw2320	sdah2	temp_0_130	raw2058	sduf12
icust2_0_80	raw2392	sdo2	temp_0_131	raw2059	sduh12
icust2_0_81	raw2393	sdot2	temp_0_132	raw2060	sduj12
icust2_0_82	raw2394	sdov2	temp_0_133	raw2061	sdu12
icust2_0_83	raw2395	sdox2	temp_0_134	raw2062	sdun12
icust2_0_84	raw2396	sdpc2	temp_0_135	raw2063	sdup12
icust2_0_85	raw2397	sdpe2	temp_0_136	raw2064	sdur12
icust2_0_86	raw2398	sdpg2	temp_0_137	raw2065	sdu12
icust2_0_87	raw2399	sdpl2	temp_0_138	raw2066	sduv12
icust2_0_88	raw2400	sdpn2	temp_0_139	raw2067	sdux12
icust2_0_89	raw2401	sdpp2	temp_0_14	raw1942	sdba12
icust2_0_9	raw2321	sdam2	temp_0_140	raw2068	sduz12
icust2_0_90	raw2402	sdpw2	temp_0_141	raw2069	sdvb12
icust2_0_91	raw2403	sdpy2	temp_0_142	raw2070	sdvd12
icust2_0_92	raw2404	sdqa2	temp_0_143	raw2071	sdvf12
icust2_0_93	raw2405	sdqf2	temp_0_144	raw2072	sdvh12
icust2_0_94	raw2406	sdqh2	temp_0_145	raw2073	sdvj12
icust2_0_95	raw2407	sdqj2	temp_0_146	raw2074	sdvl12
icust2_0_96	raw2408	sdqo2	temp_0_147	raw2075	sdvq12
icust2_0_97	raw2409	sdqq2	temp_0_148	raw2076	sdvs12
icust2_0_98	raw2410	sdqs2	temp_0_149	raw2077	sdvu12
icust2_0_99	raw2411	sdqx2	temp_0_15	raw1943	sdbf12
idist_0_0	raw2699	sdu6	temp_0_150	raw2078	sdvz12
iitem_0_0	raw2700	sdw6	temp_0_151	raw2079	sdwb12
iordr2_0_0	raw2120	sd11	temp_0_152	raw2080	sdwd12
iordr2_0_1	raw2121	sdn1	temp_0_153	raw2081	sdwi12
iordr2_0_10	raw2130	sdao1	temp_0_154	raw2082	sdwk12
iordr2_0_100	raw2220	sdqz1	temp_0_155	raw2083	sdwm12
iordr2_0_101	raw2221	sdrb1	temp_0_156	raw2084	sdwt12
iordr2_0_102	raw2222	sdri1	temp_0_157	raw2085	sdwv12
iordr2_0_103	raw2223	sdrk1	temp_0_158	raw2086	sdwx12
iordr2_0_104	raw2224	sdrm1	temp_0_159	raw2087	sdxc12
iordr2_0_105	raw2225	sdr1	temp_0_16	raw1944	sdbh12
iordr2_0_106	raw2226	sdr1	temp_0_160	raw2088	sdxe12
iordr2_0_107	raw2227	sdrv1	temp_0_161	raw2089	sdxc12
iordr2_0_108	raw2228	sdsa1	temp_0_162	raw2090	sdxl12
iordr2_0_109	raw2229	sdsc1	temp_0_163	raw2091	sdxn12
iordr2_0_11	raw2131	sdaq1	temp_0_164	raw2092	sdxp12
iordr2_0_110	raw2230	sdse1	temp_0_165	raw2093	sdxu12
iordr2_0_111	raw2231	sdsl1	temp_0_166	raw2094	sdxw12
iordr2_0_112	raw2232	sdsl1	temp_0_167	raw2095	sdxy12
iordr2_0_113	raw2233	sdsn1	temp_0_168	raw2096	sdyd12
iordr2_0_114	raw2234	sdss1	temp_0_169	raw2097	sdyl12
iordr2_0_115	raw2235	sdsu1	temp_0_17	raw1945	sdbj12
iordr2_0_116	raw2236	sdsu1	temp_0_170	raw2098	sdyh12
iordr2_0_117	raw2237	sdtb1	temp_0_171	raw2099	sdym12
iordr2_0_118	raw2238	sdt1	temp_0_172	raw2100	sdyo12
iordr2_0_119	raw2239	sdtf1	temp_0_173	raw2101	sdyp12
iordr2_0_12	raw2132	sdax1	temp_0_174	raw2102	sdyv12
iordr2_0_120	raw2240	sdvi1	temp_0_175	raw2103	sdyx12
iordr2_0_121	raw2241	sdvk1	temp_0_176	raw2104	sdyz12
iordr2_0_122	raw2242	sdvm1	temp_0_177	raw2105	sdze12
iordr2_0_123	raw2243	sdvr1	temp_0_178	raw2106	sdzg12
iordr2_0_124	raw2244	sdvt1	temp_0_179	raw2107	sdzi12

datafile name	raw number	device name	datafile name	raw number	device name
iorder2_0_125	raw2245	sdv1	temp_0_18	raw1946	sdfg12
iorder2_0_126	raw2246	sdwa1	temp_0_180	raw2108	sdzp12
iorder2_0_127	raw2247	sdwc1	temp_0_181	raw2109	sdzr12
iorder2_0_128	raw2248	sdwe1	temp_0_182	raw2110	sdzt12
iorder2_0_129	raw2249	sdwj1	temp_0_183	raw2111	sdzy12
iorder2_0_13	raw2133	sdaz1	temp_0_184	raw2112	sdaaa12
iorder2_0_130	raw2250	sdwl1	temp_0_185	raw2113	sdaac12
iorder2_0_131	raw2251	sdwn1	temp_0_186	raw2114	sdaah12
iorder2_0_132	raw2252	sdwu1	temp_0_187	raw2115	sdaaj12
iorder2_0_133	raw2253	sdww1	temp_0_188	raw2116	sdaal12
iorder2_0_134	raw2254	sdwy1	temp_0_189	raw2117	sdaaq12
iorder2_0_135	raw2255	sdxd1	temp_0_19	raw1947	sdfi12
iorder2_0_136	raw2256	sdxfl	temp_0_190	raw2118	sdaas12
iorder2_0_137	raw2257	sdxh1	temp_0_191	raw2119	sdaau12
iorder2_0_138	raw2258	sdxm1	temp_0_2	raw1930	sdo12
iorder2_0_139	raw2259	sdxo1	temp_0_20	raw1948	sdfk12
iorder2_0_14	raw2134	sdbb1	temp_0_21	raw1949	sdfp12
iorder2_0_140	raw2260	sdxq1	temp_0_22	raw1950	sdf12
iorder2_0_141	raw2261	sdxv1	temp_0_23	raw1951	sdf12
iorder2_0_142	raw2262	sdx1	temp_0_24	raw1952	sdhj12
iorder2_0_143	raw2263	sdxz1	temp_0_25	raw1953	sdhl12
iorder2_0_144	raw2264	sdye1	temp_0_26	raw1954	sdhn12
iorder2_0_145	raw2265	sdyg1	temp_0_27	raw1955	sdhu12
iorder2_0_146	raw2266	sdyl1	temp_0_28	raw1956	sdhw12
iorder2_0_147	raw2267	sdyn1	temp_0_29	raw1957	sdhy12
iorder2_0_148	raw2268	sdyp1	temp_0_3	raw1931	sdt12
iorder2_0_149	raw2269	sdyr1	temp_0_30	raw1958	sdt12
iorder2_0_15	raw2135	sdbg1	temp_0_31	raw1959	sdiv12
iorder2_0_150	raw2270	sdyl1	temp_0_32	raw1960	sdx12
iorder2_0_151	raw2271	sdyy1	temp_0_33	raw1961	sdjc12
iorder2_0_152	raw2272	sdza1	temp_0_34	raw1962	sdje12
iorder2_0_153	raw2273	sdzf1	temp_0_35	raw1963	sdjg12
iorder2_0_154	raw2274	sdzh1	temp_0_36	raw1964	sdjl12
iorder2_0_155	raw2275	sdzj1	temp_0_37	raw1965	sdjn12
iorder2_0_156	raw2276	sdzq1	temp_0_38	raw1966	sdjp12
iorder2_0_157	raw2277	sdzs1	temp_0_39	raw1967	sdju12
iorder2_0_158	raw2278	sdzu1	temp_0_4	raw1932	sdv12
iorder2_0_159	raw2279	sdzz1	temp_0_40	raw1968	sdjw12
iorder2_0_16	raw2136	sdbi1	temp_0_41	raw1969	sdjy12
iorder2_0_160	raw2280	sdaab1	temp_0_42	raw1970	sdkd12
iorder2_0_161	raw2281	sdaad1	temp_0_43	raw1971	sdkf12
iorder2_0_162	raw2282	sdaai1	temp_0_44	raw1972	sdkh12
iorder2_0_163	raw2283	sdaak1	temp_0_45	raw1973	sdkm12
iorder2_0_164	raw2284	sdaam1	temp_0_46	raw1974	sdko12
iorder2_0_165	raw2285	sdaar1	temp_0_47	raw1975	sdkq12
iorder2_0_166	raw2286	sdaat1	temp_0_48	raw1976	sdkv12
iorder2_0_167	raw2287	sdaav1	temp_0_49	raw1977	sdkx12
iorder2_0_168	raw2288	sdtk1	temp_0_5	raw1933	sdx12
iorder2_0_169	raw2289	sdtm1	temp_0_50	raw1978	sdkz12
iorder2_0_17	raw2137	sdbk1	temp_0_51	raw1979	sdle12
iorder2_0_170	raw2290	sdto1	temp_0_52	raw1980	sdlg12
iorder2_0_171	raw2291	sdtq1	temp_0_53	raw1981	sdli12
iorder2_0_172	raw2292	sdt1	temp_0_54	raw1982	sdln12
iorder2_0_173	raw2293	sdtu1	temp_0_55	raw1983	sdlp12
iorder2_0_174	raw2294	sdt1	temp_0_56	raw1984	sdlr12
iorder2_0_175	raw2295	sdua1	temp_0_57	raw1985	sdlw12
iorder2_0_176	raw2296	sduc1	temp_0_58	raw1986	sdly12
iorder2_0_177	raw2297	sdue1	temp_0_59	raw1987	sdma12
iorder2_0_178	raw2298	sdug1	temp_0_6	raw1934	sdac12
iorder2_0_179	raw2299	sdui1	temp_0_60	raw1988	sdmf12

datafile name	raw number	device name	datafile name	raw number	device name
iorder2_0_18	raw2138	sdfh1	temp_0_61	raw1989	sdmh12
iorder2_0_180	raw2300	sduk1	temp_0_62	raw1990	sdmj12
iorder2_0_181	raw2301	sdum1	temp_0_63	raw1991	sdmo12
iorder2_0_182	raw2302	sduo1	temp_0_64	raw1992	sdmq12
iorder2_0_183	raw2303	sduq1	temp_0_65	raw1993	sdms12
iorder2_0_184	raw2304	sdus1	temp_0_66	raw1994	sdmx12
iorder2_0_185	raw2305	sduu1	temp_0_67	raw1995	sdmz12
iorder2_0_186	raw2306	sduw1	temp_0_68	raw1996	sdnb12
iorder2_0_187	raw2307	sduy1	temp_0_69	raw1997	sdng12
iorder2_0_188	raw2308	sdva1	temp_0_7	raw1935	sdae12
iorder2_0_189	raw2309	sdvc1	temp_0_70	raw1998	sdni12
iorder2_0_19	raw2139	sdfj1	temp_0_71	raw1999	sdnk12
iorder2_0_190	raw2310	sdve1	temp_0_72	raw2000	sdnr12
iorder2_0_191	raw2311	sdvg1	temp_0_73	raw2001	sdnt12
iorder2_0_2	raw2122	sdp1	temp_0_74	raw2002	sdnv12
iorder2_0_20	raw2140	sdfi1	temp_0_75	raw2003	sdoa12
iorder2_0_21	raw2141	sdfq1	temp_0_76	raw2004	sdoc12
iorder2_0_22	raw2142	sdfs1	temp_0_77	raw2005	sdoe12
iorder2_0_23	raw2143	sdfu1	temp_0_78	raw2006	sdoj12
iorder2_0_24	raw2144	sdhk1	temp_0_79	raw2007	sdol12
iorder2_0_25	raw2145	sdhm1	temp_0_8	raw1936	sdag12
iorder2_0_26	raw2146	sdho1	temp_0_80	raw2008	sdon12
iorder2_0_27	raw2147	sdhv1	temp_0_81	raw2009	sdos12
iorder2_0_28	raw2148	sdhx1	temp_0_82	raw2010	sdou12
iorder2_0_29	raw2149	sdhz1	temp_0_83	raw2011	sdow12
iorder2_0_3	raw2123	sdu1	temp_0_84	raw2012	sdpb12
iorder2_0_30	raw2150	sdiu1	temp_0_85	raw2013	sdpd12
iorder2_0_31	raw2151	sdiw1	temp_0_86	raw2014	sdpf12
iorder2_0_32	raw2152	sdii1	temp_0_87	raw2015	sdpk12
iorder2_0_33	raw2153	sdjd1	temp_0_88	raw2016	sdpm12
iorder2_0_34	raw2154	sdjf1	temp_0_89	raw2017	sdpo12
iorder2_0_35	raw2155	sdjh1	temp_0_9	raw1937	sdal12
iorder2_0_36	raw2156	sdjm1	temp_0_90	raw2018	sdpv12
iorder2_0_37	raw2157	sdjo1	temp_0_91	raw2019	sdpx12
iorder2_0_38	raw2158	sdjq1	temp_0_92	raw2020	sdpz12
iorder2_0_39	raw2159	sdjv1	temp_0_93	raw2021	sdqe12
iorder2_0_4	raw2124	sdw1	temp_0_94	raw2022	sdqg12
iorder2_0_40	raw2160	sdjx1	temp_0_95	raw2023	sdqi12
iorder2_0_41	raw2161	sdjz1	temp_0_96	raw2024	sdqn12
iorder2_0_42	raw2162	sdke1	temp_0_97	raw2025	sdqp12
iorder2_0_43	raw2163	sdkg1	temp_0_98	raw2026	sdqr12
iorder2_0_44	raw2164	sdki1	temp_0_99	raw2027	sdqw12
iorder2_0_45	raw2165	sdkn1	tpccaux	raw2892	sdw7
iorder2_0_46	raw2166	sdkp1	ware_0_0	raw2696	sd16

4.3 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, DL/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.*

Oracle is a relational DBMS.

The interface used was Oracle stored procedures accessed using the Oracle Call Interface (OCI) embedded in C code.

4.4 Database Mapping

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

Horizontal partitioning was used for one (hist) of the tables and one (iordr2) of the indices. The detail of this partitioning can be understood by examining the table and index definition statements in Appendix E. Vertical partitioning and replications were not used in this implementation.

4.5 60 Day Space

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

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

- The size of the redo log was queried from the Oracle catalog.
- A full performance run was executed.
- The increase in size to the redo logs was divided by the number of transactions, giving bytes used per new order.
- This amount was multiplied by the reported tpm rate times 480 minutes, giving total space needed for 8 hours.

For the dynamic tables the following steps were followed:

1. The database was queried for the size of the dynamic tables.
2. The sum of D_NEXT_O_ID was queried from the DISTRICT table.
3. A full performance run was executed.
4. Steps 1 & 2 were repeated.
5. The change in the size of the dynamic tables was divided by the number of new orders in the run giving growth per new order.
6. The number in the pervious step was multiplied by the reported tpm rate times 480 minutes.
7. The numbers in steps 1 & 5 were added giving space needed for 8 hours.
8. The space allocated was verified to be larger than the space needed.

The 60 day space requirement is shown in Appendix F.

Clause 5 Related Items

5.1 Throughput

Measured tpmC must be reported.

Measured tpmC: 2,382,032 tpmC
Price per tpmC: \$3.76 USD per tpmC

5.2 Response Times

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

An emulation delay of 0.1 second is included in response time and menu time to compensate for browser delay.

Table 5.1 Response Times in Seconds

Type	Average	90th %	Maximum
New-Order	0.835	1.597	20.834
Payment	0.821	1.586	20.815
Order-Status	0.828	1.593	20.737
Interactive Delivery	0.103	0.104	0.328
Deferred Delivery	0.726	1.491	20.472
Stock-Level	0.811	1.576	20.788
Menu	0.103	0.104	0.706

5.3 Keying and Think Times

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

Table 5.2 Keying Times

Type	Minimum	Average	Maximum
New-Order	18.002	18.012	18.035
Payment	3.002	3.012	3.036
Order-Status	2.003	2.012	2.030
Interactive Delivery	2.003	2.012	2.035
Stock-Level	2.003	2.012	2.031

Table 5.3 Think Times

Type	Minimum	Average	Maximum
New-Order	0.000	12.015	120.202
Payment	0.000	12.016	120.201
Order-Status	0.000	10.013	100.193
Interactive Delivery	0.000	5.017	50.200
Stock-Level	0.000	5.017	50.199

5.4 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.

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

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

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

Figure 5.1: New Order Response Time Distribution

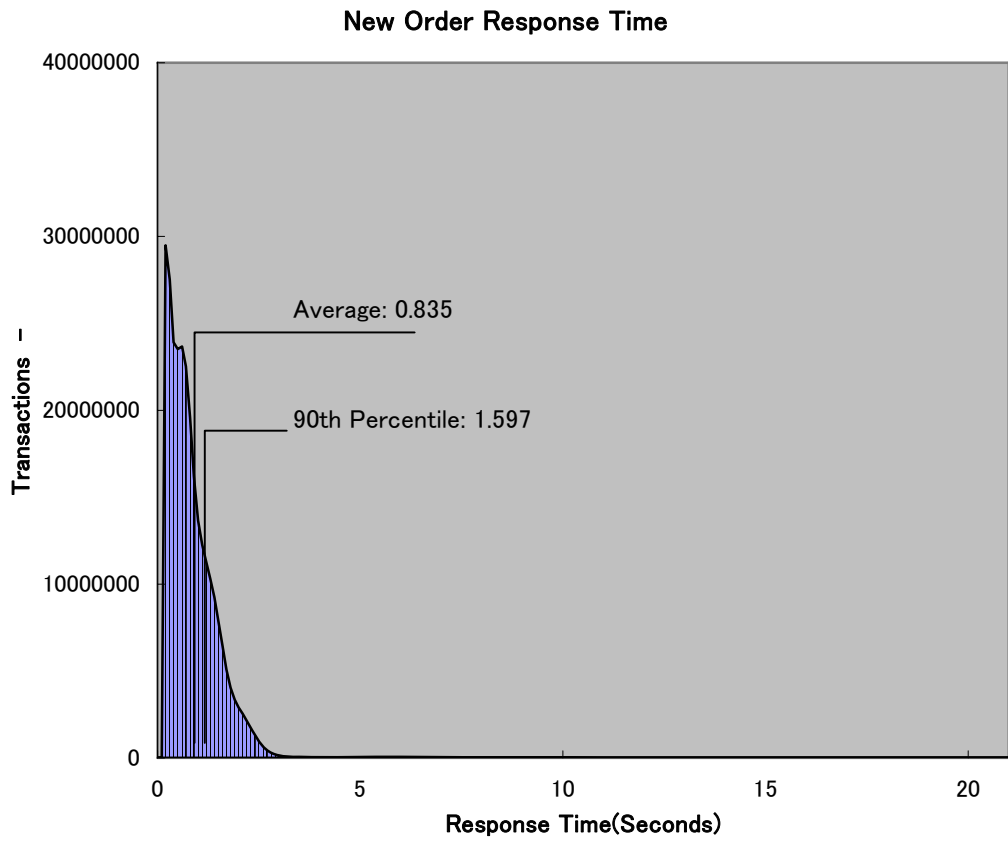


Figure 5.2: Payment Response Time Distribution

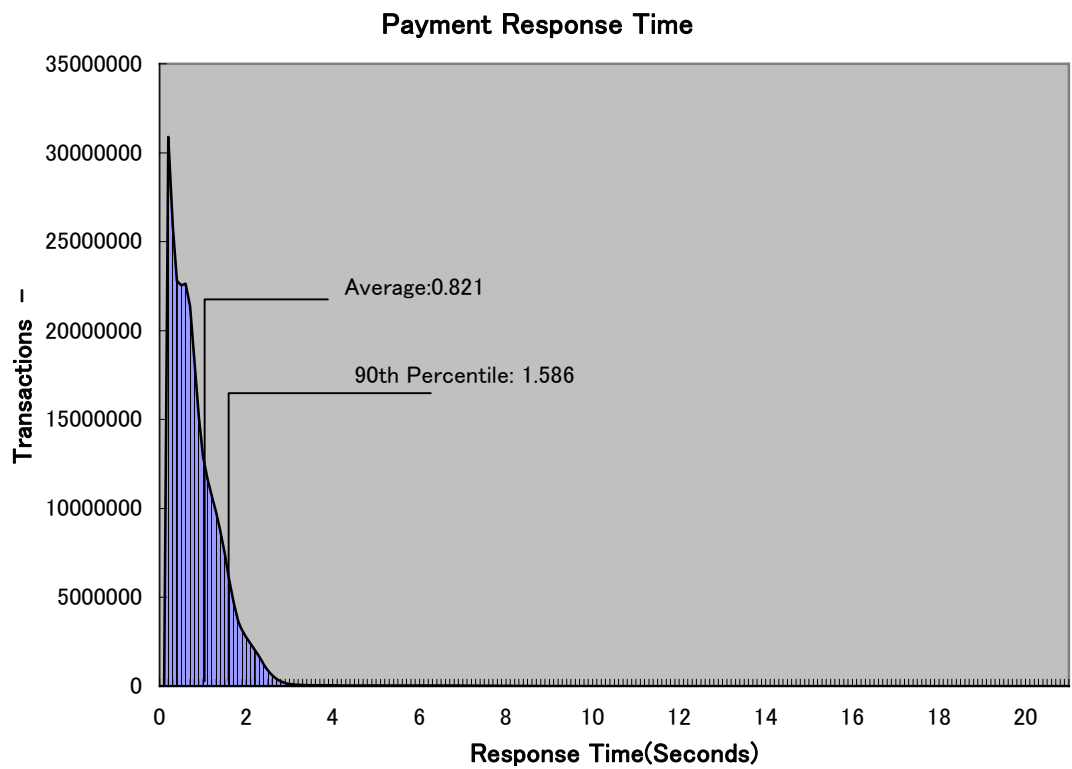


Figure 5.3: Order Status Response Time Distribution

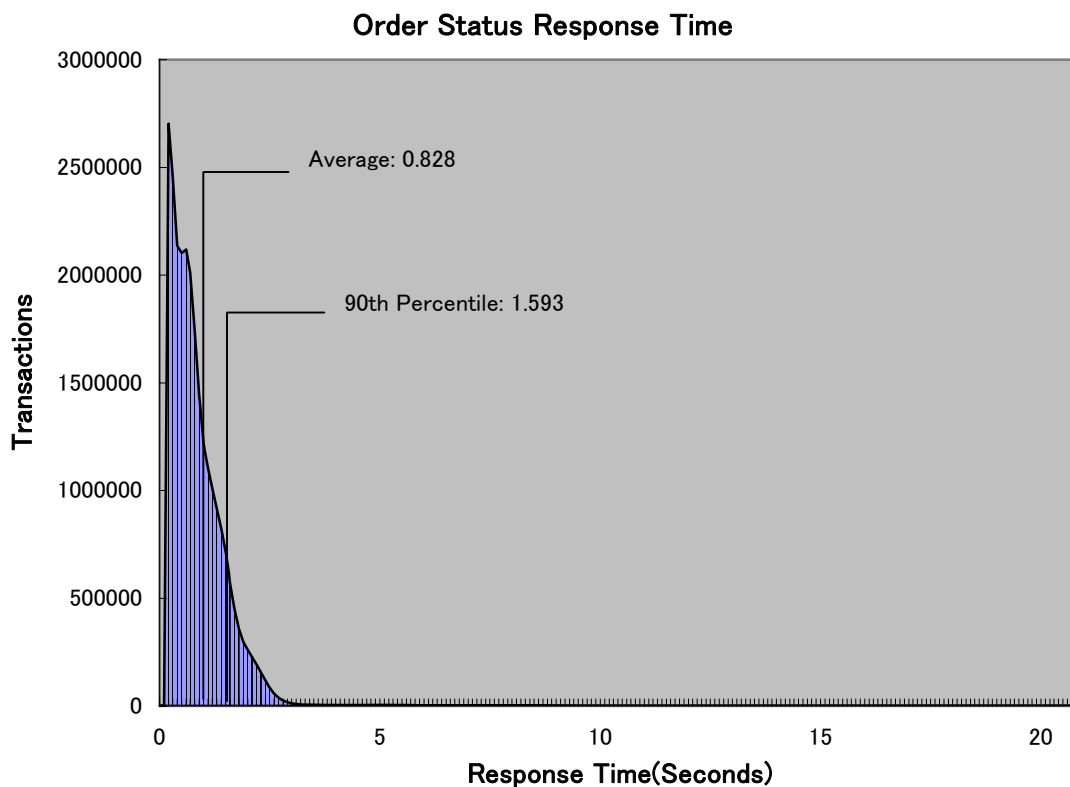


Figure 5.4: Delivery Response Time Distribution

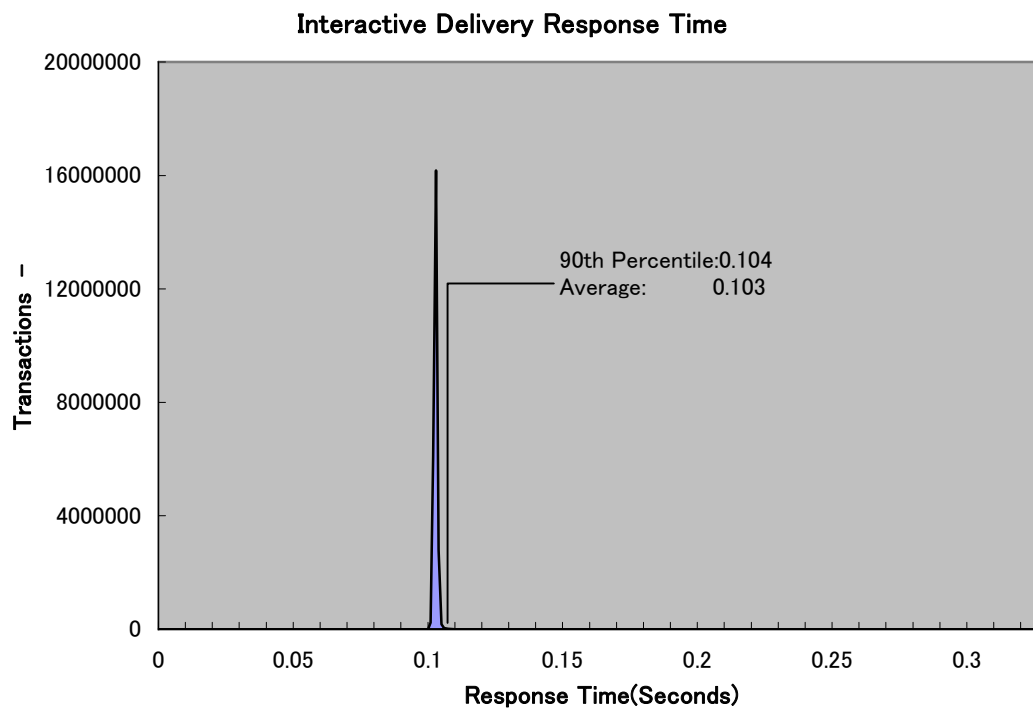


Figure 5.5: Stock Level Response Time Distribution

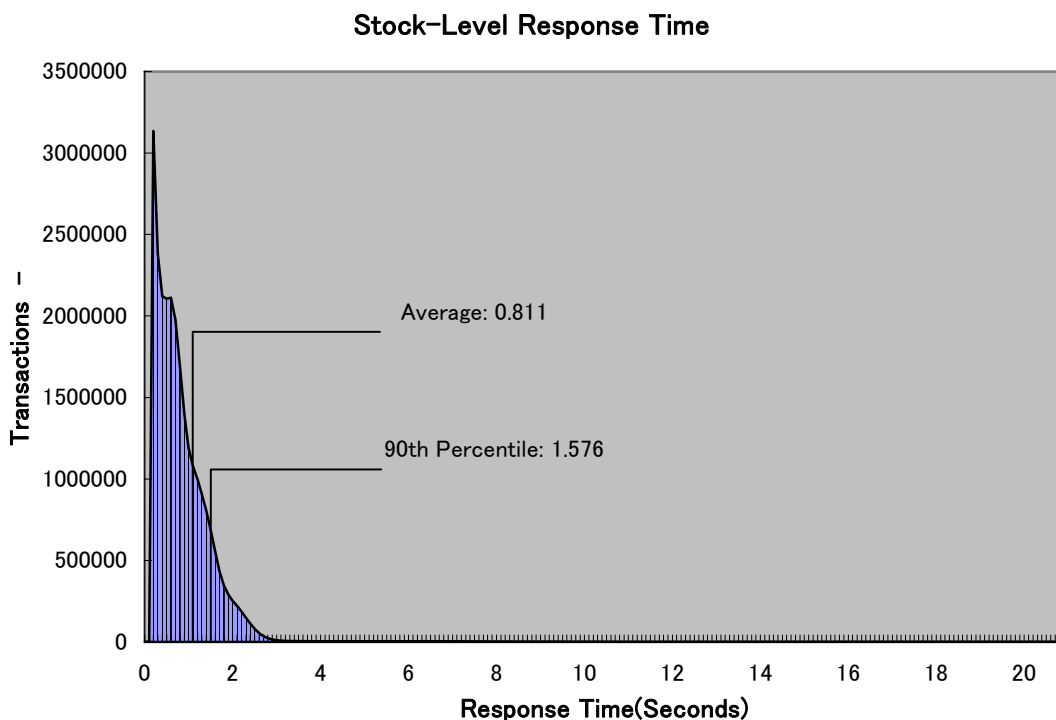


Figure 5.6: New Order Think Time Frequency Distribution

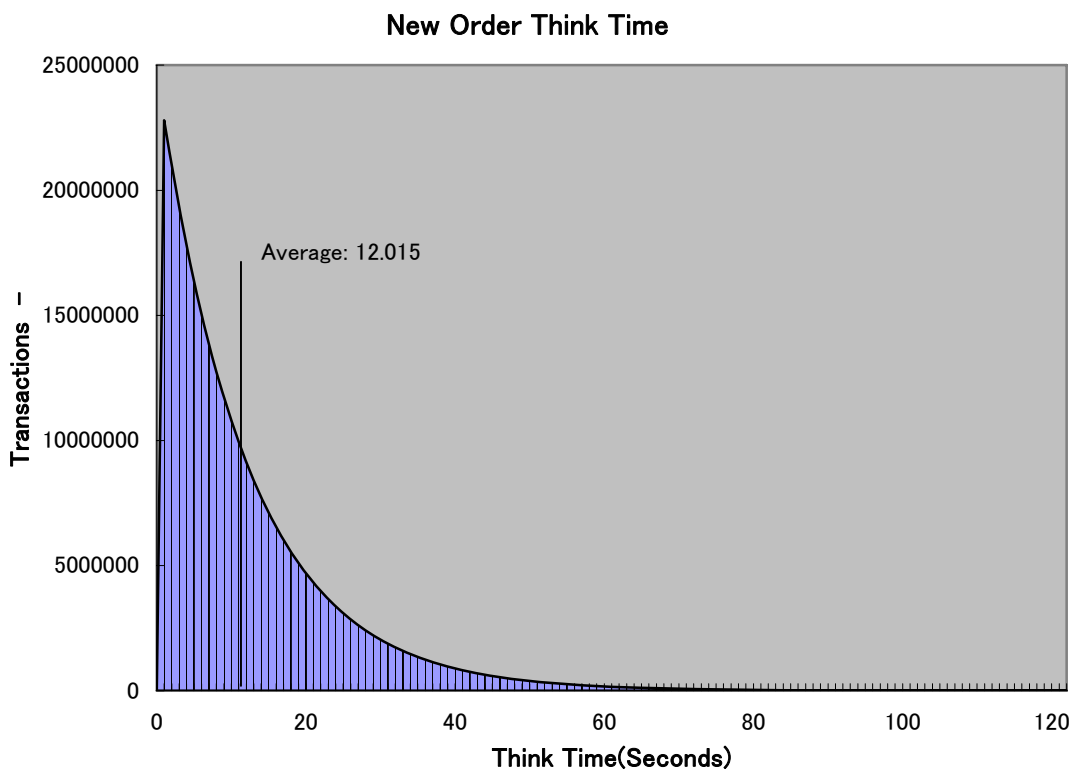


Figure 5.7: New-Order Response time vs. Throughput
Response Time vs. Throughput

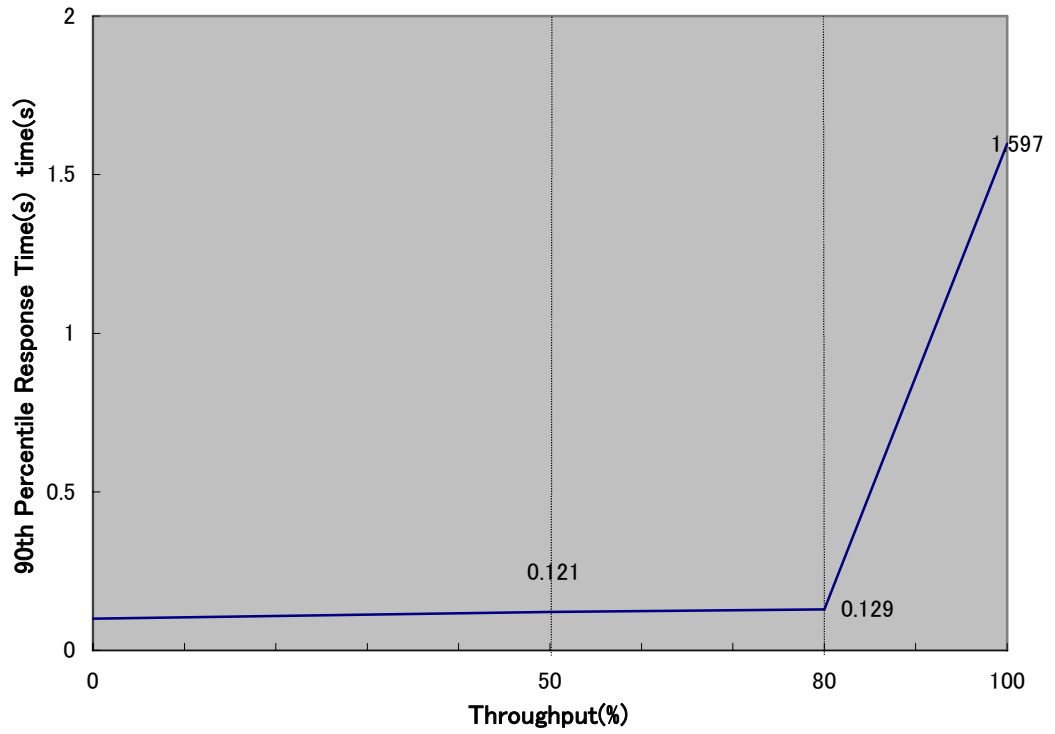
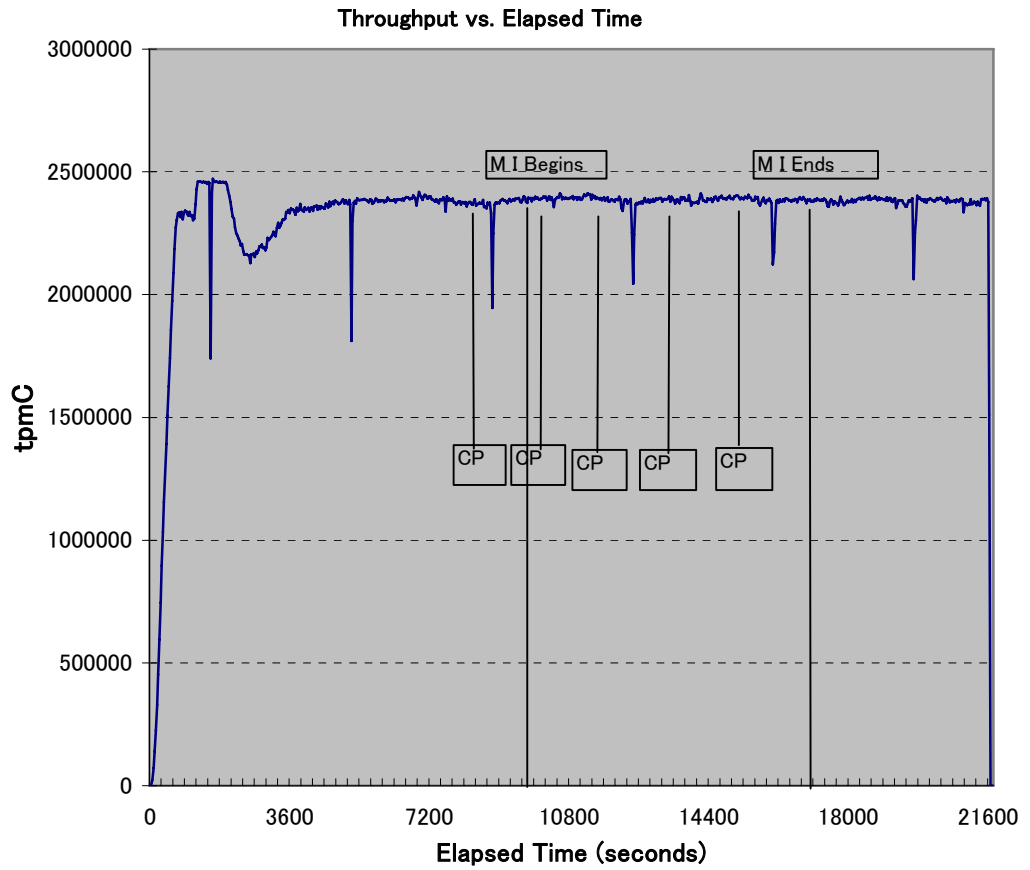


Figure 5.8: Throughput versus Elapsed Time



5.5 Steady State Determination

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

Steady state was determined by examining data reported for each 30-second interval over the duration of the measured run. Steady state was further confirmed by the throughput data collected during the run and graphed in Figure 5.8.

5.6 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.

The Oracle logical log is on a RAID0+1 array. When one log file becomes full or a time specified by parameters comes, Oracle Database 10g starts a checkpoint process. Oracle automatically logs all checkpoints to an alert file on the server. We configured log files and parameters so that checkpoints would occur in 30 minutes interval. Oracle Database 10g performed 4 times of Log file Switches during MI. At each checkpoint, Oracle wrote to disk all buffer pages that had been updated but not yet physically written to disk.

For the priced system, the logical log space for an 8-hour period is priced.

Serializable Transactions:

Oracle supports serializable transaction isolation in full compliance with the SQL92 and TPC-C requirements. This is implemented by extending multiple concurrency control mechanisms long supported by Oracle.

Oracle queries take no read locks and see only data committed as of the beginning of the query's execution. This means that the readers and writers coexist without blocking one another, providing a high degree of concurrency and consistency. While this mode does prevent reading dirty data, Oracle's default isolation level also permits a transaction that issues a query twice to see non-repeatable reads and phantoms, as defined in SQL92 and TPC-C.

Beginning with Oracle7 release 7.3, a transaction may request a higher degree of isolation with the command `SET TRANSACTION ISOLATION LEVEL SERIALIZABLE` as defined in SQL92. This command will prevent read/write and write/write conflicts that would cause serializability failures.

A session can establish this mode as its default mode, so the `SET TRANSACTION` command need not be issued in each transaction.

Oracle implements `SERIALIZABLE` mode by extending the scope of read consistency from individual query to the entire transaction itself. ALL reads by serializable transactions are therefore repeatable, as the transaction will access prior versions of data changed (or deleted) by other transactions after the start of serializable transactions.

Thus, a serializable transaction sees a fixed snapshot of the database, established at the beginning of the transaction.

To ensure proper isolation, a serializable transaction cannot modify the rows that were changed by other transactions after the beginning of a serializable transaction, or an update (or delete) statement will fail with error `ORA_08177: "cannot serialize access"` and the statement will rollback.

When a serializable transaction fails with this error, the application may either commit the work executed to that point, execute additional statements, or rollback the entire transaction. Repeated attempts to execute the same statement will always fail with the error "can't serialize access" unless the other transaction has rolled back and released its lock. This error and these recovery options are similar to the treatment of deadlocks in systems that use read locks to ensure serializable execution.

In both cases, conflicts between transactions rollback and restarts or commits without re-executing the statement receiving the error.

5.7 Reproducibility

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

No reproducibility run is needed in this revision of the benchmark.

5.8 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 120 minutes long.

5.9 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.

5.10 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.

Table 5.4: Transaction Statistics

Statistics		Value
Transaction Mix	New Order	44.94%
	Payment	43.01%
	Order status	4.02%
	Delivery	4.02%
	Stock level	4.01%
New Order	Home warehouse order lines	99.00%
	Remote warehouse order lines	1.00%
	Rolled back transactions	1.00%
	Average items per order	10.00
Payment	Home warehouse	85.00%
	Remote warehouse	15.00%
	Accessed by last name	60.00%
Order Status	Accessed by last name	60.00%
Delivery	Skipped transactions	None

5.11 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.

One checkpoint was recorded before the measured window opened and four checkpoints were started inside the measured window.

The start time and duration in seconds of at least the four (4) longest checkpoints during the Measurement Interval must be disclosed (see Clause 5.5.2.2 (2)).

	start	end	duration
measurement	20:20:44	22:20:44	120 minutes
	start	End	duration
checkpoint 0	19:53:22	20:22:39	29:17
checkpoint 1	20:22:56	20:52:05	29:09
checkpoint 2	20:52:21	21:21:33	29:12
checkpoint 3	21:21:58	21:51:16	29:18
checkpoint 4	21:51:25	22:20:40	29:15

Clause 6 Related Items

6.1 RTE Descriptions

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

The RTE used is proprietary to Fujitsu. Appendix C contains the profile used as input to this RTE.

6.2 Loss of Terminal Connections

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

No terminal connections were lost.

6.3 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.

There were no emulated components in the benchmark configuration other than the emulated users' workstations.

6.4 Functional Diagrams

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 driver system performed the data generation and input functions of the display device. It also captured the input and output data and timestamps for post-processing of the reported metrics. No other functionality was included on the driver system

The abstract at the beginning of this report contains detailed diagrams of both the benchmark configuration and the priced configuration, including the driver system.

6.5 Networks

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 replace with the Driver System must be disclosed(see Clause 6.6.4).

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

Five 1Gbps ethernet LAN connections were used between the server and five switches, to which clients were connected by 99(ninety-nine) 100Mbps ethernet LAN connections. Another 99(ninety-nine) 100Mbps ethernet LAN connections were used between the clients and the switches connected to the emulated users.

6.6 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, other than beginning the checkpointing process.

Clause 7 Related Items

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

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

A detailed price list is included in the abstract at the beginning of this report.

7.2 Availability

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

The total solution as priced will be available by December 4, 2008.

7.3 Throughput, and Price Performance

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

Maximum Qualified Throughput :	2,382,032 tpmC
Price per tpmC :	\$3.76 per tpmC
Three-year cost of ownership :	\$8,950,005 USD

7.4 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.

7.5 Usage Pricing

For any usage pricing, the sponsor must disclose:

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

The component pricing based on usage is shown below:

- 32 Oracle Database 10g Enterprise Edition, Per Processor, Unlimited Users, 3 years
- 1 Red Hat Enterprise Linux AS (Standard)
- 99 Red Hat Enterprise Linux ES (Standard)
- 99 Tuxedo Core Functionality Services(CFS-R)

7.6 System Pricing

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

Clause 6.1 describes the Server and Client components.

System pricing must include line item indication where non-sponsoring companies' part numbers are used. System pricing must also include line item indication of third party pricing.

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

Clause 9 Related Items

9.1 Auditor's Report

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

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

9.2 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
Presidio of San Francisco
Building 572B Ruger St. (surface)
P.O.Box 29920 (mail)
San Francisco, CA 94129-0920
Voice: 415-561-6272
Fax: 415-561-6120
Email: info@tpc.org

Appendix A: Client Source Code

```

.....
common/GetPrivateProfileString.c
.....

/*****
*****
*
*          *
*   TPC-C Client Application Program Source
*
*          *
*   Entry Functions
*   (1) GetPrivateProfileString
*
*          *
*   CREATE by TSL 2003.12.18
*
*          *
*   All Right Reserved, Copyright Co. FUJITSU
*   LIMITED 2003-2004 *
*****
*****/

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

/*****
*****
*   Get data string corresponded key in
*   cogfiguration file.
*   *
*   Return Value
*   *
*   Get string length
*   *
*****
*****/

int GetPrivateProfileString(char*
section_name, /* Section name
*/
name          char* key_name, /* Key
*/
string, if kye nothing char* default_str, /* Default
*/
data          char* key_data, /* Key
*/
size of key data int buf_size, /* Buffer
*/
name          char* file_name){ /* File
*/

FILE* prof_file;
char read_buf[256];
char search[32];
char* get_str;
char* key_pos=0;
int get_cnt;
int i;

/* Open profile file */
if ((prof_file = fopen(file_name, "r")) == NULL)
{
goto DEFAULT_STRING;
}

/* Make searching section name "[section
name]" */

```

```

search[0] = '[';
strcpy(&search[1], section_name);
strcat(search, "]");

/* Search section name */
while((get_str = fgets(read_buf,
sizeof(read_buf), prof_file)) != NULL) {

/* Search section name form to be read one
line */
if ((char*)strstr(read_buf, search) == NULL)
{
/* No match section name, next line read
*/
continue;
}
break;
}
if (get_str == NULL) {
/* Found EOF or read error */
goto DEFAULT_STRING_FCLOSE;
}

/* Make searching key name "key_name=" */
strcpy(search, key_name);
strcat(search, "=");

/* Search key name in this section */
while((get_str = fgets(read_buf,
sizeof(read_buf), prof_file)) != NULL) {
for (i = 0; read_buf[i] == ' ' || read_buf[i] ==
'\t'; i++);
if (read_buf[i] == '[') {
/* Other section started, undefined key
name */
goto DEFAULT_STRING_FCLOSE;
}
if ((key_pos = (char*)strstr(read_buf,
search)) == NULL) {
/* No match key name */
continue;
}
}
break;
}
if (get_str == NULL) {
/* Found EOF or read error */
goto DEFAULT_STRING_FCLOSE;
}

fclose(prof_file);

/* Get key_value, fixed format "key value" */
for (; *key_pos != ""; key_pos++);
key_pos++;
for (get_cnt = 0; *key_pos != ""; key_pos++) {
/* Get & set key value */
*key_data = *key_pos;
key_data++;
get_cnt++;
if (get_cnt >= (buf_size - 1)) {
/* Key data buffer full */
break;
}
}
*key_data = '\0';
return(get_cnt);

DEFAULT_STRING_FCLOSE:
fclose(prof_file);

DEFAULT_STRING:
strncpy(key_data, default_str, buf_size-1);
return(strlen(key_data));
}

```

```

.....
common/MakeShell
.....

#!/bin/sh
cd /home/tpc/client_apl/common
make > make_result.txt 2>&1

.....
common/Makefile
.....

#-----
# Makefile : Makefile for common of TPAPL and
SVRAPL.
#
# Created by TSL 2003.12.17
#
# All Right Rserverd, Copyright Co, FUJITSU
LIMITED 2003-2004.
#-----

# GCC compile configurations
AR = ar
ARFLAGS = rv
CFLAGS = -Wall -O2
CC = gcc

# MACRO definition
DMACRO = -DTSL -DPLSQLFLAG=1 -DTUX

# home directory
ORADIR = /usr/local/oracle
TUXDIR = /usr/local/BEA/tuxedo8.1
SVRDIR = /home/tpc/client_apl/svrapl

# include directory
ORA_INC = -I$(ORADIR)/rdbms/demo -
I$(ORADIR)/rdbms/public
COM_INC = -I$(SVRDIR)/common
SVR_INC = -I$(SVRDIR)
TUX_INC = -I$(TUXDIR)/include
INCLUDE = $(COM_INC) $(SVR_INC)
$(ORA_INC) $(TUX_INC)

# target object
COMOBS = log.o sema.o
GetPrivateProfileString.o shmем.o
COMLIB = libcom.a

INCFILES = log.h sema.h forlinux.h shmем.h

$(COMLIB) : $(COMOBS)
$(AR) $(ARFLAGS) $(COMLIB) $(COMOBS)

.SUFFIXES : .o .c
.c.o:
$(CC) -o $@ -c $(CFLAGS) $(INCLUDE)
$(DMACRO) $<

$(COMOBS) : $(INCFILES)

clean:
rm $(COMLIB) $(COMOBS)

```

```

.....
common/forlinux.h
.....

/*****
*****
*
* TPC-C Client Application Program Source
*
*
* Entry Functions
* definition for converting Linux.
*
*
* CREATE by TSL 2003.05.16
*
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2003-2004 *
*****
*****/
/* forlinux.h */

#include <limits.h>
#define MAX_PATH PATH_MAX /*
Windows:MAX_PATH , Linux:PATH_MAX */
#define Sleep(x) poll(0, 0, x); /* sleep unit is
a msec. */

.....
common/log.c
.....

/*****
*****
*
* TPC-C Client Application Program Source
*
*
* Entry Functions
* Log is outputted to a file.
*
*
* CREATE by TSL 2002.11.29
*
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2002-2004 *
*****
*****/
#include "forlinux.h"
#include <stdio.h>
#include <string.h>
#include <time.h>
#include <sys/types.h>
#include <stdarg.h>
#include <unistd.h>
#include <pthread.h>
#include <sys/types.h>
#include <sys/stat.h>
#include "sema.h"

#define LOG_MODULE
#include "log.h"

void TpcUserLog(char* file_name, int line_no,
char* type_name, char* ftmp, ...)
{
FILE* fp;
pid_t pid;
pthread_t tld;

```

```

char* fname;
int stat;

/* -- BEGIN -- Modified by Hayashi for thread-
safe. 2006/02/13 */
#if 0
! struct tm *nowtime;
#else
struct tm tt;
struct tm *nowtime=&tt;
#endif
/* -- END -- Modified by Hayashi for thread-safe.
2006/02/13 */

time_t long_time;
va_list va;

if (strcmp(type_name, "LCK") != 0) {
/* Lock semaphore */
stat = LockSem(GLB_LogSemId);
}
/* Get current time. */

time( &long_time );

/* -- BEGIN -- Modified by Hayashi for thread-
safe. 2006/02/13 */
#if 0
! nowtime = localtime( &long_time );
#else
localtime_r( &long_time, nowtime );
#endif
/* -- END -- Modified by Hayashi for thread-safe.
2006/02/13 */

/* Get process Id. */
pid = getpid();

/* Get thread Id. */
tld = pthread_self();

/* Get just file name from a path. */
fname = (char*)strchr(file_name, (int)'/');
if (fname == NULL) {
fname = file_name;
} else {
fname = fname + 1;
}

va_start(va, ftmp);

fp = fopen(GLB_LogFilePath, "a");
fprintf(fp, "%02d:%02d:%02d [%6d:%08x] %-
32s(%4d) :%s: ",
nowtime->tm_hour, nowtime->tm_min,
nowtime->tm_sec, pid, (int)tld, fname, line_no,
type_name);
vfprintf(fp, ftmp, va);

if (*(ftmp + strlen(ftmp) - 1) != '\n')
fprintf(fp, "\n");

va_end(va);

fclose(fp);

/* change mode which all users can read and
write. */
chmod(GLB_LogFilePath, S_IRUSR
|S_IWUSR|S_IRGRP|S_IWGRP|S_IROTH|
S_IWOTH);

if (strcmp(type_name, "LCK") != 0) {
// Unlock semaphore
stat = UnlockSem(GLB_LogSemId);

```

```

}
return;
}

.....
common/log.h
.....

/*****
*****
*
* TPC-C Client Application Program Source
*
*
* Entry Functions
* Log is outputted to a file.
*
*
* CREATE by TSL 2002.11.29
*
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2003-2004 *
*****
*****/

void TpcUserLog (char *file_name, int line_no,
char* type_name, char* ftmp, ...);

extern char GLB_LogFilePath[MAX_PATH];
extern int GLB_LogSemId;

#define DEFAULT_SVRAPL_LOG_PATH
"/home/tpc/log/DBDepend_Userlog.log"
#define DEFAULT_TPAPL_LOG_PATH
"/home/tpc/log/userlog.log"

#define LOG_ERR __FILE__, __LINE__, "ERR"
#define LOG_INF __FILE__, __LINE__, "INF"
#define LOG_WRN __FILE__, __LINE__,
"WRN"
#define LOG_LCK __FILE__, __LINE__, "LCK"

#define LOG_FILE_INF __FILE__,
__LINE__, "INF"
#define LOG_FILE_LINE __FILE__,
__LINE__

.....
common/sema.c
.....

/*****
*****
*
* TPC-C Client Application Program Source
*
*
* Filename :
* sema.c
*
* Entry Functions :
* There are functions to control semaphore.
*
*
* CREATE by TSL 2003.12.18
*
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2003-2004 *

```

```

*****
*****/

#include "forlinux.h"
#include <stdio.h>
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/sem.h>
#include <errno.h>
#include "log.h"
#include "sema.h"

/*****
*****/
* Initialize semaphore.
* Return Value
* > 0 semaphore Id. (always over 0)
*
* < 0 fail.
*

*****
*****/
int InitSem(char *path, int projectId)
{
    int sid;
    union semun{
        int val;
        struct semid_ds *buf;
        ushort *array;
    } c_arg;

    TpcUserLog(LOG_LCK, "InitSem: start
path<%s> projectId=%d\n",
    path, projectId);

    if ((sid = GetSem(path, projectId)) == -1) {
        TpcUserLog(LOG_LCK, "GetSem() fail,
path<%s> projectId=%d\n",
        path, projectId);
        return(-1);
    }
    c_arg.val=1;
    if (semctl(sid,0,SETVAL,c_arg)==-1) {
        TpcUserLog(LOG_LCK, "semctl fail,
sid=%d\n",sid);
        return(-1);
    }
    TpcUserLog(LOG_LCK, "InitSem: Get
semid =%d\n",sid);

    return(sid);
}
/*****
*****/
* Get semaphore.
* Return Value
* > 0 semaphore Id. (always over 0)
*
* < 0 fail.
*

*****
*****/
int GetSem(char *path, int projectId)
{
    int sid;
    int key;

    if ((key = ftok(path,projectId)) == -1) {
        TpcUserLog(LOG_LCK, "ftok() fail,
path<%s> projectId=%d errno=%d\n",
        path, projectId, errno);
        return(-1);
    }

```

```

    if ((sid=semget(key,1,0666|IPC_CREAT))== -
1){
        TpcUserLog(LOG_LCK, "semget() fail,
key=%d errno=%d\n",key, errno);
        return(-1);
    }

    return(sid);
}
/*****
*****/
* Reuire to lock semaphore.
*
* Return Value
* 1 success.
* -1 fail.
*

*****
*****/
int LockSem(int sid)
{
    struct sembuf sb;

    sb.sem_num=0;
    sb.sem_op=-1;
    sb.sem_flg=0;
    if(semop(sid,&sb,1)== -1) {
        TpcUserLog(LOG_LCK, "semop() fail,
sid=%d\n",sid);
        return(-1);
    }
    return(1);
}
/*****
*****/
* Reuire to unlock semaphore.
*
* Return Value
* 1 success.
* -1 fail.
*

*****
*****/
int UnlockSem(int sid)
{
    struct sembuf sb;

    sb.sem_num=0;
    sb.sem_op=1;
    sb.sem_flg=0;
    if(semop(sid,&sb,1)== -1){
        TpcUserLog(LOG_LCK, "semop() fail,
sid=%d\n",sid);
        return(-1);
    }
    return(1);
}

.....
common/sema.h
.....

/*****
*****/
*
* TPC-C Client Application Program Source
*
*
* Entry Functions
* Semaphore control.
*
*
* CREATE by TSL 2003.12.19
*

```

```

*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2003-2004 *

*****/

/*****
*****/
/*== project Id =====*/
#define SEM_SVRAPL_PROJID
(int)'S'
#define SEM_TPAPL_PROJID (int)'T'
#define SEM_SAMPLING_PERFORMANCE
(int)'P'

/*****
*****/
/*=====
=====*/
/*
/* prototype definition */
/*=====
=====*/
int InitSem(char *path, int projectId);
int GetSem(char *path, int projectId);
int LockSem(int sid);
int UnlockSem(int sid);

.....
common/Shmem.c
.....

/*****
*****/
*
* TPC-C Client Application Program Source
*
*
* Filename :
* sema.c
*
* Entry Functions :
* There are functions to control shared
memory.
*
* CREATE by TSL 2004.01.15
*
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2003-2004 *

*****/

#include "forlinux.h"
#include <stdio.h>
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/shm.h>
#include <errno.h>
#include "log.h"

/*****
*****/
* Initialize shared memory.
* Return Value
* > 0 shared memory address. (always over
0)
* < 0 fail.
*

*****
*****/
char* InitShmem(char *path, int projectId, int
size)
{
    int shmId;
    int key;

```

```

char *shmaddr;

TpcUserLog(LOG_LCK, "InitShmem: start
path<%s> projectId=%d\n",
    path, projectId);

if ((key = ftok(path,projectId)) == -1) {
    TpcUserLog(LOG_LCK, "ftok() fail,
path<%s> projectId=%d errno=%d\n",
    path, projectId, errno);
    return((char *)-1);
}
if
((shmId=shmget(key,size,IPC_CREAT|0666))==
-1){
    TpcUserLog(LOG_LCK, "shmget() fail,
key=%d errno=%d",key, errno);
    return((char *)-1);
}
if( (shmaddr = (char *)shmat(shmId, NULL, 0))
== (char *)-1) {
    TpcUserLog(LOG_LCK, "shmat() fail,
shmId=%d path<%s> projectId=%d errno=%d\n",
    shmId, path, projectId, errno);
    return ((char *)-1);
}

TpcUserLog(LOG_LCK, "InitShmem: Get
shmId =%d shmaddr = %08x\n",shmId,
shmaddr);

return(shmaddr);
}
/*****
* Get shared memory.
* Return Value
* > 0 shared memory address. (always over
0)
* < 0 fail.
*****/

char* GetShmem(char *path, int projectId, int
size)
{
    int shmId;
    int key;
    char *shmaddr;

    if ((key = ftok(path,projectId)) == -1) {
        TpcUserLog(LOG_LCK, "ftok() fail,
path<%s> projectId=%d errno=%d\n",
        path, projectId,errno);
        return((char *)-1);
    }
    if ((shmId=shmget(key,size, 0))== -1){
        TpcUserLog(LOG_LCK, "shmget() fail,
key=%d errno=%d\n",key,errno);
        return((char *)-1);
    }
    if ((shmaddr = (char *)shmat(shmId, NULL, 0))
== (char *)-1) {
        TpcUserLog(LOG_LCK, "shmat() fail,
shmId=%d path<%s> projectId=%d errno=%d\n",
        shmId, path, projectId, errno);
        return ((char *)-1);
    }

    return(shmaddr);
}

.....:
common/shmem.h

```

```

.....:
/*****
*
* TPC-C Client Application Program Source
*
* Entry Functions
* Shared memory control.
*
* CREATE by TSL 2004.01.15
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2003-2004 *
*****/

/!== project Id =====*/
#define
SHMEM_SAMPLING_PERFORMANCE
(int)P'

/!=====
====*/
/! prototype definition */
/!=====

char* InitShmem(char *path, int projectId, int
size);
char* GetShmem(char *path, int projectId, int
size);

.....:
tpapl/ClientMonitor.c
.....:

/*****
*
* TPC-C Client Application Program Source
*
* Entry Functions
* (1) ClientMonitor
* (2) ClientLogCheck
* (3) ClientShutdown
* (4) ClientInfSample
* (5) ClientSampleInit
* (6) ClientSampleSelfCsv
*
* CREATE by TSL 2004.01.18
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2004 *
*****/

#include "forlinux.h"
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <time.h>
#include <unistd.h>
#include "sema.h"
#include "shmem.h"
#include "SampleInfo.h"
#include "log.h"

```

```

/* Global area */
extern char GLB_TpAplLogPath[];
extern char GLB_SvrAplLogPath[];
MAC_SampleGlobalArea;

/*****
* Client monitoring function.
* Return Value
* 0 : Normal end
* !0: Illegal function no.
* Return Information
* HTML document
*****/

int ClientMonitor(int func_no, char* html_buf) {

/* Dispatch function by function no. */
switch(func_no) {

/* Client startup function */
case -1:
    ClientLogCheck(html_buf);
    break;

/* Client shutdown */
case -2:
    ClientSetSample(html_buf);
    break;

/* Client monitor */
case -3:
    ClientInfSample(html_buf);
    break;

default:
/* Error return */
return -1;
break;
}

return 0;
}

/*****
* Check client's log files.
* Check files are ...
* use:log.log : TpApl log
* DBDepend_Userlog.log : SvrApl log
*
* Return Value
* NONE
* Return Information
* HTML document
*****/

void ClientLogCheck(char* html_buf) {

int CheckLogFile(char* file_path, char*
key_word);

#define NO_ERROR_LOG "No error found."
#define CLIENT_LOG_CHECK "\
<HTML><HEAD><TITLE>Client Log
Check</TITLE></HEAD><BODY>\r\n\
<P> \
The %s check log files.\r\n\
Result : %s \r\n\
</P></BODY></HTML>\r\n"

```

```

char host_name[32];

/* Get host name */
host_name[0] = '\0';
gethostname(host_name, sizeof(host_name));

/* Check TpApl log file */
if (CheckLogFile(GLB_TpAplLogPath,
":ERR:") == 0) {
    /* No error */
    sprintf(html_buf, CLIENT_LOG_CHECK,
host_name, NO_ERROR_LOG);
} else {
    /* Error found */
    sprintf(html_buf, CLIENT_LOG_CHECK,
host_name, "Error in userlog.log");
    return;
}

/* Check SvrApl log file */
if (CheckLogFile(GLB_SvrAplLogPath,
":ERR:") == 0) {
    /* No error */
    sprintf(html_buf, CLIENT_LOG_CHECK,
host_name, NO_ERROR_LOG);
} else {
    /* Error found */
    sprintf(html_buf, CLIENT_LOG_CHECK,
host_name, "Error in DBDepend_Userlog.log");
}

/*****
* Check log files has error key word.
*
* Return Value
* >0 : found number of keywords
*
* -1 : file open error (maybe no exist
*
*****/
int CheckLogFile(char* file_path, char*
key_word) {

    FILE* log_file;
    char rd_buff[256];
    int find_words = 0;

    if ((log_file = fopen(file_path, "r")) == NULL) {
        /* Open error */
        return -1;
    }

    while (fgets(rd_buff, sizeof(rd_buff),
log_file) != NULL) {

        if (strstr(rd_buff, key_word) != NULL) {
            find_words++;
        }
    }
    fclose(log_file);
    return find_words;
}

/*****
* Set sampling disable.
* Return Value
* NONE
* Return Information

```

```

* HTML document
*
*****/
void ClientSetSample(char* html_buf) {

#define CLIENT_DIRECT "\
<HTML><HEAD><TITLE>Client sampling
disable</TITLE></HEAD><BODY>\r\n\
<P> \
The %s set sampling disable.\r\n\
Result : No error found.\r\n\
</P></BODY></HTML>\r\n"

    char host_name[32];

    GLBSMP_shared_mem->DataSampling =
DATASAMPLE_DISABLE;

    host_name[0] = '\0';
    gethostname(host_name, sizeof(host_name));
    sprintf(html_buf, CLIENT_DIRECT,
host_name);
}

/*****
* Client performance information Sampling
*
* Return Value
* NONE
* Return Information
* HTML document
*
*****/
void ClientInfSample(char* html_buf) {

#define CLIENT_SAMPLE "\
<HTML><HEAD><TITLE>Client Sampling
information</TITLE></HEAD><BODY>\
<PRE>\
Information of %s\r\n\
\r\n\
TpApl performance \r\n\
Sto\r\n\
    Num of waiting process %-7d %-7d %-
7d %-7d\r\n\
    Answer to RTE (ms) %-7d %-7d %-
7d %-7d\r\n\
\r\n\
SvrApl performance \r\n\
    SMAN MAX AVR
TRX\r\n\
    New Order Response %-7d %-7d %-
7d %-7d\r\n\
    Payment Response %-7d %-7d %-
7d %-7d\r\n\
    Order Status Response %-7d %-7d %-
7d %-7d\r\n\
    Derivery Response %-7d %-7d %-7d %-
7d\r\n\
    Stock Level Response %-7d %-7d %-
7d %-7d\r\n\
</PRE></BODY></HTML>\r\n"

#define EXT_FUNC3_ERROR "\
<HTML><HEAD><TITLE>Error
information</TITLE></HEAD><BODY>\
<PRE>\
Failure create SvrAPL object(Extended function
= -3)\
</PRE></BODY></HTML>\r\n"


```

```

#if 0
#define CLIENT_SAMPLE "\
<HTML><HEAD><TITLE>Client Sampling
information</TITLE></HEAD><BODY>\
<PRE>\
Information of CL001\r\n\
\r\n\
TpApl performance \r\n\
    New Pay Odr Del
Sto\r\n\
    Num of waiting process 10 20 30
40 50\r\n\
    Answer to RTE (ms) 110 220 330
440 550\r\n\
\r\n\
SvrApl performance \r\n\
    SMAN MAX AVR
TRX\r\n\
    New Order Response 10 11 12
13\r\n\
    Payment Response 110 111 112
113\r\n\
    Order Status Response 210 211 212
213\r\n\
    Derivery Response 310 311 312
313\r\n\
    Stock Level Response 410 411 412
413\r\n\
</PRE></BODY></HTML>\r\n"
#endif

char host_name[32];
unsigned int ans_new_avr, ans_pay_avr,
ans_odr_avr, ans_del_avr, ans_sto_avr;
unsigned int rsp_new_avr, rsp_pay_avr,
rsp_odr_avr, rsp_del_avr, rsp_sto_avr;

    SAMPLING_DATA sampling_data;

/* Get host name, inserting to HTML */
host_name[0] = '\0';
gethostname(host_name, sizeof(host_name));

/* copy sampling information into own area */
LockSem(GLBSMP_semid);
memcpy((void*)&sampling_data,
(void*)GLBSMP_shared_mem,
(size_t)sizeof(SAMPLING_DATA));

/* Clear sampling information for next
sampling interval */
memset((void*)GLBSMP_shared_mem, 0x00,
(unsigned int)&(SAMPLING_DATA*0)-
>MaxRspTimeNewOrder);

    UnlockSem(GLBSMP_semid);

/* Compute average data */
ans_new_avr =
sampling_data.NumReqNewOrder != 0?
sampling_data.AnsNewOrder /
sampling_data.NumReqNewOrder : 0;
ans_pay_avr =
sampling_data.NumReqPayment != 0?
sampling_data.AnsPayment /
sampling_data.NumReqPayment : 0;
ans_odr_avr =
sampling_data.NumReqOrderStatus != 0?
sampling_data.AnsOrderStatus /
sampling_data.NumReqOrderStatus : 0;
ans_del_avr =
sampling_data.NumReqDelivery != 0?
sampling_data.AnsDelivery /
sampling_data.NumReqDelivery : 0;

```



```

ans_sto_avr =
sampling_data.NumReqStockLevel != 0?
    sampling_data.AnsStockLevel /
sampling_data.NumReqStockLevel : 0;

rsp_new_avr =
sampling_data.NumNewOrder != 0?
    sampling_data.RspTimeNewOrder /
sampling_data.NumNewOrder : 0;
rsp_pay_avr =
sampling_data.NumPayment != 0?
    sampling_data.RspTimePayment /
sampling_data.NumPayment : 0;
rsp_odr_avr =
sampling_data.NumOrderStatus != 0?

sampling_data.RspTimeOrderStatus /
sampling_data.NumOrderStatus : 0;
rsp_del_avr = sampling_data.NumDelivery !=
0?
    sampling_data.RspTimeDelivery /
sampling_data.NumDelivery : 0;
rsp_sto_avr =
sampling_data.NumStockLevel != 0?
    sampling_data.RspTimeStockLevel
/ sampling_data.NumStockLevel : 0;

sprintf(html_buf, CLIENT_SAMPLE ,
    host_name,
    sampling_data.NumQueNewOrder,
    sampling_data.NumQuePayment,
    sampling_data.NumQueOrderStatus,
    sampling_data.NumQueDelivery,
    sampling_data.NumQueStockLevel,
    ans_new_avr, ans_pay_avr, ans_odr_avr,
    ans_del_avr, ans_sto_avr,

    sampling_data.SMaxRspTimeNewOrder,
    sampling_data.MaxRspTimeNewOrder,
    rsp_new_avr,
    sampling_data.NumNewOrder,
    sampling_data.SMaxRspTimePayment,
    sampling_data.MaxRspTimePayment,
    rsp_pay_avr,
    sampling_data.NumPayment,
    sampling_data.SMaxRspTimeOrderStatus,
    sampling_data.MaxRspTimeOrderStatus,
    rsp_odr_avr,
    sampling_data.NumOrderStatus,
    sampling_data.SMaxRspTimeDelivery,
    sampling_data.MaxRspTimeDelivery,
    rsp_del_avr,
    sampling_data.NumDelivery,
    sampling_data.SMaxRspTimeStockLevel,
    sampling_data.MaxRspTimeStockLevel,
    rsp_sto_avr,
    sampling_data.NumStockLevel);
}

/*****
* Initialize sampling
* Return Value
* NONE
*****/

void ClientSampleInit() {
#define SAMPLING_CONF_FILE
"/home/tpc/conf/sampling.conf"
#define DEFAULT_CSV_FILE
"/home/tpc/log/sampling.csv"
#define DEFAULT_SAMPLING_INTERVAL 5

FILE* conf_file;

```

```

char rd_buff[MAX_PATH];
int i;

/* Initialize shared memory */
MAC_SampleInitParent;

/* Setup sampling configuration */
if ((conf_file = fopen(SAMPLING_CONF_FILE,
"r")) == NULL) {
    GLBSMP_shared_mem-
>SelfSamplingOutput =
SELFOUTPUT_DISABLE;
    return;
}
GLBSMP_shared_mem->SelfSamplingOutput
= SELFOUTPUT_ENABLE;

/* CSV file path */
if (fgets(rd_buff, sizeof(rd_buff), conf_file) ==
NULL) {
    strcpy(GLBSMP_shared_mem-
>CsvFilePath, DEFAULT_CSV_FILE);
    GLBSMP_shared_mem->SamplingInterval
= DEFAULT_SAMPLING_INTERVAL;
    goto FILE_CLOSE;
}
for(i = 0; !(rd_buff[i] == '\n' || rd_buff[i] == '\0') ;
i++);
rd_buff[i] = '\0';
strcpy(GLBSMP_shared_mem->CsvFilePath,
rd_buff);

/* Sampling interval */
if (fgets(rd_buff, sizeof(rd_buff), conf_file) ==
NULL) {
    GLBSMP_shared_mem->SamplingInterval
= DEFAULT_SAMPLING_INTERVAL;
    goto FILE_CLOSE;
}
GLBSMP_shared_mem->SamplingInterval =
atoi(rd_buff);

FILE_CLOSE:
fclose(conf_file);
}

/*****
*****/
* Self CSV data output
* Return Value
* NONE
*****/

void ClientSampleSelfCsv(time_t cur_sec) {

    FILE* csv_file;

#define TITLE_LINE
"time,num_thread,stay_New,stay_Pay,stay_Odr,
stay_Del,stay_Sto,\n"

"resp_New,num_New,resp_Pay,num_Pay,resp_
Odr,num_Odr,resp_Del,num_Del,resp_Sto,num
_Sto,\n"

"imax_New,imax_Pay,imax_Odr,imax_Del,imax
_Sto,\n"

"max_New,max_Pay,max_Odr,max_Del,max_St
o,\n"

"ans_New,nas_Pay,ans_Odr,ans_Del,ans_Sto,c
onnect\n"

```

```

/* -- BEIGN -- Modified by Hayashi for thread-
safe. 2006/02/13 */
#if 0
! struct tm *nowtime;
#else
struct tm tt;
struct tm *nowtime= &tt;
#endif
/* -- END -- Modified by Hayashi for thread-safe.
2006/02/13 */

unsigned int ans_new_avr, ans_pay_avr,
ans_odr_avr, ans_del_avr, ans_sto_avr;
unsigned int rsp_new_avr, rsp_pay_avr,
rsp_odr_avr, rsp_del_avr, rsp_sto_avr;

SAMPLING_DATA sampling_data;

if (GLBSMP_shared_mem-
>SelfSamplingOutput ==
SELFOUTPUT_DISABLE) {
/* Output disable */
return;
}

LockSem(GLBSMP_semaphore);
if ((cur_sec - GLBSMP_shared_mem-
>CsvOutTime) < GLBSMP_shared_mem-
>SamplingInterval) {
/* No output timing */
goto UNLOCK_SEM;
}

/* Output CSV data */
if ((csv_file = fopen(GLBSMP_shared_mem-
>CsvFilePath, "a")) == NULL) {
goto UNLOCK_SEM;
}

if (GLBSMP_shared_mem->CsvOutTime ==
0) {
/* First time, output header data */
fprintf(csv_file, TITLE_LINE);
fclose(csv_file);
GLBSMP_shared_mem->CsvOutTime =
cur_sec;
goto UNLOCK_SEM;
}
GLBSMP_shared_mem->CsvOutTime =
cur_sec;

/* copy sampling information into own area */
memcpy((void*)&sampling_data,
(void*)GLBSMP_shared_mem,
(size_t)sizeof(SAMPLING_DATA));

/* Clear sampling information for next
sampling interval */
memset((void*)GLBSMP_shared_mem, 0x00,
(unsigned int)&((SAMPLING_DATA*)0)-
>MaxRspTimeNewOrder);

/* Compute average data */
ans_new_avr =
sampling_data.NumReqNewOrder != 0?
    sampling_data.AnsNewOrder /
sampling_data.NumReqNewOrder : 0;
ans_pay_avr =
sampling_data.NumReqPayment != 0?

```

```

        sampling_data.AnsPayment /
sampling_data.NumReqPayment : 0;
    ans_odr_avr =
sampling_data.NumReqOrderStatus != 0?
        sampling_data.AnsOrderStatus /
sampling_data.NumReqOrderStatus : 0;
    ans_del_avr =
sampling_data.NumReqDelivery != 0?
        sampling_data.AnsDelivery /
sampling_data.NumReqDelivery : 0;
    ans_sto_avr =
sampling_data.NumReqStockLevel != 0?
        sampling_data.AnsStockLevel /
sampling_data.NumReqStockLevel : 0;

    rsp_new_avr =
sampling_data.NumNewOrder != 0?
        sampling_data.RspTimeNewOrder /
sampling_data.NumNewOrder : 0;
    rsp_pay_avr =
sampling_data.NumPayment != 0?
        sampling_data.RspTimePayment /
sampling_data.NumPayment : 0;
    rsp_odr_avr =
sampling_data.NumOrderStatus != 0?

sampling_data.RspTimeOrderStatus /
sampling_data.NumOrderStatus : 0;
    rsp_del_avr = sampling_data.NumDelivery !=
0?
        sampling_data.RspTimeDelivery /
sampling_data.NumDelivery : 0;
    rsp_sto_avr =
sampling_data.NumStockLevel != 0?
        sampling_data.RspTimeStockLevel
/ sampling_data.NumStockLevel : 0;

/* Output sampling data */
/* -- BEIGN -- Modified by Hayashi for thread-
safe. 2006/02/13 */
#if 0
! nowtime = localtime( &cur_sec );
#else
    localtime_r( &cur_sec, nowtime );
#endif
/* -- END -- Modified by Hayashi for thread-safe.
2006/02/13 */

fprintf(csv_file,
        "%02d-%02d %02d:%02d:%02d,",
        nowtime->tm_mon+1, nowtime->tm_mday,
nowtime->tm_hour, nowtime->tm_min, nowtime-
>tm_sec);

/* Number of thread (no sampling information)
*/
fprintf(csv_file, "%d,", 0);

/* Waiting process queue */
fprintf(csv_file, "%d,",
sampling_data.NumQueueNewOrder);
fprintf(csv_file, "%d,",
sampling_data.NumQueuePayment);
fprintf(csv_file, "%d,",
sampling_data.NumQueueOrderStatus);
fprintf(csv_file, "%d,",
sampling_data.NumQueueDelivery);
fprintf(csv_file, "%d,",
sampling_data.NumQueueStockLevel);

/* Responce time & number of proesing
trasection */
fprintf(csv_file, "%3f,", (float)rsp_new_avr /
1000.0);

```

```

        fprintf(csv_file, "%d,",
sampling_data.NumNewOrder);
        fprintf(csv_file, "%3f,", (float)rsp_pay_avr /
1000.0);
        fprintf(csv_file, "%d,",
sampling_data.NumPayment);
        fprintf(csv_file, "%3f,", (float)rsp_odr_avr /
1000.0);
        fprintf(csv_file, "%d,",
sampling_data.NumOrderStatus);
        fprintf(csv_file, "%3f,", (float)rsp_del_avr /
1000.0);
        fprintf(csv_file, "%d,",
sampling_data.NumDelivery);
        fprintf(csv_file, "%3f,", (float)rsp_sto_avr /
1000.0);
        fprintf(csv_file, "%d,",
sampling_data.NumStockLevel);

/* Max processing time in sampling interval */
fprintf(csv_file, "%3f,",
(float)sampling_data.SMaxRspTimeNewOrder /
1000.0);
        fprintf(csv_file, "%3f,",
(float)sampling_data.SMaxRspTimePayment /
1000.0);
        fprintf(csv_file, "%3f,",
(float)sampling_data.SMaxRspTimeOrderStatus
/ 1000.0);
        fprintf(csv_file, "%3f,",
(float)sampling_data.SMaxRspTimeDelivery /
1000.0);
        fprintf(csv_file, "%3f,",
(float)sampling_data.SMaxRspTimeStockLevel /
1000.0);

/* Max processing time in all time */
fprintf(csv_file, "%3f,",
(float)sampling_data.MaxRspTimeNewOrder /
1000.0);
        fprintf(csv_file, "%3f,",
(float)sampling_data.MaxRspTimePayment /
1000.0);
        fprintf(csv_file, "%3f,",
(float)sampling_data.MaxRspTimeOrderStatus /
1000.0);
        fprintf(csv_file, "%3f,",
(float)sampling_data.MaxRspTimeDelivery /
1000.0);
        fprintf(csv_file, "%3f,",
(float)sampling_data.MaxRspTimeStockLevel /
1000.0);

/* Ans time to RTE */
fprintf(csv_file, "%3f,", (float)ans_new_avr /
1000.0);
        fprintf(csv_file, "%3f,", (float)ans_pay_avr /
1000.0);
        fprintf(csv_file, "%3f,", (float)ans_odr_avr /
1000.0);
        fprintf(csv_file, "%3f,", (float)ans_del_avr /
1000.0);
        fprintf(csv_file, "%3f,", (float)ans_sto_avr /
1000.0);

/* Number of connection (no sampling) */
fprintf(csv_file, "%d", 0);

fprintf(csv_file, "\n");

fclose(csv_file);

UNLOCK_SEM:
    UnlockSem(GLBSPM_semid);
return;

```

```

}

.....:
tpapl/ConvInt.c
.....:

/*****
****
*
*
* TPC-C Client Application Program Source
*
*
*
* Entry Functions
* (1) str2int
* (2) str2short
* (3) str2dbl
*
*
* CREATE by TSL 2002.10.01
*
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2002 *

*****
****/
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#define numcheck(num) (0x30 <= num && num
<= 0x39) /* 0 - 9 */
#define alpcheck(num) (0x41 <= num && num
<= 0x5a) /* A - Z */

/*
str2int :
takes a string, makes sure it's not too long,
and ensures that it
represents an integer.
If it does, the corresponding int value is
returned.

-3: there is not string data.
-2: find not character data.
-1: string data is too many long
*/
int str2int(char *str, int field_len) {
    int x;

//for warning
// if(str == 0 || !(x = strlen(str))) return -3;
if(str == 0 || (x = strlen(str)) == 0) return -3;

if(x > field_len){
    if (strchr (str, '%') != 0) /* 98.8.3 :-----
----- */
        return -2;
    else
        return -1;
}
else{
    for( ; x > 0; x--){
        if (numcheck(str[x-1])) {
            return -2;
        }
    }
    return atoi(str);
}

/*
str2short :
takes a string, makes sure it's not too long,
and ensures that it

```

```

represents an integer.
If it does, the corresponding short value is
returned.

-3: there is not string data.
-2: find not character data.
-1: string data is too many long
*/
short str2short(char *str, int field_len) {
    int x;

    //for warning
    // if(str == 0 || !(x = strlen(str))) return -3;
    if(str == 0 || (x = strlen(str)) == 0) return -3;

    if(x > field_len){
        if (strchr (str, '%') != 0) /* 98.8.3 :-----
    ----- */
            return -2;
        else
            return -1;
    }
    else {
        for( ; x; x--){
            if (!numcheck(str[x-1]))
                return -2;
        }
    }
    x = atoi(str);
    return (short)x;
}

/*
str2dbl :
takes a string, makes sure it's not too long,
and makes sure that it
represents a floating point number.
If so, delete the decimal point.
As a result, the value is increased hundredfold.
this function is returned integer value.

!! This function use Payment transaction only.

-3: there is not string data.
-2: find not character data.
-1: string data is too many long

*/
int str2dbl(char *str, int field_len) {
    int x, len, cnt;
    /* Replaced T.Kato 03.08.20 Bug Fix --over 5
    column integer is memory crush -- */
    /* total 5+2+1(NULL)bytes
    but editing area is 7bytes */
    /* char NUM[7];*/
    char NUM[16];
    /* Replaced end */

    char pointf = 0;
    int fcnt = 2; /* */

    //for warning
    // if(str == 0 || !(x = strlen(str))) return -3;
    if(str == 0 || (x = strlen(str)) == 0) return -3;

    len = x;

    if(x > field_len){
        if (strchr (str, '%') != 0) /* 98.8.3 :-----
    ----- */
            return -2;
        else
            return -1;
    }
}

```

```

else{
    /* check string data */
    for(;x;x--) {
        if(numcheck(str[x-1]));
        else if((str[x-1] == '.') && ((len - x) < 3));
        else if((str[x-1] == '-') && (x == 1));
        else if((str[x-1] == '+') && (x == 1));
        else return -2;
    }
}

/* delete the decimal point. As a result,do
hundredfold the value.*/
for (cnt = 0, x = 0; x < len; x++){

    if ( str[x] == '.' ){
        /* find the decimal point. set point flag.*/
        pointf = 1;
    } else {
        /* set character to work buffer.*/
        NUM[cnt] = str[x]; cnt++;

        /* The figure below the decimal point was
detected */
        if ( pointf == 1 ) {fcnt--;}
    }

    if ( pointf == 1 && fcnt > 0 ){
        /*There was no figure below the decimal
point or only one digit was
found.: ----- */
        for ( ; fcnt > 0; fcnt-- ) {
            NUM[cnt++] = '0';
        }
    }
    else if ( pointf == 0 ) {
        /* There is no decimal point.: -----
--- */
        NUM[cnt++] = '0'; NUM[cnt++] = '0';
    }

    NUM[cnt] = 0;

    return (atoi(NUM));
}

.....:
tpapl/ConvOther.c
.....:

/*****
****
*
*
* TPC-C Client Application Program Source
*
*
* Entry Functions *
* (1) para_split *
* (2) checkHTMLform *
* (3) convert_time *
* (4) convert_date *
*
* CREATE by TSL 2002.10.01
*
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2002 *
*****
****/
#include <stdio.h>
#include <stdlib.h>

```

```

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

/*
para_split :
----- (QueryString)-----
-----
-----: -----NULL-----
---
-----NULL-----

Split divides up a string based on the first
instance of a specified
delimiter ('sp'). The first instance of 'sp' is
converted to a NULL
and the address of the first character of the
second half is returned.
Thus the user has the first half (which he
passed in and still has) and
the second half (which was returned) with a
NULL between them. Yay.
(Yes, strtok does this, sort of, but I can't nest
strtok calls.)
*/
char *para_split(char *para, char delimita) {
    char *point = para;;

    /* The address of the delimitation character is
calculated */
    /* ----- */
    // if ((point = strchr (para, delimita)) == NULL)
    // return (char *)0;

    for(; !(*point == '\0' || *point == delimita);
point++);
    if (*point == '\0')
        return (char *)0;

    /* The delimitation character is replaced with
NULL*/
    *point = '\0'; /* -----NULL----- */

    /* The first position of the analyzed variable is
returned.*/
    return (point + 1); /* ----- */
}

/*
check HTML form

*/
int checkHTMLform( char *str, char *buffer)
{
    char* src = str;
    char* dst = buffer;

    while (*src != '\0'){
        if ( *(src) == '&' ){
            *(dst) = '&'; dst++;
            *(dst) = 'a'; dst++;
            *(dst) = 'm'; dst++;
            *(dst) = 'p'; dst++;
            *(dst) = ';'; dst++;
        }
        else if ( *(src) == '<' ) {
            *(dst) = '&'; dst++;
            *(dst) = 'l'; dst++;
            *(dst) = 't'; dst++;
        }
    }
}

```

```

        *(dst) = ':'; dst++;
    }
    else if ( *(src) == '>' ) {
        *(dst) = '&'; dst++;
        *(dst) = 'g'; dst++;
        *(dst) = 't'; dst++;
        *(dst) = ':'; dst++;
    }
    else if ( *(src) == "" ) {
        *(dst) = '&'; dst++;
        *(dst) = 'q'; dst++;
        *(dst) = 'u'; dst++;
        *(dst) = 'a'; dst++;
        *(dst) = 't'; dst++;
        *(dst) = ':'; dst++;
    }
    else {
        *(dst) = *(src);
        dst++;
    }
    src++;
}

*(dst) = 0;
return ( (unsigned long)dst - (unsigned
long)buffer );
}

//
// The date data is converted. (The time data is
not contained.)
// Numeric data is converted into character string
data.
//
void convert_time( char *save_p, double time )
{
/* Replaced T.Kato 20005.01.21 For thread safe
*/
#if 0
! struct tm* tim;
! time_t tt = ( time_t )time;
!
! tim = localtime( &tt );
#endif
struct tm tm_data;
struct tm* tim = &tm_data;
time_t tt = ( time_t )time;

localtime_r( &tt, tim );
/* Replaced end */

sprintf( save_p, "%02d-%02d-%02d-
%04d %02d:%02d:%02d",
tim->tm_mday, tim->tm_mon+1, tim-
>tm_year + 1900,
tim->tm_hour, tim->tm_min, tim->tm_sec );
}

//
// The date data is converted. (The time data is
contained.)
// Numeric data is converted into character string
data.
//
void convert_date( char *save_p, double time )
{
/* Replaced T.Kato 2005.01.21 For thread safe */
#if 0
! struct tm* tim;
! time_t tt = ( time_t )time;
!
! tim = localtime( &tt );
#endif

```

```

struct tm tm_data;
struct tm* tim = &tm_data;
time_t tt = ( time_t )time;

tim = localtime_r( &tt, tim );
/* Replaced end */

sprintf( save_p, "%02d-%02d-%04d",
tim->tm_mday, tim->tm_mon + 1, tim-
>tm_year + 1900 );
}

.....
tpapl/ConvString.c
.....

/******
****
*
* TPC-C Client Application Program Source
*
*
* Entry Functions
* (1) int2str
* (2) int3str
* (3) dec2str
* (4) sigdec2str
* (5) str2str
* (6) alp2str
* (7) date2str
* (8) zip2str
* (9) phone2str
*
* CREATE by TSL 2002.10.01
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2002 *
*****
****/
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

/*
int2str : Converts an integer value to a string of
a specified length and
outputs the string to the memory buffer
supplied.

field = the destination field
field_size = number of characters to output
value = integer to be displayed
*/
void int2str(char *str, int len, int num)
{
int cnt;

for (cnt = len - 1; cnt >= 0; cnt--){

str[cnt] = (char)((num % 10) + '0');
num /= 10;
}

for (cnt = 0; cnt < len-1; cnt++){

if (str[cnt] == '0')
str[cnt] = ' ';
else
return;
}
}

```

```

/*
int3str : Converts an integer value to a string of
a specified length and
outputs the string to the memory buffer
supplied.

field = the destination field
field_size = number of characters to output
value = integer to be displayed
*/
void int3str(char *str, int len, int num)
{
int cnt;

for (cnt = len - 1; cnt >= 0; cnt--){

str[cnt] = (char)((num % 10) + '0');
num /= 10;
}
}

/*
dec2str:
Converts a double precision floating point
value to a string of
a specified length and outputs the string to the
memory buffer supplied.
This routine assumes the following restrictions
apply:
Precision is fixed at 2 places to the right of the
decimal point.
No string length will be less than 4.

field = the destination field
field_size = number of characters to output
value = floating point number to be displayed
*/
void dec2str(char *str, int len, double num)
{
int dec, sign, i, cnt;

/* Replaced T.Kato 2005.01.21 For thread safe */
#if 0
! char *string;
!
! string = ecvt(num, len-1, &dec, &sign);
#endif

char string_buf[17];
char *string = string_buf;

ecvt_r(num, len-1, &dec, &sign, string,
sizeof(string_buf)-1);
/* Replaced end */

/* dec = -----,sign = ---0,---1,string=-----
----- */

if ( dec > 0 ) {
/* if the integer part is not zero ..
Exsample :num data is 1234.56 */
cnt = (len - 3) - dec;

/* -----: "0012" -> " 12"
*/
/* If the high-order digit is zero , zero is
changed at the blank */
for (i = 0; i < cnt; i++){
/* pad with blank in the high part of the
number */
str[i] = ' ';
}
}
}

```

```

    /* The high-order digit set to the output
area: -----*/
    for (; i < (len - 3); i++){
        str[i] = *(string++);
    }
}
else {
    /* If the integer part is zero ... Example:
num data is 0.12 */
    cnt = len - 4;

    for (i = 0; i < cnt; i++){
        /* pad with blank in the high part of the
number */
        str[i] = ' ';
    }
    str[i++] = '0';
}

str[i++] = '.';

for (; dec < 0 && i < len; dec++, i++){
    /* pad with 0's in the high part of the fraction
*/
    str[i] = '0';
}

for (; i < len; i++){
    /* copy the decimal portion (2 places) */
    str[i] = *(string++);
}
}

/*
sigdec2str:
Converts a double precision floating point
value to a string of
a specified length and outputs the string to the
supplied buffer.
If the value is negative, the first character will
be a minus sign (-).

field = the destination field
field_size = number of characters to output
value = floating point number to be displayed
*/
void sigdec2str(char *str, int len, double num)
{
    if (num >= 0.0) {
        str[0] = '.';
        dec2str (&str[1], len - 1, num);
    } else {
        str[0] = '-';
        dec2str (&str[1], len - 1, -num);
    }
}

/*
str2str :
makes sure the string exists and isn't too long.

-1: string data is too many long
-2: find not figure data.
0: there is not string data.
1: normal end
*/
int str2str(char *str, int field_len) {
    int x;

    //for warning
    // if (str == 0 || !(x = strlen (str))) return 0;
    if (str == 0 || (x = strlen (str)) == 0) return 0;

    if(x > field_len) {

```

```

        if ( strchr (str, '%') != 0) /* 98.8.3 :-----
----- */
            return -2;
        else
            return -1;
    }
}
/*
else {
    for(; x ; x--){
        if (!alpcheck(str[x-1]))
            return -2;
    }
}
*/
return 1;
}

/*
alp2str : Outputs a string into the memory
space supplied.

field = the destination field
field_size = number of characters to output
string = alpha string to be displayed
*/
void alp2str(char *str, int len, char *alp)
{
    int cnt;

    cnt = strlen (alp);
    strncpy (str, alp, len); /* copy to destination
area */

    /* len-----*/
    /* If not coming up to the specified length then
set the blank. */
    if (len - cnt > 0)
        memset (&str[cnt], ' ', len - cnt);
}

/*
date2str : Outputs a date in the supplied buffer
in the following format:
DD-MM-YYYY

field = the destination field
date = date to be converted and displayed
*/
void date2str(char *str, char *time)
{
    int year, month, day;

#ifdef DBPRT
    fprintf (test_fp, "date2: %s\n", time);
#endif
    sscanf( time, "%d-%d-%d", &day, &month,
&year );

    int3str (str, 2, day);
    str[2] = '-';
    int3str (&str[3], 2, month);
    str[5] = '-';
    int3str (&str[6], 4, year);
}

/*
zip2str:
Outputs a zipcode in the supplied buffer in the
following format:
XXXXX-XXXX

str = the destination field
zip = the zipcode to be output
*/
void zip2str (char *str, char *zip)

```

```

{
    alp2str (str, 5, zip);
    str[5] = '-';
    alp2str (&str[6], 4, &zip[5]);
}

/*
phone2str:
Outputs a phone number in the supplied buffer
in the following format:
XXXXXX-XXX-XXX-XXXX

str = the destination field
phone = the phone number to be output
*/
void phone2str(char *str, char *phone)
{
    alp2str (str, 6, phone);
    str[6] = '-';

    alp2str (&str[7], 3, &phone[6]);
    str[10] = '-';

    alp2str (&str[11], 3, &phone[9]);
    str[14] = '-';

    alp2str (&str[15], 4, &phone[12]);
}

.....
tpapl/ErrPage.c
.....

/*****
*
* TPC-C Client Application Program Source
*
* Entry Functions
* (1) set_errHTML
* (2) set_SvrApplErr
* (3) set_errpage
*
* CREATE by TSL 2002.10.01
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2002
*****/
#include "forlinux.h"

#include <stdio.h>
#include <string.h>
#include "tpcweb.h"

#include "tpapl.h"
#include <pthread.h>
#include <atmi.h>
#include "GlobalArea.h"

/*
set_errHTML :
this function make error message of
application program.
*/
int set_errHTML (char *page, char *err_inf, int
cookie, char *errname) {

```

```

    printf(page, errorpage, errname, err_inf,
    SOPATH, cookie);

    return 0;
}

#if 0
!/* #ifdef symfo"-----Oracle--Symfo-----
------(set_errHTML)--
! set_or Kerr :
! this function make error message of the
Oracle application program.
!*/
lint set_or Kerr (char *page, char *err_inf, int
cookie) {
!
!#ifdef Symfo
! printf(page, symfoerr, err_inf, SOPATH,
cookie);
!#endif
!
! return 0;
!}
#endif

/*
set_tuxerr :
this function make error message of the TP-
application program.
*/
/* Replaced 03.01.15 */
#if 0
lint set_tuxerr (char *page, char *err_inf, int
cookie) {
#endif
int set_SvrAplErr (char *page, char *err_inf, int
cookie) {
/* Replaced end */

    printf(page, tuxerr, err_inf, SOPATH, cookie);

    return 0;
}

/* Error message list : these are notified from
CLINET to RTE */
/* 98.8.3 : ----- */
char errstrings[23][166] = {
"The function you selected doesn't exist.\n"
"Don't enter URLs manually!\n%s",
/* 0 */

"You seem to have responded to a form that
doesn't exist.\n"
"Don't enter URLs manually!\n%s",
/* 1 */

"The District ID you entered isn't valid.\n%s\n"
"It must be an integer in the range 1 to 10.\n",
/* 2 */

"The threshold value you entered isn't
valid.\n%s\n"
"It must be an integer in the range 10 to 20.\n",
/* 3 */

"The terminal number you entered isn't
valid.\n%s\n"
"It must be an integer in the range 1 to %d.\n",
/* 4 */

"The Carrier ID you entered isn't valid.\n%s\n"

```

```

"It must be an integer in the range 1 to 10.\n",
/* 5 */

"The Customer ID you entered isn't
valid.\n%s\n"
"It must be an integer of 4 or fewer digits.\n",
//It must be an integer in the 1 to 3000.\n",
/* 6 */

"The Customer Last Name you entered isn't
valid.\n%s\n"
"It must be a string shorter than 16
characters.\n", /* 7 */

"The Payment Amount you entered isn't
valid.\n%s\n"
"It must be a dollar amount, without the dollar
sign,"
" between $1.00 and $5000.00.\n",
/* 8 */

"The Customer Warehouse ID you entered isn't
valid.\n%s\n"
"It must be an integer in the range 1 to %d.\n",
/* 9 */

"The Customer District ID you entered isn't
valid.\n%s\n"
"It must be an integer in the 1 to 10.\n",
/* 10 */

"You must enter either a Customer ID or a
Customer Last Name.\n"
"You left both fields blank.\n%s",
/* 11 */

"The Warehouse ID you entered isn't
valid.\n%s\n"
"It must be an integer in the range 1 to %d.\n",
/* 12 */

"On entry line %d, the data you entered for
the %s field isn't valid.\n%s\n", /* 13 */

"Supply Warehouse ID",
/* 14 */

"Item ID", /* 15 */
*/

"Quantity", /*
16 */
*/
"Your entry was outside the range.",
/* 17 */
*/
"You didn't entry anything for the field.",
/* 18 */
*/
"Your entry contained too many characters.",
/* 19 */
*/
"The input data is wrong data type, must be
numeric.", /* 20 */
*/
"It must be an integer in the range 1 to %d.",
/* 21 */
*/
"The input data is wrong data type, must be
english capital letter.", /* 22 */
*/
};

/*
set_errpage:

RTE-----
-----

a generic error page generator. If the user
does anything screwy,

```

```

s/he gets here. The function generates an
error page based on the
two errlvl arguments and returns it for the user..

When err_no is 13 or more, Order Line Data is
Abnormal.
(err_no is the error data line number)

98.8.3 : -----
*/
int set_errpage (char *buf, int user, int err_no, int
err_inf, int sub_inf, int sub_inf2) {
char errmsg[1024];
int nchar;
int length;

//for warning
sub_inf;
nchar;

if(err_no >= 13) { /* OrderLine
Data(Neworder) is Abnormal */
switch(err_inf) {
case -5: /* S_W_ID data is abnormal
*/
printf(errmsg, errstrings[13], err_no-
12, errstrings[14], errstrings[20]);
sub_inf2 = GLB_Numwh;
break;
case -8: /* S_W_ID data is uninput */
printf(errmsg, errstrings[13], err_no-
12, errstrings[14], errstrings[18]);
sub_inf2 = GLB_Name;
break;
case -15: /* S_W_ID data is outside
range */
printf(errmsg, errstrings[13], err_no-
12, errstrings[14], errstrings[17]);
sub_inf2 = GLB_Numwh;
break;

case -1: /* L_ID data is uninput */
printf(errmsg, errstrings[13], err_no-
12, errstrings[15], errstrings[18]);
sub_inf2 = 100000;
break;
case -6: /* L_ID data is abnormal */
printf(errmsg, errstrings[13], err_no-
12, errstrings[15], errstrings[20]);
sub_inf2 = 100000;
break;
case -16: /* L_ID data is outside
range */
printf(errmsg, errstrings[13], err_no-
12, errstrings[15], errstrings[17]);
sub_inf2 = 100000;
break;

case -7: /* Quantity data is abnormal
*/
printf(errmsg, errstrings[13], err_no-
12, errstrings[16], errstrings[20]);
sub_inf2 = 10;
break;
case -2: /* Quantity data is uninput */
printf(errmsg, errstrings[13], err_no-
12, errstrings[16], errstrings[18]);
sub_inf2 = 10;
break;
case -17: /* Quantity data is outside
range */
printf(errmsg, errstrings[13], err_no-
12, errstrings[16], errstrings[17]);
sub_inf2 = 10;
break;

```

```

default:
    break;
}

length = strlen(errmsg);
sprintf(&errmsg[length], errstrings[21],
sub_inf2);
sprintf(buf, errhtml, errmsg, SOPATH,
user);
}
else if ( err_no == 4 || err_no == 9 || err_no
== 12 ) {

    switch(err_inf) {
    case -3: /* There is not Input data */
        sprintf(errmsg, errstrings[err_no],
errstrings[18], sub_inf2);
        break;

    case -1: /* too many characters */
        sprintf(errmsg, errstrings[err_no],
errstrings[19], sub_inf2);
        break;

    case -2: /* Not all digits */
        sprintf(errmsg, errstrings[err_no],
errstrings[20], sub_inf2 );
        break;

    case -4: /* nothing sub message */
        sprintf(errmsg, errstrings[err_no], " ",
sub_inf2);
        break;

    default: /* Other error */
        sprintf(errmsg, errstrings[err_no],
errstrings[17], sub_inf2);
        break;
    }

    sprintf(buf, errhtml, errmsg, SOPATH,
user);
    // printf("%s", buf);
}
else{
    switch(err_inf) {
    case -3: /* There is not Input data */
        sprintf(errmsg, errstrings[err_no],
errstrings[18]);
        break;

    case -1: /* too many characters */
        sprintf(errmsg, errstrings[err_no],
errstrings[19]);
        break;

    case -2: /* Not all digits */
        if (err_no == 7)
            sprintf(errmsg, errstrings[err_no],
errstrings[22]);
        else
            sprintf(errmsg, errstrings[err_no],
errstrings[20]);

        break;

    case -4: /* nothing sub message */
        sprintf(errmsg, errstrings[err_no], " ");
        break;

    default: /* Other error */

```

```

        sprintf(errmsg, errstrings[err_no],
errstrings[17]);
        break;
    }

    sprintf(buf, errhtml, errmsg, SOPATH,
user);
    // printf("%s", buf);
}

// DBGR(sprintf(test_fp, "This Transaction is
parameter ERROR\n"));
return 0;
}

.....
tpapl/GetTerminalInfo.c
.....

/*****
****
*
* TPC-C Client Application Program Source
*
*
* Entry Functions
* (1) GetTerminalInfo
* (2) GetConfigFileInfo
*
* CREATE by TSL 2002.12.27
*
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2002 *
****
****/
#include "forlinux.h"
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>
#include <pthread.h>
#include <atmi.h>

#include "GlobalArea.h"
#include "log.h"
#include "log_level.h"

int GetPrivateProfileString(char* section_name,
char* key_name,
char* default_str, char*
key_data,
int buf_size, char* file_name);
int GetConfFileInfo_GetInt(char* section_name,
char* key_name);
int GetConfFileInfo_GetStr(char* section_name,
char* key_name, char* str);

/*****
****
* Get configuration file information.
*
* Return Value
* None
*
****
****/

void GetConfFileInfo() {

    /* Check INI file exist */

```

```

if (access(GLB_ConfigFilePath, 0x00) != 0) {
    /* INI file no exist, using default value */
    TpcUserLog(LOG_LCK, "INI file nothing,
using default value");
    GLB_TermBase =
DEFAULT_TERMBASE;
    GLB_Numwh =
DEFAULT_MAXWH;
    GLB_Maxconnect =
DEFAULT_MAXCONNECT;
    GLB_Maxterm =
DEFAULT_MAXTERM;
    GLB_C_FLAG =
DEFAULT_CFLAG;
    strcpy(GLB_TpAplLogPath,
DEFAULT_TPAPL_LOG_PATH);
    strcpy(GLB_SvrAplLogPath,
DEFAULT_SVRAPL_LOG_PATH);
    strcpy(GLB_LogFilePath,
DEFAULT_SVRAPL_LOG_PATH);
    return;
}
TpcUserLog(LOG_LCK, "INI file exist, using
spacified parameter\n");

/* Get execution informations
*/
/* If undefined key and illigal value, using
default value */
if ((GLB_TermBase =
GetConfFileInfo_GetInt("TPAPL_INFO",
"Term_Base")) <= 0) {
    GLB_TermBase = DEFAULT_TERMBASE;
}
if ((GLB_Numwh =
GetConfFileInfo_GetInt("TPAPL_INFO",
"NumWarehouses")) <= 0) {
    GLB_Numwh = DEFAULT_MAXWH;
}
if ((GLB_Maxconnect =
GetConfFileInfo_GetInt("TPAPL_INFO",
"MaxUsers")) <= 0) {
    GLB_Maxconnect =
DEFAULT_MAXCONNECT;
}
if ((GLB_Maxterm =
GetConfFileInfo_GetInt("TPAPL_INFO",
"MaxTerm of Client")) <= 0) {
    GLB_Maxterm = DEFAULT_MAXTERM;
}
if ((GLB_C_FLAG =
GetConfFileInfo_GetInt("TPAPL_INFO",
"CONTROL_Flag")) == -1) {
    GLB_C_FLAG = DEFAULT_CFLAG;
}
if (GetConfFileInfo_GetStr("TPAPL_INFO",
"LogPath", GLB_TpAplLogPath) != 0) {
    strcpy(GLB_TpAplLogPath,
DEFAULT_TPAPL_LOG_PATH);
}
if (GetConfFileInfo_GetStr("SVRAPL_INFO",
"LogPath", GLB_SvrAplLogPath) != 0) {
    strcpy(GLB_SvrAplLogPath,
DEFAULT_SVRAPL_LOG_PATH);
}

strcpy(GLB_LogFilePath,
GLB_TpAplLogPath);
}

/*-----*/
/* Get information in the CONFIG file for integer
value */
/*-----*/

```

```

int GetConfFileInfo_GetInt(char* section_name,
char* key_name) {

    char value_buf[64];
    int i;

    for (i = 0; i < 3; i++) {
        GetPrivateProfileString(section_name,
key_name, "",
                                value_buf, sizeof(value_buf),
GLB_ConfigFilePath);
        if (value_buf[0] == '*') {
            /* if Key is nothing, retry getting */
            continue;
        }
        break;
    }
#ifdef PUT_INF_LOG
    TpcUserLog(LOG_LCK, "CONFIG file
information [%s %s]=[%s]", section_name,
key_name, value_buf);
#endif
    if (value_buf[0] == '*') {
        /* Target key was nothing */
        return (-1);
    }
    return(atoi(value_buf));
}

/*-----*/
/* Get information in the CONFIG file for string
value */
/*-----*/
int GetConfFileInfo_GetStr(char* section_name,
char* key_name, char* str) {

    int i;
    char value_buf[1024];

    for (i = 0; i < 3; i++) {
        GetPrivateProfileString(section_name,
key_name, "",
                                value_buf, sizeof(value_buf),
GLB_ConfigFilePath);
        if (value_buf[0] == '*') {
            /* if Key is nothing, retry getting */
            continue;
        }
        break;
    }
#ifdef PUT_INF_LOG
    TpcUserLog(LOG_LCK, "CONFIG file
information [%s %s]=[%s]", section_name,
key_name, value_buf);
#endif
    if (value_buf[0] == '*') {
        /* Target key was nothing */
        return (-1);
    }
    strcpy(str, value_buf);
    return(strlen(value_buf));
}

.....
tpapl/GlobalArea.c
.....

/*-----*/
*****
*
* TPC-C Client Application Program Source
*
*
* Entry Functions
* Global Area definition for common.
*
*
* CREATE by TSL 2003.12.15
*
*
* Entry Functions

```

```

* Global Area definition for common.
*
*
* CREATE by TSL 2003.12.15
*
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2003 *

*****
****/
#include "forlinux.h"

#include <pthread.h>
#include <atmi.h>

#ifdef DBPRT /* for debug */
FILE *test_fp;
#endif

/* Environment of operation */
int GLB_TermBase;
int GLB_Numwh;
int GLB_Maxconnect;
int GLB_Maxterm;
int GLB_C_FLAG;
char GLB_TpAplLogPath[MAX_PATH];
char
GLB_SvrAplLogPath[MAX_PATH];

/* Configuration file path */
char
GLB_ConfigFilePath[MAX_PATH];

/* Thread key */
pthread_key_t GLB_ThreadKey;

/* Log information */
char GLB_LogFilePath[MAX_PATH];
int GLB_LogSemId;

/* TUXEDO context */
#if 0 /* * 2006.03.29 T.Motoo: Changed the type
of "GLB_TpContext". */
!TPCONTEXT_T GLB_TpContext = 0;
#endif
TPCONTEXT_T *GLB_TpContext =
NULL;

/*
* 2006.03.29 T.Motoo: Added.
*/
int GLB_ThreadLimit = 1;

.....
tpapl/GlobalArea.h
.....

/*-----*/
*****
*
* TPC-C Client Application Program Source
*
*
* Entry Functions
* Global Area definition for common.
*
*
* CREATE by TSL 2003.12.15
*
*
* Entry Functions

```

```

* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2003 *

*****
****/

#ifndef GLOBALAREA_H
#define GLOBALAREA_H

#ifdef DBPRT /* for debug */
extern FILE *test_fp;
#endif

extern int GLB_TermBase;
#define DEFAULT_TERMBASE 1
extern int GLB_Numwh;
#define DEFAULT_MAXWH 2000
extern int GLB_Maxconnect;
#define DEFAULT_MAXCONNECT
20000
extern int GLB_Maxterm;
#define DEFAULT_MAXTERM 2000
extern int GLB_C_FLAG;
#define DEFAULT_CFLAG 0
extern char
GLB_TpAplLogPath[MAX_PATH];
extern char
GLB_SvrAplLogPath[MAX_PATH];

/* Configuration file path */
extern char
GLB_ConfigFilePath[MAX_PATH];

/* Thread key */
extern pthread_key_t GLB_ThreadKey;

/* Log information */
extern char
GLB_LogFilePath[MAX_PATH];
extern int
GLB_LogSemId;

/* TUXEDO context */
#if 0 /* * 2006.03.29 T.Motoo: Changed the type
of "GLB_TpContext". */
!extern TPCONTEXT_T
GLB_TpContext;
#endif
extern TPCONTEXT_T* GLB_TpContext;

/*
* 2006.03.29 T.Motoo: "GLB_ThreadLimit" and
"TUXCDPERCTXT" were added.
*/
extern int
GLB_ThreadLimit;

/*
* Call descriptors per context (TUXEDO)
*/
#define TUXCDPERCTXT 50

#endif // GLOBALAREA_H

.....
tpapl/InitThreadEnv.c
.....

/*-----*/
*****
*
* TPC-C Client Application Program Source
*
*

```



```

*
* Entry Functions
* (1) GetThreadKey
* (2) CreateTuxEnv
* (3) DestroyThread
* (4) FreeThreadKey
* (5) GetThreadCntl
* (6) RegisTuxApl
* (7) TermChildProcess
* (7) PlainCleanup
*
* CREATE by TSL 2003.12.16
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2003
*
*****
****/
#include "forlinux.h"
#include <pthread.h>
#include <atmi.h>
#include <unistd.h>

#include "httpd.h"
#include "http_config.h"
#include "http_protocol.h"
#include "ap_config.h"
#include "ap_compat.h"
#include "ap_mpm.h" /* 2006.03.29 T.Motoo:
Added for ap_mpm_query */

#include "tpccinf.h"
#include "trans.h"
#include "ThreadCntl.h"
#include "GlobalArea.h"
#include "TpAplDBDependPrototype.h"
#include "log.h"
#include "log_level.h"

/*****
* Get thread key.
* Return Value
* 0 : Success
* !0 : Fail
*
*****
****/
int GetThreadKey() {

    int ret_code;
    void DestroyThread(void* p);

#ifdef PUT_INF_LOG
    TpcUserLog (LOG_INF, "Thread key
creating start [GetThreadKey]\n");
#endif

    /* Create the thread key */
    if ((ret_code =
pthread_key_create(&GLB_ThreadKey,
DestroyThread)) != 0) {
        TpcUserLog (LOG_ERR, "Thread key fail
to creat [error:%d]\n", ret_code);
        return -1;
    }

#ifdef PUT_INF_LOG
    TpcUserLog (LOG_INF, "Thread key
creating end [GetThreadKey= %d]\n",
GLB_ThreadKey);
#endif

    return 0;
}

```

```

}

/*****
* Initialize environment for Thread.
*
* Return Value
* !0 : Success(pointer of
THREAD_CNTL_INFO)
* 0 : Fail
*
*****
****/
/*
* 2006.03.29 T.Motoo: The argument was
added. "id" is ID of connection managed
* by apache. Unique at any point in
time.
*/
#if 0
!THREAD_CNTL_INFO* CreateThreadEnv() {
#endif
THREAD_CNTL_INFO* CreateThreadEnv(int id)
{

    THREAD_CNTL_INFO* ThreadCntlInfo;

    void* ift_buf;
    char* resp_buf;
    char* query_str;
    int buf_leng;

#define BUF_TYPE "CARRAY"

    if ((ThreadCntlInfo =
(THREAD_CNTL_INFO*)pthread_getspecific(GL
B_ThreadKey)) == NULL) {

#ifdef PUT_INF_LOG
        TpcUserLog (LOG_INF, "Thread initialize
started \n");
#endif

        /* First execution in this thread */
#ifdef SCRTEST
        /* Regist context */
        /*
* 2006.03.29 T.Motoo: Modified because child
process came to have one or more
* contexts.
*/
        #if 0
        ! if (tpsetctx(GLB_TpContext, 0) == -1) {
        #endif
            if (tpsetctx(GLB_TpContext{((id %
GLB_ThreadLimit) /
TUXCDPERCTXT), 0) ==
-1) {
                TpcUserLog (LOG_ERR, "tpsetctx()
failed\n");
                return(0);
            }
        #endif

        /* Get query data area */
#ifdef USEPOOL_QUERY
        if ((query_str =
(char*)malloc(QUERY_STR_SIZE)) == NULL) {
            TpcUserLog (LOG_ERR, "malloc() failed
for query string buffer (size=%d)\n",
QUERY_STR_SIZE);
            return(0);
        }
        #else
        query_str = NULL;

```

```

#endif

        /* Get response editing area */
        if ((resp_buf =
(char*)malloc(RESF_BUF_SIZE)) == NULL) {
            TpcUserLog (LOG_ERR, "malloc() failed
for response editing buffer (size=%d)\n",
RESP_BUF_SIZE);
            return(0);
        }

        /* Get Thread control information area */
        if ((ThreadCntlInfo =
(THREAD_CNTL_INFO*)malloc(sizeof(THREAD
_CNTL_INFO))) == NULL) {
            TpcUserLog (LOG_ERR, "malloc() failed
for THREAD_CNTL_INFO (size=%d)\n",
sizeof(THREAD_CNTL_INFO));
            return(0);
        }

        /* Get the TUXEDO interface data area */
        buf_leng = (GetGenericDataLen() + 16) &
0xfffffff;

#ifdef CONST_TUX_BUF

#ifdef SCRTEST
        if ((ift_buf = (void *)tpalloc("CARRAY",
NULL, buf_leng)) == NULL) {
            TpcUserLog (LOG_ERR, "tpalloc() failed
for interface data buffer (size=%d)\n", buf_leng);
            return(0);
        }
    #else
        if ((ift_buf = (void *)calloc (buf_leng, 1)) ==
NULL) {
            TpcUserLog (LOG_ERR, "calloc() failed
for interface data buffer (size=%d)\n", buf_leng);
            return(0);
        }
    #endif

    #else
        ift_buf = 0;
    #endif

    /* Set each pointer */
    ThreadCntlInfo->TrxDat = ift_buf;
    ThreadCntlInfo->TrxDatLeng = buf_leng;
    ThreadCntlInfo->QueryData = query_str;
    ThreadCntlInfo->RespBuf = resp_buf;

    /* Set thread data pointer */
    if (pthread_setspecific(GLB_ThreadKey,
(void*)ThreadCntlInfo) != 0) {
        TpcUserLog (LOG_ERR,
"pthread_setspecific() failed for
THREAD_CNTL_INFO setting \n");
#ifdef CONST_TUX_BUF

#ifdef SCRTEST
            tpfree(ift_buf);
        #else
            free(ift_buf);
        #endif

    #endif

        return(0);
    }

#ifdef PUT_INF_LOG
    TpcUserLog (LOG_INF, "Thread initialize
ended [thread key:%d]\n", GLB_ThreadKey);
#endif
}

```

```

return(ThreadCntlInfo);
}

/*****
* Destroy thread, then free allocate area.
*
* Return Value
* NONE
*****/

void DestroyThread(void* p) {

    THREAD_CNTL_INFO* ThreadCntlInfo;

#ifdef PUT_INF_LOG
    TpcUserLog(LOG_INF, "Thread
terminated start\n");
#endif

    if (p != NULL) {
        ThreadCntlInfo =
(THREAD_CNTL_INFO*)p;

        if (ThreadCntlInfo->TrxDData != 0)
#ifdef SCRTEST
            tpfree(ThreadCntlInfo->TrxDData);
#else
            free(ThreadCntlInfo->TrxDData);
#endif

#ifdef USEPOOL_QUERY
        if (ThreadCntlInfo->QueryData != 0)
            free((void*)ThreadCntlInfo->QueryData);
#endif

        if (ThreadCntlInfo->RespBuf != 0)
            free((void*)ThreadCntlInfo->RespBuf);
        free((void*)ThreadCntlInfo);
        ThreadCntlInfo = 0;
        if (pthread_setspecific(GLB_ThreadKey,
(void*)ThreadCntlInfo) != 0) {
            TpcUserLog(LOG_ERR,
"pthread_setspecific() failed for Thread
destroyed\n");
        }
    }

#ifdef PUT_INF_LOG
    TpcUserLog(LOG_INF, "Thread terminate
ended [TSD value:%08x]\n", (unsigned long)p);
    return;
#endif
}

/*****
* Free thread key.
* Return Value
* NONE
*****/

void FreeThreadKey() {
    int ret_code;

    if ((ret_code =
pthread_key_delete(GLB_ThreadKey)) != 0) {
        TpcUserLog(LOG_ERR,
"pthread_key_delete() failed [ret_code=%d]\n",
ret_code);
    }
}

```

```

/*****
****
* Get Thread_CNTL_INFO pointer in my thread.
*
* Return Value
* !0 : Success(pointer of
THREAD_CNTL_INFO)
* 0 : Fail
*****/

THREAD_CNTL_INFO* GetThreadCntl() {
    THREAD_CNTL_INFO* ThreadCntlInfo;

    if ((ThreadCntlInfo =
(THREAD_CNTL_INFO*)pthread_getspecific(GL
B_ThreadKey)) == NULL) {
        TpcUserLog(LOG_ERR, "Thread cntrol
information is not allocated.\n");
        return 0;
    }

#ifdef CONST_TUX_BUF
    /* Nothing to do */

#else
#ifdef SCRTEST
        if (ThreadCntlInfo->TrxDData = (char
*)tpalloc("CARRAY", NULL, ThreadCntlInfo-
>TrxDDataLeng)) == NULL ) {
            TpcUserLog(LOG_ERR, "tpalloc() failed
for interface data buffer (size=%d)\n",
ThreadCntlInfo->TrxDDataLeng);
            return(0);
        }
    #else
        if (ThreadCntlInfo->TrxDData = (char
*)calloc( ThreadCntlInfo->TrxDDataLeng, 1)) ==
NULL ) {
            TpcUserLog(LOG_ERR, "calloc() failed for
interface data buffer (size=%d)\n",
ThreadCntlInfo->TrxDDataLeng);
            return(0);
        }
    #endif
#endif

    return ThreadCntlInfo;
}

/*****
****
* Free TUXEDO interface buffer.
*
* Return Value
* NONE
*****/

void FreeTuxBuffer(THREAD_CNTL_INFO*
ThreadCntlInfo) {

#ifdef CONST_TUX_BUF
    /* No free buffer */
#else
    if (ThreadCntlInfo->TrxDData != 0) {
#ifdef SCRTEST
        tpfree(ThreadCntlInfo->TrxDData);
    #else
        free(ThreadCntlInfo->TrxDData);
    #endif
    #endif
        ThreadCntlInfo->TrxDData = 0;
    }
}

#endif

```

```

return;
}

/*****
****
* Regist TUXEDO aplication.
*
* Return Value
* !0 : Success
* 0 : Fail
*****/

/*
* 2006.03.29 T.Motoo: The argument was
added. "p" is pool of apache. Moreover,
* some variables were added.
*/
#if 0
ITPCONTEXT_T RegistTuxApl() {
    !
    ! TPCONTEXT_T ctx = 0;
    #endif
    TPCONTEXT_T *RegistTuxApl(void *p) {

        TPCONTEXT_T *ctx = NULL; /* Contexts
*/
        int num_of_ctx = 0; /* Contexts per
child */
        int thr_per_child = 0; /* Threads per child
*/
        int i; /* Uses as counter */

        static TPINIT *tpinf = 0;

        if (tpinf == 0) {
            /* Get Initialize information area for tpinit() */
            if ((tpinf = (TPINIT *)tpalloc("TPINIT", NULL,
sizeof(TPINIT))) == NULL) {
                TpcUserLog(LOG_ERR, "tpalloc failed
for tpinit() (%s)\n", tpstrerror(tperrno));
                return 0;
            }
        }

        /* Execute tpinit() (Regist TUXEDO
aplication)*/
        memset((void*)tpinf, 0x00, sizeof(TPINIT));
        tpinf->flags=TPMULTICONTEXTS;

        #if 0 /* 2006.03.29 T.Motoo: Changed to get one
or more contexts. */
            ! if (tpinit(tpinf) < 0) {
                ! /* tpinit() abnormal end */
                ! TpcUserLog(LOG_ERR, "tpinit() faild
(%s)\n", tpstrerror(tperrno));
                ! return 0;
                ! }
            ! }
            !
            ! /* Get my context */
            ! if (tpgetctx(&ctx, 0) == -1) {
                ! TpcUserLog(LOG_ERR, "Failed to get
Tuxedo context (%s)\n", tpstrerror(tperrno));
                ! return 0;
                ! }
            #endif
        /*
        * Gets "ThreadsPerChild" and
        "ThreadLimit".
        */

        ap_mpm_query(AP_MPMO_MAX_THREADS,
&thr_per_child);

```

```
ap_mpm_query(AP_MPMQ_HARD_LIMIT_THR
EADS, &GLB_ThreadLimit);
```

```
/*
 * Gets the number of contexts.
 */
num_of_ctx = ((thr_per_child - 1) /
TUXCDPERCTXT) + 1;

/*
 * Allocates the memory for contexts in the
pool.
 */
ctx = (TPCONTEXT_T
*)ap_palloc((apr_pool_t*)p,
sizeof(TPCONTEXT_T)
* num_of_ctx);

if (ctx == NULL) {
    TpcUserLog(LOG_ERR, "ap_palloc
failed for contexts\n");
    return 0;
}

for (i = 0; i < num_of_ctx; i++) {
    /*
     * Joins the TUXEDO.
     */
    if (tpinit(tpinf) < 0) {
        /* tpinit() abnormal end */
        TpcUserLog(LOG_ERR, "tpinit() failed
(%)s\n",
                tpstrerror(tperrno));
        return 0;
    }

    /*
     * Gets the context.
     */
    if (tpgetctx((ctx + i), 0) == -1) {
        TpcUserLog(LOG_ERR, "Failed to
get Tuxedo context (%)s\n",
                tpstrerror(tperrno));
        return 0;
    }
}

return ctx;
}

/*****
 * Termmnate child process.
 *
 * Return Value
 * Always SUCCESS
 */
/*****
apr_status_t TermChildProcess(void* p) {

#ifdef PUT_INF_LOG
    TpcUserLog(LOG_INF, "Child process
terminated start. \n");
#endif

    /* Leave from TUXEDO aplication */
    if (GLB_TpContext != 0) {
        if (tpterm() == -1) {
            TpcUserLog(LOG_ERR, "tpterm() failed
for Thread destroyed\n");
        }
        GLB_TpContext = 0;
    }
}
```

```
/* Delete TSD key */
FreeThreadKey();

#ifdef PUT_INF_LOG
    TpcUserLog(LOG_INF, "Child process
terminated end. \n");
#endif

    return(APR_SUCCESS);
}

/*****
 * Plain cleanup.
 * Return Value
 * Always SUCCESS
 */
/*****/
apr_status_t PlainCleanup(void* p) {

    /* Notheng to do */
    return(APR_SUCCESS);
}

:-----:
tpapl/MakeShell_lib
:-----:

#!/bin/sh
cd /home/tpc/client_apl/tpapl
make -f Makefile_lib > make_result.txt 2>&1

:-----:
tpapl/MakeShell_tpapl
:-----:

#!/bin/sh

# Output object from library
cd /home/tpc/client_apl/tpapl/trnexe
echo "==" > ./make_result.txt
echo "=====" Output object"
>> ./make_result.txt
rm *.o >> ./make_result.txt
ar -xv libtrnexe_$1.a ConvTime.o
>> ./make_result.txt
ar -xv libtrnexe_$1.a CreateTranErrReason.o
>> ./make_result.txt
ar -xv libtrnexe_$1.a TestFunction.o
>> ./make_result.txt
ar -xv libtrnexe_$1.a TransactionDataLen.o
>> ./make_result.txt
ar -xv libtrnexe_$1.a TrxDelivery.o
>> ./make_result.txt
ar -xv libtrnexe_$1.a TrxNewOrder.o
>> ./make_result.txt
ar -xv libtrnexe_$1.a TrxOrderStatus.o
>> ./make_result.txt
ar -xv libtrnexe_$1.a TrxPayment.o
>> ./make_result.txt
ar -xv libtrnexe_$1.a TrxStockLevel.o
>> ./make_result.txt

# Make library
cd /home/tpc/client_apl/tpapl
echo "==" >> make_result.txt
echo "=====" Remake library" >>
make_result.txt
```

```
ar -rv libtpapl.a ./trnexe/ConvTime.o
>> make_result.txt
ar -rv libtpapl.a ./trnexe/CreateTranErrReason.o
>> make_result.txt
ar -rv libtpapl.a ./trnexe/TestFunction.o
>> make_result.txt
ar -rv libtpapl.a ./trnexe/TransactionDataLen.o
>> make_result.txt
ar -rv libtpapl.a ./trnexe/TrxDelivery.o
>> make_result.txt
ar -rv libtpapl.a ./trnexe/TrxNewOrder.o
>> make_result.txt
ar -rv libtpapl.a ./trnexe/TrxOrderStatus.o
>> make_result.txt
ar -rv libtpapl.a ./trnexe/TrxPayment.o
>> make_result.txt
ar -rv libtpapl.a ./trnexe/TrxStockLevel.o
>> make_result.txt

ar -rv libtpapl.a ../common/log.o
>> make_result.txt
ar -rv libtpapl.a ../common/sema.o
>> make_result.txt
ar -rv libtpapl.a ../common/shmem.o
>> make_result.txt
ar -rv
libtpapl.a ../common/GetPrivateProfileString.o
>> make_result.txt

# Make TPAPL
echo "==" >> make_result.txt
echo "=====" Make " >>
make_result.txt
touch mod_tpapl.c >> make_result.txt
make -f Makefile_tpapl >>
make_result.txt 2>&1
## Not install ##make -f Makefile_tpapl install
>> make_result.txt 2>&1

# Check undefined symbol
echo "==" >> make_result.txt
echo "=====" mod_tpapl.so information "====="
>> make_result.txt
ldd -r ./libs/mod_tpapl.so >> make_result.txt
2>&1

:-----:
tpapl/Makefile
:-----:

##
## Makefile -- Build procedure for sample tpapl
Apache module
## Autogenerated via ``apxs -n tpapl -g``.
##

builddir=.
top_srcdir=/etc/httpd
top_builddir=/etc/httpd
include /usr/lib/httpd/build/special.mk

# the used tools
APXS=apxs
APACHECTL=apachectl

# additional defines, includes and libraries
#DEFS=-Dmy_define=my_value
#INCLUDES=-Imy/include/dir
#LIBS=-Lmy/lib/dir -lmylib

# the default target
all: local-shared-build
```

```

# install the shared object file into Apache
install: install-modules

# cleanup
clean:
  -rm -f mod_tpapl.o mod_tpapl.lo
  mod_tpapl.slo mod_tpapl.la

# simple test
test: reload
  lynx -mime_header http://localhost/tpapl

# install and activate shared object by reloading
Apache to
# force a reload of the shared object file
reload: install restart

# the general Apache start/restart/stop
# procedures
start:
  $(APACHECTL) start
restart:
  $(APACHECTL) restart
stop:
  $(APACHECTL) stop

.....
tpapl/Makefile_lib
.....

#-----
-----
# Makefile : Makefile for TpApl library on Linux.
#
# Created by TSL 2003.12.22
#
# All Right Rserverd, Copyright Co, FUJITSU
LIMITED 2003-2004.
#-----
-----

# GCC compile configurations
AR = ar
ARFLAGS = rv

#CFLAGS note:
# CONST_TUX_BUF defined : TUXEDO
interface buffer is created when thread initialize.
# CONST_TUX_BUF undefined : TUXEDO
interface buffer is created when transaction
processing start,
# and freed when transaction
procesing end.
# USEPOOL_QUERY define : Use query data
area in apache pool.
# USEPOOL_QUERY undefined : Allocate the
query data area, and copied query data form
apache pool.
#CFLAGS = -Wall
#CFLAGS = -Wall -DCONST_TUX_BUF
CFLAGS = -Wall -O2 -DCONST_TUX_BUF -
DUSEPOOL_QUERY
CC = gcc

# Define macros
DMACRO =

# home directory.
TOPDIR = /home/tpc/client_apl
TUXDIR = /usr/local/BEA/tuxedo8.1
APADIR = /usr/include/httpd
APAODIR = /usr/include/apr-0
APLDIR = $(TOPDIR)/tpapl

```

```

# include directory
COM_INC = -I$(TOPDIR)/common
TUX_INC = -I$(TUXDIR)/include
APA_INC = -I$(APADIR)
APA0_INC = -I$(APAODIR)
APL_INC = -I$(APLDIR)

# header file directory
HDFDIR = $(APLDIR)
COMDIR = $(TOPDIR)/common

INCLUDE = $(APL_INC) $(COM_INC) -
I$(APAODIR) $(APA_INC) $(TUX_INC)
INCFILE = $(APLDIR)/delpage.h \
  $(APLDIR)/GlobalArea.h \
  $(APLDIR)/log_level.h \
  $(APLDIR)/menupage.h \
  $(APLDIR)/newpage.h \
  $(APLDIR)/odrpge.h \
  $(APLDIR)/paypage.h \
  $(APLDIR)/stpage.h \
  $(APLDIR)/ThreadCntl.h \
  $(APLDIR)/tpapl.h \
  $(APLDIR)/TpAplDBDependPrototype.h \
  $(APLDIR)/TpAplPrototype.h \
  $(APLDIR)/tpccinf.h \
  $(APLDIR)/tpcinweb.h \
  $(APLDIR)/tpcweb.h \
  $(APLDIR)/trans.h \
  $(APLDIR)/SampleInfo.h \
  $(COMDIR)/log.h \
  $(COMDIR)/sema.h

# target object
OBS = TpAplHandler.o ClientMonitor.o
ConvInt.o ConvOther.o ConvString.o \
  ErrPage.o GetTerminalInfo.o GlobalArea.o
InitThreadEnv.o tpaplFunction.o
ARCH_LIB = $(APLDIR)/libtpapl.a

$(ARCH_LIB) : $(OBS) $(INCFILE)
  $(AR) $(ARFLAGS) $(ARCH_LIB) $(OBS)

.SUFFIXES: .o .c
.c.o:
  $(CC) -o $@ -c $(CFLAGS) $(INCLUDE)
$(DMACRO) $<

$(OBS) : $(INCFILE)

clean:
  rm $(TIER_ARCH_LIB) $(TIER_OBJS)

.....
tpapl/Makefile_tpapl
.....

#-----
-----
# Makefile : Makefile for TpApl library on Linux.
#
# Created by TSL 2003.12.18
#
# All Right Rserverd, Copyright Co, FUJITSU
LIMITED 2003-2004.
#-----
-----

builddir=.

```

```

top_srcdir=/etc/httpd
top_builddir=/etc/httpd
include /usr/lib/httpd/build/special.mk

# the used tools
APXS=apxs
APACHECTL=apachectl

# additional defines, includes and libraries
#DEFS=-Dmy_define=my_value
#INCLUDES=-Imy/include/dir
#LIBS=-Lmy/lib/dir -lmylib

TPAHOME = /home/tpc/client_apl
TUXHOME = /usr/local/BEA/tuxedo8.1

#LIBS=-L$(TPAHOME)/tpapl -L$(TUXHOME)/lib
\
# -ltpapl \
# -ltux -lbuft -lfml -lfml32 -lengine \
# -ldl -lpthread

LIBS=-L$(TPAHOME)/tpapl -L$(TUXHOME)/lib \
-ltpapl \
-ltux

# the default target
all: local-shared-build

# install the shared object file into Apache
install: install-modules

# cleanup
clean:
  -rm -f mod_tpapl.o mod_tpapl.lo
  mod_tpapl.slo mod_tpapl.la

# simple test
test: reload
  lynx -mime_header http://localhost/tpapl

# install and activate shared object by reloading
Apache to
# force a reload of the shared object file
reload: install restart

# the general Apache start/restart/stop
# procedures
start:
  $(APACHECTL) start
restart:
  $(APACHECTL) restart
stop:
  $(APACHECTL) stop

.....
tpapl/SampleInfo.h
.....

/*****
*
* TPC-C Client Application Program Source
*
*
* Entry Functions
* Performance information definition
*
* CREATE by TSL 2004.01.18
*
*
*****/

```

```

* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2004 *

*****
*/
/* Performans sampling faunctions */
int ClientMonitor(int func_no, char*
html_buf);
void ClientLogCheck(char* html_buf);
void ClientSetSample(char* html_buf);
void ClientInfSample(char* html_buf);
void ClientSampleInit();
void ClientSampleSelfCsv(time_t cur_sec);

/* Structure of performance sampling area */
typedef struct _sampling_data {
// Number of DB server executed transactions
unsigned int NumNewOrder;
unsigned int NumPayment;
unsigned int NumOrderStatus;
unsigned int NumDelivery;
unsigned int NumStockLevel;

// Response time (ms) from DB server (total
time in sampling interval)
unsigned int RspTimeNewOrder;
unsigned int RspTimePayment;
unsigned int RspTimeOrderStatus;
unsigned int RspTimeDelivery;
unsigned int RspTimeStockLevel;

// Max response time (ms) from DB server
(total time in sampling interval)
unsigned int SMaxRspTimeNewOrder;
unsigned int SMaxRspTimePayment;
unsigned int SMaxRspTimeOrderStatus;
unsigned int SMaxRspTimeDelivery;
unsigned int SMaxRspTimeStockLevel;

// Number of request from RTE
unsigned int NumReqNewOrder;
unsigned int NumReqPayment;
unsigned int NumReqOrderStatus;
unsigned int NumReqDelivery;
unsigned int NumReqStockLevel;

// Answer time (ms) to RTE (total time in
sampling interval)
unsigned int AnsNewOrder;
unsigned int AnsPayment;
unsigned int AnsOrderStatus;
unsigned int AnsDelivery;
unsigned int AnsStockLevel;

// NOTE : Under the members are not cleared
by sampling interval.
// Max response time (ms) from DB server (all
of sampling time)
unsigned int MaxRspTimeNewOrder;
unsigned int MaxRspTimePayment;
unsigned int MaxRspTimeOrderStatus;
unsigned int MaxRspTimeDelivery;
unsigned int MaxRspTimeStockLevel;

// Number of executing and waiting
transactions
unsigned int NumQueNewOrder;
unsigned int NumQuePayment;
unsigned int NumQueOrderStatus;
unsigned int NumQueDelivery;
unsigned int NumQueStockLevel;

// Self sampling information
char CsvFilePath[MAX_PATH];
unsigned int CsvOutTime;

```

```

unsigned int SamplingInterval;
int SelfSamplingOutput;
#define SELFOUTPUT_ENABLE 1
#define SELFOUTPUT_DISABLE 0
int DataSampling;
#define DATASAMPLE_ENABLE 0
#define DATASAMPLE_DISABLE 1

// wait timer for 2tier.
unsigned int WaitTimer;

} SAMPLING_DATA;

/* ===== */
/* Macros */
/* ===== */
/* Path */
#define SAMPLING_SEMPATH
"/home/tpc/conf"
#define SAMPLING_SHMPATH
"/home/tpc/bin"

/* Sampling informaion */
#define MAC_SampleGlobalArea \
int GLBSMP_shared_mem = 0; \
SAMPLING_DATA* \
GLBSMP_shared_mem = 0;

extern int GLBSMP_shared_mem;
extern SAMPLING_DATA* \
GLBSMP_shared_mem;

/* Initialize semafore and shared memory */
#define MAC_SampleInitParent \
GLBSMP_shared_mem = \
InitSem(SAMPLING_SEMPATH, \
SEM_SAMPLING_PERFORMANCE); \
GLBSMP_shared_mem = \
(SAMPLING_DATA*)InitShmem(SAMPLING_SH \
MPATH, \
SHMEM_SAMPLING_PERFORMANCE, \
sizeof(SAMPLING_DATA)); \
memset(GLBSMP_shared_mem, 0x00, \
sizeof(SAMPLING_DATA));

#define MAC_SampleInitChild \
GLBSMP_shared_mem = \
GetSem(SAMPLING_SEMPATH, \
SEM_SAMPLING_PERFORMANCE); \
GLBSMP_shared_mem = \
(SAMPLING_DATA*)GetShmem(SAMPLING_S \
HMPATH, \
SHMEM_SAMPLING_PERFORMANCE, \
sizeof(SAMPLING_DATA));

#define MAC_SampleInitPerformance \
ClientSampleInit();

/* Fancions work area */
#define MAC_SampleWork \
struct timeval \
sample_start_time; \
struct timeval \
sample_end_time; \
unsigned int el_time;

/* Get start time */
#define MAC_SampleStartTime \
if (GLBSMP_shared_mem- \
>DataSampling == DATASAMPLE_ENABLE) { \
gettimeofday(&sample_start_time, \
NULL); \
}

```

```

/*sleep(10);*/ \
}

/* Transaction queue up/down */
#define MAC_SampleQueueUp(count_area) \
if (GLBSMP_shared_mem- \
>DataSampling == DATASAMPLE_ENABLE) { \
LockSem(GLBSMP_shared_mem); \
GLBSMP_shared_mem- \
>count_area++; \
UnlockSem(GLBSMP_shared_mem); \
/*sleep(10);*/ \
}

#define \
MAC_SampleQueueDown(count_area) \
if (GLBSMP_shared_mem- \
>DataSampling == DATASAMPLE_ENABLE) { \
LockSem(GLBSMP_shared_mem); \
GLBSMP_shared_mem->count_area-- \
; \
UnlockSem(GLBSMP_shared_mem); \
}

/* Compute execution time */
#define MAC_SampleExecuteTime \
if (GLBSMP_shared_mem- \
>DataSampling == DATASAMPLE_ENABLE) { \
gettimeofday(&sample_end_time, \
NULL); \
el_time = ((unsigned \
int)sample_end_time.tv_sec*1000 + (unsigned \
int)sample_end_time.tv_usec/1000) \
- ((unsigned \
int)sample_start_time.tv_sec*1000 + (unsigned \
int)sample_start_time.tv_usec/1000); \
}

/* SvrApl sampling sequence
* (1) MAC_SampleWork
* (2) MAC_SampleStartTime
* (3) Processing transaction on DB server
* (4) Except Delivery MAC_SampleDBSrvResp
* Only Delivery
MAC_SampleDBSrvRespDel
*/
#define \
MAC_SampleRespMax(max_resp_time, \
smp_max_resp_time) \
if (GLBSMP_shared_mem- \
>max_resp_time < el_time) \
GLBSMP_shared_mem- \
>max_resp_time = el_time; \
if (GLBSMP_shared_mem- \
>smp_max_resp_time < el_time) \
GLBSMP_shared_mem- \
>smp_max_resp_time = el_time;

/* For except Delivery */
#define MAC_SampleDBSrvResp(resp_time, \
max_resp_time, smp_max_resp_time, \
proc_trans) \
if (GLBSMP_shared_mem- \
>DataSampling == DATASAMPLE_ENABLE) { \
MAC_SampleExecuteTime; \
LockSem(GLBSMP_shared_mem); \
GLBSMP_shared_mem->resp_time \
+= el_time; \
MAC_SampleRespMax(max_resp_time, \
smp_max_resp_time); \
GLBSMP_shared_mem- \
>proc_trans++; \
UnlockSem(GLBSMP_shared_mem); \
}

```

```

    }

/* For only Delivery */
#define MAC_SampleDBSrvRespDel() \
    if (GLBSMP_shared_mem-
>DataSampling == DATASAMPLE_ENABLE) { \
    MAC_SampleExecuteTime; \
    LockSem(GLBSMP_semid); \
    GLBSMP_shared_mem-
>RspTimeDelivery += el_time; \

MAC_SampleRespMax(MaxRspTimeDelivery,
SMaxRspTimeDelivery); \
    GLBSMP_shared_mem-
>NumDelivery++; \
    GLBSMP_shared_mem-
>NumQueDelivery--; \
    UnlockSem(GLBSMP_semid); \
}

/* TpApl sampling sequence for except Delivery
* (1) MAC_SampleWork
* (2) MAC_SampleStartTime
* (3) MAC_SampleQueueUp
* (4) Processing transaction on TUXEDO and
DB server
* (5) Except Delivery MAC_SampleTuxResp
* Only Delivery MAC_SampleTuxRespDel
*/
/* For except Delivery */
#define MAC_SampleTuxResp(ans_time,
proc_trans, trans_que) \
    if (GLBSMP_shared_mem-
>DataSampling == DATASAMPLE_ENABLE) { \
    MAC_SampleExecuteTime; \
    LockSem(GLBSMP_semid); \
    GLBSMP_shared_mem->ans_time +=
el_time; \
    GLBSMP_shared_mem-
>proc_trans++; \
    GLBSMP_shared_mem->trans_que--;
\
    UnlockSem(GLBSMP_semid); \
}

/* For only Delivery */
#define MAC_SampleTuxRespDel \
    if (GLBSMP_shared_mem-
>DataSampling == DATASAMPLE_ENABLE) { \
    MAC_SampleExecuteTime; \
    LockSem(GLBSMP_semid); \
    GLBSMP_shared_mem->AnsDelivery
+= el_time; \
    GLBSMP_shared_mem-
>NumReqDelivery++; \
    UnlockSem(GLBSMP_semid); \
}

/*
* Output Self pafrotmance log
*/
#define MAC_SampleOutPutCsvLog \

ClientSampleSelfCsv(sample_end_time.tv_sec)

.....
tpapl/ThreadCntl.h
.....

/*****
*
* TPC-C Client Application Program Source
*
*****/

```

```

* Entry Functions
* Function definition for TUXEDO control
information.
*
* CREATE by TSL 2003.12.26
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2003
*

*****
*****/

typedef struct _THREAD_CNTL_INFO {
    void* TrxDData;
    int TrxDDataLeng;
    char* QueryData;
    char* RespBuf;
} THREAD_CNTL_INFO;

.....
tpapl/TpAplDBDependPrototype.h
.....

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

* TPC-C Client Application Program Source
*
*
* Entry Functions
* Function definition for common.
*
* CREATE by TSL 2002.10.01
*
* GHANGE by TSL 2003.12.15 for COM+ -->
TUXEDO
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2002
*

*****
*****/

int str2int(char *str, int field_len);
short str2short(char *str, int field_len);
int str2dbl(char *str, int field_len);

void int2str(char *str, int len, int num);
void int3str(char *str, int len, int num);
void dec2str(char *str, int len, double num);
void sigdec2str(char *str, int len, double num);
int str2str(char *str, int field_len);
void alp2str(char *str, int len, char *alp);
void date2str(char *str, char *time);
void zip2str(char *str, char *zip);
void phone2str(char *str, char *phone);

char* para_split(char *para, char delimita);
int checkHTMLform(char *str, char *buffer);
void convert_time(char *save_p, double
time);
void convert_date(char *save_p, double
time);

void time2str(char *str, char *time);

int set_errHTML(char *page, char *err_inf,
int cookie, char *errname);
/*
!int set_oraerr(char *page, char *err_inf, int
cookie);

```

```

*/
/* Replaced 03.01.15 */
#if 0
!int set_tuxerr(char *page, char *err_inf, int
cookie);
#endif
int set_SvrAplErr(char *page, char *err_inf,
int cookie);
/* Replaced end */
int set_errpage(char *buf, int user, int
err_no, int err_inf, int sub_inf2);

int NewOrder(char *s_buf,
RTE_INPUT_DATA *in_data, int cookie);
int Delivery(char *s_buf,
RTE_INPUT_DATA *in_data, int cookie);
int Payment(char *s_buf,
RTE_INPUT_DATA *in_data, int cookie);
int StockLevel(char *s_buf,
RTE_INPUT_DATA *in_data, int cookie);
int OrderStatus(char *s_buf,
RTE_INPUT_DATA *in_data, int cookie);
long GetGenericDataLen();

THREAD_CNTL_INFO* GetThreadCntl();
void FreeTuxBuffer(THREAD_CNTL_INFO*
ThreadCntlInfo);

.....
tpapl/TpAplHandler.c
.....

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

* TPC-C Client Application Program Source
*
*
* Entry Functions
* (1) TpAplHandler
* (2) OutputResultForm
* (3) GetConfigInfo
* (4) InitNewChildCreate
* (5) CreateTpAplSvrConf
*
* CREATE by TSL 2003.12.17
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2003
*

*****
*****/
#include "forlinux.h"
#include <sys/types.h>
#include <unistd.h>
#include <atmi.h>

#include "stdio.h"
#include "htpd.h"
#include "http_config.h"
#include "http_protocol.h"
#include "ap_config.h"
#include "ap_compat.h"

#include "trans.h"
#include "ThreadCntl.h"
#include "GlobalArea.h"
#include "TpAplPrototype.h"
#include "log_level.h"
#include "log.h"
#include "menupage.h"
#include "sema.h"
#include "shmem.h"

```

```

#include "SampleInfo.h"

/*****
* TpApl HTTP processing handler
*
* Return Value
* OK      : Normal end
* DECLINED : Abnormal end
*
*****/
int TpAplHandler(request_rec *r)
{
    int cookie = -1;
    int rtn;
    char* S_BUF;

    RTE_INPUT_DATA in_data_area;
    THREAD_CNTL_INFO* ThreadCntlInfo;

    void OutputResultForm(request_rec *r, char*
buf_body);

    /* Check handler executing conditions */
    if (strcmp(r->handler, "tpapl") {
        return DECLINED;
    }

#ifdef PUT_INF_LOG
    TpcclUserLog (LOG_INF, "#####
TpAplHandler start #####\n");
#endif

    if (r->header_only) {
        /* Request is header only */
        TpcclUserLog (LOG_WRN, "Request is http
header only.\n");
        r->content_type = "text/html";
        goto OK_RETURN;
    }

    /* Initialize thread environment */
    #if 0 /* 2006.03.29 T.Motoo: Modified because
the argument had been changed. */
    ! ThreadCntlInfo = CreateThreadEnv();
    #endif
    ThreadCntlInfo = CreateThreadEnv(r-
>connection->id);

    if (ThreadCntlInfo == 0) {
        TpcclUserLog (LOG_ERR, "Can't
Initialize\n");
        /* Initialization failure */
        OutputResultForm(r, initerr);
        goto OK_RETURN;
    }
    S_BUF = (char*)ThreadCntlInfo->RespBuf;

    /* Get Query string in to own area & analyze
requested data */
    #ifndef USEPOOL_QUERY
    strcpy(ThreadCntlInfo->QueryData, r->args);
    #else
    ThreadCntlInfo->QueryData = r->args;
    #endif

#ifdef PUT_INF_LOG
    TpcclUserLog (LOG_INF, "Recieved request
[%dbytes][%s]\n",
                strlen(ThreadCntlInfo-
>QueryData), ThreadCntlInfo->QueryData);
#endif

    memset(&in_data_area, 0x00,
sizeof(in_data_area));
    cookie = anly_para ((char *)ThreadCntlInfo-
>QueryData, &in_data_area );

    /* Terminal Number Check
* If terminal number is not valid then send
error message.
*/
    if ( cookie < GLB_TermBase || cookie >=
(GLB_TermBase + GLB_Maxterm) ){

        if (ClientMonitor(cookie, S_BUF) == 0) {
            if (cookie != -3) /* -3:reuest od
performance sampling */
                TpcclUserLog (LOG_INF, "Extended
function executing [function number:%d]\n",
cookie);
        }
        else {
            sprintf (S_BUF, badterm,
GLB_TermBase, GLB_TermBase +
GLB_Maxterm - 1, cookie);
            TpcclUserLog (LOG_ERR, "Terminal
number over the range[Terminal number:%d]\n",
cookie);
        }

        OutputResultForm(r, S_BUF);
        goto OK_RETURN;
    }

    /* Execute the taransaction data */
    rtn = select_trn ( &in_data_area, S_BUF,
cookie );

    /* Response output form */
    OutputResultForm(r, S_BUF);

OK_RETURN:
#ifdef PUT_INF_LOG
    TpcclUserLog (LOG_INF, "=====  

TpAplHandler end =====\n");
#endif
    return OK;
}

/*****
* Output Processing result form.
*
* Argument
* buf_body :
* Output message on screen
*
* Return Value
* NONE
*
*****/
void OutputResultForm(request_rec *r, char*
buf_body) {
    //int len=strlen(buf_body);

    r->content_type = "text/html";
    // ap_send_http_header(r);
    ap_rputs(buf_body, r);
    //buf_body[100]=0;
    //TpcclUserLog (LOG_INF, "Content len=%d
data=(%s)\n", len,buf_body);
    return;
}

}

/*****
* Get configuration information
*
* Return Value
* char* NULL : allways
*
*****/
module tpapl_module;

char* GetConfigInfo(cmd_parms* parms, void*
mconfig, char* path) {
    char work_path[MAX_PATH];
    int i;
    char *conf;

    /* Set default log path */
    strcpy(GLB_TpAplLogPath,
DEFAULT_TPAPL_LOG_PATH);
    strcpy(GLB_LogFilePath,
DEFAULT_TPAPL_LOG_PATH);
    TpcclUserLog (LOG_LCK, "Directive
processing start [GetConfigInfo]\n");

    /* Get configuration informaion (set to global
area) */
    strcpy(GLB_ConfigFilePath, path);
    GetConfigFile();

    /* Initialize TPAPL semaphore for log */
    strcpy(work_path, GLB_TpAplLogPath);
    for(i = strlen(work_path) - 1; i > 0 &&
work_path[i] != '/' ; i--);
    work_path[i] = '\0';

    if ((GLB_LogSemId = InitSem(work_path,
SEM_TPAPL_PROJID)) == -1) {
        TpcclUserLog (LOG_LCK, "InitSem() faille for
TpApl log\n");
        return NULL;
    }

    /* Initialize SVRAPL semaphore for log */
    strcpy(work_path, GLB_SvrAplLogPath);
    for(i = strlen(work_path) - 1; i > 0 &&
work_path[i] != '/' ; i--);
    work_path[i] = '\0';

    if (InitSem(work_path,
SEM_SVRAPL_PROJID) == -1) {
        TpcclUserLog (LOG_LCK, "InitSem() faille for
SvrApl log\n");
        return NULL;
    }

    /* Set server configuration */
    conf = (char*)ap_get_module_config(parms-
>server->module_config, &tpapl_module);
    strcpy(conf, path);

    /* Initialize client performance monitor */
    MAC_SampleInitPerformance;

    TpcclUserLog (LOG_INF, "Directive processing
ended [GetConfigInfo]\n");
    return NULL;
}

/*****
* Initialize child process creates.
*
*****/

```



```

Date: - - : : \r\n
\r\n
Warehouse:          District: \r\n
"

#define h_pay4 "\r\n
\r\n
\r\n
Customer:  Cust-Warehouse:  Cust-
District: \r\n
Name:           Since: - -
\r\n"
/*
                Credit: \r\n
                %%Disc: . \r\n
*/

#define h_pay5 "\r\n
\r\n
\r\n
                Phone: - - -
\r\n
\r\n
\r\n
Amount Paid:  $ .     New Cust-Balance:
$ . \r\n
Credit Limit: $ . \r\n
\r\n
Cust-Data: "
/*
Cust-Data:          \r\n
\r\n
\r\n
\r\n"
*/

/* Trailer data */
#define h_pay3 "\r\n
</PRE>\r\n
<FORM ACTION=\"%s\" METHOD=\"GET\">\r\n
<INPUT TYPE=\"hidden\" NAME=\"c\"
VALUE=%d>\r\n
<INPUT TYPE=\"submit\" NAME=\"b\"
VALUE=\"New order\">\r\n
<INPUT TYPE=\"submit\" NAME=\"b\"
VALUE=\"Payment\">\r\n
<INPUT TYPE=\"submit\" NAME=\"b\"
VALUE=\"Delivery\">\r\n
<INPUT TYPE=\"submit\" NAME=\"b\"
VALUE=\"Order Status\">\r\n
<INPUT TYPE=\"submit\" NAME=\"b\"
VALUE=\"Stock Level\">\r\n
<INPUT TYPE=\"submit\" NAME=\"b\"
VALUE=\"Quit\">\r\n
</FORM></BODY></HTML>"

/* Offset to field which should set data */
int payp[] = {
0x06,
0x29, 0x51,
0x55, 0x7e,
0x94, 0xbd,
0xd3, 0xe8, 0xeb, 0xf1, 0xfc, 0x111, 0x114,
0x11a,
0x12c, 0x142, 0x158,
0x164, 0x175, 0x178, 0x195, /* 18 - 21 */
0x1a8, 0x1d9, /* 22, 23 */
0x1e5, 0x216, /* 24, 25 */
0x225, 0x23a, 0x23d, 0x256,
0x284, 0x2a4,
0x2c5,
0x2e1, /* offset 0x3e */
0x320,
0x35f,
0x39e};

```

```

:
:
: tpapl/stopage.h
:
:
:
:*****
:
: * COPYRIGHT FUJITSU LIMITED 2002
: * CREATE:1999.08.19 FJH
: *
:*****/
:
:/* -----
:-----
:  stopage.h
:  data of Stock Level transaction result screen
:(HTML form)
:----- */
:
:/* Header data */
#define h_stock1 "\r\n
<HTML><HEAD><TITLE>TPC-
WINDOW</TITLE></HEAD><BODY>\r\n
<CENTER>Stock-Level<BR></CENTER>\r\n
<font size=4>\r\n<PRE>"
:
:/* Screen data*/
#define h_stock2 "\r\n
Warehouse:  District: \r\n
\r\n
\r\n
Stock Level Threshold: \r\n
\r\n
\r\n
low stock: \r\n
\r\n\r\n\r\n\r\n\r\n\r\n\r\n\r\n\r\n\r\n\r\n\r\n\r\n\r\n"
:
:/* Trailer data */
#define h_stock3 "\r\n
</PRE>\r\n
<FORM ACTION=\"%s\" METHOD=\"GET\">\r\n
<INPUT TYPE=\"hidden\" NAME=\"c\"
VALUE=%d>\r\n
<INPUT TYPE=\"submit\" NAME=\"b\"
VALUE=\"New order\">\r\n
<INPUT TYPE=\"submit\" NAME=\"b\"
VALUE=\"Payment\">\r\n
<INPUT TYPE=\"submit\" NAME=\"b\"
VALUE=\"Delivery\">\r\n
<INPUT TYPE=\"submit\" NAME=\"b\"
VALUE=\"Order Status\">\r\n
<INPUT TYPE=\"submit\" NAME=\"b\"
VALUE=\"Stock Level\">\r\n
<INPUT TYPE=\"submit\" NAME=\"b\"
VALUE=\"Quit\">\r\n
</FORM></BODY></HTML>\n"
:
:/* Offset to field which should set data */
int stockp[] = {
0x0A, 0x1C,
0x39,
0x4a};
:
:
:*****
:
: * TPC-C Client Application Program Source
:
: * Entry Functions
: * struct definition.
:
:*****
:
:*****
:
: tpapl/tpapl.h
:
:*****
:
: * TPC-C Client Application Program Source
:
: * Entry Functions
: * struct definition.

```

```

*
*
* CREATE by TSL 2003.12.22
*
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2003 *
*****
****/
/* Http SO file path */
#define SOPATH "tpapl"
/* HTML editing buffer size */
#define WORK_S 2400
/* Flags */
#define OK 1
#define NG 0
/* Make w_id d_id form terminal no. */
#define MAC_w_id(cookie) (cookie - 1)/10 + 1
#define MAC_d_id(cookie) (cookie - 1)%10 + 1
:
: tpapl/tpaplFunction.c
:
:*****
:
: * TPC-C Client Application Program Source
:
: * Entry Functions
: * (1) only_para
: * (2) select_trn
: * (3) fast_menu
:
: * CREATE by TSL 2002.10.01
:
: * All Right Reserved, Copyright Co. FUJITSU
LIMITED 2002 *
*****
****/
#include "forlinux.h"
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <pthread.h>
#include <sys/time.h>
#include <atmi.h>
#include "trans.h"
#include "log.h"
// HTML-Page Data
#include "tpcweb.h"
#include "tpcinweb.h"
#include "menupage.h"
#include "tpccinf.h"
#include "tpapl.h"
#include "ThreadCntl.h"
#include "GlobalArea.h"
#include "TpApIxDBDependPrototype.h"
#include "sema.h"
#include "shmem.h"
#include "SampleInfo.h"
/*
only_para :
QueryString:-----

```

```

-----
-----: -----

Gets the query string and finds every
variable=value pair contained
within it. For every pair, it runs the variable
name through a really
big compound switch statement that matches
for specific variables we
want to catch. When we find a known variable
name, we stick a pointer
to its corresponding value into the appropriate
member of 'ptrs.'

query - a 1024 byte buffer that contains the
query string.
ptrs - a raw_form_data structure to hold
pointers.
*/
int only_para (char *para, RTE_INPUT_DATA
*in_data) {
    char *val, *rest;

    if(!para) return 0;
    if(*para == '\0') return 0;

    while(para) {

        rest = para_split(para, '&'); /* next parameta
point */
        val = para_split(para, '='); /* now value
point */

        switch(para[0]) {
        case 'c':
            in_data->cookie = val;    break;

        case 'b':
            in_data->button = val;    break;

        case 'f':
            in_data->form = val;    break;

        case 't':
            in_data->threshold = val;    break;

        case 'D':
            in_data->D_ID = val;    break;

        case 'H':
            in_data->H_AMOUNT = val;    break;

        case 'C':
            switch(para[1]) {
            case 'I':
                in_data->C_ID = val;    break;

            case 'W':
                in_data->C_W_ID = val;    break;

            case 'L':
                in_data->C_LAST = val;    break;

            case 'D':
                in_data->C_D_ID = val;    break;
            }
            break;

        case 'O':
            switch(para[1]) {
            case 'C':
                in_data->O_CARRIER_ID = val;
break;

```

```

        case 'S':
            switch(para[2]) {
            case 'O':
                if (para[3] >= 0x31 && para[3] <=
0x39){
                    //num = (int)(para[3] - 0x30);
                    if (strlen(val) != 0)
                        in_data-
>OL_SUPPLY_W_ID[(int)(para[3] - 0x30) - 1] =
val;
                }
                break;

            case '1':
                if (para[3] >= 0x30 && para[3] <=
0x35 ){
                    //num = (int)(para[3] - 0x30) + 10;
                    if (strlen(val) != 0)
                        in_data-
>OL_SUPPLY_W_ID[(int)(para[3] - 0x30) + 10 -
1] = val;
                }
                break;
            }
            break;

        case 'I':
            switch(para[2]) {
            case 'O':
                if (para[3] >= 0x31 && para[3] <=
0x39){
                    //num = (int)(para[3] - 0x30);
                    if (strlen(val) != 0)
                        in_data->OL_ID[(int)(para[3] -
0x30) - 1] = val;
                }
                break;

            case '1':
                if (para[3] >= 0x30 && para[3] <=
0x35 ){
                    //num = (int)(para[3] - 0x30) + 10;
                    if (strlen(val) != 0)
                        in_data->OL_ID[(int)(para[3] -
0x30) + 10 - 1] = val;
                }
                break;
            }
            break;

        case 'Q':
            switch(para[2]) {
            case 'O':
                if (para[3] >= 0x31 && para[3] <=
0x39 ){
                    //num = (int)(para[3] - 0x30);
                    if (strlen(val) != 0)
                        in_data-
>OL_QUANTITY[(int)(para[3] - 0x30) - 1] = val;
                }
                break;

            case '1':
                if (para[3] >= 0x30 && para[3] <=
0x35 ){
                    //num = (int)(para[3] - 0x30) + 10;
                    if (strlen(val) != 0)
                        in_data-
>OL_QUANTITY[(int)(para[3] - 0x30) + 10 - 1] =
val;
                }
                break;
            }
            break;

```

```

        }
        break;
    }

    para = rest;
}

if (in_data->cookie != 0)
    return(atoi (in_data->cookie));
else
    return(0);
}

/* -----
-
select_trn:
RTE-----
-----: -----
s_buf-----HTML-----
-----

interprets information from the user's input
data to determine which
page should be displayed back to the user.

query - the query string that comes back from
ParseFormData
ptrs - a pointer to a raw_form_data structure
with pointers
to values in 'query'.

-----
-----
*/
int select_trn ( RTE_INPUT_DATA *in_data,
char *s_buf, int cookie ) {

    int length = 0;
    int rtn = 0;

    MAC_SampleWork; /* Performance sampling
work area */

    if (in_data->form && (in_data->form[0] != 'M') )
    {

        if (in_data->form[0] == 'I'){
            /* send the transaction select screen
page */
            /* Replaced T,Kato 03.07,28 Speed up */
            /* rtn = fast_menu (s_buf, in_data,
cookie);*/
            sprintf(s_buf, h_menu, SOPATH,
cookie);
            /* Replaced end */
            return rtn;
        }
        else{

            MAC_SampleStartTime;

            /* check transaction type */
            switch(in_data->form[0]) {

                case 'N':
                    MAC_SampleQueueUp(NumQueNewOrder);
                    rtn = NewOrder (s_buf, in_data,
cookie);
                    MAC_SampleTuxResp(AnsNewOrder,
NumReqNewOrder, NumQueNewOrder);
                    break;

```

```

case 'D':
MAC_SampleQueueUp(NumQueDelivery);
    rtn = Delivery(s_buf, in_data, cookie);
    MAC_SampleTuxRespDel;
    break;

case 'P':
MAC_SampleQueueUp(NumQuePayment);
    rtn = Payment (s_buf, in_data, cookie);
    MAC_SampleTuxResp(AnsPayment,
NumReqPayment, NumQuePayment);
    break;

case 'S':
MAC_SampleQueueUp(NumQueStockLevel);
    rtn = StockLevel(s_buf, in_data,
cookie);
    MAC_SampleTuxResp(AnsStockLevel,
NumReqStockLevel, NumQueStockLevel);
    break;

case 'O':
MAC_SampleQueueUp(NumQueOrderStatus);
    rtn = OrderStatus (s_buf, in_data,
cookie);

MAC_SampleTuxResp(AnsOrderStatus,
NumReqOrderStatus, NumQueOrderStatus);
    break;

default:
    /* uninput transaction type */
    set_errpage(s_buf, cookie, 1, -4, 0, 0);
    rtn = 1;
    break;
}
/* Output self performance log */
MAC_SampleOutPutCsvLog;

return rtn;
}
else if(in_data->button) {

    /* send the data input screen page */
    switch(in_data->button[0]) {
case 'N':
        /*length = sprintf(s_buf, in_newpage,
SOPATH, cookie, srv->m_tcctxt[user_id].w_id);*/
        length = sprintf(s_buf, in_newpage,
SOPATH, cookie, MAC_w_id(cookie));
        strcpy(s_buf+length-1, in_newpage2);
        break;

case 'D':
        /*sprintf(s_buf, in_delpage, SOPATH,
cookie, srv->m_tcctxt[user_id].w_id);*/
        sprintf(s_buf, in_delpage, SOPATH,
cookie, MAC_w_id(cookie));
        break;

case 'P':
        /*sprintf(s_buf, in_paypage, SOPATH,
cookie, srv->m_tcctxt[user_id].w_id);*/
        sprintf(s_buf, in_paypage, SOPATH,
cookie, MAC_w_id(cookie));
        break;

case 'S':
        /*sprintf(s_buf, in_stkpage, SOPATH,
cookie,

```

```

        srv->m_tcctxt[user_id].w_id, srv-
>m_tcctxt[user_id].d_id);*/
        sprintf(s_buf, in_stkpage, SOPATH,
cookie, MAC_w_id(cookie), MAC_d_id(cookie));
        break;

case 'O':
        /*sprintf(s_buf, in_odrpage, SOPATH,
cookie, srv->m_tcctxt[user_id].w_id);*/
        sprintf(s_buf, in_odrpage, SOPATH,
cookie, MAC_w_id(cookie));
        break;

case 'Q':
        sprintf (s_buf, loginpage , VLDATA,
SOPATH);
        /* Replaced 03.01.15 Can't LeaveCriticalSection
*/
        #if 0
        ! return rtn;
        #endif
        break;
        /* Replaced end */

default:
        /* uninput transaction type */
        set_errpage(s_buf, cookie, 0, -4, 0, 0);
        break;
    }
    return rtn;
}
else {

        /* if there is not parameter then send login
page data.
        this part use WWW browser only */
        sprintf (s_buf, loginpage, VLDATA,
SOPATH);
        return 0;
    }
}

/* Deleted T,Kato 03.07.28 Speed up */
#if 0
/*
! fast_menu:
! This function reads a user's responses to the
login form, sets
! up the user context, and returns the menu
page.
!*/
!
!int fast_menu ( char *s_buf, RTE_INPUT_DATA
*in_data, int cookie){
!
! //for warning
! in_data;
!
! sprintf(s_buf, h_menu, SOPATH, cookie);
! return 0;
!}
#endif

.....
tpapl/tpccinf.h
.....
/******
*
* COPYRIGHT FUJITSU LIMITED 2002
* CREATE:1999.11.19 FJH
*
* Modified TSL 2003.12.22

```

```

.....*/
/*=====
=====+
FILENAME : tpccinf.h
DESCRIPTION

+=====
=====*/

#ifndef TPCCINF_H
#define TPCCINF_H

#define QUERY_STR_SIZE 1024
#define RESP_BUF_SIZE 4096

#define VLDATA "Ver 1.0 Linux & Tuxedo"

#ifndef SCRTEST

#ifndef DBPRT
#define MDDATA "SCR And DP"
#else
#define MDDATA "SCR"
#endif

#else

#ifndef NOSCR
#define MDDATA "DBG"
#else
#define MDDATA "REL"
#endif

#endif

#endif

.....
tpapl/tpcinweb.h
.....

/******
*
* COPYRIGHT FUJITSU LIMITED 2002
* CREATE:1999.08.19 FJH
*
*.....*/

/* -----
-----
tpcinweb.h
Transaction input data screen data
----- */

/* -----
-----*/

#define in_delpage "\
<HTML><HEAD><TITLE>TPC-C:
Delivery</TITLE></HEAD>\r\n\
<BODY><FORM ACTION=\"%s\"
METHOD=\"GET\">\r\n\
<INPUT TYPE=\"hidden\" NAME=\"f\"
VALUE=\"D\">\r\n\
<INPUT TYPE=\"hidden\" NAME=\"c\"
VALUE=\"%d\">\r\n\
<center>Delivery<br></center>\r\n\
<font size=4><PRE>Warehouse:%6d\r\n\
\r\n\

```



```
Customer: <INPUT NAME="CI" SIZE=4
maxlength=4> Cust-Warehouse: <INPUT
NAME="CW" SIZE=5 maxlength=6> Cust-
District: <INPUT NAME="CD" SIZE=2
maxlength=2>\r\n
Name: <INPUT NAME="CL"
SIZE=17 maxlength=16> Since:\r\n
Credit:\r\n
%%Disc:\r\n
Phone:\r\n
\r\n
Amount Paid $<INPUT NAME="H"
SIZE=7 maxlength=7> New Cust-
Balance:\r\n
Credit Limit:\r\n
\r\n
Cust-Data:\r\n
\r\n
\r\n
</PRE>\r\n
<INPUT
TYPE="submit"></FORM></BODY></HTML>"
```

```
/* -----
stock level page
* -----*/
#define in_stkpage "\
<HTML><HEAD><TITLE>TPC-C: Stock-
Level</TITLE></HEAD>\r\n
<BODY><FORM ACTION="%s"
METHOD="GET">\r\n
<INPUT TYPE="hidden" NAME="f"
VALUE="S">\r\n
<INPUT TYPE="hidden" NAME="c"
VALUE=%d>\r\n
<center>Stock-Level<br></center>\r\n
<font size=4><PRE>Warehouse:%6d
District: %2d\r\n
\r\n
Stock Level Threshold: <INPUT NAME="t"
SIZE=2 maxlength=2>\r\n
\r\n
low stock:\r\n
</PRE>\r\n
<INPUT
TYPE="submit"></FORM></BODY></HTML>"
```

```
#define in_stkpage2 "\
<HTML><HEAD><TITLE>TPC-C: Stock-
Level</TITLE></HEAD>\r\n
<BODY><FORM ACTION="%s"
METHOD="GET">\r\n
<INPUT TYPE="hidden" NAME="f"
VALUE="S">\r\n
<INPUT TYPE="hidden" NAME="c"
VALUE=%d>\r\n
<center>Stock Level<br></center>\r\n
<font size=3><PRE>\r\n
Warehouse:%6d District:%2d\r\n
\r\n
Stock Level Threshold:<INPUT NAME="t"
SIZE=2 maxlength=2>\r\n
\r\n
low stock:\r\n
</PRE>\r\n
<INPUT
TYPE="submit"></FORM></BODY></HTML>\r
n"
```

```
.....
tpapl/tpcweb.h
.....
/*-----
```

```
*
* COPYRIGHT FUJITSU LIMITED 2002
* CREATE:1998.08.06 FJH
*
*****/
/* -----
-----
tpcweb.h
-----*/
```

```
/* If transaction input data is abnormal then use
this format. */
#define errhtml "\
<HTML><HEAD><TITLE>ERROR: TPC-
C</TITLE></HEAD><BODY>\
<p>You did something bad. The error message
was:</p>\
<PRE>%s</PRE>\
<p>Either hit the "back" button on your browser
and fix the problem, \
or hit the "Quit" button below to terminate this
session. </P><HR>\
<P><FORM ACTION="%s"
METHOD="GET">\
<INPUT TYPE="hidden" NAME="c"
VALUE=%d>\
<INPUT TYPE="submit" NAME="b"
VALUE="Quit">\
</FORM></P></BODY></HTML>\r\n"
```

```
/* If TP application terminated abnormally then
use this format. */
#define tuxerr "\
<HTML><HEAD><TITLE>ERROR: Tuxedo
</TITLE></HEAD><BODY>\
<P>The database could not process your
request. \
tpcall terminated abnormally.</P>\
<HR><PRE>%s</PRE><HR>\
<FORM ACTION="%s" METHOD="GET">\
<INPUT TYPE="hidden" NAME="c"
VALUE=%d>\
<INPUT TYPE="submit" NAME="b"
VALUE="Quit">\
</BODY></HTML>"
```

```
/* If application terminated abnormally then use
this format. */
#define errorpage "\
<HTML><HEAD><TITLE>ERROR: %s
</TITLE></HEAD><BODY>\
<P>The database could not process your
request. \
Transaction terminated abnormally.</P>\
<HR><PRE>%s</PRE><HR>\
<FORM ACTION="%s" METHOD="GET">\
<INPUT TYPE="hidden" NAME="c"
VALUE=%d>\
<INPUT TYPE="submit" NAME="b"
VALUE="Quit">\
</BODY></HTML>"
```

```
#if 0 /* oraerr.symfoerr -> errorpage */
/* [oraerr]-[symfoerr]--"TITLE"-----
-----*/
/* Since "TITLE" was only different, [oraerr] and
[symfoerr] were changed so that it might be
common and could use.*/
/* If Oracle application terminated abnormally
then use this format. */
#define oraerr "\
!<HTML><HEAD><TITLE>ERROR: ORACLE
</TITLE></HEAD><BODY>\
```

```
<P>The database could not process your
request. \
!Transaction terminated abnormally.</P>\
!<HR><PRE>%s</PRE><HR>\
!<FORM ACTION="%s" METHOD="GET">\
!<INPUT TYPE="hidden" NAME="c"
VALUE=%d>\
!<INPUT TYPE="submit" NAME="b"
VALUE="Quit">\
!</BODY></HTML>"
!
/* If SymfoWare application terminated
abnormally then use this format. */
#define symfoerr "\
!<HTML><HEAD><TITLE>ERROR:
SYMFOWARE</TITLE></HEAD><BODY>\
!<P>The database could not process your
request. \
!Transaction terminated abnormally.</P>\
!<HR><PRE>%s</PRE><HR>\
!<FORM ACTION="%s" METHOD="GET">\
!<INPUT TYPE="hidden" NAME="c"
VALUE=%d>\
!<INPUT TYPE="submit" NAME="b"
VALUE="Quit">\
!</BODY></HTML>"
#endif
```

```
/* If TPINIT() abnormally then use this format. */
#define tuxierr "\
<HTML><HEAD><TITLE>ERROR: Tuxedo-init
</TITLE></HEAD><BODY>\
<P>The database could not process your
request. \
%s terminated abnormally.</P>\
</BODY></HTML>"
```

```
.....
tpapl/trans.h
.....
/*-----
*
* COPYRIGHT FUJITSU LIMITED 2002
* CREATE:1999.10.28 FJH
*
*****/
/*=====
=====+
FILENAME : trans.h
the work struct according to transaction is
declared.
+=====
=====*/
```

```
/* RTE - Client interface struct */
typedef struct {
char *button,
*cookie,
*form,
*O_CARRIER_ID,
*threshold,
*D_ID,
*C_ID,
*C_W_ID,
*C_D_ID,
*C_LAST,
*H_AMOUNT,
*OL_SUPPLY_W_ID[15],
*OL_ID[15],
```

```

*OL_QUANTITY[15];
} RTE_INPUT_DATA;
//) rte_input_data;

.....
tpapl/trnexe/ConvTime.c
.....

/*****
*
*          *
*   TPC-C Client Application Program Source
*
*          *
* Entry Functions
* (1) time2str
*
* CREATE by TSL 2002.10.01
*
*          *
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2002 *
*****
*****/

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "forlinux.h"

#include "trans.h"
#include "ThreadCntl.h"
#include "TpAplDBDependPrototype.h"

/*
time2str:
Outputs a date and time in the supplied buffer
in the following format:
DD-MM-YYYY hh:mm:ss

field = the destination field
date = date and time to be converted and
displayed
*/
void time2str (char *str, char *time)
{
short mon;
int year, day, hour, min, sec;
char month[4];

#ifdef DBPRT
printf (test_fp, "time2: %s\n", time);
#endif
/* Modified by TSL -- BEGIN -- 2006.03.17 */
#if 0
! sscanf( time, "%2d-%3s-%2d.%2d:%2d:%2d",
! &day, month, &year, &hour, &min,
&sec );

! if(strcmp(month, "jan") == 0)
! strcpy(month, "01");
! if(strcmp(month, "feb") == 0)
! strcpy(month, "02");
! if(strcmp(month, "mar") == 0)
! strcpy(month, "03");
! if(strcmp(month, "apr") == 0)
! strcpy(month, "04");
! if(strcmp(month, "may") == 0)
! strcpy(month, "05");
! if(strcmp(month, "jun") == 0)
! strcpy(month, "06");
! if(strcmp(month, "jul") == 0)
! strcpy(month, "07");

```

```

! if(strcmp(month, "aug") == 0)
! strcpy(month, "08");
! if(strcmp(month, "sep") == 0)
! strcpy(month, "09");
! if(strcmp(month, "oct") == 0)
! strcpy(month, "10");
! if(strcmp(month, "nov") == 0)
! strcpy(month, "11");
! if(strcmp(month, "dec") == 0)
! strcpy(month, "12");
!
! int3str (str, 2, day);
! str[2] = '-';
!
! mon = atoi(month);
! int3str (&str[3], 2, mon);
! str[5] = '-';
!
! /* ----- */
! if ( year >= 70 )
! year += 1900;
! else
! year += 2000;
!
! int3str (&str[6], 4, year);
!
#endif
sscanf( time, "%2d-%2d-%4d.%2d:%2d:%2d",
&day, &mon, &year, &hour, &min, &sec );

int3str (str, 2, day);
str[2] = '-';
int3str (&str[3], 2, mon);
str[5] = '-';
int3str (&str[6], 4, year);

/* Modified by TSL -- END -- 2006.03.17 */

str[10] = ' ';

int3str (&str[11], 2, hour);
str[13] = ' ';

int3str (&str[14], 2, min);
str[16] = ' ';

int3str (&str[17], 2, sec);
}

.....
tpapl/trnexe/CreateTranErrReason.c
.....

/*****
*
*          *
*   TPC-C Client Application Program Source
*
*          *
* Entry Functions
* (1) CreateTranErrReason
*
* CREATE by TSL 2003.12.15
*
*          *
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2003 *
*****
*****/
#include <stdio.h>
#include <stdlib.h>

```

```

#include <string.h>
#include "forlinux.h"

#include "atmi.h"
#include "tpcc.h"

int CreateTranErrReason(long errno_code, int
reason_code, char** reason_message) {
/* errno_code ..... return value of "tpcall" or
"tpacall"
* reason_code ..... xxxout.terror
* reason message ... convert message
*/
switch (errno_code) {

/* tpcall/tpacall error */
case -1:
TpccUserLog (LOG_ERR, "tpcall/tpacall
execution error occurred. [errno_code=%d]\n",
errno_code);
*reason_message = "Irrecoverable error in
tpcall/tpacall.";
return -2;
break;

/* Normal end */
default:
switch(reason_code) {
/* Normaol end */
case NOERR:
return 0;

/* Irrecoverable error */
case IRRECERR:
TpccUserLog (LOG_ERR, "Transaction
processing error [IRRECERR] occurred.\n");
*reason_message = "Irrecoverable error
in transaction processing.";
return -1; /* Execution error */

/* Retry */
default:
return 1;
}
}
}

.....
tpapl/trnexe/MakeShell
.....

#!/bin/sh
cd /home/tpc/client_apl/tpapl/trnexe
echo ""-----"" >
make_result.txt
echo ""----FOR WARE HOUSE BIND-----"" >>
make_result.txt
echo ""-----"" >>
make_result.txt
make BIND_TYPE="WH_BIND" >>
make_result.txt 2>&1
echo ""-----"" >>
make_result.txt
echo ""----FOR TRANSACTION BIND-----"" >>
make_result.txt
echo ""-----"" >>
make_result.txt
rm *.o >> make_result.txt
2>&1
make BIND_TYPE="TRNS_BIND" >>
make_result.txt 2>&1

```



```

.....
tpapl/trnexe/Makefile
.....

#-----
-----
# Makefile : Makefile for TpApl library on Linux.
#
# Created by TSL 2003.12.18
#
# All Right Rserverd, Copyright Co, FUJITSU
LIMITED 2003-2004.
#-----
-----

# GCC compile configurations
AR = ar
ARFLAGS = rv
CFLAGS = -Wall -O2
CC = gcc

# MACRO definition (input parameter)
# BIND_TYPE = TRNS_BIND ... Transaction
bind
# WH_BIND ..... Ware house bind
DMACRO = -D$(BIND_TYPE)

# home directory.
TOPDIR = /home/tpc/client_apl
TUXDIR = /usr/local/BEA/tuxedo8.1
APADIR = /usr/include/httpd
APLDIR = $(TOPDIR)/tpapl
SVRDIR = $(TOPDIR)/svrapl
ORADIR = /usr/local/oracle

# include directory
TPA_INC = -I$(APLDIR)/trnexe
COM_INC = -I$(TOPDIR)/common
TUX_INC = -I$(TUXDIR)/include
APA_INC = -I$(APADIR)
APL_INC = -I$(APLDIR)
SVR_INC = -I$(SVRDIR)
ORA_INC = -I$(ORADIR)/rdbms/demo -
-I$(ORADIR)/rdbms/public

# header file directory
HDFDIR = $(APLDIR)/trnexe
COMDIR = $(TOPDIR)/common

INCLUDE = $(TPA_INC) $(COM_INC)
$(APA_INC) $(TUX_INC) $(APL_INC)
$(SVR_INC) $(ORA_INC)
INCFILE = $(SVRDIR)/tpcc_info.h \
$(HDFDIR)/OracleInfo.h \
$(HDFDIR)/OracleFunction.h \
$(HDFDIR)/log_level.h \
$(APLDIR)/GlobalArea.h \
$(APLDIR)/trans.h \
$(APLDIR)/tpcweb.h \
$(APLDIR)/TpAplDBDependPrototype.h \
$(APLDIR)/tpapl.h \
$(APLDIR)/ThreadCntl.h \
$(APLDIR)/stpage.h \
$(APLDIR)/paypage.h \
$(APLDIR)/odrpage.h \
$(APLDIR)/newpage.h \
$(APLDIR)/delpage.h \
$(COMDIR)/log.h \
$(COMDIR)/forlinux.h \

# target object
OBS = ConvTime.o CreateTranErrReason.o
TestFunction.o TransactionDataLen.o \

```

```

TrxDelivery.o TrxNewOrder.o
TrxOrderStatus.o TrxPayment.o TrxStockLevel.o
ARCH_LIB =
$(APLDIR)/trnexe/libtrnexe_$(BIND_TYPE).a

$(ARCH_LIB) : $(OBS)
$(AR) $(ARFLAGS) $(ARCH_LIB) $(OBS)

.SUFFIXES: .o .c
.c.o:
$(CC) -o $@ -c $(CFLAGS) $(INCLUDE)
$(DMACRO) $<

$(OBS) : $(INCFILE)

clean:
# rm $(ARCH_LIB) $(OBS)

.....
.....
tpapl/trnexe/OracleFunction.h
.....

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

*
*
* TPC-C Client Application Program Source
*
*
* Entry Functions
* Function definition for Oracle.
*
* CREATE by TSL 2002.10.01
*
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2002 *

*****
*****/

// -----
// TrxNewOrder.cpp
// -----
int chk_NOdata (NewOrderData *bp, int cnt,
RTE_INPUT_DATA *in_data, int svcnt);
int setNOdata (char *s_work, int OF, int cnt,
NewOrderData *bp, RTE_INPUT_DATA
*in_data);

// -----
// TestProc.cpp
// -----
void dummy_delivery ( DeliveryData *bp );
void dummy_stocklvl ( StockLevelData *bp );
void dummy_payment ( PaymentData *bp );
void dummy_orderstat ( OrderStatusData *bp );
void dummy_neworder ( NewOrderData *bp );
void oder_dsp (RTE_INPUT_DATA *in_data,
OrderStatusData *bp, int w_id, int d_flag);
void pay_dsp (RTE_INPUT_DATA *in_data,
PaymentData *bp, int w_id, int d_flag);
void sto_dsp (RTE_INPUT_DATA *in_data,
StockLevelData *bp, int w_id, int d_id, int
d_flag);
void new_dsp (RTE_INPUT_DATA *in_data,
NewOrderData *bp, int w_id, int d_flag, int cnt);

int CreateTranErrReason (long errno_code, int
reason_code, char** reason_message);

// -----Oracle--Symfo-----
// used in common by Oracle and Symfo.

```

```

#define MAC_errHTML(page, err_inf, cookie )
set_errHTML(page, err_inf, cookie, "ORACLE" );
#define MAC_errHTML_TUXEDO(page, err_inf,
cookie ) set_errHTML(page, err_inf, cookie,
"TUXEDO" );

.....
.....
tpapl/trnexe/OracleInfo.h
.....

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

*
*
* TPC-C Client Application Program Source
*
*
* Entry Functions
* Oracle Area definition.
*
* CREATE by TSL 2002.10.01
*
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2002 *

*****
*****/

#ifndef ORACLEINFO_H
#define ORACLEINFO_H

#define INTNULL 0

#endif

.....
.....
tpapl/trnexe/TestFunction.c
.....

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

*
*
* TPC-C Client Application Program Source
*
*
* Entry Functions
* (1) get_datetimestr
* (2) get_datestr
* (3) dummy_delivery
* (4) dummy_stocklvl
* (5) dummy_payment
* (6) dummy_orderstat
* (7) dummy_neworder
* (8) oder_dsp
* (9) pay_dsp
* (10) sto_dsp
* (11) new_dsp
* (12) tsp
*
* CREATE by TSL 2002.10.01
*
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2002 *

*****
*****/

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "forlinux.h"

```

```
#include "trans.h"
#include "tpcc_info.h"

//
// dmy.h : DLL-----
// -----
// ----- --SCRTEST--#define-----
// -----DBPRT--#define-----
//

//
// -----
//

#ifdef SCRTEST

char *get_datetimestr( char *buf )
{
    struct tm    *tm;
    time_t      tim;

    time( &tim );
    tm = localtime( &tim );

    sprintf( buf, "%2d-%2d-%4d %2d:%2d:%2d",
tm->tm_mday, tm->tm_mon+1,
tm->tm_year+1900, tm->tm_hour, tm->tm_min, tm->tm_sec );

    return buf;
}

char *get_datestr( char *buf )
{
    struct tm    *tm;
    time_t      tim;

    time( &tim );
    tm = localtime( &tim );

    sprintf( buf, "%2d-%2d-%4d",
tm->tm_mday, tm->tm_mon+1, tm->tm_year+1900 );
    return buf;
}

void dummy_delivery( DeliveryData *bp )
{
    bp->delout.terror = NOERR;

    return;
}

void dummy_stocklvl( StockLevelData *bp )
{
    int i;

    bp->stoout.terror = NOERR;

    do{
        i = rand()%1000;
    } while ( i > bp->stoin.threshold );

    bp->stoout.low_stock = i;

    return;
}

```

```
void dummy_payment( PaymentData *bp )
{
    bp->payout.terror = NOERR;
    strcpy( bp->payout.h_date, "11-oct-02.16:37:15" );
    strcpy( bp->payout.w_street_1, "Baker street" );
    strcpy( bp->payout.w_street_2, "221B" );
    strcpy( bp->payout.w_city, "London" );
    strcpy( bp->payout.w_state, "GB" );
    strcpy( bp->payout.w_zip, "88033000" );

    strcpy( bp->payout.d_street_1, "Minato-ku" );
    strcpy( bp->payout.d_street_2, "Azabu 10" );
    strcpy( bp->payout.d_city, "Tokyo" );
    strcpy( bp->payout.d_state, "JP" );
    strcpy( bp->payout.d_zip, "102 1234" );

    bp->payout.c_id = 777;
    strcpy( bp->payout.c_first, "John" );
    strcpy( bp->payout.c_middle, "H" );
    strcpy( bp->payout.c_last, "Watson" );
    strcpy( bp->payout.c_street_1, "Baker street" );
    strcpy( bp->payout.c_street_2, "221B" );
    strcpy( bp->payout.c_credit, "GC" );
    bp->payout.c_discount = (float)20.00;
// check
    strcpy( bp->payout.c_city, "London" );
    strcpy( bp->payout.c_state, "GB" );
    strcpy( bp->payout.c_zip, "888 1234" );
    strcpy( bp->payout.c_phone, "12345678901234567" );
    bp->payout.c_balance = 67876;
    bp->payout.c_credit_lim = 77777;
    strcpy( bp->payout.c_since, "11-10-2002" );

    strcpy( bp->payout.c_data, "Migyamigyamigyamigyamigyamigya" );
    return;
}

void dummy_orderstat( OrderStatusData *bp )
{
    int i, j;

    bp->ordout.terror = NOERR;
    bp->ordout.c_id = rand()%10000;
    strcpy( bp->ordout.c_first, "Robert" );
    strcpy( bp->ordout.c_middle, "L" );
    strcpy( bp->ordout.c_last, "Fish" );
    bp->ordout.c_balance = ( ( rand()*rand()%19999999 )-9999999 ) / (double)100.0;

    bp->ordout.o_id = rand()%10000;
    strcpy( bp->ordout.o_entry_d, "11-oct-02.16:25:45" );
    bp->ordout.o_carrier_id = rand()%100;

    bp->ordout.o_ol_cnt = ( rand()%11 )+5;
    j = bp->ordout.o_ol_cnt;
    for ( i = 0; i < j; i++ )
    {
        bp->ordout.ol_supply_w_id[i] = ( rand()%100000 )+1;
        bp->ordout.ol_i_id[i] = ( rand()%100000 )+1;
        bp->ordout.ol_quantity[i] = ( rand()%99 )+1;
        bp->ordout.ol_amount[i] = (float)rand();
// check

        sprintf( bp->ordout.ol_delivery_d[i], "%02d-10-2002",i + 1 );
    }
}

```

```

}

return;
}

void dummy_neworder( NewOrderData *bp )
{
    static int o_id = 3001;
    int i;

    bp->newout.terror = NOERR;

    strcpy( bp->newout.c_last, "Holmes" );
    strcpy( bp->newout.c_credit, "GC" );
    bp->newout.o_id = o_id++;

    strcpy( bp->newout.o_entry_d, "11-oct-02.15:10:30" );
    bp->newout.c_discount = (float)(( rand()%101 )/10000.0); // check
    bp->newout.w_tax = (float)(( rand()%2001 )/10000.0); // check
    bp->newout.d_tax = (float)(( rand()%2001 )/10000.0); // check
    bp->newout.total_amount = 0; // check

    for ( i = 0; i < 15; i++ ){
        if ( bp->newin.ol_supply_w_id[i] == 0 ) {
            break;
        }
        if ( bp->newin.ol_i_id[i] == -1 ) {
        }

        sprintf( bp->newout.i_name[i], "ItemName%02d", i );
        bp->newout.s_quantity[i] = ( rand()%10 )+1;
        bp->newout.brand_generic[i] = ( rand()%26 )+'A';
        bp->newout.i_price[i] = (float)(( ( rand()%10000 )+1 )/100.0); // check
        bp->newout.ol_amount[i] = bp->newout.i_price[i] * bp->newin.ol_quantity[i]; // check
        bp->newout.total_amount += bp->newout.ol_amount[i]; // check
    }
    bp->newout.o_ol_cnt = i;

    return;
}

#ifdef
//
// -----
//

#ifdef DBPRT
void oder_dsp( RTE_INPUT_DATA *in_data, OrderStatusData *bp, int w_id, int d_flag )
{
    int i;

    if ( d_flag == 0 ){
        fprintf( test_fp, "----- in data area -----\n\n" );
        fprintf( test_fp, "w_id = %d ", w_id );
        fprintf( test_fp, "d_id = %s ", in_data->D_ID );

        if ( in_data->C_ID != 0 )
            fprintf( test_fp, "c_id = %s\n", in_data->C_ID );
        if ( in_data->C_LAST != 0 )

```

```

    fprintf(test_fp, "c_last = %s \n", in_data-
>C_LAST);

    fprintf(test_fp, "----- trans buf area -----
\n\n");
    fprintf(test_fp, "w_id = %d ", bp->w_id);
    fprintf(test_fp, "d_id = %d ", bp->d_id);
    fprintf(test_fp, "c_id = %d\n", bp->c_id);
}
else {
    fprintf(test_fp, "----- trans buf area (after) ----
\n\n");
    fprintf(test_fp, "w_id = %d ", bp->w_id);
    fprintf(test_fp, "d_id = %d ", bp->d_id);
    fprintf(test_fp, "c_id = %d\n", bp->c_id);
    fprintf(test_fp, "c_first=%s ", bp->c_first);
    fprintf(test_fp, "c_middl=%s ", bp-
>c_middle);
    fprintf(test_fp, "c_last =%s\n", bp->c_last);

    fprintf(test_fp, "c_balan=%f ", bp-
>c_balance);
    fprintf(test_fp, "o_id =%d ", bp->o_id);
    fprintf(test_fp, "o_entry_d=%s\n", bp-
>o_entry_d); // check

    if ( bp->o_carrier_id != 0 ) {
        fprintf(test_fp, "o_carrier_id=%d\n", bp-
>o_carrier_id);
    }

    for( i = 0; i < bp->o_ol_cnt; i++ ){
        fprintf(test_fp, "ol_supp=%d ", bp-
>ol_supply_w_id[i]);
        fprintf(test_fp, "ol_i_id=%d ", bp-
>ol_i_id[i]);
        fprintf(test_fp, "ol_quan=%d ", bp-
>ol_quantity[i]);
        fprintf(test_fp, "ol_amou=%f\n", bp-
>ol_amount[i]);
    }
}

void pay_dsp(RTE_INPUT_DATA *in_data,
PaymentData *bp, int w_id, int d_flag)
{
    int i;

    if (d_flag == 0){
        fprintf(test_fp, "----- in data area -----
\n\n");

        fprintf(test_fp, "w_id = %d ", w_id);
        fprintf(test_fp, "d_id = %s ", in_data->D_ID);
        fprintf(test_fp, "c_w_id=%s ", in_data-
>C_W_ID);
        fprintf(test_fp, "c_d_id=%s ", in_data-
>C_D_ID);
        fprintf(test_fp, "h_amount=%s \n", in_data-
>H_AMOUNT);

        if (in_data->C_ID != 0)
            fprintf(test_fp, "c_id = %s \n", in_data-
>C_ID);
        if (in_data->C_LAST != 0)
            fprintf(test_fp, "c_last = %s \n", in_data-
>C_LAST);

        fprintf(test_fp, "----- trans buf area -----
\n\n");
        fprintf(test_fp, "w_id = %d ", bp->w_id);
        fprintf(test_fp, "d_id = %d ", bp->d_id);
        fprintf(test_fp, "c_id = %d ", bp->c_id);
        fprintf(test_fp, "c_w_id=%d ", bp->c_w_id);

```

```

    fprintf(test_fp, "c_d_id=%d ", bp->c_d_id);
    fprintf(test_fp, "h_amount=%f \n", bp-
>h_amount);
}
else {
    fprintf(test_fp, "----- trans buf area (after) ---
\n\n");
    fprintf(test_fp, "w_id = %d ", bp->w_id);
    fprintf(test_fp, "d_id = %d ", bp->d_id);
    fprintf(test_fp, "c_id = %d\n", bp->c_id);

    fprintf(test_fp, "w_str_1=%s ", bp-
>w_street_1);
    fprintf(test_fp, "w_str_2=%s\n", bp-
>w_street_2);
    fprintf(test_fp, "d_str_1=%s ", bp-
>d_street_1);
    fprintf(test_fp, "d_str_2=%s\n", bp-
>d_street_2);
    fprintf(test_fp, "w_city=%s ", bp->w_city);
    fprintf(test_fp, "w_state=%s\n", bp->w_state);
    fprintf(test_fp, "d_city=%s ", bp->d_city);
    fprintf(test_fp, "d_state=%s\n", bp->d_state);

    fprintf(test_fp, "c_w_id=%d ", bp->c_w_id);
    fprintf(test_fp, "d_w_id=%d\n", bp->c_d_id);

    fprintf(test_fp, "c_first=%s ", bp->c_first);
    fprintf(test_fp, "c_middl=%s ", bp-
>c_middle);
    fprintf(test_fp, "c_last =%s\n", bp->c_last);

    fprintf(test_fp, "c_str_1=%s ", bp-
>c_street_1);
    fprintf(test_fp, "c_str_2=%s\n", bp-
>c_street_2);
    fprintf(test_fp, "c_city=%s\n", bp->c_city);
    fprintf(test_fp, "c_credi=%s ", bp->c_credit);
    fprintf(test_fp, "c_state=%s\n", bp->c_state);

    fprintf(test_fp, "c_balan=%f\n", bp-
>c_balance);

    i = strlen( bp->c_data );
    fprintf(test_fp, "c_date=%s\n", bp->c_data);
}

void sto_dsp(RTE_INPUT_DATA *in_data,
StockLevelData *bp, int w_id, int d_id, int
d_flag)
{
    if (d_flag == 0){
        fprintf(test_fp, "----- in data area -----
\n\n");

        fprintf(test_fp, "w_id = %d ", w_id);
        fprintf(test_fp, "d_id = %d ", d_id);
        fprintf(test_fp, "threshold= %s \n", in_data-
>threshold);

        fprintf(test_fp, "----- trans buf area -----
\n\n");
        fprintf(test_fp, "w_id = %d ", bp->w_id);
        fprintf(test_fp, "d_id = %d ", bp->d_id);
        fprintf(test_fp, "threshold= %d \n", bp-
>threshold);
    }
    else{
        fprintf(test_fp, "----- trans buf area (after) ---
\n\n");

        fprintf(test_fp, "w_id = %d ", bp->w_id);
        fprintf(test_fp, "d_id = %d ", bp->d_id);

```

```

        fprintf(test_fp, "threshold= %d ", bp-
>threshold);
        fprintf(test_fp, "low_stock= %d \n", bp-
>low_stock);
    }
}

void new_dsp(RTE_INPUT_DATA *in_data,
NewOrderData *bp, int w_id, int d_flag,
int cnt)
{
    int i, loop;

    if (d_flag == 0){
        fprintf(test_fp, "----- in data area -----
\n\n");

        fprintf(test_fp, "w_id = %d ", w_id);
        fprintf(test_fp, "d_id = %s ", in_data->D_ID);
        fprintf(test_fp, "c_id = %s \n", in_data-
>C_ID);

        for (i = 0; i < cnt; i++){

            if (in_data->OL_SUPPLY_W_ID[i] != 0 ){
                fprintf(test_fp, "ol_sup_w_id=%s
", in_data->OL_SUPPLY_W_ID[i]);
            }

            if (in_data->OL_I_ID[i] != 0 ){
                fprintf(test_fp, "ol_i_id=%s ", in_data-
>OL_I_ID[i]);
            }

            if (in_data->OL_QUANTITY[i] != 0 ){
                fprintf(test_fp, "ol_quan=%s\n",
in_data->OL_QUANTITY[i]);
            }
        }

        fprintf(test_fp, "----- trans buf area -----
\n\n");
        fprintf(test_fp, "w_id = %d ", bp->w_id);
        fprintf(test_fp, "d_id = %d ", bp->d_id);
        fprintf(test_fp, "c_id = %d\n", bp->c_id);

        for (i = 0; i <= cnt; i++){

            fprintf(test_fp, "ol_sup_w_id=%d ", bp-
>ol_supply_w_id[i]);
            fprintf(test_fp, "ol_i_id=%d ", bp-
>ol_i_id[i]);
            fprintf(test_fp, "ol_quan=%d\n", bp-
>ol_quantity[i]);
        }
    }
    else{
        fprintf(test_fp, "----- trans buf area (after) ---
\n\n");

        fprintf(test_fp, "c_last=%s ", bp->c_last);
        fprintf(test_fp, "c_credit=%s\n", bp-
>c_credit);
        fprintf(test_fp, "o_id=%d ", bp->o_id);

        fprintf(test_fp, "o_entry_d=%s\n", bp-
>o_entry_d); // check
        fprintf(test_fp, "c_discnt=%f\n", bp-
>c_discount * 100.0);

        fprintf(test_fp, "o_ol_cnt=%d ", bp-
>o_ol_cnt);

        fprintf(test_fp, "w_tax=%f ", bp->w_tax *
100.0);

```

```

    fprintf (test_fp, "d_tax=%f\n", bp->d_tax *
100.0);

    loop = bp->o_ol_cnt;
    for (i = 0; i < loop; i++) {

        fprintf(test_fp, "-----
no_sup_w_id=%d ",
            bp->ol_supply_w_id[i]);
        fprintf(test_fp, "o_i_id=%d ", bp-
>o_i_id[i]);
        fprintf(test_fp, "i_name=%s\n", &bp-
>i_name[i][0]);
        fprintf(test_fp, "o_quant=%d ", bp-
>o_ol_quantity[i]);
        fprintf(test_fp, "s_quant=%d ", bp-
>s_quantity[i]);
        fprintf(test_fp, "brand=%c ", bp-
>brand_generic[i]);
        fprintf(test_fp, "i_price=%f ", bp-
>i_price[i]); // check
        fprintf(test_fp, "ol_amnt=%f\n", bp-
>o_ol_amount[i]); // check
    }
    fprintf (test_fp, "total_a=%f\n", bp-
>total_amount); // check
}

#endif

#ifdef TIMEST
int tsp(int id, char flag, char type){

// struct tm times;
SYSTEMTIME systemTime; // for IIS Version

GetLocalTime(&systemTime);

fprintf (TIMES, "ID=%d, FL=%d,
T=%c : %d:%d:%d.%d\n",
    id, flag, type, (int)systemTime.wHour,
    (int)systemTime.wMinute,
    (int)systemTime.wSecond,
    (long)systemTime.wMilliseconds);

fflush (TIMES);
return 0; }

#endif

.....
tpapl/trnexe/TransactionDataLen.c
.....

/*****
*
* TPC-C Client Application Program Source
*
* Entry Functions
* (1) GetGenericDataLen
* (2) GetDeliveryDataLen
*
* CREATE by TSL 2002.10.01
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2002 *
*****/

```

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "forlinux.h"

#include "tpcc_info.h"

/*****
****
* Get transaction data size.
* Return Value
* transaction data size
*****/

long GetGenericDataLen() {
    long max_len = 0;

    if (max_len < sizeof(NewOrderData)) max_len
= sizeof(NewOrderData);
    if (max_len < sizeof(OrderStatusData))
max_len = sizeof(OrderStatusData);
    if (max_len < sizeof(PaymentData)) max_len
= sizeof(PaymentData);
    if (max_len < sizeof(StockLevelData))
max_len = sizeof(StockLevelData);
    if (max_len < sizeof(DeliveryData)) max_len =
sizeof(DeliveryData);

    return max_len;
}

/*****
****
* Get delivery transaction data size.
*
* Return Value
* Delivery transaction data size
*****/

long GetDeliveryDataLen() {
    return sizeof(struct delstruct);
}

.....
tpapl/trnexe/TrxDelivery.c
.....

/*****
****
*
* TPC-C Client Application Program Source
*
* Entry Functions
* (1) Delivery
*
* CREATE by TSL 2003.12.15
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2003 *
*****/

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "forlinux.h"

#include <sys/time.h>

```

```

#include "atmi.h"

#include "trans.h"
#include "tpcc_info.h"
#include "delpage.h"

#include "ThreadCntl.h"
#include "TpAplDBDependPrototype.h"
#include "log.h"
#include "log_level.h"

#include "tpapl.h"
// #include "GlobalArea.h" // Common
#include "OracleFunction.h"

/*****
-----
Delivery : this function processes the delivery
transaction.
----- */
int Delivery (char *s_buf, RTE_INPUT_DATA
"in_data, int cookie)
{
    DeliveryData *bp;
    char S_WORK[WORK_S];

    struct timeval tv;

#ifdef TRNS_BIND
static char* svr_name = "DELIVERY";
#else
static char* svr_name = "OPSTUXSERVER";
#endif

    int h_del1_len;
    int h_del2_len;
    int h_del3_len;

    THREAD_CNTL_INFO* ThreadCntlInfo;

    //SvrAPL return value
#ifdef SCRTST
int ret_val;
#endif

    MAC_PutFncEntryLog("Delivery");

    /* Create execution environment */
    ThreadCntlInfo = GetThreadCntl();
    if (ThreadCntlInfo == 0) {
        sprintf(S_WORK, "thread control
information is not allocated [DEL]\n");
        MAC_errHTML(s_buf, S_WORK, cookie);
        TpccUserLog (LOG_ERR, S_WORK);
        return (-1);
    }
    bp = (DeliveryData*)ThreadCntlInfo->TrxDData;
    memset(bp, 0x00, sizeof(DeliveryData));

    /* ----- Check
the Input data */
    bp->delin.w_id = MAC_w_id(cookie);

    bp->delin.o_carrier_id = str2short (in_data-
>O_CARRIER_ID, 2);

    if (bp->delin.o_carrier_id < 1 || bp-
>delin.o_carrier_id > 10) {
        TpccUserLog (LOG_ERR, "Input data error
[DEL] (o_carrier_id = %s)[Return_Value:%d]\n",
            in_data->O_CARRIER_ID, bp-
>delin.o_carrier_id);
    }
}

```

```

    return set_errpage(s_buf, cookie, 5, (int)bp-
>delin.o_carrier_id, 0, 0);
}

/* ----- Execute
Delivery transaction */

/* Get Derivery start time */
gettimeofday(&tv, NULL);
bp->delin.startsec = (long)tv.tv_sec;
bp->delin.startusec = (long)(tv.tv_usec /
1000);

#ifdef SCRTEST

/* Replaced 2003.12.15 Transaction processeing
interface COM+ --> TUXEDO */
#ifdef TRNS_BIND
/* Set transaction type for Warehouse bind */
bp->retval = 4;
#endif

resend_delivery:
ret_val = tpacall(svr_name, (char*)bp,
sizeof(NewOrderData), 0 | TPNOTIME |
TPNOREPLY);
if (ret_val == -1) {
/* Display messege */
switch ( tpermo ) {
case TPELIMIT: /* -----
.... */
case TPETIME: /* -----
*/
case TPGOTSIG: /* ----- */
/* Because it is an executable again error,
processing is executed again. */
TpccUserLog (LOG_WRN, "Delivery
retry reason by termno=%d\n", tpermo);
goto resend_delivery;
break;

default:
/* The error which was not able to be
executed again occurred */
sprintf( S_WORK, "tpacall failed in
Delivery: tpermo = %d\n"
" svc = '%s' carrier = %d\n", tpermo,
svr_name, bp->delin.o_carrier_id );

MAC_errHTML_TUXEDO( s_buf,
S_WORK, cookie );
TpccUserLog (LOG_ERR, S_WORK);
FreeTuxBuffer(ThreadCntlInfo);
return (-1);
}
}
#else
dummy_delivery(bp);
#endif

/* ----- The execution result data notified RTE
is made by the HTML form */
/* Replaced T.kato 03.04.18 Speed up */
//sprintf (S_WORK, h_del2);
strcpy(S_WORK, h_del2);
h_del2_leng = strlen(S_WORK);
/* Replaced end */

int2str ((S_WORK + delp[0]), 6, (int)bp-
>delin.w_id);

int2str ((S_WORK + delp[1]), 2, (int)bp-
>delin.o_carrier_id);
alp2str ((S_WORK + delp[2]), 25, "Delivery
has been queued");

```

```

/* Replaced T.Kato 03.04.18 */
#if 0
! sprintf(s_buf, h_del1);
! strcat (s_buf, S_WORK);
!
! sprintf(S_WORK, h_del3, SOPATH, cookie);
! strcat (s_buf, S_WORK);
#endif
strcpy(s_buf, h_del1);
h_del1_leng = strlen(s_buf);
memcpy(s_buf + h_del1_leng, S_WORK,
h_del2_leng);
h_del3_leng = sprintf(S_WORK, h_del3,
SOPATH, cookie);
memcpy(s_buf + h_del1_leng + h_del2_leng,
S_WORK, h_del3_leng);
*(s_buf + h_del1_leng + h_del2_leng +
h_del3_leng) = '\0';
/* Replaced end */

FreeTuxBuffer(ThreadCntlInfo);
return 0;
}

.....:
tpapl/trnexe/TrxNewOrder.c
.....:

/*****
*
* TPC-C Client Application Program Source
*
* Entry Functions *
* (1) NewOrder *
* (2) chk_NOData *
* (3) setNOData *
*
* CREATE by TSL 2002.10.01
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2002 *
*****/
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "forlinux.h"

#include "atmi.h"

#include "trans.h"
#include "tpcc_info.h"
#include "newpage.h"

#include "ThreadCntl.h"
#include "TpAplDBDependPrototype.h"
#include "log.h"
#include "log_level.h"

#include "tpapl.h"
//include "GlobalArea.h" // Common
#include "OracleFunction.h"

/* Added T.Kato 04.05.13 Speed up */
int leng_h_new1 = strlen(h_new1);
int leng_h_new2 = strlen(h_new2);
/* Added end */

```

```

/*-----
-----

NewOrder : this function processes the
NewOrder transaction.

-----*/
int NewOrder (char *s_buf, RTE_INPUT_DATA
*in_data, int cookie)
{
NewOrderData *bp;

/*int user_id, i;*/
int i;
int ol_cnt, cnt, rtn;

char S_WORK[WORK_SZ];

#ifdef TRNS_BIND
static char* svr_name = "NEWORDER";
#else
static char* svr_name = "OPSTUXSERVER";
#endif
long olen;

int h_new1_leng;
int h_new2_leng;
int h_new3_leng;

//SvrAPL return value
#ifdef SCRTEST
int ret_value;
int ret_val;
char* tran_errmsg;
#endif

THREAD_CNTL_INFO* ThreadCntlInfo;
int return_value;

MAC_PutFncEntryLog("NewOrder");

/*user_id = cookie - GLB_TermBase;*/

/* Create execution environment */
ThreadCntlInfo = GetThreadCntl();
if (ThreadCntlInfo == 0) {
sprintf( S_WORK, "thread contorl
information is not allocated [NEW]\n");
MAC_errHTML( s_buf, S_WORK, cookie );
TpccUserLog (LOG_ERR, S_WORK);
return (-1);
}
bp = ( NewOrderData * )ThreadCntlInfo-
>TrxData;
memset(bp, 0x00, sizeof(NewOrderData));

/* ----- check
the Input data */
bp->newin.w_id = MAC_w_id(cookie);

if((bp->newin.d_id = str2int (in_data->D_ID,
2)) < 1 ) {
TpccUserLog (LOG_ERR, "Input data error
[NEW] (d_id = %s)[Retuen_value:%d]\n",
in_data->D_ID, bp-
>newin.d_id);
FreeTuxBuffer(ThreadCntlInfo);
return set_errpage(s_buf, cookie, 2, (int)bp-
>newin.d_id, 0, 0);
}

if((bp->newin.c_id = str2int (in_data->C_ID,
4)) < 0 ) {

```

```

    TpcUserLog (LOG_ERR, "Input data error
[NEW] (c_id = %s)[Return_value:%d]\n",
    in_data->C_ID, bp-
>newin.c_id);
    FreeTuxBuffer(ThreadCntlInfo);
    return set_errpage(s_buf, cookie, 6, bp-
>newin.c_id, 0, 0);
}

    ol_cnt = 0;
    for (cnt = 0; cnt < 15; cnt++){

        if ((rtn = chk_NOdata( bp, cnt, in_data,
ol_cnt)) < 0){
            TpcUserLog (LOG_ERR, "Error end
chk_NOdata() [NEW]
(Line:%d)[Return_Value:%d]\n",
                cnt, rtn);
            FreeTuxBuffer(ThreadCntlInfo);
            return set_errpage(s_buf, cookie, 13 +
cnt, rtn, 0, 0);
        }
        else if (rtn == 1){
            ol_cnt++;
        }
    }

    /* nothing order line data */
    if ( cnt >= 15 && ol_cnt == 0 ) {
        TpcUserLog (LOG_ERR, "nothing order
line data [NEW]\n");
        FreeTuxBuffer(ThreadCntlInfo);
        return set_errpage(s_buf, cookie, 13, -8, 0,
0);
    }

    /* if ol_cnt < 15 then the last order line set
NULL */
    if ( ol_cnt < 15 ){
        bp->newin.ol_i_id[ol_cnt] = 0;
        bp->newin.ol_quantity[ol_cnt] = 0;
        bp->newin.ol_supply_w_id[ol_cnt] = 0;
    }

    bp->newout.o_ol_cnt = ol_cnt;

    /* ----- Execute
NewOrder transaction */
    #ifndef SCRTST
    resend_neworder;

    /* Replaced 2003.12.15 Transaction processeing
interface COM+ --> TUXEDO */
    #ifndef TRNS_BIND
        /* Set transaction type for Warehouse bind */
        bp->retval = 1;
    #endif

        ret_val = tpcall(svr_name,
(char*)ThreadCntlInfo->TrxData,
sizeof(NewOrderData),
                (char*)&ThreadCntlInfo-
>TrxData, &olen, 0)TPNOTIME);
        bp = ( NewOrderData *)ThreadCntlInfo-
>TrxData;
        ret_value = CreateTranErrReason(ret_val, bp-
>newout.terror, &tran_errmsg);

        switch(ret_value) {
        case 0:
            /* Success */
            break;

        case 1:

```

```

/* Retry NewOrder transaction */
    TpcUserLog (LOG_WRN, "NewOrder
retry\n");
    goto resend_neworder;

    case -1:
        /* Oracle failed */
        sprintf( S_WORK, "Oracle failed to process
NewOrder Transaction.(%s)\n"
                "ret_value = %d d_id = %d c_id = %d
lines = %d cookie = %d\n",
                tran_errmsg, ret_value,
                bp->newin.d_id, bp->newin.c_id,
ol_cnt, cookie );

        MAC_errHTML( s_buf, S_WORK, cookie );
        TpcUserLog (LOG_ERR, S_WORK);
        FreeTuxBuffer(ThreadCntlInfo);
        return (-1);

    default:
        /* Tuxedo failed */
        sprintf( S_WORK, "tpcall failed to process
NewOrder Transaction.(tperno=%d)\n"
                "ret_value = %d d_id = %d c_id = %d
lines = %d cookie = %d\n",
                tperno, ret_value,
                bp->newin.d_id, bp->newin.c_id,
ol_cnt, cookie );

        MAC_errHTML_TUXEDO( s_buf, S_WORK,
cookie );
        TpcUserLog (LOG_ERR, S_WORK);
        FreeTuxBuffer(ThreadCntlInfo);
        return (-1);
    }
}
/* Changed end */
#else
    dummy_neworder( bp );
#endif

/* Replaced T.Kato 04.05.13 Speed up */
#if 0
/* Replaced T.Kato 03.04.18 Speed up */
! //sprintf (S_WORK, h_new2);
! strcpy(S_WORK, h_new2);
! h_new2_leng = strlen(S_WORK);
/* Replaced end */
#endif

    strcpy(S_WORK, h_new2);
    h_new2_leng = leng_h_new2;
/* Replaced end */

    int2str ((S_WORK + newp[0]), 6, (int)bp-
>newin.w_id);

    int2str ((S_WORK + newp[1]), 2, (int)bp-
>newin.d_id);
    int2str ((S_WORK + newp[3]), 4, bp-
>newin.c_id);

    alp2str ((S_WORK + newp[4]), 16, bp-
>newout.c_last);
    alp2str ((S_WORK + newp[5]), 2, bp-
>newout.c_credit);
    int2str ((S_WORK + newp[7]), 8, (int)bp-
>newout.o_id);

    cnt = bp->newout.o_ol_cnt;

    time2str((S_WORK + newp[2]),bp-
>newout.o_entry_d);
    dec2str ((S_WORK + newp[6]),5,(double)(bp-
>newout.c_discount*100.0));

```

```

    int2str ((S_WORK + newp[8]),2,(int)bp-
>newout.o_ol_cnt);
    dec2str ((S_WORK + newp[9]),5, (double)(bp-
>newout.w_tax * 100.0));
    dec2str ((S_WORK +
newp[10]),5,(double)(bp->newout.d_tax *
100.0));

    for ( i = 0; i < cnt; i++ ) {
        return_value = setNOdata (S_WORK,
0x50*i, i, bp, in_data);
        if (return_value != 0) {
            TpcUserLog (LOG_ERR, "Error end
setNOdata() [NEW]
(Line:%d)[Return_Value:%d]\n",
                i, return_value);
        }
    }

    /* "Item number is not valid" or "" ('0) */
    // Oracle Web Server use
    if (strcmp(bp->newout.status, "I") > 0)
        alp2str ((S_WORK + newp[19]), 24, bp-
>newout.status);

    dec2str ((S_WORK + newp[20]), 8,
(double)(bp->newout.total_amount)); // check

    /* ----- The execution result data notified RTE
is make by the HTML form */

    /* Replaced T.Kato 04.05.13 Speed up */
    #if 0
    /* Replaced T.Kato 03.04.18 Speed up */
    #if 0
    !! sprintf(s_buf, h_new1);
    !! strcat (s_buf, S_WORK);
    !!
    !! sprintf(S_WORK, h_new3, SOPATH,
cookie);
    !! strcat (s_buf, S_WORK);
    #endif
    ! strcpy(s_buf, h_new1);
    ! h_new1_leng = strlen(s_buf);
    ! memcpy(s_buf + h_new1_leng, S_WORK,
h_new2_leng);
    ! h_new3_leng = sprintf(S_WORK, h_new3,
SOPATH, cookie);
    ! memcpy(s_buf + h_new1_leng +
h_new2_leng, S_WORK, h_new3_leng);
    ! *(s_buf + h_new1_leng + h_new2_leng +
h_new3_leng) = '\0';
    /* Replaced end */
    #endif
    strcpy(s_buf, h_new1);
    h_new1_leng = leng_h_new1;
    memcpy(s_buf + h_new1_leng, S_WORK,
h_new2_leng);
    h_new3_leng = sprintf(S_WORK, h_new3,
SOPATH, cookie);
    memcpy(s_buf + h_new1_leng +
h_new2_leng, S_WORK, h_new3_leng);
    *(s_buf + h_new1_leng + h_new2_leng +
h_new3_leng) = '\0';
    /* Replaced end */

    FreeTuxBuffer(ThreadCntlInfo);
    return (0);
}

#define SUPPLY_NG 0x01
#define I_ID_NG 0x02
#define QUANTITY_NG 0x04

```

```

/* -----
-----
chk_NOdata :
VerifyNewOrderLine verifies that a user's
inputs for a line in
the New Order form are okay.
return -5 : w_id abnormal value : Not
Number
return -6 : i_id abnormal value : Not
Number
return -7 : ol_quantity abnormal value : Not
Number

98.8.3 : ----- (-15, -16, -17----:
outside range )

-----
----- */
int chk_NOdata (NewOrderData *bp, int cnt,
RTE_INPUT_DATA *in_data, int svcnt)
{

char flag = 0;

if( in_data->OL_SUPPLY_W_ID[cnt] == 0 &&
in_data->OL_I_ID[cnt] == 0 &&
in_data->OL_QUANTITY[cnt] == 0 ){
/* Order line nothing : 1----- */
return 16; /* change return code */
}

if( in_data->OL_SUPPLY_W_ID[cnt] != 0 ){
if((bp->newin.ol_supply_w_id[svcnt] =
str2int( in_data->OL_SUPPLY_W_ID[cnt],
6) < 1 )
return -5; /* w_id abnormal */
}
else {
flag |= SUPPLY_NG;
}

if( in_data->OL_I_ID[cnt] != 0 ){

if((bp->newin.ol_i_id[svcnt] =
str2int( in_data->OL_I_ID[cnt], 6) < 0 )
return -6; /* i_id abnormal value
*/

/* sv-apl ----- 99.12.20 */
else if (bp->newin.ol_i_id[svcnt] == 0)
bp->newin.ol_i_id[svcnt] = -1;
}
else{
flag |= I_ID_NG;
}

if( in_data->OL_QUANTITY[cnt] != 0 ){
if(((bp->newin.ol_quantity[svcnt] =
str2int( in_data->OL_QUANTITY[cnt], 2))
< 1) ||
bp->newin.ol_quantity[svcnt] > 10 ){

if ( bp->newin.ol_quantity[svcnt] < 0 )
return -7; /* ol_quantity
abnormal value */
else
return -17; /* outside range */
}
}
else{
flag |= QUANTITY_NG;
}

if (flag != 0){

```

```

/* the order lien data is abnormal : there is a
uninput item */
if((flag & SUPPLY_NG) != 0 ) return -8;
if((flag & I_ID_NG) != 0 ) return -1;
if((flag & QUANTITY_NG) != 0) return -2;
return 1;
}
else{
/* the order lien data is normal */
return 1;
}
}

/* -----
-----
setNOdata : This function set the execution
result data of the TP
applicatin program.

OF is an offset value to the next line data.
cnt is line number

-----
----- */
int setNOdata (char *s_work,int OF,int cnt,
NewOrderData *bp,RTE_INPUT_DATA
*in_data)
{
/*for warning
in_data;

if((bp->newin.ol_i_id[cnt] ) {
alp2str ((s_work + OF + newp[11]), 78, " ");
return -1;
}
else {
int2str((s_work + OF + newp[11]), 6, (int)bp-
>newin.ol_supply_w_id[cnt]);

if (bp->newin.ol_i_id[cnt] == -1 )
bp->newin.ol_i_id[cnt] = 0;
int2str((s_work + OF + newp[12]), 6, bp-
>newin.ol_i_id[cnt]);

alp2str((s_work + OF + newp[13]), 24, bp-
>newout.i_name[cnt]);

int2str((s_work + OF + newp[14]), 2, (int)bp-
>newin.ol_quantity[cnt]);
int2str((s_work + OF + newp[15]), 3, (int)bp-
>newout.s_quantity[cnt]);
alp2str((s_work + OF + newp[16]), 1, &bp-
>newout.brand_generic[cnt]);

dec2str((s_work + OF + newp[17]),
6,(double)bp->newout.i_price[cnt]); // check
dec2str((s_work + OF + newp[18]),
7,(double)bp->newout.ol_amount[cnt]); // check
return 0;
}
}

.....
tpapl/trnexe/TrxOrderStatus.c
.....

/*****
****
*
* TPC-C Client Application Program Source
*
*
* Entry Functions
*

```

```

* (1) OrderStatus
*
* CREATE by TSL 2003.12.15
*
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2002 *

-----
****/
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "forlinux.h"

#include "atmi.h"

#include "trans.h"
#include "tpcc_info.h"
#include "odpage.h"

#include "ThreadCntl.h"
#include "TpAplDBDependPrototype.h"
#include "log.h"
#include "log_level.h"

#include "tpapl.h"
// #include "GlobalArea.h" // Common
#include "OracleInfo.h"
#include "OracleFunction.h"

/* Added T.Kato 04.05.13 Speed up */
int leng_h_order1 = strlen(h_order1);
int leng_h_order2 = strlen(h_order2);
/* Added end */

/*-----
-----
OrderStatus : this function processes the
Orderstatus transaction

-----
----- */
int OrderStatus (char *s_buf,
RTE_INPUT_DATA *in_data, int cookie)
{
OrderStatusData *bp;
int i, rtn;

char S_WORK[WORK_S];
char c_id_flag = NG;

#ifdef TRNS_BIND
static char* svr_name = "ORDERSTATUS";
#else
static char* svr_name = "OPSTUXSERVER";
#endif

long olen;

int h_order1_leng;
int h_order2_leng;
int h_order3_leng;

//SvrAPL return value
#ifdef SCRTEST
int ret_value;
int ret_val;
char* tran_errmsg;
#endif

THREAD_CNTL_INFO* ThreadCntlInfo;

MAC_PutFuncEntryLog("OrderStatus");

ThreadCntlInfo = GetThreadCntl();

```

```

if (ThreadCntlInfo == 0) {
    sprintf( S_WORK, "thread contorl
information is not allocated [ODR]\n");
    MAC_errHTML( s_buf, S_WORK, cookie );
    TpcUserLog (LOG_ERR, S_WORK);
    return (-1);
}
bp = ( OrderStatusData * )ThreadCntlInfo-
>TrxData;
memset(bp, 0x00, sizeof(OrderStatusData));

/* ----- check
the Input data */
bp->ordin.w_id = MAC_w_id(cookie);

/* check d_id data */
if ((bp->ordin.d_id = str2short (in_data->D_ID,
2)) < 1) {
    TpcUserLog (LOG_ERR, "Input data error
[ORD] (d_id = %s)[Return_Value:%d]\n",
in_data->D_ID, bp->ordin.d_id);
    FreeTuxBuffer(ThreadCntlInfo);
    return set_errpage(s_buf, cookie, 2, (int)bp-
>ordin.d_id, 0, 0);
}

if ((bp->ordin.c_id = str2int(in_data->C_ID,
4)) != -3){

    if (bp->ordin.c_id < 0) {
        TpcUserLog (LOG_ERR, "Input data
error [ORD] (c_id = %s)[Return_Value:%d]\n",
in_data->C_ID, bp-
>ordin.c_id);
        FreeTuxBuffer(ThreadCntlInfo);
        return set_errpage(s_buf, cookie, 6, bp-
>ordin.c_id, 0, 0);
    }
    else{
        c_id_flag = OK;
    }
}
else{
    bp->ordin.c_id = 0;
}

/* check c_last data */
if((rtn = str2str(in_data->C_LAST, 16)) < 0){
    c_id_flag = OK;
}
else{
    if ( rtn == 0 || *(in_data->C_LAST) == '\0' ) {
        bp->ordin.bylastname = 0; /* Oracle
use only */
        bp->ordin.c_last[0] = '\0';
    }
    else {
        strcpy (bp->ordin.c_last, in_data-
>C_LAST);
        bp->ordin.bylastname = 1; /* Oracle
use only */
        c_id_flag = OK;
    }
}

/* c_id and c_last is nothing */
if (c_id_flag == NG) {
    TpcUserLog (LOG_ERR, "c_id and c_last
is nothing [ORD]\n");
    FreeTuxBuffer(ThreadCntlInfo);
    return set_errpage(s_buf, cookie, 11, -4, 0,
0);
}

```

```

}

/* ----- Execute
Orderstatus transaction */
#ifdef SCRTST
resend_orderstatus:

/* Replaced 2003.12.15 Transaction
processeing interface COM+ --> TUXEDO */
#ifdef TRNS_BIND
/* Set transaction type for Warehouse bind */
bp->retval = 3;
#endif

    retval = tpcall(svr_name,
(char*)ThreadCntlInfo->TrxData,
sizeof(NewOrderData),
(char*)&ThreadCntlInfo-
>TrxData, &olen, 0|TPNOTIME);
    bp = ( OrderStatusData * )ThreadCntlInfo-
>TrxData;
    ret_value = CreateTranErrReason(retval, bp-
>ordout.terror, &tran_errmsg);

    switch(retval) {
    case 0:
        /* Success */
        break;

    case 1:
        /* Retry OrderStatus transaction */
        TpcUserLog (LOG_WRN, "OrderStatus
retry\n");
        goto resend_orderstatus;

    case -1:
        /* Oracle failed */
        sprintf( S_WORK, "Oracle failed to
process Order Status Transaction.(%s)\n"
"ret_value = %d d_id = %d c_id = %d
c_last = %s' cookie = %d\n",
tran_errmsg, ret_value, bp-
>ordin.d_id, bp->ordin.c_id,
bp->ordin.c_last, cookie );

        MAC_errHTML( s_buf, S_WORK, cookie );
        TpcUserLog (LOG_ERR, S_WORK);
        FreeTuxBuffer(ThreadCntlInfo);
        return (-1);

    default:
        /* Tuxedo failed */
        sprintf( S_WORK, "tpcall failed to process
NewOrder Transaction.(tperno=%d)\n"
"ret_value = %d d_id = %d c_id = %d
c_last = %s' cookie = %d\n",
tperno, ret_value, bp->ordin.d_id, bp-
>ordin.c_id,
bp->ordin.c_last, cookie );

        MAC_errHTML_TUXEDO( s_buf, S_WORK,
cookie );
        TpcUserLog (LOG_ERR, S_WORK);
        FreeTuxBuffer(ThreadCntlInfo);
        return (-1);
    }
}
/* Changed end */

/* ----- Check the
execution result */

#else
    dummy_orderstat( bp );
#endif

```

```

/* Replaced T.Kato 04.05.13 Speed up */
#if 0
/* Replaced T.Kato 03.04.18 Speed up */
! //sprintf(S_WORK, h_order2);
! strcpy(S_WORK, h_order2);
! h_order2_leng = strlen(S_WORK);
/* Replaced end */
#endif
    strcpy(S_WORK, h_order2);
    h_order2_leng = leng_h_order2;
/* Relaced end */

    int2str ((S_WORK + orderp[0]), 6, (int)bp-
>ordin.w_id);
    int2str ((S_WORK + orderp[1]), 2, (int)bp-
>ordin.d_id);
    int2str ((S_WORK + orderp[2]), 4, bp-
>ordout.c_id);
    alp2str ((S_WORK + orderp[3]), 16, bp-
>ordout.c_first);
    alp2str ((S_WORK + orderp[4]), 2, bp-
>ordout.c_middle);
    alp2str ((S_WORK + orderp[5]), 16, bp-
>ordout.c_last);
    sigdec2str ((S_WORK + orderp[6]), 9, bp-
>ordout.c_balance);
    int2str ((S_WORK + orderp[7]), 8, (int)bp-
>ordout.o_id );
    time2str ((S_WORK + orderp[8]), bp-
>ordout.o_entry_d );

    if ( bp->ordout.o_carrier_id != INTNULL ) {
        int2str ((S_WORK + orderp[9]), 2, bp-
>ordout.o_carrier_id);
    }

    /* 0x39 is an offset value to the same filed of
the next line */
    for ( i = 0; i < bp->ordout.o_ol_cnt; i++ ){

        int2str ((S_WORK+i*0x3a+orderp[10]), 6,
(int)bp->ordout.ol_supply_w_id[i]);

        int2str ((S_WORK+i*0x3a+orderp[11]), 6,
(int)bp->ordout.ol_i_id[i]);
        int2str ((S_WORK+i*0x3a+orderp[12]), 2,
(int)bp->ordout.ol_quantity[i]);
        sigdec2str ((S_WORK+i*0x3a+orderp[13]),
8, (double)bp->ordout.ol_amount[i]);

        if( strcmp( bp->ordout.ol_delivery_d[i],
"NOT DELIVR", 10) != 0 ){

            date2str ((S_WORK+i*0x3a+orderp[14]),
bp->ordout.ol_delivery_d[i]);
        }
    }

    /* ----- The execution result data notified RTE
is make by the HTML form */
/* Replaced T.Kato 04.05.13 Speed up */
#if 0
/* Replaced T.Kato 03.04.18 Speed up */
! #if 0
!! sprintf(s_buf, h_order1); /* set Header Data
*/
!! strcat (s_buf, S_WORK); /* set Result
Data */
!!
!! sprintf (S_WORK, h_order3, SOPATH,
cookie); /* set Tailer Data */
!! strcat (s_buf, S_WORK);
! #endif
! strcpy(s_buf, h_order1);
! h_order1_leng = strlen(s_buf);

```



```
! memcpy(s_buf + h_order1_leng, S_WORK,
h_order2_leng);
! h_order3_leng = sprintf (S_WORK, h_order3,
SOPATH, cookie);
! memcpy(s_buf + h_order1_leng +
h_order2_leng, S_WORK, h_order3_leng);
! *(s_buf + h_order1_leng + h_order2_leng +
h_order3_leng) = '\0';
!/* Replaced end */
#endif
strcpy(s_buf, h_order1);
h_order1_leng = leng_h_order1;
memcpy(s_buf + h_order1_leng, S_WORK,
h_order2_leng);
h_order3_leng = sprintf (S_WORK, h_order3,
SOPATH, cookie);
memcpy(s_buf + h_order1_leng +
h_order2_leng, S_WORK, h_order3_leng);
*(s_buf + h_order1_leng + h_order2_leng +
h_order3_leng) = '\0';
!/* Replaced end */
```

```
FreeTuxBuffer(ThreadCntlInfo);
return 0;
}
```

```
.....:
tpapl/trnexe/TrxPayment.c
.....:
```

```
/*.....:
*****
*
* TPC-C Client Application Program Source
*
*
* Entry Functions *
* (1) Payment *
*
* CREATE by TSL 2003.12.15
*
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2003 *
*****
****/
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "forlinux.h"

#include "atmi.h"

#include "trans.h"
#include "tpcc_info.h"
#include "paypage.h"

#include "ThreadCntl.h"
#include "TpAplDBDependPrototype.h"
#include "log.h"
#include "log_level.h"

#include "tpapl.h"
#include "GlobalArea.h" // Common
#include "OracleFunction.h"

!/* Added T.Kato 04.03.10 Speed up */
#define SP1_DATA " "
#define SP2_DATA " "
#define SP3_DATA " "
#define CREDIT_DATA " Credit:
"
#define DISC_DATA " %Disc: "
```

```
int leng_h_pay1 = strlen(h_pay1);
int leng_h_pay2 = strlen(h_pay2);
int leng_h_pay4 = strlen(h_pay4);
int leng_h_pay5 = strlen(h_pay5);
int leng_sp1_data = strlen(SP1_DATA);
int leng_sp2_data = strlen(SP2_DATA);
int leng_sp3_data = strlen(SP3_DATA);
int leng_credit_data =
strlen(CREDIT_DATA);
int leng_disc_data = strlen(DISC_DATA);
!/* Added end */
```

```
!-----:
-----:
```

Payment : this function processes the Payment transaction.

```
-----*/
```

```
int Payment (char *s_buf, RTE_INPUT_DATA
*in_data, int cookie)
{
PaymentData *bp;
int i, rtn;

float h_amount; /* For work */

char c_id_flag = NG;
char S_WORK[WORK_SZ];

char buffer[128]; /* check HTML form */
char buffer2[128];
char buffer3[512];
int newlength;
```

```
#ifdef TRNS_BIND
static char* svr_name = "PAYMENT";
#else
static char* svr_name = "OPSTUXSERVER";
#endif
long olen;

//SvrAPL return value
#ifndef SCRTEST
int ret_value;
int ret_val;
char* tran_errmsg;
#endif
```

```
THREAD_CNTL_INFO* ThreadCntlInfo;
#ifndef SCRTEST
#endif

!/* Added T.Kato 04.03.10 */
int next_pos;
int swork_pos;
!/* Added end */
```

```
MAC_PutFncEntryLog("Payment");

ThreadCntlInfo = GetThreadCntl();
if (ThreadCntlInfo == 0) {
sprintf(S_WORK, "thread contorl
information is not allocated [PAY]\n");
MAC_errHTML(s_buf, S_WORK, cookie);
TpccUserLog (LOG_ERR, S_WORK);
return (-1);
}
bp = ( PaymentData *)ThreadCntlInfo-
>TrxDat;
memset(bp, 0x00, sizeof(PaymentData));
```

```
!/* -----: check
the Input data */
bp->payin.w_id = MAC_w_id(cookie);
```

```
!/* check d_id data */
if((bp->payin.d_id = str2short (in_data->D_ID,
2)) < 1 ) {
TpccUserLog (LOG_ERR, "Input data error
[PAY] (d_id = %s)[Return_Value:%d]\n",
in_data->D_ID, bp->payin.d_id);
FreeTuxBuffer(ThreadCntlInfo);
return set_errpage(s_buf, cookie, 2, (int)bp-
>payin.d_id, 0, 0);
}
```

```
!/* check c_id data */
if((bp->payin.c_id = str2int (in_data->C_ID,
4)) != -3){
```

```
if (bp->payin.c_id < 0) {
TpccUserLog (LOG_ERR, "Input data
error [PAY] (c_id = %s)[Return_Value:%d]\n",
in_data->C_ID, bp-
>payin.c_id);
FreeTuxBuffer(ThreadCntlInfo);
return set_errpage(s_buf, cookie, 6, bp-
>payin.c_id, 0, 0);
}
else{
c_id_flag = OK;
}
}
else{
bp->payin.c_id = 0;
}
```

```
!/* check c_last data */
if((rtn = str2str(in_data->C_LAST, 16)) < 0){
c_id_flag = OK;
}
else{
```

```
if (rtn == 0 || *(in_data->C_LAST) == '\0') {
bp->payin.bylastname = 0; /*
Oracle use only */
bp->payin.c_last[0] = '\0';
} else {
strcpy (bp->payin.c_last, in_data-
>C_LAST);
bp->payin.bylastname = 1; /*
Oracle use only */
c_id_flag = OK;
}
}
```

```
!/* c_id and c_last data is nothing */
if (c_id_flag == NG) {
TpccUserLog (LOG_ERR, "c_id and c_last
data is nothing [PAY]\n");
FreeTuxBuffer(ThreadCntlInfo);
return set_errpage(s_buf, cookie, 11, -4, 0,
0);
}
```

```
!/* check c_w_id data */
!/* Replaced T.Kato 03.08.20 Bug fix --effect
floating point-- */
!/* if((bp->payin.c_w_id = str2dbl (in_data-
>C_W_ID, 5) / 100) < 1) {*/
```

```
if((bp->payin.c_w_id = str2int (in_data-
>C_W_ID, 6)) < 1) {
!/* Replaced end */
```

```

    TpccUserLog (LOG_ERR, "Input data error
[PAY] (c_w_id = %s)[Return_Value:%d]\n",
    in_data->C_W_ID, bp-
>payin.c_w_id);
    FreeTuxBuffer(ThreadCntlInfo);
    return set_errpage(s_buf, cookie, 9, (int)bp-
>payin.c_w_id, 0, GLB_Numwh);
}

/* check c_d_id data */
if((bp->payin.c_d_id = str2short (in_data-
>C_D_ID, 2)) < 1 ) {
    TpccUserLog (LOG_ERR, "Input data error
[PAY] (c_d_id = %s)[Return_Value:%d]\n",
    in_data->C_D_ID, bp-
>payin.c_d_id);
    FreeTuxBuffer(ThreadCntlInfo);
    return set_errpage(s_buf, cookie, 10,
(int)bp->payin.c_d_id, 0, 0);
}

if((bp->payin.h_amount = (long)str2dbl
(in_data->H_AMOUNT, 7)) < 100 ||
    bp->payin.h_amount > 500000) {
    TpccUserLog (LOG_ERR, "Input data error
[PAY] (h_amount = %s)[Return_Value:%d]\n",
    in_data->H_AMOUNT, bp-
>payin.h_amount);
    FreeTuxBuffer(ThreadCntlInfo);
    return set_errpage(s_buf, cookie, 8, (int)bp-
>payin.h_amount, 0, 0);
}

/* ----- Execute
Payment transaction */
#ifdef SCRTTEST
resend_payment:

/* Replaced 2003.12.15 Transaction processeing
interface COM+ --> TUXEDO */
#ifdef TRNS_BIND
/* Set transaction type for Warehouse bind */
    bp->retval = 2;
#endif

    ret_val = tpcall(svr_name,
(char*)ThreadCntlInfo->TrxDData,
sizeof(NewOrderData),
(char*)&ThreadCntlInfo-
>TrxDData, &olen, 0|TPNOTIME);
    bp = (PaymentData *)ThreadCntlInfo-
>TrxDData;
    ret_value = CreateTranErrReason(ret_val, bp-
>payout.terror, &tran_errmsg);

    switch(ret_value) {
    case 0:
        /* Success */
        break;

    case 1:
        /* Retry Payment transaction */
        TpccUserLog (LOG_WRN, "Payment
retry\n");
        goto resend_payment;

    case -1:
        /* Oracle failed */
        sprintf (S_WORK, "Oracle failed to
process Payment Transaction.(%s)\n"
            "ret_value = %d d_id = %d c_id = %d
c_last = %s\n"
            "c_w_id = %d, c_d_id = %d, h_amount
= %d cookie = %d\n",

```

```

    tran_errmsg, ret_value,
    bp->payin.d_id, bp->payin.c_id, bp-
>payin.c_last,
    bp->payin.c_w_id, bp->payin.c_d_id,
    bp->payin.h_amount, cookie );

    MAC_errHTML( s_buf, S_WORK, cookie );
    TpccUserLog (LOG_ERR, S_WORK);
    FreeTuxBuffer(ThreadCntlInfo);
    return (-1);

    default:
        /* Tuxedo failed */
        sprintf( S_WORK, "tpcall failed to process
NewOrder Transaction.(tperno=%d)\n"
            "ret_value = %d d_id = %d c_id = %d
c_last = %s\n"
            "c_w_id = %d, c_d_id = %d, h_amount
= %d cookie = %d\n",
            tperno, ret_value,
            bp->payin.d_id, bp->payin.c_id, bp-
>payin.c_last,
            bp->payin.c_w_id, bp->payin.c_d_id,
            bp->payin.h_amount, cookie );
        MAC_errHTML_TUXEDO( s_buf, S_WORK,
cookie );
        TpccUserLog (LOG_ERR, S_WORK);
        FreeTuxBuffer(ThreadCntlInfo);
        return (-1);
    }
/* Changed end */

#else
    dummy_payment( bp );
#endif

/* Replaced T.Kato 04.03.10 Speed up */
#if 0
/* Replaced T.Kato 03.04.18 Speed up */
! //sprintf (S_WORK, h_pay2);
! strcpy(S_WORK, h_pay2);
/* Replaced end */
#endif

    memcpy(S_WORK, h_pay2, leng_h_pay2+1);
    swork_pos = leng_h_pay2;
/* Replaced end */

    time2str ((S_WORK + payp[0]), bp-
>payout.h_date );
    int2str ((S_WORK + payp[1]), 6, (int)bp-
>payin.w_id);
    int2str ((S_WORK + payp[2]), 2, (int)bp-
>payin.d_id);

    // check HTML form

    alp2str (&buffer2[0], 20, bp-
>payout.w_street_1);
    buffer2[20] = 0;

/* Replaced T.kato 04.03.10 Speed up */
#if 0
! newlength = checkHTMLform ( &buffer2[0],
&buffer[0]);
! strcpy (&buffer3[0], &buffer[0]);
! strcat (buffer3, " ");
#endif

    newlength = checkHTMLform ( buffer2,
buffer3);
    memcpy(buffer3+newlength, SP1_DATA,
leng_sp1_data+1);
    next_pos = newlength + leng_sp1_data;
/* Replaced end */

```

```

    alp2str (buffer2, 20, bp->payout.d_street_1);
    buffer2[20] = 0;
    newlength = checkHTMLform ( &buffer2[0],
&buffer[0]);

/* Replaced T.Kato 04.03.10 Speed up */
#if 0
! strcat (buffer3, &buffer[0]);
! strcat (buffer3, "\r\n");
#endif

    memcpy(buffer3+next_pos, buffer,
newlength+1);
    next_pos += newlength;
    memcpy(buffer3+next_pos, "\r\n", 2+1);
    next_pos += 2;
/* Replaced end */

    alp2str (buffer2, 20, bp->payout.w_street_2);
    buffer2[20] = 0;
    newlength = checkHTMLform ( &buffer2[0],
&buffer[0]);

/* Replaced T.Kato 04.03.10 Speed up */
#if 0
! strcat (buffer3, &buffer[0]);
! strcat (buffer3, " ");
#endif

    memcpy(buffer3+next_pos, buffer,
newlength+1);
    next_pos += newlength;
    memcpy(buffer3+newlength, SP1_DATA,
leng_sp1_data+1);
    next_pos = newlength + leng_sp1_data;
/* Replaced end */

    alp2str (buffer2, 20, bp->payout.d_street_2);
    buffer2[20] = 0;
    newlength = checkHTMLform ( &buffer2[0],
&buffer[0]);

/* Replaced T.Kato 04.03.10 Speed up */
#if 0
! strcat (buffer3, &buffer[0]);
! strcat ( S_WORK, buffer3 );
#endif

    memcpy(buffer3+next_pos, buffer,
newlength+1);
    next_pos += newlength;
    memcpy(buffer3+next_pos, "\r\n", 2+1);
    next_pos += 2;

    memcpy(S_WORK+swork_pos, buffer3,
next_pos+1);
    swork_pos += next_pos;
/* Replaced end */

    // check HTML form
/* Replaced T.Kato 04.03.10 Speed up */
#if 0
/* Replaed T.Kato 03.04.18 Speed up */
! //sprintf ( buffer3, h_pay4 );
! strcpy ( buffer3, h_pay4 );
/* Replaced end */
#endif

    memcpy(buffer3, h_pay4, leng_h_pay4+1);
/* Replaced end */

    alp2str ((&buffer3[0] + payp[7] - 0xd3), 20, bp-
>payout.w_city);

```

```

    alp2str ((&buffer3[0] + payp[8] - 0xd3), 2, bp-
->payout.w_state);
    zip2str ((&buffer3[0] + payp[9] - 0xd3), bp-
->payout.w_zip);
    alp2str ((&buffer3[0] + payp[11] - 0xd3), 20,
bp->payout.d_city);
    alp2str ((&buffer3[0] + payp[12] - 0xd3), 2, bp-
->payout.d_state);
    zip2str ((&buffer3[0] + payp[13] - 0xd3), bp-
->payout.d_zip);

    int2str ((&buffer3[0] + payp[15] - 0xd3), 4, bp-
->payout.c_id);
    int2str ((&buffer3[0] + payp[16] - 0xd3), 6,
(int)bp->payin.c_w_id);
    int2str ((&buffer3[0] + payp[17] - 0xd3), 2,
(int)bp->payin.c_d_id);

    alp2str ((&buffer3[0] + payp[18] - 0xd3), 16,
bp->payout.c_first);
    alp2str ((&buffer3[0] + payp[19] - 0xd3), 2, bp-
->payout.c_middle);
    alp2str ((&buffer3[0] + payp[20] - 0xd3), 16,
bp->payout.c_last);

    date2str ((&buffer3[0] + payp[21] - 0xd3), bp-
->payout.c_since);

/* Replaced T.Kato 04.03.10 Speed up */
/* strcat (S_WORK, buffer3);*/

    memcpy(S_WORK+swork_pos, buffer3,
leng_h_pay4+1);
    swork_pos += leng_h_pay4;
/* Replaced end */

/* Replaced T.Kato 04.03.10 Speed up*/
/* strcpy (&buffer3[0], " ");*/

    memcpy(buffer3, SP2_DATA,
leng_sp2_data+1);
    next_pos = leng_sp2_data;
/* Replaced end */

    alp2str (buffer2, 20, bp->payout.c_street_1);
    buffer2[20] = 0;
    newlength = checkHTMLform ( &buffer2[0],
&buffer[0]);

/* Replaced T.Kato 04.03.10 */
#if 0
!   strcat (buffer3, &buffer[0]);
!   strcat (buffer3, "          Credit: ");
#endif

    memcpy(buffer3+next_pos, buffer,
newlength+1);
    next_pos += newlength;
    memcpy(buffer3+next_pos, CREDIT_DATA,
leng_credit_data+1);
    next_pos += leng_credit_data;
/* Replaced end */

    alp2str (buffer2, 2, bp->payout.c_credit);
    buffer2[2] = 0;

/* Replaced T.Kato 04.03.10 */
#if 0
!   strcat (buffer3, &buffer2[0]);
!   strcat (buffer3, "\r\n");
#endif

    memcpy(buffer3+next_pos, buffer2, 2);
    memcpy(buffer3+next_pos+2, "\r\n", 3);
    next_pos += 4;

/* Replaced end */

/* Replaced T.Kato 40.03.10 */
/* strcat (buffer3, " ");*/

    memcpy(buffer3+next_pos, SP2_DATA,
leng_sp2_data+1);
    next_pos += leng_sp2_data;
/* Replaced end */

    alp2str (buffer2, 20, bp->payout.c_street_2);
    buffer2[20] = 0;
    newlength = checkHTMLform ( &buffer2[0],
&buffer[0]);

/* Replaced T.Kato 04.03.10 */
#if 0
!   strcat (buffer3, &buffer[0]);
!   strcat (buffer3, "          %Disc: ");
!   strcat (S_WORK, buffer3);
#endif

    memcpy(buffer3+next_pos, buffer,
newlength+1);
    next_pos += newlength;
    memcpy(buffer3+next_pos, DISC_DATA,
leng_disc_data+1);
    next_pos += leng_disc_data;

    memcpy(S_WORK+swork_pos, buffer3,
next_pos+1);
    swork_pos += next_pos;
/* Replaced end */

    dec2str (&buffer3[0], 5,
(double)((double)(bp->payout.c_discount) *
(double)100.0));

/* Replaced T.Kato 04.03.10 Speed up */
#if 0
!   sprintf (&buffer3[5], "\r\n");
!   strcat (S_WORK, buffer3);
#endif

    buffer3[5] = '\r';
    buffer3[6] = '\n';
    buffer3[7] = '\0';

    memcpy(S_WORK+swork_pos, buffer3, 7+1);
    swork_pos += 7;
/* Replaced end */

/* Replaced T.Kato 04.03.10 Speed up */
#if 0
!/* Replaced T.Kato 03.04.18 Speed up */
!   //sprintf (buffer3, h_pay5);
!   strcpy (buffer3, h_pay5);
!/* Replaced end */
#endif

    memcpy(buffer3, h_pay5, leng_h_pay5+1);
/* Replaced end */

    alp2str ((&buffer3[0] + payp[26] - 0x21D), 20,
bp->payout.c_city);
    alp2str ((&buffer3[0] + payp[27] - 0x21D), 20,
bp->payout.c_state);
    zip2str ((&buffer3[0] + payp[28] - 0x21D), bp-
->payout.c_zip);
    phone2str ((&buffer3[0] + payp[29] - 0x21D),
bp->payout.c_phone);

    h_amount = (float)bp->payin.h_amount /
(float)100;

    dec2str ((&buffer3[0] + payp[30] - 0x21D), 7,
(double)h_amount);

    sigdec2str ((&buffer3[0] + payp[31] - 0x21D),
14, bp->payout.c_balance);
    dec2str ((&buffer3[0] + payp[32] - 0x21D), 13,
bp->payout.c_credit_lim);

/* Replaced T.Kato 04.03.10 */
/* strcat (S_WORK, buffer3);*/

    memcpy(S_WORK+swork_pos, buffer3,
leng_h_pay5+1);
    swork_pos += leng_h_pay5;
/* Replaced end */

    if ( (i = strlen (bp->payout.c_data )) <= 0 ) {

/* Replaced T.Kato 04.03.10 Speed up */
/* sprintf (&buffer3[0], "\r\n\r\n\r\n");*/

    memcpy(buffer3, "\r\n\r\n\r\n\r\n", 8+1);
/* Replaced end */

    }
    else{
        alp2str (buffer2, 50, bp->payout.c_data);
        buffer2[50] = 0;
        newlength = checkHTMLform ( &buffer2[0],
&buffer[0]);

/* Replaced T.Kato 04.03.10 Speed up */
#if 0
!   strcpy (&buffer3[0], &buffer[0]);
!   strcat (buffer3, "\r\n");
#endif

        memcpy(buffer3, buffer, newlength+1);
        memcpy(buffer3+newlength, "\r\n", 2+1);
        next_pos = newlength + 2;
/* Replaced end */

        if (i > 50){

            alp2str (buffer2, 50, &bp-
->payout.c_data[50]);
            buffer2[50] = 0;
            newlength = checkHTMLform
( &buffer2[0], &buffer[0]);

/* Replaced T.Kato 04.03.10 Speed up */
#if 0
!   strcat (buffer3, " ");
!   strcat (buffer3, &buffer[0]);
!   strcat (buffer3, "\r\n");
#endif

            memcpy(buffer3+next_pos, SP3_DATA,
leng_sp3_data+1);
            next_pos += leng_sp3_data;
            memcpy(buffer3+next_pos, buffer,
newlength+1);
            next_pos += newlength;
            memcpy(buffer3+next_pos, "\r\n", 2+1);
            next_pos += 2;
/* Replaced end */
            if (i > 100){

                alp2str (buffer2, 50, &bp-
->payout.c_data[100]);
                buffer2[50] = 0;
                newlength = checkHTMLform
( &buffer2[0], &buffer[0]);

```

```

/* Replaced T.Kato 04.03.10 Speed up */
#if 0
!      strcat (buffer3, "      ");
!      strcat (buffer3, &buffer[0]);
!      strcat (buffer3, "\r\n");
#endif

      memcpy(buffer3+next_pos,
SP3_DATA, leng_sp3_data+1);
      next_pos += leng_sp3_data;
      memcpy(buffer3+next_pos, buffer,
newlength+1);
      next_pos += newlength;
      memcpy(buffer3+next_pos, "\r\n",
2+1);
      next_pos += 2;
/* Replaced end */

      if (i > 150){

          alp2str (buffer2, 50, &bp-
> payout.c_data[150]);
          buffer2[50] = 0;
          newlength = checkHTMLform
(&buffer2[0], &buffer[0]);

/* Replaced T.Kato 04.03.10 Speed up */
#if 0
!      strcat (buffer3, "      ");
!      strcat (buffer3, &buffer[0]);
!      strcat (buffer3, "\r\n");
#endif

      memcpy(buffer3+next_pos,
SP3_DATA, leng_sp3_data+1);
      next_pos += leng_sp3_data;
      memcpy(buffer3+next_pos, buffer,
newlength+1);
      next_pos += newlength;
      memcpy(buffer3+next_pos, "\r\n",
2+1);
      next_pos += 2;
/* Replaced end */

      }
      else {

/* Replaced T.Kato 04.03.10 Speed up */
/*strcat ( buffer3, "\r\n\r\n");*/

      memcpy(buffer3+next_pos, "\r\n",
2+1);
      next_pos += 2;
/* Replaced end */
      }
      else {

/* Replaced T.Kato 04.03.10 Speed up */
/*strcat ( buffer3, "\r\n\r\n\r\n");*/

      memcpy(buffer3+next_pos, "\r\n\r\n",
4+1);
      next_pos += 4;
/* Replaced end */
      }
      }

/* Added T.Kato 04.03.10 Speed up */
else {
      memcpy(buffer3+next_pos,
"\r\n\r\n\r\n", 6+1);
      next_pos += 6;
}
/* Added end */

```

```

}

/* Replaced T.Kato 04.03.10 Speed up */
/*strcat (S_WORK, buffer3);*/

      memcpy(S_WORK+swork_pos, buffer3,
next_pos);
      swork_pos += next_pos;
/* Replaced end */

/* ----- The execution result data notified RTE
is make by the HTML form */
/* Replaced T.Kato 04.03.10 Speed up */
#if 0
/* Replaced T.Kato 03.04.18 Speed up */
! //sprintf(s_buf, h_pay1); /* set Header Data
*/
! strcpy(s_buf, h_pay1); /* set Header Data */
/* Replaced end */
!
! strcat (s_buf, S_WORK); /* set Result Data
*/
!
! sprintf(S_WORK, h_pay3, SOPATH, cookie);
/* set Tailer Data */
! strcat (s_buf, S_WORK);
#endif

      memcpy(s_buf, h_pay1, leng_h_pay1+1);
      memcpy(s_buf+leng_h_pay1, S_WORK,
swork_pos+1);
      next_pos = sprintf(S_WORK, h_pay3,
SOPATH, cookie); /* set Tailer Data */
      memcpy(s_buf+leng_h_pay1+swork_pos,
S_WORK, next_pos+1);
/* Replaced end */

      FreeTuxBuffer(ThreadCntlInfo);
      return (0);
}

.....
tpapl/trnexe/TrxStockLevel.c
.....

/*****
*
* TPC-C Client Application Program Source
*
* Entry Functions
* (1) StockLevel
*
* CREATE by TSL 2003.12.15
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2003 *
*****/
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "forlinux.h"
#include "atmi.h"

#include "trans.h"
#include "tpcc_info.h"
#include "stpage.h"

```

```

#include "ThreadCntl.h"
#include "TpAplDBDependPrototype.h"
#include "log.h"
#include "log_level.h"

#include "tpapl.h"
#include "GlobalArea.h" // Common
#include "OracleFunction.h"

/* Added T.Kato 04.05.13 Speed up */
int leng_h_stock1 = strlen(h_stock1);
int leng_h_stock2 = strlen(h_stock2);
/* Added end */

/*-----
-----

StockLevel : this function processes the
StockLevel transaction.

-----*/
int StockLevel (char *s_buf, RTE_INPUT_DATA
*in_data, int cookie)
{
      StockLevelData *bp;

      char S_WORK[WORK_S];

#ifdef TRNS_BIND
static char *svr_name = "STOCKLEVEL";
#else
static char *svr_name = "OPSTUXSERVER";
#endif
      long olen;

      int h_stock1_leng;
      int h_stock2_leng;
      int h_stock3_leng;

      //SvrAPL return value
#ifdef SCRTEST
      int ret_value;
      int ret_val;
      char* tran_errmsg;
#endif

      THREAD_CNTL_INFO* ThreadCntlInfo;

      MAC_PutFncEntryLog("StockLevel");

      ThreadCntlInfo = GetThreadCntl();
      if (ThreadCntlInfo == 0) {
          sprintf (S_WORK, "thread contorl
information is not allocated [STO]\n");
          MAC_errHTML( s_buf, S_WORK, cookie );
          TpcUserLog (LOG_ERR, S_WORK);
          return (-1);
      }
      bp = ( StockLevelData *)ThreadCntlInfo-
>TrxDData;
      memset(bp, 0x00, sizeof(StockLevelData));

/* ----- check
the Input data */
      bp->stoin_w_id = MAC_w_id(cookie);
      bp->stoin_d_id = MAC_d_id(cookie);

      bp->stoin.threshold = (long)str2short(in_data-
>threshold, 2);

      if(bp->stoin.threshold < 10 || bp-
>stoin.threshold > 20) {

```

```

    TpcUserLog (LOG_ERR, "Input data error
[STO] (threshold = %s)[Return_Value:%d]\n",
    in_data->threshold, bp-
>stoin.threshold);
    return set_errpage(s_buf, cookie, 3, (int)bp-
>stoin.threshold, 0, 0);
}

/* ----- Execute Stock
Level transaction */
#ifdef SCRTST
resend_stock:

/* Replaced 2003.12.15 Transaction processeing
interface COM+ --> TUXEDO */
#ifdef TRNS_BIND
    /* Set transaction type for Warehouse bind */
    bp->retval = 5;
#endif

    ret_val = tpcall(svr_name,
(char*)ThreadCntlInfo->TrxDData,
sizeof(NewOrderData),
(char*)&ThreadCntlInfo-
>TrxDData, &oLen, 0|TPNOTIME);
    bp = ( StockLevelData *)ThreadCntlInfo-
>TrxDData;
    ret_value = CreateTranErrReason(ret_val, bp-
>stout.error, &tran_errmsg);

    switch(ret_value) {
    case 0:
        /* Success */
        break;

    case 1:
        /* Retry Paymant transaction */
        TpcUserLog (LOG_WRN, "StockLevel
retry\n");
        goto resend_stock;

    case -1:
        /* Oracle failed */
        sprintf( S_WORK, "Oracle failed to
process StockLevel Transaction.(%s)\n"
"ret_value = %d threshold = %d cookie
= %d\n",
            tran_errmsg, ret_value, bp-
>stoin.threshold, cookie );

        MAC_errHTML( s_buf, S_WORK, cookie );
        TpcUserLog (LOG_ERR, S_WORK);
        FreeTuxBuffer(ThreadCntlInfo);
        return (-1);

    default:
        /* Tuxedo failed */
        sprintf( S_WORK, "tpcall failed to process
NewOrder Transaction.(tperrno=%d)\n"
"ret_value = %d threshold = %d cookie
= %d\n",
            tperrno, ret_value, bp-
>stoin.threshold, cookie );

        MAC_errHTML_TUXEDO( s_buf, S_WORK,
cookie );
        TpcUserLog (LOG_ERR, S_WORK);
        FreeTuxBuffer(ThreadCntlInfo);
        return (-1);
    }
/* Changed end */

#else
dummy_stocklvl ( bp );
#endif

```

```

/* Replaced T.Kato 04.05.13 Speed up */
#if 0
/* Replaced T.Kato 03.04.18 Speed up */
! //sprintf (S_WORK, h_stock2);
! strcpy (S_WORK, h_stock2);
! h_stock2_leng = strlen(S_WORK);
/* Replaced end */
#endif

    strcpy (S_WORK, h_stock2);
    h_stock2_leng = leng_h_stock2;
/* Replaced end */

    int2str ((S_WORK + stockp[0]), 6, (int)bp-
>stoin.w_id);

    int2str ((S_WORK + stockp[1]), 2, (int)bp-
>stoin.d_id);
    int2str ((S_WORK + stockp[2]), 2, (int)bp-
>stoin.threshold);
    int2str ((S_WORK + stockp[3]), 3, (int)bp-
>stout.low_stock);

/* ----- The execution result data notified RTE
is make by the HTML form */

/* Replaced T.Kato 04.05.13 Speed up */
#if 0
/* Replaced T.Kato 03.04.18 Speed up */
!#if 0
!! sprintf(s_buf, h_stock1); /* Set Header data
*/
!! strcat (s_buf, S_WORK); /* Set Result
data */
!!
!! sprintf(S_WORK, h_stock3, SOPATH,
cookie); /* Set Tailer data */
!! strcat (s_buf, S_WORK);
!#endif
! strcpy(s_buf, h_stock1);
! h_stock1_leng = strlen(s_buf);
! memcpy(s_buf + h_stock1_leng, S_WORK,
h_stock2_leng);
! h_stock3_leng = sprintf(S_WORK, h_stock3,
SOPATH, cookie);
! memcpy(s_buf + h_stock1_leng +
h_stock2_leng, S_WORK, h_stock3_leng);
! *(s_buf + h_stock1_leng + h_stock2_leng +
h_stock3_leng) = '\0';
/* Replaced end */
#endif
    strcpy(s_buf, h_stock1);
    h_stock1_leng = leng_h_stock1;
    memcpy(s_buf + h_stock1_leng, S_WORK,
h_stock2_leng);
    h_stock3_leng = sprintf(S_WORK, h_stock3,
SOPATH, cookie);
    memcpy(s_buf + h_stock1_leng +
h_stock2_leng, S_WORK, h_stock3_leng);
    *(s_buf + h_stock1_leng + h_stock2_leng +
h_stock3_leng) = '\0';
/* Replaced end */

    FreeTuxBuffer(ThreadCntlInfo);
    return (0);
}

```

```

.....
tpapl/trnexe/log_level.h
.....

```

```

/******
****

```

```

*
* TPC-C Client Application Program Source
*
*
* CREATE by TSL 2003.02.07
*
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2003 *

*****
****/

#define PUT_INF_LOG //
Information log
#define PUT_FNC_ENTRY_LOG //
Function entry point log
#define PUT_FNC_EXIT_LOG //
Function exit log

/* Function entry point log macro */
#ifdef PUT_FNC_ENTRY_LOG
#define MAC_PutFncEntryLog(func)
TpcUserLog(LOG_INF, ">>>>> "func" start
>>>>>");
#else
#define MAC_PutFncEntryLog(func) ;
#endif

/* Function exit point log */
#ifdef PUT_FNC_EXIT_LOG
#define MAC_PutFncExitLog(func)
TpcUserLog(LOG_INF, "<<<<< "func" end
<<<<<");
#else
#define MAC_PutFncExitLog(func) ;
#endif

```

Appendix B: Server Source Code

```

.....
blocks/paynz.sql
.....

DECLARE /* paynz */
  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
  inittpcc.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
  inittpcc.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;

  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 = inittpcc.cust_rowid

```

```

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
  inittpcc.dist_name, :d_street_1, :d_street_2, :d_cit
y, :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,
:c_date, inittpcc.ware_name || ' ' ||
  inittpcc.dist_name);
  EXIT;

  EXCEPTION
  WHEN not_serializable OR deadlock OR
snapshot_too_old THEN
    ROLLBACK;
    :retry := :retry + 1;
  END;

  END LOOP;
END;

.....
blocks/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 inittpcc.ware_name,
      :w_street_1, :w_street_2, :w_city, :w
_state, :w_zip;

    SELECT rowid
  BULK COLLECT INTO inittpcc.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;

  inittpcc.c_num := sql%rowcount;

```

```

  inittpcc.cust_rowid :=
  inittpcc.row_id/(inittpcc.c_num) / 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 = inittpcc.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_st
ate,
      :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 = inittpcc.cust_rowid
      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
  inittpcc.dist_name, :d_street_1, :d_street_2, :d_c
ity,
      :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,
:c_date, inittpcc.ware_name || ' ' ||
  inittpcc.dist_name);
  EXIT;

  EXCEPTION
  WHEN not_serializable OR deadlock OR
snapshot_too_old THEN
    ROLLBACK;
    :retry := :retry + 1;
  END;

  END LOOP;
END;

  TPC Benchmark C Full Disclosure

```

```

.....
blocks/tkvcin.sql
.....

-- The initnew package for storing variables used
in the
-- New Order anonymous block

CREATE OR REPLACE PACKAGE inittpcc
AS
  TYPE intarray IS TABLE OF INTEGER INDEX
  BY BINARY_INTEGER;
  TYPE distarray IS TABLE OF VARCHAR(24)
  INDEX BY BINARY_INTEGER;
  nulldate DATE;
  TYPE rowidarray IS TABLE OF ROWID INDEX
  BY PLS_INTEGER;
  s_dist distarray;
  idx1arr intarray;
  s_remote intarray;
  dist intarray;
  row_id rowidarray;
  cust_rowid rowid;
  dist_name VARCHAR2(11);
  ware_name VARCHAR2(11);
  c_num PLS_INTEGER;

  PROCEDURE init_no(idxarr intarray);
  PROCEDURE init_del;
  PROCEDURE init_pay;
END inittpcc;
/
show errors;

CREATE OR REPLACE PACKAGE BODY
inittpcc AS
  PROCEDURE init_no (idxarr intarray)
  IS
  BEGIN
    -- initialize null date
    nulldate := TO_DATE('01-01-1811', 'MM-DD-
YYYY');
    idx1arr := idxarr;
  END init_no;

  PROCEDURE init_del
  IS
  BEGIN
    FOR i IN 1 .. 10 LOOP
      dist(i) := i;
    END LOOP;
  END init_del;

  PROCEDURE init_pay IS
  BEGIN
    NULL;
  END init_pay;

END inittpcc;
/
show errors
exit

.....
blocks/tkvcpdel.sql
.....

declare
  TYPE numarray IS TABLE OF NUMBER
  INDEX BY BINARY_INTEGER;
  TYPE numlist is varray (10) of number;
  dist numarray;

```

```

amt numarray ;
cnt pls_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);

BEGIN
  LOOP BEGIN
    FORALL d IN 1..10
      DELETE FROM nord N
        WHERE no_d_id = inittpcc.dist(d)
          AND no_w_id = :w_id
          AND no_o_id = (select min (no_o_id)
            from nord
              where no_d_id = N.no_d_id
                and no_w_id = N.no_w_id)
        RETURNING no_d_id, no_o_id BULK
        COLLECT INTO :d_id, :order_id;

    :ordcnt := SQL%ROWCOUNT;

    FORALL o in 1.. :ordcnt
      UPDATE ordr SET o_carrier_id = :carrier_id
        WHERE o_id = :order_id (o)
          AND o_d_id = :d_id(o)
          AND o_w_id = :w_id
        RETURNING o_c_id BULK COLLECT
        INTO :o_c_id;

    FORALL o in 1.. :ordcnt
      UPDATE ordl SET ol_delivery_d = :now
        WHERE ol_w_id = :w_id
          AND ol_d_id = :d_id(o)
          AND ol_o_id = :order_id(o)
        RETURNING sum(ol_amount) BULK
        COLLECT INTO :sums;

    FORALL c IN 1.. :ordcnt
      UPDATE cust
        SET c_balance = c_balance + :sums(c),
          c_delivery_cnt = c_delivery_cnt +
1
        WHERE c_w_id = :w_id
          AND c_d_id = :d_id(c)
          AND c_id = :o_c_id(c);
      COMMIT;
      EXIT;
      EXCEPTION
        WHEN not_serializable OR deadlock OR
snapshot_too_old
        THEN
          ROLLBACK;
          :retry := :retry + 1;
        END;

    END LOOP; -- for retry
  END;

.....
blocks/tkvcpnew.sql
.....

-- New Order Anonymous block

DECLARE
  idx PLS_INTEGER;
  dummy_local PLS_INTEGER;

```

```

cache_ol_cnt PLS_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 .. cache_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 = (CASE WHEN s_quantity
< :ol_quantity (idx) + 10
          THEN s_quantity +91
        ELSE s_quantity
          END) - :ol_quantity(idx)
        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,
          i_price*:ol_quantity(idx),
          CASE WHEN i_data NOT LIKE
'%ORIGINAL%'
            THEN 'G'
          ELSE (CASE WHEN s_data NOT
LIKE '%ORIGINAL%'
            THEN 'G'
              ELSE 'B'
            END)
          END
        BULK COLLECT
        INTO :i_price, :i_name, :s_quantity,
inittpcc.s_dist,
          :ol_amount, :brand_generic;
  END u1;

PROCEDURE u2 IS
BEGIN
  FORALL idx IN 1 .. cache_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 = (CASE WHEN s_quantity
< :ol_quantity (idx) + 10
          THEN s_quantity +91
        ELSE s_quantity
          END) - :ol_quantity(idx)
        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,
          i_price*:ol_quantity(idx),
          CASE WHEN i_data NOT LIKE
'%ORIGINAL%'
            THEN 'G'
          ELSE (CASE WHEN s_data NOT
LIKE '%ORIGINAL%'
            THEN 'G'
              ELSE 'B'
            END)
          END
        BULK COLLECT
        INTO :i_price, :i_name, :s_quantity,
inittpcc.s_dist,
          :ol_amount, :brand_generic;
  END u2;

```



```

        END
        BULK COLLECT
    INTO :i_price, :i_name, :s_quantity,
    inittpcc.s_dist,
        :ol_amount, :brand_generic;
    END u9;

    PROCEDURE u10 IS
    BEGIN
        FORALL idx IN 1 .. cache_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 = (CASE WHEN s_quantity
        < :ol_quantity (idx) + 10
            THEN s_quantity +91
            ELSE s_quantity
            END) - :ol_quantity(idx)
        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,
            i_price*ol_quantity(idx),
            CASE WHEN i_data NOT LIKE
        '%ORIGINAL%'
            THEN 'G'
            ELSE (CASE WHEN s_data NOT
        LIKE '%ORIGINAL%'
            THEN 'G'
            ELSE 'B'
            END)
        END
        BULK COLLECT
    INTO :i_price, :i_name, :s_quantity,
    inittpcc.s_dist,
        :ol_amount, :brand_generic;
    END u10;

    PROCEDURE fix_items IS
    rows_lost          PLS_INTEGER;
    max_index          PLS_INTEGER;
    temp_index         PLS_INTEGER;
    BEGIN
        idx := 1;
        rows_lost := 0;
        max_index := dummy_local;

        WHILE (max_index != cache_ol_cnt) LOOP

            WHILE (idx <= sql%rowcount AND
                sql%bulk_rowcount(idx +
        rows_lost) = 1)
                LOOP
                    idx := idx + 1;
                END LOOP;

            temp_index := max_index;
            WHILE (temp_index >= idx + rows_lost)
        LOOP
                :ol_amount(temp_index +
        1) := :ol_amount(temp_index);
                :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);
                inittpcc.s_dist(temp_index + 1) :=
        inittpcc.s_dist(temp_index);
                :brand_generic(temp_index +
        1) := :brand_generic(temp_index);
                temp_index := temp_index - 1;
            END LOOP;
        
```

```

    IF (idx + rows_lost <= cache_ol_cnt) THEN
        :i_price(idx + rows_lost) := 0;
        :i_name(idx + rows_lost) := 'NO
    ITEM';
        :s_quantity(idx + rows_lost) := 0;
        inittpcc.s_dist(idx + rows_lost) := NULL;
        :brand_generic(idx + rows_lost) := '';
        :ol_amount(idx + rows_lost) := 0;
        rows_lost := rows_lost + 1;
        max_index := max_index + 1;
    END IF;

    END LOOP;
    END fix_items;

    BEGIN
    LOOP BEGIN
        cache_ol_cnt := :o_ol_cnt;

        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
        INTO :c_discount, :c_last, :c_credit
        FROM cust
        WHERE c_id = :c_id AND c_d_id = :d_id
        AND c_w_id = :w_id;

        SELECT w_tax
        INTO :w_tax
        FROM ware
        WHERE 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);

        dummy_local := :d_id;

        IF (dummy_local < 6) THEN
            IF (dummy_local < 3) THEN
                IF (dummy_local = 1) THEN
                    u1;
                ELSE
                    u2;
                END IF;
            ELSE
                IF (dummy_local = 3) THEN
                    u3;
                ELSIF (dummy_local = 4) then
                    u4;
                ELSE
                    u5;
                END IF;
            END IF;
        ELSE
            IF (dummy_local < 8) THEN
                IF (dummy_local = 6) THEN
                    u6;
                ELSE
                    u7;
                END IF;
            ELSE
        
```

```

            IF (dummy_local = 8) THEN
                u8;
            ELSIF (dummy_local = 9) then
                u9;
            ELSE
                u10;
            END IF;
        END IF;
        END IF;

        dummy_local := sql%rowcount;

        IF (dummy_local != cache_ol_cnt ) THEN
        fix_items; END IF;

        FORALL idx IN 1..dummy_local
            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,
                inittpcc.idx1arr(idx), inittpcc.nulldate,
                :ol_i_id(idx), :ol_supply_w_id(idx),
                :ol_quantity(idx), :ol_amount(idx),
                inittpcc.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;

    .....
    common/GetPrivateProfileString.c
    .....

    /*****
    *****
    *
    * TPC-C Client Application Program Source
    *
    *
    * Entry Functions
    * (1) GetPrivateProfileString
    *
    *
    * CREATE by TSL 2003.12.18
    *
    *
    * All Right Reserved, Copyright Co. FUJITSU
    LIMITED 2003-2004 *
    *****
    *****/

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

    /*****
    *****
    * Get data string corresponded key in
    cogfiguration file. *
    
```

```

* Return Value          *
*   Get string length  *
*****
****/
int GetPrivateProfileString(char*
section_name, /* Section name
*/
name          char* key_name, /* Key
*/
char* default_str, /* Default
string, if key nothing */
char* key_data, /* Key
data */
int buf_size, /* Buffer
size of key data */
char* file_name) { /* File
name */

FILE* prof_file;
char read_buf[256];
char search[32];
char* get_str;
char* key_pos=0;
int get_cnt;
int i;

/* Open profile file */
if ((prof_file = fopen(file_name, "r")) == NULL)
{
goto DEFAULT_STRING;
}

/* Make searching section name "[section
name]" */
search[0] = '\0';
strcpy(&search[1], section_name);
strcat(search, ".ini");

/* Search section name */
while((get_str = fgets(read_buf,
sizeof(read_buf), prof_file)) != NULL) {

/* Search section name form to be read one
line */
if ((char*)strstr(read_buf, search) == NULL)
{
/* No match section name, next line read
*/
continue;
}
break;
}
if (get_str == NULL) {
/* Found EOF or read error */
goto DEFAULT_STRING_FCLOSE;
}

/* Make searching key name "key_name=" */
strcpy(search, key_name);
strcat(search, "=");

/* Search key name in this section */
while((get_str = fgets(read_buf,
sizeof(read_buf), prof_file)) != NULL) {
for (i = 0; read_buf[i] == ' ' || read_buf[i] ==
'\t'; i++);
if (read_buf[i] == '[') {
/* Other section started, undefined key
name */
goto DEFAULT_STRING_FCLOSE;
}
if ((key_pos = (char*)strstr(read_buf,
search)) == NULL) {

```

```

/* No match key name */
continue;
}
break;
}
if (get_str == NULL) {
/* Found EOF or read error */
goto DEFAULT_STRING_FCLOSE;
}

fclose(prof_file);

/* Get key_value, fixed format "key value" */
for (; *key_pos != '\0'; key_pos++)
key_pos++;
for (get_cnt = 0; *key_pos != '\0'; key_pos++) {
/* Get & set key value */
*key_data = *key_pos;
key_data++;
get_cnt++;
if (get_cnt >= (buf_size - 1)) {
/* Key data buffer full */
break;
}
}
*key_data = '\0';
return(get_cnt);

DEFAULT_STRING_FCLOSE:
fclose(prof_file);

DEFAULT_STRING:
strcpy(key_data, default_str, buf_size-1);
return(strlen(key_data));
}

.....
common/MakeShell
.....

#!/bin/sh
cd /home/tpc/client_apl/common
make > make_result.txt 2>&1

.....
common/Makefile
.....

#-----
-----
# Makefile : Makefile for common of TPAPL and
SVRAPL.
#
# Created by TSL 2003.12.17
#
# All Right Rserverd, Copyright Co, FUJITSU
LIMITED 2003-2004.
#-----
-----

# GCC compile configurations
AR = ar
ARFLAGS = rv
CFLAGS = -Wall -O2
CC = gcc

# MACRO definition
DMACRO = -DTSL -DPLSQLFLAG=1 -DTUX

# home directory
ORADIR = /usr/local/oracle

```

```

TUXDIR = /usr/local/BEA/tuxedo8.1
SVRDIR = /home/tpc/client_apl/svrpl

# include directory
ORA_INC = -I$(ORADIR)/rdbms/demo -
I$(ORADIR)/rdbms/public
COM_INC = -I$(SVRDIR)/common
SVR_INC = -I$(SVRDIR)
TUX_INC = -I$(TUXDIR)/include
INCLUDE = $(COM_INC) $(SVR_INC)
$(ORA_INC) $(TUX_INC)

# target object
COMOBSJ = log.o sema.o
GetPrivateProfileString.o shmем.o
COMLIB = libcom.a

INCFILS = log.h sema.h forlinux.h shmем.h

$(COMLIB) : $(COMOBSJ)
$(AR) $(ARFLAGS) $(COMLIB) $(COMOBSJ)

.SUFFIXES : .o .c
.c.o:
$(CC) -o $@ -c $(CFLAGS) $(INCLUDE)
$(DMACRO) $<

$(COMOBSJ) : $(INCFILS)

clean:
rm $(COMLIB) $(COMOBSJ)

.....
common/forlinux.h
.....

/*****
*****
*
* TPC-C Client Application Program Source
*
*
* Entry Functions
* definition for converting Linux.
*
*
* CREATE by TSL 2003.05.16
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2003-2004 *
*****
*****/
/* forlinux.h */

#include <limits.h>
#define MAX_PATH PATH_MAX /*
Windows:MAX_PATH , Linux:PATH_MAX */
#define Sleep(x) poll(0, 0, x); /* sleep unit is
a msec. */

.....
common/log.c
.....

/*****
*****

```

```

*
* TPC-C Client Application Program Source
*
* Entry Functions
* Log is outputted to a file.
*
* CREATE by TSL 2002.11.29
*
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2002-2004 *
*****/
#include "forlinux.h"
#include <stdio.h>
#include <string.h>
#include <time.h>
#include <sys/types.h>
#include <stdarg.h>
#include <unistd.h>
#include <pthread.h>
#include <sys/types.h>
#include <sys/stat.h>
#include "sema.h"

#define LOG_MODULE
#include "log.h"

void TpcUserLog(char* file_name, int line_no,
char* type_name, char* fmt, ...)
{
FILE* fp;
pid_t pid;
pthread_t tld;
char* fname;
int stat;

/* -- BEGIN -- Modified by Hayashi for thread-
safe. 2006/02/13 */
#if 0
! struct tm *nowtime;
#else
struct tm tt;
struct tm *nowtime=&tt;
#endif
/* -- END -- Modified by Hayashi for thread-safe.
2006/02/13 */

time_t long_time;
va_list va;

if (strcmp(type_name, "LCK") != 0) {
/* Lock semaphore */
stat = LockSem(GLB_LogSemId);
}
/* Get current time. */

time(&long_time);

/* -- BEGIN -- Modified by Hayashi for thread-
safe. 2006/02/13 */
#if 0
! nowtime = localtime(&long_time);
#else
localtime_r(&long_time, nowtime);
#endif
/* -- END -- Modified by Hayashi for thread-safe.
2006/02/13 */

/* Get process Id. */
pid = getpid();

/* Get thread Id. */

```

```

tld = pthread_self();

/* Get just file name from a path. */
fname = (char*)strchr(file_name, (int)'/');
if (fname == NULL) {
fname = file_name;
} else {
fname = fname + 1;
}

va_start(va, fmt);

fp = fopen(GLB_LogFilePath, "a");
fprintf(fp, "%02d:%02d:%02d [%6d:%08x] %-
32s(%4d) :%s: ",
nowtime->tm_hour, nowtime->tm_min,
nowtime->tm_sec, pid, (int)tld, fname, line_no,
type_name);
vfprintf(fp, fmt, va);

if (*(fmt + strlen(fmt) - 1) != '\n')
fprintf(fp, "\n");

va_end(va);

fclose(fp);

/* change mode which all users can read and
write. */
chmod(GLB_LogFilePath, S_IRUSR
|S_IWUSR |S_IRGRP|S_IWGRP| S_IROTH |
S_IWOTH);

if (strcmp(type_name, "LCK") != 0) {
// Unlock semaphore
stat = UnlockSem(GLB_LogSemId);
}

return;
}

.....
common/log.h
.....

/*****
*
* TPC-C Client Application Program Source
*
* Entry Functions
* Log is outputted to a file.
*
* CREATE by TSL 2002.11.29
*
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2003-2004 *
*****/

void TpcUserLog (char *file_name, int line_no,
char * type_name, char * fmt, ...);

extern char GLB_LogFilePath[MAX_PATH];
extern int GLB_LogSemId;

#define DEFAULT_SVRAPL_LOG_PATH
"/home/tpc/log/DBDepend_Userlog.log"
#define DEFAULT_TPAPL_LOG_PATH
"/home/tpc/log/userlog.log"

```

```

#define LOG_ERR __FILE__, __LINE__, "ERR"
#define LOG_INF __FILE__, __LINE__, "INF"
#define LOG_WRN __FILE__, __LINE__,
"WRN"
#define LOG_LCK __FILE__, __LINE__, "LCK"

#define LOG_FILE_INF __FILE__,
__LINE__, "INF"
#define LOG_FILE_LINE __FILE__,
__LINE__

.....
common/sema.c
.....

/*****
*
* TPC-C Client Application Program Source
*
* Filename :
* sema.c
* Entry Functions :
* There are functions to control semaphore.
*
* CREATE by TSL 2003.12.18
*
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2003-2004 *
*****/

#include "forlinux.h"
#include <stdio.h>
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/sem.h>
#include <errno.h>
#include "log.h"
#include "sema.h"

/*****
****
* Initialize semaphore.
* Return Value
* > 0 semaphore Id. (always over 0)
*
* < 0 fail.
*
*****
****/

int InitSem(char *path, int projectId)
{
int sid;
union semun{
int val;
struct semid_ds *buf;
ushort *array;
} c_arg;

TpcUserLog(LOG_LCK, "InitSem: start
path<-%s> projectId=%d\n",
path, projectId);

if ((sid = GetSem(path, projectId)) == -1) {
TpcUserLog(LOG_LCK, "GetSem() fail,
path<-%s> projectId=%d\n",

```

```

        path, projectId);
    return(-1);
}
c_arg.val=1;
if (semctl(sid,0,SETVAL,c_arg)==-1) {
    TpccUserLog(LOG_LCK, "semctl fail,
sid=%d\n",sid);
    return(-1);
}
TpccUserLog(LOG_LCK, "InitSem: Get
semid=%d\n",sid);

    return(sid);
}
/*****
* Get semaphore.
* Return Value
* > 0 semaphore Id. (always over 0)
*
* < 0 fail.
*
*****/
int GetSem(char *path, int projectId)
{
    int sid;
    int key;

    if ((key = ftok(path,projectId)) == -1) {
        TpccUserLog(LOG_LCK, "ftok() fail,
path<%s> projectId=%d errno=%d\n",
        path, projectId, errno);
        return(-1);
    }
    if ((sid=semget(key,1,0666|IPC_CREAT))== -
1){
        TpccUserLog(LOG_LCK, "semget() fail,
key=%d errno=%d\n",key, errno);
        return(-1);
    }

    return(sid);
}
/*****
* Reuire to lock semaphore.
*
* Return Value
* 1 success.
* -1 fail.
*
*****/
int LockSem(int sid)
{
    struct sembuf sb;

    sb.sem_num=0;
    sb.sem_op=-1;
    sb.sem_flg=0;
    if(semop(sid,&sb,1)== -1) {
        TpccUserLog(LOG_LCK, "semop() fail,
sid=%d\n",sid);
        return(-1);
    }
    return(1);
}
/*****
* Reuire to unlock semaphore.
*
* Return Value
* 1 success.
* -1 fail.
*
*****/

```

```

*****
*****/
int UnlockSem(int sid)
{
    struct sembuf sb;

    sb.sem_num=0;
    sb.sem_op=1;
    sb.sem_flg=0;
    if(semop(sid,&sb,1)== -1){
        TpccUserLog(LOG_LCK, "semop() fail,
sid=%d\n",sid);
        return(-1);
    }
    return(1);
}

:-----:
common/sem.h
:-----:

/*****
*
* TPC-C Client Application Program Source
*
* Entry Functions
* Semaphore control.
*
* CREATE by TSL 2003.12.19
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2003-2004 *
*****/

/*== project Id =====*/
#define SEM_SVRAPL_PROJID
(int)'S'
#define SEM_TPAPL_PROJID (int)'T'
#define SEM_SAMPLING_PERFORMANCE
(int)'P'

/*=====
====*/
/* prototype definition */
/*=====
====*/
int InitSem(char *path, int projectId);
int GetSem(char *path, int projectId);
int LockSem(int sid);
int UnlockSem(int sid);

:-----:
common/shmem.c
:-----:

/*****
*
* TPC-C Client Application Program Source
*
*
* Filename :
* sema.c
* Entry Functions :
*
*****/

```

```

* There are functions to control shared
memory.
*
* CREATE by TSL 2004.01.15
*
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2003-2004 *
*****/
#include "forlinux.h"
#include <stdio.h>
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/shm.h>
#include <errno.h>
#include "log.h"

/*****
*
* Initialize shared memory.
* Return Value
* > 0 shared memory address. (always over
0)
* < 0 fail.
*
*****/
char* InitShmem(char *path, int projectId, int
size)
{
    int shmId;
    int key;
    char *shmaddr;

    TpccUserLog(LOG_LCK, "InitShmem: start
path<%s> projectId=%d\n",
    path, projectId);

    if ((key = ftok(path,projectId)) == -1) {
        TpccUserLog(LOG_LCK, "ftok() fail,
path<%s> projectId=%d errno=%d\n",
        path, projectId, errno);
        return((char *)-1);
    }
    if
((shmId=shmget(key,size,IPC_CREAT|0666))==
-1){
        TpccUserLog(LOG_LCK, "shmget() fail,
key=%d errno=%d",key, errno);
        return((char *)-1);
    }
    if( shmaddr = (char *)shmat(shmId, NULL, 0)
== (char *)-1 ) {
        TpccUserLog(LOG_LCK, "shmat() fail,
shmId=%d path<%s> projectId=%d errno=%d\n",
        shmId, path, projectId, errno);
        return ((char *)-1);
    }

    TpccUserLog(LOG_LCK, "InitShmem: Get
shmId =%d shmaddr = %08x\n",shmId,
shmaddr);

    return(shmaddr);
}
/*****
*
* Get shared memory.
* Return Value
* > 0 shared memory address. (always over
0)
* < 0 fail.
*
*****/

```

```

*****
****/
char* GetShmem(char *path, int projectld, int
size)
{
    int shmld;
    int key;
    char *shmaddr;

    if ((key = ftok(path,projectld)) == -1) {
        TpcUserLog(LOG_LCK, "ftok() fail,
path<%s> projectld=%d errno=%d\n",
        path, projectld,errno);
        return((char *)-1);
    }
    if ((shmld=shmget(key,size, 0))==-1){
        TpcUserLog(LOG_LCK, "shmget() fail,
key=%d errno=%d\n",key,errno);
        return((char *)-1);
    }
    if ((shmaddr = (char *)shmat(shmld, NULL, 0))
== (char *)-1) {
        TpcUserLog(LOG_LCK, "shmat() fail,
shmld=%d path<%s> projectld=%d errno=%d\n",
        shmld, path, projectld, errno);
        return ((char *)-1);
    }

    return(shmaddr);
}

.....
common/shmem.h
.....

/*****
*****
*
*
* TPC-C Client Application Program Source
*
*
* Entry Functions
* Shared memory control.
*
*
*
* CREATE by TSL 2004.01.15
*
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2003-2004 *
*****
*****/

/== project ld =====*/
#define
SHMEM_SAMPLING_PERFORMANCE
(int)'P'

/=====
====*/
/* prototype definition */
/=====
====*/
char* InitShmem(char *path, int projectld, int
size);
char* GetShmem(char *path, int projectld, int
size);

.....

```

```

svrapl/GlobalArea.c
.....
/*****
*****
*
* TPC-C Client Application Program Source
*
*
* Entry Functions
* Global Area definition.
*
* CREATE by TSL 2003.05.16
*
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2003-2004 *
*****
*****/
#include "forlinux.h"
#include "tpcc.h"
#include "tpccflags.h"
#include "TrnCntrlInfo.h"

char GLB_LogFilePath[MAX_PATH];
char GLB_ConfigFilePath[MAX_PATH];
int GLB_LogSemld;

/* Global area for Oracle interfase. */
/* ----- */
/* Delivery (pldel.cpp) */
/* ----- */
pldelctx *pldctx;
delctx *dctx;
#ifdef DMLRETDEL
amtctx *actx;
#endif
/* ----- */
/* NewOrder (plnew.cpp) */
/* ----- */
newctx *nctx;
/* ----- */
/* OrderStatus (plord.cpp) */
/* ----- */
ordctx *octx;
defctx cbctx;

/* Added T.Kato 2004.12.21 New Oracle10g tool
kit */
int ordcount = 0;
#ifdef DEBUG
int trace_on = 0;
#endif
/* Added end */

/* ----- */
/* Payment (plpay.cpp) */
/* ----- */
payctx *pctx;
/* ----- */
/* StockLevel (plsto.cpp) */
/* ----- */
stoctx *sctx;
/* ----- */
/* (tpccpl.cpp) */
/* ----- */
FILE *lfp;
/* Deleted T.Kato 02.10.23 for warning
!FILE *fopen ();
Deleted end */

/* Added t.Kato 02.10.24 for Delivery logging file
control */
int iflg; /* Delivery log initialize flag */

```

```

/* Added end */
int proc_no;
int logon;
int new_init;
int pay_init;
int ord_init;

#ifdef DEL_ORA8I
int del_init;
#else
int del_init_oci;
int del_init_plsql;
#endif

int sto_init;
int res_init;

int execstatus;
int errcode;

OCIEnv *tpcenv;
OCIServer *tpcsrv;
OCIError *errhp;
OCISvcCtx *tpcsvc;
OCISession *tpcusr;
OCIStmt *curi;

/* for stock-level transaction */
int w_id;
int d_id;
int c_id;

/* Replaced T.kato 03.09.09 Oracle10g tool kit */
#if 0
! int threshold;
#endif

#ifdef USE_IEEE_NUMBER
float threshold;
#else
int threshold;
#endif /* USE_IEEE_NUMBER */
/* Replaced end */

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];

/* Replaced T.kato 03.09.09 Oracle10g tool kit */
#if 0
! int ol_quantity[15];
! int ol_amount[15];
#endif

#ifdef USE_IEEE_NUMBER
float ol_quantity[15];
float ol_amount[15];
#else

```

```

int ol_quantity[15];
int ol_amount[15];
#endif /* USE_IEEE_NUMBER */
/* Replaced end */

ub4 ol_del_len[15];
text ol_delivery_d[15][11];
/* xnie - begin */
OCIRowid *o_rowid;
/* xnie - end */

/* for payment transaction */
int c_w_id;
int c_d_id;

/* Replaced T.kato 03.09.09 Oracle10g tool kit */
#if 0
! int h_amount;
#endif

#ifdef USE_IEEE_NUMBER
float h_amount;
#else
int h_amount;
#endif /* USE_IEEE_NUMBER */
/* Replaced end */

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 sincelen;
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];

/* Replaced T.kato 03.09.09 Oracle10g tool kit */
#if 0
! int nol_quantity[15];
! int nol_amount[15];
! int s_quantity[15];
! int i_price[15];
#endif

#ifdef USE_IEEE_NUMBER
float nol_quantity[15];
float nol_amount[15];
float s_quantity[15];
float i_price[15];
#else
int nol_quantity[15];
int nol_amount[15];
int s_quantity[15];
int i_price[15];

```

```

#endif /* USE_IEEE_NUMBER */
/* Replaced end */

int nol_quant10[15];
int nol_quant91[15];
int nol_ytdqty[15];
int o_all_local;
float w_tax;
float d_tax;
/* Deleted T.Kato 02.11.13
!float total_amount;
Deleted end */
char i_name[15][25];
char brand_gen[15];
char brand_generic[15][1];
int status;
int tracelevel;

OCIDate cr_date;
OCIDate c_since;
OCIDate o_entry_d_base;
OCIDate ol_d_base[15];
dvoid *xmem;
/* ----- */
/* (tpccsvr.cpp) */
/* ----- */
/* set up pointers for type casting */
struct newstruct *newinfo;
struct paystruct *payinfo;
struct ordstruct *ordinfo;
struct delstruct *delinfo;
struct stostruct *stoinfo;

#ifdef AVOID_DEADLOCK
int indx[NITEMS], ordl_cnt;
#endif

.....
svrapl/GlobalArea.h
.....

/*****
*****
*
* TPC-C Client Application Program Source
*
*
* Entry Functions *
* Global Area definition. *
*
* CREATE by TSL 2003.05.16
*
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2003-2004 *
*****
*****/

#include "tpccflags.h"
#include "TrnCntrlInfo.h"

extern char GLB_LogFilePath[MAX_PATH];
extern char GLB_ConfigFilePath[MAX_PATH];
extern int GLB_LogSemId;

#define TPCC_CONF_FILE
"/home/tpccconf/tpapl.conf"

```

```

#define LOG_FILE_NAME_THREAD
"log\\SvrThread%05d.log"

/* Global area for Oracle interfase. */
/* ----- */
/* Delivery (pldel.cpp) */
/* ----- */
extern pldelctx *pldctx;
extern delctx *dctx;
#ifdef DMLRETDDEL
extern amtctx *actx;
#endif
/* ----- */
/* NewOrder (plnew.cpp) */
/* ----- */
extern newctx *nctx;
/* ----- */
/* OrderStatus (plord.cpp) */
/* ----- */
extern ordctx *octx;
extern defctx *cbctx;

/* Added T.Kato 2004.12.21 New Oracle10g tool
kit */
extern int ordcount;
#ifdef DEBUG
extern int trace_on;
#endif
/* Added end */

/* ----- */
/* Payment (plpay.cpp) */
/* ----- */
extern payctx *pctx;
/* ----- */
/* StockLevel (plsto.cpp) */
/* ----- */
extern stoctx *sctx;
/* ----- */
/* (tpccpl.cpp) */
/* ----- */
extern FILE *lfp;
/* Deleted T.Kato 02.10.23 for warning
!FILE *fopen ();
Deleted end */

/* Added t.Kato 02.10.24 for Delivery logging file
control */
extern int iflg; /* Delivery log initialize flag */
/* Added end */
extern int proc_no;
extern int logon;
extern int new_init;
extern int pay_init;
extern int ord_init;

#ifdef DEL_ORA81
extern int del_init;
#else
extern int del_init_oci;
extern int del_init_plsql;
#endif

extern int sto_init;
extern int res_init;

extern int execstatus;
extern int errcode;

extern OCIEnv *tpcenv;
extern OCIServer *tpcsrv;

```

```

extern OCIError *errhp;
extern OCISvcCtx *tpcsvc;
extern OCISession *tpcusr;
extern OCISmt *curi;

/* for stock-level transaction */
extern int w_id;
extern int d_id;
extern int c_id;

/* Replaced T.kato 03.09.09 Oracle10g tool kit */
#if 0
! extern int threshold;
#endif

#ifdef USE_IEEE_NUMBER
extern float threshold;
#else
extern int threshold;
#endif /* USE_IEEE_NUMBER */
/* Replaced end */

extern int low_stock;

/* for delivery transaction */
extern int del_o_id[10];
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 ub4 datelen;
extern int o_carrier_id;
extern int o_ol_cnt;
extern int ol_supply_w_id[15];
extern int ol_i_id[15];

/* Replaced T.kato 03.09.09 Oracle10g tool kit */
#if 0
! extern int ol_quantity[15];
! extern int ol_amount[15];
#endif

#ifdef USE_IEEE_NUMBER
extern float ol_quantity[15];
extern float ol_amount[15];
#else
extern int ol_quantity[15];
extern int ol_amount[15];
#endif /* USE_IEEE_NUMBER */
/* Replaced end */

extern ub4 ol_del_len[15];
extern text ol_delivery_d[15][11];
/* xnie - begin */
extern OCIRowid *o_rowid;
/* xnie - end */

/* for payment transaction */
extern int c_w_id;
extern int c_d_id;

/* Replaced T.kato 03.09.09 Oracle10g tool kit */
#if 0
! extern int h_amount;
#endif

#ifdef USE_IEEE_NUMBER
extern float h_amount;

```

```

#else
extern int h_amount;
#endif /* USE_IEEE_NUMBER */
/* Replaced end */

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 ub4 sincelen;
extern text c_since_d[11];
extern float c_discount;
extern char c_credit[3];
extern int c_credit_lim;
extern char c_data[201];
extern ub4 hlen;
extern text h_date[20];

/* for new order transaction */

extern int nol_i_id[15];
extern int nol_supply_w_id[15];

/* Replaced T.kato 03.09.09 Oracle10g tool kit */
#if 0
! extern int nol_quantity[15];
! extern int nol_amount[15];
! extern int s_quantity[15];
! extern int i_price[15];
#endif

#ifdef USE_IEEE_NUMBER
extern float nol_quantity[15];
extern float nol_amount[15];
extern float s_quantity[15];
extern float i_price[15];
#else
extern int nol_quantity[15];
extern int nol_amount[15];
extern int s_quantity[15];
extern int i_price[15];
#endif /* USE_IEEE_NUMBER */
/* Replaced end */

extern int nol_quant10[15];
extern int nol_quant19[15];
extern int nol_ytdqty[15];
extern int o_all_local;
extern float w_tax;
extern float d_tax;
/* Deleted T.Kato 02.11.13
!float total_amount;
Deleted end */
extern char i_name[15][25];
extern char brand_gen[15];
extern char brand_generic[15][1];
extern int status;
extern int tracelevel;

extern OCIDate cr_date;
extern OCIDate c_since;
extern OCIDate o_entry_d_base;
extern OCIDate ol_d_base[15];

```

```

extern dvoid *xmem;
/* ----- */
/* (tpccsvr.cpp) */
/* ----- */
/* set up pointers for type casting */
extern struct newstruct *newinfo;
extern struct paystruct *payinfo;
extern struct ordstruct *ordinfo;
extern struct delstruct *delinfo;
extern struct stostruct *stoinfo;

#ifdef AVOID_DEADLOCK
int indx[NITEMS], ordl_cnt;
#endif

:-----:
svrapl/MakeShell
:-----:

#! /bin/sh
cd /home/tpc/client_apl/svrapl
make > make_result.txt 2>&1

:-----:
svrapl/Makefile
:-----:

#-----
-----
# Makefile : Makefile for 3 tier and 2 tier
executing files on Linux.
#
# Created by TSL 2003.12.17
#
# All Right Rserverd, Copyright Co, FUJITSU
LIMITED 2003-2004.
#-----
-----

# GCC compile configurations
AR = ar
ARFLAGS = rv
CFLAGS = -Wall -O2
CC = gcc
LD = gcc

# MACRO definition
#DMACRO = -DTSL -DPLSQLFLAG=1 -DTUX
-DDGLDEF
DMACRO = -DPLSQLFLAG=1 -DTUX

# home directory.
ORADIR = /usr/local/oracle
TUXDIR = /usr/local/BEA/tuxedo8.1
SVRDIR = /home/tpc/client_apl/svrapl
TPDIR = /home/tpc/client_apl/tpapl
COMDIR = /home/tpc/client_apl/common
SVRCOMDIR = $(COMDIR)

# include directory
ORA_INC = -I$(ORADIR)/rdbms/demo -
I$(ORADIR)/rdbms/public
COM_INC = -I$(COMDIR)
TUX_INC = -I$(TUXDIR)/include
TP_INC = -I$(TPDIR)
INCLUDE = $(COM_INC) $(ORA_INC)
$(TUX_INC) $(TP_INC)
OBJDIR = $(SVRDIR)/bin

# target object

```

```

3TIERDIR = /home/tpc/client_apl/svrapl/3tier
COMDIR = /home/tpc/client_apl/common
COMOBS = tpccsvr.o GlobalArea.o
initsvrconfig.o
ALLOBS = $(COMOBS) $(MAIN_WHBOBJ)
$(MAIN_NEWOBJ) $(MAIN_PAYOBJ)
$(MAIN_DELOBJ) \
$(MAIN_STOOBJ) $(MAIN_ORDOBJ)
3TIERLIB = $(3TIERDIR)/libtier.a
COMLIB = $(COMDIR)/libcom.a

# depend on include file.
INCFIL = $(SVRDIR)/tpcc.h
$(SVRDIR)/GlobalArea.h $(SVRDIR)/prototype.h \
$(SVRDIR)/tpccflags.h
$(SVRDIR)/tpcc_info.h $(SVRDIR)/TrnCntrlInfo.h
$(SVRDIR)/tpcc_info.h \
$(COMDIR)/log.h $(COMDIR)/sema.h
$(COMDIR)/forlinux.h $(TPDIR)/SampleInfo.h

#---- transaction or warehouse main object.
MAIN_WHBOBJ = bs-whb.o
MAIN_NEWOBJ = bs-new.o
MAIN_PAYOBJ = bs-pay.o
MAIN_DELOBJ = bs-del.o
MAIN_STOOBJ = bs-sto.o
MAIN_ORDOBJ = bs-ord.o

# tuxedo
TUXLIBS = $(TUXDIR)/lib/libtux.a
$(TUXDIR)/lib/libbuft.a $(TUXDIR)/lib/libfml.a \
$(TUXDIR)/lib/libfml32.a
$(TUXDIR)/lib/libengine.a -pthread -ldl
#TUXLIBS = -L$(TUXDIR)/lib/ -ltux -lbuft -lfml -lfml32
# Oracle
#ORALIB = -L$(ORADIR)/rdbms/demo
#ORALIBS = $(ORADIR)/lib/libocci10.a
#ORALIBS = $(ORADIR)/rdbms/lib/defopt.o
$(ORADIR)/lib/libclntst10.a
#ORALIBS = $(ORADIR)/lib/libclntst10.a

#---- execute file for 3 tier.
TARGET_WHB_3TIER =
$(OBJDIR)/3tier_tpccfmlw
TARGET_NEW_3TIER =
$(OBJDIR)/3tier_tpccfmln
TARGET_PAY_3TIER =
$(OBJDIR)/3tier_tpccfmlp
TARGET_DEL_3TIER =
$(OBJDIR)/3tier_tpccfmlf
TARGET_STO_3TIER =
$(OBJDIR)/3tier_tpccfmls
TARGET_ORD_3TIER =
$(OBJDIR)/3tier_tpccfmlf

3TIERTARGETS = $(TARGET_WHB_3TIER)
$(TARGET_NEW_3TIER)
$(TARGET_PAY_3TIER) \
$(TARGET_DEL_3TIER)
$(TARGET_ORD_3TIER)
TARGETS = $(3TIERTARGETS)

# link library.
#LDFLAGS=-L$(ORACLE_HOME)/rdbms/lib/ -
L$(ORACLE_HOME)/lib/ -dy \
# -L$(ORACLE_HOME)/rdbms/lib/ -
L$(ORACLE_HOME)/lib/ \
# $(ORACLE_HOME)/rdbms/lib/defopt.o -
lclntsh \
# -ldl -lm -pthread -lnsl

```

```

LDFLAGS=-L$(ORACLE_HOME)/rdbms/lib/ -
L$(ORACLE_HOME)/lib/ -dy \
-L$(ORACLE_HOME)/rdbms/lib/ -
L$(ORACLE_HOME)/lib/ \
$(ORACLE_HOME)/rdbms/lib/defopt.o -
lclntsh \
-ldl -lm -pthread -lnsl

$(TARGETS) : $(ALLOBS) $(3TIERLIB)
$(COMLIB)
$(LD) -o $(TARGET_WHB_3TIER)
$(MAIN_WHBOBJ) $(COMOBS) $(COMLIB)
$(3TIERLIB) $(TUXLIBS) $(ORALIBS)
$(LDFLAGS)
$(LD) -o $(TARGET_NEW_3TIER)
$(MAIN_NEWOBJ) $(COMOBS) $(COMLIB)
$(3TIERLIB) $(TUXLIBS) $(ORALIBS)
$(LDFLAGS)
$(LD) -o $(TARGET_PAY_3TIER)
$(MAIN_PAYOBJ) $(COMOBS) $(COMLIB)
$(3TIERLIB) $(TUXLIBS) $(ORALIBS)
$(LDFLAGS)
$(LD) -o $(TARGET_DEL_3TIER)
$(MAIN_DELOBJ) $(COMOBS) $(COMLIB)
$(3TIERLIB) $(TUXLIBS) $(ORALIBS)
$(LDFLAGS)
$(LD) -o $(TARGET_STO_3TIER)
$(MAIN_DELOBJ) $(COMOBS) $(COMLIB)
$(3TIERLIB) $(TUXLIBS) $(ORALIBS)
$(LDFLAGS)
$(LD) -o $(TARGET_ORD_3TIER)
$(MAIN_ORDOBJ) $(COMOBS) $(COMLIB)
$(3TIERLIB) $(TUXLIBS) $(ORALIBS)
$(LDFLAGS)

.SUFFIXES: .o .c
.c.o:
$(CC) -o @$@ -c $(CFLAGS) $(INCLUDE)
$(DMACRO) $<

$(ALLOBS) : $(INCFIL)
$(ALLOBS) : Makefile

clean:
rm $(ALLOBS) $(TARGETS)

.....
svrapl/TrnCntrlInfo.h
.....

/*****
****
*
*
* TPC-C Client Application Program Source
*
*
* Entry Functions *
* Transaction structure object definition.
*
*
* CREATE by TSL 2003.05.16
*
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2003 *
****/

/* ----- */
/* Delivery Struct */
/* ----- */

```

```

struct delctx {
sb2 del_o_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];

ub4 del_o_id_len[NDISTS];
ub4 c_id_len[NDISTS];
int oid_ctx;
int cid_ctx;
OCIBind *olamt_bp;

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 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];

int del_o_id[NDISTS];
int del_d_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];
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;
struct pldelctx {

    ub2 del_d_id_len[NDISTS];
    ub2 del_o_id_len[NDISTS];

    ub2 w_id_len;
    ub2 d_id_len[NDISTS];
    ub2 o_c_id_len[NDISTS];
    ub2 sums_len[NDISTS];
    ub2 carrier_id_len;
    ub2 ordcnt_len;
    ub2 del_date_len;

    int del_o_id[NDISTS];
    int del_d_id[NDISTS];
    int o_c_id[NDISTS];

    /* Replaced T.kato 03.09.09 Oracle10g tool kit */
    #if 0
    /* Replaced T.kato 03.07.18 Replaced New
    Oracle10i tool kit */
    /* int sums[NDISTS]; */
    #ifndef TSL
    ! int sums[NDISTS];
    #else
    ! float sums[NDISTS];
    #endif
    /* Replaced end */
    #endif

    #ifdef USE_IEEE_NUMBER
    float sums[NDISTS];
    #else
    int sums[NDISTS];
    #endif
    /* Replaced end */

    OCIDate del_date;
    int carrier_id;
    int ordcnt;

    ub4 del_o_id_rcnt;
    ub4 del_d_id_rcnt;
    ub4 o_c_id_rcnt;
    ub4 sums_rcnt;

    int retry;
    OCISmt *curp1;
    OCISmt *curp2;
    OCIBind *w_id_bp;
    OCIBind *d_id_bp;
    OCIBind *o_id_bp;
    OCIBind *o_c_id_bp;
    OCIBind *ordcnt_bp;
    OCIBind *sums_bp;
    OCIBind *del_date_bp;
    OCIBind *carrier_id_bp;
    OCIBind *retry_bp;

    int norow;
};
typedef struct pldelctx pldelctx;

```

```

#ifdef DMLRETEL
struct amtctx {
    int ol_amt[NITEMS];
    sb2 ol_amt_ind[NITEMS];
    ub4 ol_amt_len[NITEMS];
    ub2 ol_amt_rcode[NITEMS];
    int ol_cnt;
};
typedef struct amtctx amtctx;
#endif

/* ----- */
/* NewOrder Struct */
/* ----- */
struct newctx {

    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 s_dist_info_len[NITEMS];
    ub2 ol_o_id_len[NITEMS];
    ub2 ol_number_len[NITEMS];
    ub2 s_remote_len[NITEMS];
    ub2 s_quant_len[NITEMS];
    ub2 ol_dist_info_len[NITEMS];
    ub2 s_bg_len[NITEMS];

    int ol_o_id[NITEMS];
    int ol_number[NITEMS];

    /* Replaced T.kato 03.09.09 Oracle10g tool kit */
    #if 0
    ! int s_remote[NITEMS];
    #endif

    #ifdef USE_IEEE_NUMBER
    float s_remote[NITEMS];
    #else
    int s_remote[NITEMS];
    #endif
    /* Replaced end */

    char s_dist_info[NITEMS][25];
    OCISmt *curn1;
    OCIBind *ol_i_id_bp;
    OCIBind *ol_supply_w_id_bp;
    OCIBind *i_price_bp;
    OCIBind *i_name_bp;
    OCIBind *s_bg_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;
    OCISmt *curn2;
    OCIBind *ol_quantity_bp;
    OCIBind *s_remote_bp;
    OCIBind *s_quantity_bp;
    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 *ol_o_id_bp;
OCIBind *ol_amount_bp;

/* Replaced 03.05.15 Argument error
(OCIBNDPL). */
#if 0
! sb2 w_id_len;
#endif
ub2 w_id_len;
/* Replaced end */
ub2 d_id_len;
ub2 c_id_len;
ub2 o_all_local_len;
ub2 o_ol_cnt_len;
ub2 w_tax_len;
ub2 d_tax_len;
ub2 o_id_len;
ub2 c_discount_len;
ub2 c_credit_len;
ub2 c_last_len;
ub2 retries_len;
ub2 cr_date_len;
};

typedef struct newctx newctx;

/* ----- */
/* OrderStatus Struct */
/* ----- */
struct ordctx {

    ub2 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;

    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;

    OCISmt *curo0;
    OCISmt *curo1;
    OCISmt *curo2;
    OCISmt *curo3;
    OCISmt *curo4;
    OCIBind *c_id_bp;
    OCIBind *w_id_bp[4];
    OCIBind *d_id_bp[4];
    OCIBind *c_last_bp[2];
    OCIBind *o_id_bp;
    OCIBind *c_rowid_bp;
    /* Deleted T.Kato 2004.12.21 New Oracle10g
    tool kit */
    /* OCIBind *o_rowid_bp; */
    /* Deleted end */

    OCIDefine *c_rowid_dp;
    OCIDefine *c_last_dp[2];

```

```

OCIDefine *c_id_dp;
OCIDefine *c_first_dp[2];
OCIDefine *c_middle_dp[2];
OCIDefine *c_balance_dp[2];
/* Deleted T.Kato 2004.12.21 New Oracle10g
tool kit */
/* OCIDefine *o_rowid_dp[2];*/
/* Deleted end */
OCIDefine *o_id_dp[2];
OCIDefine *o_entry_d_dp[2];
OCIDefine *o_cr_id_dp[2];
OCIDefine *o_ol_cnt_dp[2];
OCIDefine *o_d_d_dp;
OCIDefine *o_i_id_dp;
OCIDefine *o_supply_w_id_dp;
OCIDefine *o_quantity_dp;
OCIDefine *o_amount_dp;
OCIDefine *o_d_base_dp;
OCIDefine *c_count_dp;
OCIRowid *c_rowid_ptr[100];
OCIRowid *c_rowid_cust;
/* Deleted T.Kato 2004.12.21 New Oracle10g
tool kit */
/* OCIRowid *o_rowid;*/
/* Deleted end */
int cs;
int cust_idx;
int norow;
int rcount;
int somerows;
};

typedef struct ordctx ordctx;

struct defctx
{
boolean reexec;
ub4 count;
};
typedef struct defctx defctx;

/* ----- */
/* Payment Struct */
/* ----- */
struct payctx {
OCIStmt *curpi;
OCIStmt *curp0;
OCIStmt *curp1;
OCIBind *w_id_bp[2];
ub2 w_id_len;

OCIBind *d_id_bp[2];
ub2 d_id_len;

OCIBind *c_w_id_bp[2];
ub2 c_w_id_len;

OCIBind *c_d_id_bp[2];
ub2 c_d_id_len;

OCIBind *c_id_bp[2];
ub2 c_id_len;

OCIBind *h_amount_bp[2];
ub2 h_amount_len;

OCIBind *c_last_bp[2];
ub2 c_last_len;

OCIBind *w_street_1_bp[2];
ub2 w_street_1_len;

OCIBind *w_street_2_bp[2];
ub2 w_street_2_len;

```

```

OCIBind *w_city_bp[2];
ub2 w_city_len;

OCIBind *w_state_bp[2];
ub2 w_state_len;

OCIBind *w_zip_bp[2];
ub2 w_zip_len;

OCIBind *d_street_1_bp[2];
ub2 d_street_1_len;

OCIBind *d_street_2_bp[2];
ub2 d_street_2_len;

OCIBind *d_city_bp[2];
ub2 d_city_len;

OCIBind *d_state_bp[2];
ub2 d_state_len;

OCIBind *d_zip_bp[2];
ub2 d_zip_len;

OCIBind *c_first_bp[2];
ub2 c_first_len;

OCIBind *c_middle_bp[2];
ub2 c_middle_len;

OCIBind *c_street_1_bp[2];
ub2 c_street_1_len;

OCIBind *c_street_2_bp[2];
ub2 c_street_2_len;

OCIBind *c_city_bp[2];
ub2 c_city_len;

OCIBind *c_state_bp[2];
ub2 c_state_len;

OCIBind *c_zip_bp[2];
ub2 c_zip_len;

OCIBind *c_phone_bp[2];
ub2 c_phone_len;

OCIBind *c_since_bp[2];
ub2 c_since_len;

OCIBind *c_credit_bp[2];
ub2 c_credit_len;

OCIBind *c_credit_lim_bp[2];
ub2 c_credit_lim_len;

OCIBind *c_discount_bp[2];
ub2 c_discount_len;

OCIBind *c_balance_bp[2];
ub2 c_balance_len;

OCIBind *c_data_bp[2];
ub2 c_data_len;

OCIBind *h_date_bp[2];
ub2 h_date_len;

OCIBind *retries_bp[2];
ub2 retries_len;

OCIBind *cr_date_bp[2];
ub2 cr_date_len;

```

```

OCIBind *byln_bp[2];
ub2 byln_len;
};

typedef struct payctx payctx;

/* ----- */
/* StockLevel Struct */
/* ----- */
struct stockx {
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 stockx stockx;

.....
svrap/bs-del.c
.....

#include <stdio.h>
#include <xa.h>
#include <atmi.h>

#if defined(__cplusplus)
extern "C" {
#endif
extern int _tmrunserver _((int));
//extern void OPSTUXSERVER _((TPSVCINFO
*));
//extern void TPCC _((TPSVCINFO *));
extern void DELIVERY _((TPSVCINFO *));
#if defined(__cplusplus)
}
#endif

//static struct tmdspchtbl_t _tmdspchtbl[] = {
// { "OPSTUXSERVER", "OPSTUXSERVER",
(void (*) _((TPSVCINFO *)) OPSTUXSERVER,
0, 0 },
// { NULL, NULL, NULL, 0, 0 }
//};
static struct tmdspchtbl_t _tmdspchtbl[] = {
{ "DELIVERY", "DELIVERY", (void (*)
_((TPSVCINFO *)) DELIVERY, 0, 0 },
{ NULL, NULL, NULL, 0, 0 }
};

#ifdef _TMDLLIMPORT
#define _TMDLLIMPORT
#endif

_TMDLLIMPORT extern struct xa_switch_t
tmnull_switch;

struct tmsvargs_t tmsvargs = {
NULL,
&_tmdspchtbl[0],
0,
tpsvrinit,
tpsvrdone,

```

```

    _tmrunserver, /* PRIVATE */
    NULL, /* RESERVED */
    NULL, /* RESERVED */
    NULL, /* RESERVED */
    NULL /* RESERVED */
};

struct tmsvrargs_t *
#ifdef _TMPROTOTYPES
_tmgetsvrargs(void)
#else
_tmgetsvrargs()
#endif
{
    tmsvrargs.xa_switch = &tnull_switch;
    return(&tmsvrargs);
}

int
#ifdef _TMPROTOTYPES
main(int argc, char **argv)
#else
main(argc,argv)
int argc;
char **argv;
#endif
{
#ifdef TMMAINEXIT
#include "mainexit.h"
#endif

    return(_tmstartserver( argc, argv,
_tmgetsvrargs()));
}

.....:
svrapl/bs-new.c
.....:

#include <stdio.h>
#include <xa.h>
#include <atmi.h>

#if defined(__cplusplus)
extern "C" {
#endif
extern int _tmrunserver (int);
//extern void OPSTUXSERVER _((TPSVCINFO
*));
//extern void TPCC _((TPSVCINFO *));
extern void NEWORDER (TPSVCINFO *);

#if defined(__cplusplus)
}
#endif

//static struct tmdspchtbl_t _tmdspchtbl[] = {
// { "OPSTUXSERVER", "OPSTUXSERVER",
(void *) _((TPSVCINFO **)) OPSTUXSERVER,
0, 0 },
// { NULL, NULL, NULL, 0, 0 }
//};
static struct tmdspchtbl_t _tmdspchtbl[] = {
{ "NEWORDER", "NEWORDER", (void *)
_((TPSVCINFO **)) NEWORDER, 0, 0 },
{ NULL, NULL, NULL, 0, 0 }
};

#ifdef _TMDLLIMPORT
#define _TMDLLIMPORT
#endif

```

```

_TMDLLIMPORT extern struct xa_switch_t
tnull_switch;

struct tmsvrargs_t tmsvrargs = {
    NULL,
    &_tmdspchtbl[0],
    0,
    tpsvrinit,
    tpsvrdone,
    _tmrunserver, /* PRIVATE */
    NULL, /* RESERVED */
    NULL, /* RESERVED */
    NULL, /* RESERVED */
    NULL /* RESERVED */
};

struct tmsvrargs_t *
#ifdef _TMPROTOTYPES
_tmgetsvrargs(void)
#else
_tmgetsvrargs()
#endif
{
    tmsvrargs.xa_switch = &tnull_switch;
    return(&tmsvrargs);
}

int
#ifdef _TMPROTOTYPES
main(int argc, char **argv)
#else
main(argc,argv)
int argc;
char **argv;
#endif
{
#ifdef TMMAINEXIT
#include "mainexit.h"
#endif

    return(_tmstartserver( argc, argv,
_tmgetsvrargs()));
}

.....:
svrapl/bs-ord.c
.....:

#include <stdio.h>
#include <xa.h>
#include <atmi.h>

#if defined(__cplusplus)
extern "C" {
#endif
extern int _tmrunserver _((int));
//extern void OPSTUXSERVER _((TPSVCINFO
*));
//extern void TPCC _((TPSVCINFO *));
extern void ORDERSTATUS _((TPSVCINFO *));
#if defined(__cplusplus)
}
#endif

//static struct tmdspchtbl_t _tmdspchtbl[] = {
// { "OPSTUXSERVER", "OPSTUXSERVER",
(void *) _((TPSVCINFO **)) OPSTUXSERVER,
0, 0 },
// { NULL, NULL, NULL, 0, 0 }
//};
static struct tmdspchtbl_t _tmdspchtbl[] = {
{ "ORDERSTATUS", "ORDERSTATUS", (void *)
_((TPSVCINFO **)) ORDERSTATUS, 0, 0 },
{ NULL, NULL, NULL, 0, 0 }
};

```

```

};

#ifdef _TMDLLIMPORT
#define _TMDLLIMPORT
#endif

_TMDLLIMPORT extern struct xa_switch_t
tnull_switch;

struct tmsvrargs_t tmsvrargs = {
    NULL,
    &_tmdspchtbl[0],
    0,
    tpsvrinit,
    tpsvrdone,
    _tmrunserver, /* PRIVATE */
    NULL, /* RESERVED */
    NULL, /* RESERVED */
    NULL, /* RESERVED */
    NULL /* RESERVED */
};

struct tmsvrargs_t *
#ifdef _TMPROTOTYPES
_tmgetsvrargs(void)
#else
_tmgetsvrargs()
#endif
{
    tmsvrargs.xa_switch = &tnull_switch;
    return(&tmsvrargs);
}

int
#ifdef _TMPROTOTYPES
main(int argc, char **argv)
#else
main(argc,argv)
int argc;
char **argv;
#endif
{
#ifdef TMMAINEXIT
#include "mainexit.h"
#endif

    return(_tmstartserver( argc, argv,
_tmgetsvrargs()));
}

.....:
svrapl/bs-pay.c
.....:

#include <stdio.h>
#include <xa.h>
#include <atmi.h>

#if defined(__cplusplus)
extern "C" {
#endif
extern int _tmrunserver _((int));
//extern void OPSTUXSERVER _((TPSVCINFO
*));
//extern void TPCC _((TPSVCINFO *));
extern void PAYMENT _((TPSVCINFO *));
#if defined(__cplusplus)
}
#endif

//static struct tmdspchtbl_t _tmdspchtbl[] = {
// { "OPSTUXSERVER", "OPSTUXSERVER",
(void *) _((TPSVCINFO **)) OPSTUXSERVER,
0, 0 },

```

```

// { NULL, NULL, NULL, 0, 0 }
//};
static struct tmdspchtbl_t_tmdspchtbl[] = {
    { "PAYMENT", "PAYMENT", (void *)
      _((TPSVCINFO *)) PAYMENT, 0, 0 },
    { NULL, NULL, NULL, 0, 0 }
};

#ifndef _TMDLLIMPORT
#define _TMDLLIMPORT
#endif

_TMDLLIMPORT extern struct xa_switch_t
tmnull_switch;

struct tmsvrargs_t tmsvrargs = {
    NULL,
    &tmdspchtbl[0],
    0,
    tpsvrinit,
    tpsvrdone,
    _tmrunserver, /* PRIVATE */
    NULL, /* RESERVED */
    NULL, /* RESERVED */
    NULL, /* RESERVED */
    NULL /* RESERVED */
};

struct tmsvrargs_t *
#ifdef _TMDLLIMPORT
#else
#endif
_tmgetsvrargs(void)
#else
_tmgetsvrargs()
#endif
{
    tmsvrargs.xa_switch = &tmnull_switch;
    return(&tmsvrargs);
}

int
#ifdef _TMDLLIMPORT
main(int argc, char **argv)
#else
main(argc,argv)
int argc;
char **argv;
#endif
{
#ifdef TMMAINEXIT
#include "mainexit.h"
#endif

    return( _tmstartserver( argc, argv,
        _tmgetsvrargs()));
}

.....
svrapl/bs-sto.c
.....

#include <stdio.h>
#include <xa.h>
#include <atmi.h>

#ifdef __cplusplus
extern "C" {
#endif
extern int _tmrunserver _((int));
//extern void OPSTUXSERVER _((TPSVCINFO
*));
//extern void TPCC _((TPSVCINFO *));
extern void STOCKLVL _((TPSVCINFO *));
#ifdef __cplusplus
}

```

```

#endif

//static struct tmdspchtbl_t_tmdspchtbl[] = {
//    { "OPSTUXSERVER", "OPSTUXSERVER",
//      (void *) _((TPSVCINFO *)) OPSTUXSERVER,
//      0, 0 },
//    { NULL, NULL, NULL, 0, 0 }
//};

static struct tmdspchtbl_t_tmdspchtbl[] = {
    { "STOCKLVL", "STOCKLVL", (void *)
      _((TPSVCINFO *)) STOCKLVL, 0, 0 },
    { NULL, NULL, NULL, 0, 0 }
};

#ifndef _TMDLLIMPORT
#define _TMDLLIMPORT
#endif

_TMDLLIMPORT extern struct xa_switch_t
tmnull_switch;

struct tmsvrargs_t tmsvrargs = {
    NULL,
    &tmdspchtbl[0],
    0,
    tpsvrinit,
    tpsvrdone,
    _tmrunserver, /* PRIVATE */
    NULL, /* RESERVED */
    NULL, /* RESERVED */
    NULL, /* RESERVED */
    NULL /* RESERVED */
};

struct tmsvrargs_t *
#ifdef _TMDLLIMPORT
#else
#endif
_tmgetsvrargs(void)
#else
_tmgetsvrargs()
#endif
{
    tmsvrargs.xa_switch = &tmnull_switch;
    return(&tmsvrargs);
}

int
#ifdef _TMDLLIMPORT
main(int argc, char **argv)
#else
main(argc,argv)
int argc;
char **argv;
#endif
{
#ifdef TMMAINEXIT
#include "mainexit.h"
#endif

    return( _tmstartserver( argc, argv,
        _tmgetsvrargs()));
}

.....
svrapl/bs-whb.c
.....

#include <stdio.h>
#include <xa.h>
#include <atmi.h>

#include <string.h>
#include "forlinux.h"
#include "log.h"

```

```

#ifdef __cplusplus
extern "C" {
#endif
extern int _tmrunserver _((int));
extern void OPSTUXSERVER _((TPSVCINFO
*));
#ifdef __cplusplus
}
#endif

static struct tmdspchtbl_t_tmdspchtbl[] = {
    { "OPSTUXSERVER", "OPSTUXSERVER",
      (void *) _((TPSVCINFO *)) OPSTUXSERVER,
      0, 0 },
    { NULL, NULL, NULL, 0, 0 }
};

#ifndef _TMDLLIMPORT
#define _TMDLLIMPORT
#endif

_TMDLLIMPORT extern struct xa_switch_t
tmnull_switch;

struct tmsvrargs_t tmsvrargs = {
    NULL,
    &tmdspchtbl[0],
    0,
    tpsvrinit,
    tpsvrdone,
    _tmrunserver, /* PRIVATE */
    NULL, /* RESERVED */
    NULL, /* RESERVED */
    NULL, /* RESERVED */
    NULL /* RESERVED */
};

struct tmsvrargs_t *
#ifdef _TMDLLIMPORT
#else
#endif
_tmgetsvrargs(void)
#else
_tmgetsvrargs()
#endif
{
    tmsvrargs.xa_switch = &tmnull_switch;
    return(&tmsvrargs);
}

int
#ifdef _TMDLLIMPORT
main(int argc, char **argv)
#else
main(argc,argv)
int argc;
char **argv;
#endif
{
#ifdef TMMAINEXIT
#include "mainexit.h"
#endif

    return( _tmstartserver( argc, argv,
        _tmgetsvrargs()));
}

.....
svrapl/initsvrconfig.c
.....

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

```

```

* TPC-C Client Application Program Source
*
* Entry Functions
* (1) GetConfigFileInfo
*
* CREATE by TSL 2003.12.19
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2003-2004 *
*****/
#include "forlinux.h"
#include <unistd.h>
#include "tpcc.h"
#include "tpcc_info.h"
#include "GlobalArea.h"
#include "log.h"
#include "sema.h"
#include "prototype.h"
#include "shmem.h"
#include "SampleInfo.h"
/* Global area for sampling. */
MAC_SampleGlobalArea;
/*****
* Get configuration file information.
*
* Return Value
* None
*****/
int GetConfFileInfo_GetStr(char* section_name,
char* key_name, char* str);

void GetConfFileInfo()
{
    /* Check INI file exist */
    if (access(GLB_ConfigFilePath, 0x00) != 0) {
        /* INI file no exist, using default value */
        TpcUserLog(LOG_LCK, "INI file nothing,
using default value");
        strcpy(GLB_LogFilePath,
DEFAULT_SVRAPL_LOG_PATH);
        return;
    }

    /* Get execution informations
*/
    /* If undefined key and illegal value, using
default value */
    if (GetConfFileInfo_GetStr("SVRAPL_INFO",
"LogPath", GLB_LogFilePath) != 0) {
        strcpy(GLB_LogFilePath,
DEFAULT_SVRAPL_LOG_PATH);
    }
}
/*****/
/* Get information in the CONFIG file for string
value */
/*****/
int GetConfFileInfo_GetStr(char* section_name,
char* key_name, char* str) {

    int i;
    char value_buf[1024];

    for (i = 0; i < 3; i++) {
        GetPrivateProfileString(section_name,
key_name, "",

```

```

value_buf, sizeof(value_buf),
GLB_ConfigFilePath);
    if (value_buf[0] == "") {
        /* if Key is nothing, retry getting */
        continue;
    }
    break;
}
#endif
#ifdef PUT_INF_LOG
    TpcUserLog(LOG_LCK, "CONFIG file
information [%s %s]=[%s]", section_name,
key_name, value_buf);
#endif
    if (value_buf[0] == "") {
        /* Target key was nothing */
        return (-1);
    }
    strcpy(str, value_buf);
    return(strlen(value_buf));
}

/*****
* Initialize configuration information
*
* Return Value
* none.
*****/
void InitSvrConfig(char* path) {

    char work_path[MAX_PATH];
    int i;

    /* Initialize share memory for sampling of
svrapl */
    MAC_SampleInitParent;

    /* Get configuration information (set to global
area) */
    strcpy(GLB_ConfigFilePath, path);

    /* Set default log path */
    strcpy(GLB_LogFilePath,
DEFAULT_SVRAPL_LOG_PATH);

    GetConfFileInfo();

    TpcUserLog(LOG_LCK, "InitSvrConfig start
\n");

    /* Initialize SVRAPL semaphore for log */
    strcpy(work_path, GLB_LogFilePath);
    for(i = strlen(work_path) - 1; i > 0 &&
work_path[i] != '/' ; i--);
    work_path[i] = '\0';

    if ((GLB_LogSemId = InitSem(work_path,
SEM_SVRAPL_PROJID)) == -1) {
        TpcUserLog(LOG_LCK, "InitSem() faile for
SvrApl log\n");
        return;
    }

    return;
}

/*****
svrapl/log_level.h
/*****

```

```

*
* TPC-C Client Application Program Source
*
* CREATE by TSL 2003.02.07
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2003 *
*****/
#define PUT_INF_LOG //
Information log
#define PUT_FUNC_ENTRY_LOG //
Function entry point log
//define PUT_FUNC_EXIT_LOG //
Function exit log

/* Function entry point log macro */
#ifdef PUT_FUNC_ENTRY_LOG
#define MAC_PutFuncEntryLog(func)
TpcUserLog(LOG_INF, ">>>>> "func" start
>>>>>");
#else
#define MAC_PutFuncEntryLog(func) ;
#endif

/* Function exit point log */
#ifdef PUT_FUNC_EXIT_LOG
#define MAC_PutFuncExitLog(func)
TpcUserLog(LOG_INF, "<<<<< "func" end
<<<<<");
#else
#define MAC_PutFuncExitLog(func) ;
#endif

/*****
svrapl/prototype.h
/*****
*
* TPC-C Client Application Program Source
*
* Entry Functions
* Function prototype definition.
*
* CREATE by TSL 2003.12.11
*
* All Right Reserved, Copyright Co. FUJITSU
LIMITED 2003-2004 *
*****/
#include "tpccflags.h"

/*****/
/* Prototype */
/*****/
#ifdef DEL_ORA8I
int tkvcdfinit ();
int tkvcninit ();
int tkvcoint ();
int tkvcpinit (void);
int tkvcsinit ();
int tkvcd ();

```

```

int tkvcn ();
int tkvcs ();
int tkvcp ();
int tkvco ();
void tkvcddone ();
void tkvcndone ();
void tkvcsdone ();
void tkvcpdone ();
void tkvcodone ();
#else
int tkvcinit (int plsqli);
int tkvcninit ();
int tkvcinit ();
int tkvcpinit (void);
int tkvcsinit ();
int tkvcd (int plsqli);
int tkvcn ();
int tkvcs ();
int tkvcp ();
int tkvco ();
void tkvcddone (int plsqli);
void tkvcndone ();
void tkvcsdone ();
void tkvcpdone ();
void tkvcodone ();
#endif

/* pldel */
void shiftdata(int from);

/* tpccpl Prototype */
int TPCinit (int id, char* uid, char* pwd);
int TPCnew (struct newstruct* str);
int TPCdel (struct delstruct* str);
int TPCpay (struct paystruct* str);
int TPCord (struct ordstruct* str);
int TPCsto (struct stostruct* str);
void TPCexit (void);

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

#ifndef AVOID_DEADLOCK
/* Added T.Kato 02.11.22 */
void swap_item(struct newstruct *str, int i, int j);
void q_sort_item(int *arr, struct newstruct *str, int left, int right);
/* Added End */
void swap(struct newstruct *str, int i, int j);
void q_sort(int *arr, struct newstruct *str, int left, int right);
#endif

/* Added Hayashi 03.12.24 */
void InitSvrConfig(char *);
int GetPrivateProfileString(char* section_name,
char* key_name,
char* default_str, char*
key_data,
int buf_size, char*
file_name);
/* Added End */

.....
svrapl/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
*/

#include "log.h"

/* Deleted 03.05.19 No use. */
#if 0
!typedef struct cda_def csrdef;
!typedef struct cda_def ldadef;
#endif
/* Deleted end */

/* TPC-C transaction functions */

/* 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

/* Modified by TSL --- BEGIN ---2006.03.17 */
/* #define FULLDATE "dd-mon-yy.hh24:mi:ss" */

#define FULLDATE "dd-mm-yyyy.hh24:mi:ss"
/* Modified by TSL --- END ---2006.03.17 */

#define SHORTDATE "dd-mm-yyyy"

#define DELRT 80.0

/* Deleted 03.05.19 No use. */
#if 0
extern int tkvcss (); /* for alter session to get
memory size and trace */
extern boolean multitrans;
#endif
/* Deleted end */
/* Deleted 03.05.16 For warning */
#if 0
extern int ord_init;
#endif
/* Deleted end */

/* Deleted 03.05.19 No use. */
#if 0
extern void errprt ();
#endif
/* Deleted end */

/* Added T.Kato 2003.03.25 for debug */
extern void DbgLog(char* form_dat, int arg);
#ifndef DGLDEF
#define DBGLOG(format_data, arg)
TpcUserLog(LOG_INF,format_data, arg)
#else
#define DBGLOG(format_data, arg)
#endif

#ifndef 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(x) (x * )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(LOG_FILE_LINE,(errp),(function));

#define OCIBND(stmp, bndp, errp, sqlvar, progvl, progvl, ftype)\
    ocierror(LOG_FILE_LINE,(errp), \

OCIHandleAlloc((stmp),(dvoid*)&(bndp),OCI_H
TYPE_BIND,0,(dvoid**0)); \
    ocierror(LOG_FILE_LINE, (errp), \
        OCIBindByName((stmp), &(bndp), (errp), \
            (text *) (sqlvar), strlen((sqlvar)), \
            (progvl), (progvl), \
            (ftype),0,0,0,0,OCI_DEFAULT));

/* bind arrays for sql */
#define
OCIBNDRA(stmp,bndp,errp,sqlvar,progvl,progvl,f
type,indp,alen,arcode) \
    DISCARD ocierror(LOG_FILE_LINE,(errp), \

OCIHandleAlloc((stmp),(dvoid*)&(bndp),OCI_H
TYPE_BIND,0,(dvoid**0)); \
    DISCARD ocierror(LOG_FILE_LINE,(errp), \

OCIBindByName((stmp),&(bndp),(errp),(text
*)(sqlvar),strlen((sqlvar)),\

(progvl),(progvl),(ftype),(indp),(alen),(arcode),0,0,
OCI_DEFAULT));

/* use with callback data */
#define
OCIBNDRAD(stmp,bndp,errp,sqlvar,progvl,ftype
,indp,ctxp,\
    cbf_nodata,cbf_data) \
    DISCARD ocierror(LOG_FILE_LINE,(errp), \

OCIHandleAlloc((stmp),(dvoid*)&(bndp),OCI_H
TYPE_BIND,0,(dvoid**0)); \
    DISCARD ocierror(LOG_FILE_LINE,(errp), \

OCIBindByName((stmp),&(bndp),(errp),(text
*)(sqlvar), \
    strlen((sqlvar)),0,(progvl),(ftype), \

indp,0,0,0,0,OCI_DATA_AT_EXEC)); \
    DISCARD ocierror(LOG_FILE_LINE,(errp), \

OCIBindDynamic((bndp),(errp),(ctxp),(cbf_nodat
a),(ctxp),(cbf_data));

/* bind in/out for plsql without indicator and rcode
*/
#define
OCIBNDPL(stmp,bndp,errp,sqlvar,progvl,progvl,f
type,alen) \
    DISCARD ocierror(LOG_FILE_LINE,(errp), \

OCIHandleAlloc((stmp),(dvoid*)&(bndp),OCI_H
TYPE_BIND,0,(dvoid**0)); \
    DISCARD ocierror(LOG_FILE_LINE,(errp), \

OCIBindByName((stmp),&(bndp),(errp),(CONST
text *) (sqlvar), \
    (sb4)strlen((CONST char *)
(sqlvar)),(void *) (progvl), \

(progvl),(ftype),NULL,(alen),NULL,(ms),(cu),OCI
_DEFAULT));

/* bind in/out values for plsql with indicator and
rcode */
#define
OCIBNDRAA(stmp,bndp,errp,sqlvar,progvl,progvl
l,ftype,indp,alen,arcode,\
    ms,cu) \
    ocierror(LOG_FILE_LINE, (errp), \

OCIHandleAlloc((stmp),(dvoid*)&(bndp),OCI_H
TYPE_BIND,0,(dvoid**0)); \
    ocierror(LOG_FILE_LINE,(errp),\

OCIBindByName((stmp),&(bndp),(errp),(text
*)(sqlvar),strlen((sqlvar)),\

(progvl),(progvl),(ftype),(indp),(alen),(arcode),(ms
),(cu),OCI_DEFAULT));

#define
OCIDEFINE(stmp,dfnp,errp,pos,progvl,progvl,ft
ype)\

OCIDefineByPos((stmp),&(dfnp),(errp),(pos),(pro
gv),(progvl),(ftype),\
    0,0,0,OCI_DEFAULT);

#define
OCIDEF(stmp,dfnp,errp,pos,progvl,progvl,ftype) \
OCIHandleAlloc((stmp),(dvoid*)&(dfnp),OCI_HT
YPE_DEFINE,0,\
    (dvoid**0)); \

OCIDefineByPos((stmp),&(dfnp),(errp),(pos),(pro
gv),\
    (progvl),(ftype),(indp),(alen),\
    (arcode),OCI_DEFAULT);

#define
OCIDFNRA(stmp,dfnp,errp,pos,progvl,progvl,ftyp
e,indp,alen,arcode) \

OCIHandleAlloc((stmp),(dvoid*)&(dfnp),OCI_HT
YPE_DEFINE,0,\
    (dvoid**0)); \

OCIDefineByPos((stmp),&(dfnp),(errp),(pos),(pro
gv),\
    (progvl),(ftype),(indp),(alen),\
    (arcode),OCI_DEFAULT);

#define
OCIDFNDDYN(stmp,dfnp,errp,pos,progvl,progvl,ft
ype,indp,ctxp,cbf_data) \
    ocierror(LOG_FILE_LINE,(errp), \

OCIHandleAlloc((stmp),(dvoid*)&(dfnp),OCI_HT
YPE_DEFINE,0,\
    (dvoid**0)); \
    ocierror(LOG_FILE_LINE,(errp), \

OCIDefineByPos((stmp),&(dfnp),(errp),(pos),(pro
gv),(progvl),(ftype),\
    (indp),NULL,NULL,
OCI_DYNAMIC_FETCH)); \
    ocierror(LOG_FILE_LINE,(errp), \

OCIDefineDynamic((dfnp),(errp),(ctxp),(cbf_data
)));

/* Deleted T.Kato 02.10.23 Overrapped
tpcc_info.h */
#if 0
/* New order */
!struct newinstruct {
! 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 ordinstruc {
! 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 ordinstruc ordin;
! struct ordoutstruct ordout;
!};
!
!
!/* Delivery */
!
!struct delinstruc {
! int w_id;
! int o_carrier_id;
! double qtime;
! int in_timing_int;
! int plsqflag;
!};
!
!struct deloutstruct {
! int terror;
! int retry;
!};
!
!
!struct delstruct {
! struct delinstruc 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

.....:
svrapl/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;
};

/* Replaced T.Kato 02.10.24 for TPAPL interface */
#if 0
! double qtime;
! int in_timing_int;
#endif

    long startsec;
    long startusec;
/* Replaced end */

};

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

.....
svrapi/tpccflags.h
.....

#define DMLRETDEL

/* Deleted T.Kato 2004.12.21 New Oracle10g
tool kit */
#if 0
#ifndef TSL
#define USE_IEEE_NUMBER
#endif
#endif

.....
svrapi/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 <sys/time.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 <unistd.h>

#include "forlinux.h"
#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
#include "GlobalArea.h"

```

```

#include "prototype.h"
#include "sema.h"
#include "shmem.h"
#include "SampleInfo.h"

#ifdef TUX

#include <lmenv.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

#if 0
// 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 );
}
#endif

/* FUNCTION: int tpsvrit (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
* txntype: transaction type the server
will be running
* RETURNS: None
*
* COMMENTS: None
*/

int tpsvrit (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]);

/* Set default log path */
strcpy(GLB_LogFilePath,
DEFAULT_SVRAPL_LOG_PATH);
TpccUserLog(LOG_LCK, "Start tpsvrit");

/* Initialize semaphore and log. */
InitSvrConfig(TPCC_CONF_FILE);

#ifdef TUX

srand ( (unsigned)time( NULL ) );
svrcnt = rand();

/* send 6 for all txns to be init'd */
/* fix uid and pwd for now, pull out later */
/* not passing parameters through TUX yet
*/

#if 0 /* Replaced 2003/12/12 adjust
arguments */
! if (TPCinit( svrcnt, "tpcc", "tpcc", 6) ) {
#else
if (TPCinit( svrcnt, "tpcc", "tpcc" ) ) {
#endif
    TpccUserLog(LOG_FILE_INF, " FAILED
to init all txns types");
    return (-1);
}

    TpccUserLog(LOG_INF, "Finished
TPCinit(tpsvrit)");

    return 0;

#else // ifdef TUX for topend

#if 0 /* Replaced 2003/12/12 adjust
arguments */
! if (TPCinit( svrid, uid, pwd, txntype) ) {
#else
if (TPCinit( svrid, uid, pwd) ) {
#endif
    TpccUserLog(LOG_INF, "Failed in TPCinit
(probably connecting).");
    exit (1);
}

    TpccUserLog(LOG_INF, "Finished TPCinit");

    return (1);
#endif

}

void tpsvrdone ()

{
    TpccUserLog(LOG_INF, "Start tpsvrdone");

#if 0 /* Replaced 2003/12/12 adjust arguments */
! TPCexit (0);
#else
    TPCexit ();
#endif
}

```

```

    TpccUserLog(LOG_INF, "Finished
TPCexit(tpsvrdone)");
}

/* 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 (TPSCINFO *msg)
#endif

{

#ifdef TOP
int result;

result = TPCnew(neword);

return result;

#else // for Tuxedo

    MAC_SampleWork; // Sampling area

    newinfo = (struct newstruct *) msg->data;

    MAC_SampleStartTime; // Start sampling.
    newinfo->retval = TPCnew (newinfo); // set
return value to 0 or -1
    // Finish sampling.
    MAC_SampleDBSrvResp(RspTimeNewOrder,
MaxRspTimeNewOrder,
SMaxRspTimeNewOrder, NumNewOrder);

    // 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

    MAC_SampleWork; // Sampling area

    payinfo = (struct paystruct *) msg->data;
    MAC_SampleStartTime; // Start sampling.
    payinfo->retval = TPCpay (payinfo); // set
return value to 1 or 0 or -1
    // Finish sampling.
    MAC_SampleDBSrvResp(RspTimePayment,
MaxRspTimePayment, SMaxRspTimePayment,
NumPayment);

    // 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
*      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

    MAC_SampleWork; // Sampling area

    delinfo = (struct delstruct *) msg->data;

    MAC_SampleStartTime; // Start sampling.
    delinfo->retval = TPCdel (delinfo); // set return
value to 0 or -1
    MAC_SampleDBSrvRespDel(); // Finish
sampling.

    // always return tpreturn success - let client
side poll retval for actual error
    tpreturn (TPSUCCESS, 0, (char *) delinfo,
sizeof (struct delstruct), 0);

#endif

}

/* Replaced T.kato 02.10.28 old version name
used */
#if 0
/* FUNCTION: int STOCKLEVEL(CLIENTDATA
*jobData, StockLevelData *stodata, int
deadlock)*/
#endif
/* FUNCTION: int STOCKLVL(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
*
*/

/* Replaced T.kato 02.10.28 old vaersion name
used */
#if 0
#ifdef TOP
int STOCKLEVEL(CLIENTDATA *jobData,
StockLevelData *stodata, int deadlock)
#else

```

```

Ivoid STOCKLEVEL (TPSVCINFO *msg)
#endif
#endif

#ifdef TOP
int STOCKLVL(CLIENTDATA *jobData,
StockLevelData *stodata, int deadlock)
#else
void STOCKLVL (TPSVCINFO *msg)
#endif
/* Replaced end */

{

#ifdef TOP

int result;

result = TPCsto(stodata);

return result;

#else

MAC_SampleWork; // Sampling area

stoinfo = (struct stostruct *) msg->data;
MAC_SampleStartTime; // Start sampling.
stoinfo->retval = TPCsto (stoinfo); // set return
value to 0 or -1
// Finish sampling
MAC_SampleDBSrvResp(RspTimeStockLevel,
MaxRspTimeStockLevel,
SMaxRspTimeStockLevel, NumStockLevel);

// always return treturn success - let client
side poll retval for actual error
treturn (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

/* Replaced T.Kato 03.03.19 Ununique
STRUCTURE size between Derivery and
StockLevel */
#if 0
! if (msg->len == 928) { // len for neworder
! newinfo = (struct newstruct *) msg->data;
! newinfo->retval = TPCnew (newinfo); // set
return value to 0 or -1
!
! // always return treturn success - let client
side poll retval for actual error
! treturn (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 treturn success - let client
side poll retval for actual error
! treturn (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
!
! // always return treturn success - let client
side poll retval for actual error
! treturn (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 treturn success
- let client side poll retval for actual error
! treturn (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 treturn success - let
client side poll retval for actual error
! treturn (TPSUCCESS, 0, (char *)
stoinfo, sizeof (struct stostruct), 0);
! }
#endif
#endif

```

```

int trx_type = *(int*)msg->data;

MAC_SampleWork; // Sampling area

if (trx_type == 1) { // type for neworder
newinfo = (struct newstruct *) msg->data;

DBGLOG("OPS:[New]Start",0);
MAC_SampleStartTime; // Sampling start
newinfo->retval = TPCnew (newinfo); // set
return value to 0 or -1

MAC_SampleDBSrvResp(RspTimeNewOrder,
MaxRspTimeNewOrder,
SMaxRspTimeNewOrder, NumNewOrder); //
Sampling finish
DBGLOG("OPS:[New]End >%d",newinfo-
>retval);

// always return treturn success - let client
side poll retval for actual error

treturn (TPSUCCESS, 0, (char *) newinfo,
sizeof (struct newstruct), 0);
}
else
if (trx_type == 2) { // type for payment
payinfo = (struct paystruct *) msg->data;
DBGLOG("OPS:[Pay]Start",0);
MAC_SampleStartTime; // Sampling start
payinfo->retval = TPCpay (payinfo); // set
return value to 1 or 0 or -1

MAC_SampleDBSrvResp(RspTimePayment,
MaxRspTimePayment, SMaxRspTimePayment,
NumPayment); // Sampling finish
DBGLOG("OPS:[Pay]End >%d",payinfo-
>retval);

// always return treturn success - let client
side poll retval for actual error
treturn (TPSUCCESS, 0, (char *) payinfo,
sizeof (struct paystruct), 0);
}
else
if (trx_type == 3) { // type for order status
ordinfo = (struct ordstruct *) msg->data;
DBGLOG("OPS:[Ord]Start",0);
MAC_SampleStartTime; // Sampling start
ordinfo->retval = TPCord (ordinfo); // set
return value to 0 or -1

MAC_SampleDBSrvResp(RspTimeOrderStatus,
MaxRspTimeOrderStatus,
SMaxRspTimeOrderStatus, NumOrderStatus); //
Sampling finish
DBGLOG("OPS:[Ord]End >%d",ordinfo-
>retval);

// always return treturn success - let client
side poll retval for actual error
treturn (TPSUCCESS, 0, (char *) ordinfo,
sizeof (struct ordstruct), 0);
}
else
if (trx_type == 4) { // type for delivery
delinfo = (struct delstruct *) msg->data;
DBGLOG("OPS:[Del]Start",0);
MAC_SampleStartTime; // Start sampling.
delinfo->retval = TPCdel (delinfo); // set
return value to 0 or -1

```

```

    MAC_SampleDBSrvRespDel(); // Finish
sampling.
    DBGLOG("OPS:[Del]End >%d",delinfo-
>retval);

    // 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;
    DBGLOG("OPS:[Sto]Start",0);
    MAC_SampleStartTime; // Start sampling.
    stoinfo->retval = TPCsto (stoinfo); // set
return value to 0 or -1

MAC_SampleDBSrvResp(RspTimeStockLevel,
MaxRspTimeStockLevel,
SMaxRspTimeStockLevel, NumStockLevel); //
Finish sampling
    DBGLOG("OPS:[Sto]End >%d",stoinfo-
>retval);

    // always return tpreturn success - let client
side poll retval for actual error
    tpreturn (TPSUCCESS, 0, (char *) stoinfo,
sizeof (struct stostruct), 0);
}
/* Replaced end */

#endif
}

.....
svrapl/3tier/MakeShell
.....

#!/bin/sh
cd /home/tpc/client_apl/svrapl/3tier
make > make_result.txt 2>&1

.....
svrapl/3tier/Makefile
.....

#-----
-----
# Makefile : Makefile for 3 tier library on Linux.
#
# Created by TSL 2003.12.17
#
# All Right Rserverd, Copyright Co, FUJITSU
LIMITED 2003-2004.
#-----
-----

# GCC compile configurations
AR = ar
ARFLAGS = rv
CFLAGS = -Wall -O2
CC = gcc

# MACRO definition
#DMACRO = -DTSL -DPLSQLFLAG=1 -DTUX
DMACRO = -DPLSQLFLAG=1 -DTUX

# home directory.
ORADIR = /usr/local/oracle
TUXDIR = /usr/local/BEA/tuxedo8.1
SVRDIR = /home/tpc/client_apl/svrapl
COMDIR = /home/tpc/client_apl/common

```

```

# include directory
ORA_INC = -$(ORADIR)/rdbms/demo -
I$(ORADIR)/rdbms/public
COM_INC = -$(COMDIR)
SRV_COM_INC = -$(SVRDIR)
TUX_INC = -$(TUXDIR)/include
INCLUDE = $(COM_INC) $(ORA_INC)
$(TUX_INC) $(SRV_COM_INC)
SVRDIR = /home/tpc/client_apl/svrapl

# depend on include file.
INCFILE = $(SVRDIR)/tpcc.h
$(SVRDIR)/GlobalArea.h $(SVRDIR)/prototype.h
\
$(SVRDIR)/tpccflags.h
$(SVRDIR)/tpcc_info.h $(SVRDIR)/TrnCntrlInfo.h
$(SVRDIR)/tpcc_info.h \
$(COMDIR)/log.h $(COMDIR)/sema.h
$(COMDIR)/forlinux.h

# target object
TIER_OBJS = pldel.o plnew.o plord.o
plpay.o plsto.o tpccpl.o
TIER_ARCH_LIB = libtier.a

$(TIER_ARCH_LIB) : $(TIER_OBJS)
$(INCFILE)
$(AR) $(ARFLAGS) $(TIER_ARCH_LIB)
$(TIER_OBJS)

.SUFFIXES: .o .c
.c.o:
$(CC) -o $@ -c $(CFLAGS) $(INCLUDE)
$(DMACRO) <

$(TIER_OBJS) : $(INCFILE)
$(TIER_OBJS) : Makefile

clean:
rm $(TIER_ARCH_LIB) $(TIER_OBJS)

.....
svrapl/3tier/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 "forlinux.h"
#include "log.h"
#include "tpcc.h"
#include "GlobalArea.h"
#include "prototype.h"

#define SQLTXT "BEGIN inittpc.init_del ; END;"

#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 "

#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"

#define SQLTXT4 "UPDATE ordl \
SET ol_delivery_d = :c_date \
WHERE ol_w_id = :w_id AND ol_d_id = :d_id
AND ol_o_id = :o_id \
RETURNING sum(ol_amount) into :ol_amount
"

#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

#ifdef DMLRETDEL
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);
}

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);
    }

#ifdef OLD
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 = actx->ol_cnt + 1;
        *bufpp = &actx->ol_amt[index];
        *indpp = &actx->ol_amt_ind[index];
        actx->ol_amt_len[index] = sizeof(actx->ol_amt[0]);
        *alenp = &actx->ol_amt_len[index];
        *rcodepp = &actx->ol_amt_rcode[index];
        *piecep = OCI_ONE_PIECE;
        if (iter == 1)
            return (OCI_CONTINUE);
        else
            return (OCI_ERROR);
    }
#else
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;
        *bufpp = &actx->ol_amt[index];
        *indpp = &actx->ol_amt_ind[index];
        actx->ol_amt_len[index] = sizeof(actx->ol_amt[0]);
        *alenp = &actx->ol_amt_len[index];
        *rcodepp = &actx->ol_amt_rcode[index];
        *piecep = OCI_ONE_PIECE;

        return (OCI_CONTINUE);
    }
#endif

#ifdef
int tkvcddinit (int plsqliflag)
    {
        text stmbuff[SQL_BUF_SIZE];

        if (plsqliflag)
        {
            pldctx = (pldelctx *) malloc (sizeof(pldelctx));
            DISCARD
            memset(pldctx, (char)0, (ub4)sizeof(pldelctx));
            /* Initialize */
            DISCARD OCCHandleAlloc(tpcenv,
            (dvoid **)&pldctx->curp1, OCI_HTYPE_STMT, 0,
            (dvoid **)0);
            DISCARD sprintf ((char *) stmbuff, SQLTXT);

```

```

            DISCARD OCISmtPrepare(pldctx->curp1,
            errhp, stmbuff,
                (ub4) strlen((char *)stmbuff),
                OCI_NTV_SYNTAX,
                OCI_DEFAULT);
            DISCARD OCIERROR(errhp,
            OCISmtExecute(tpcscv, pldctx->curp1, errhp, 1, 0, NULLP(OCISnapshot),
            NULLP(OCISnapshot),
            OCI_DEFAULT));

            DISCARD OCCHandleAlloc(tpcenv, (dvoid **)
            &pldctx->curp2, OCI_HTYPE_STMT,
            0, (dvoid **)0);
#ifdef (ISO5) || defined(ISO6) ||
defined(ISO8)
            #if defined(ISO5)
                sqlfile("../blocks/tkvcpdel_iso5.sql", stmbuff);
            #endif
            #if defined(ISO6)
                sqlfile("../blocks/tkvcpdel_iso6.sql", stmbuff);
            #endif
            #if defined(ISO8)
                sqlfile("../blocks/tkvcpdel_iso8.sql", stmbuff);
            #endif
            #else
                /* Replaced 04.01.20 TUXEDO Client */
            #if 0
                ! sqlfile("../blocks/tkvcpdel.sql", stmbuff);
            #endif

                sqlfile("../home/tpc/blocks/tkvcpdel.sql", stmbuff);
                /* Replaced end */
            #endif
            DISCARD OCISmtPrepare(pldctx->curp2,
            errhp, stmbuff,
                (ub4)strlen((char *)stmbuff),
                OCI_NTV_SYNTAX, OCI_DEFAULT);
            OCIBNDPL(pldctx->curp2, pldctx->w_id_bp,
            errhp, "w_id",
                ADR(w_id), SIZ(int),
                SOLT_INT, &pldctx->w_id_len);
            OCIBNDPL(pldctx->curp2, pldctx->ordcnt_bp,
            errhp, "ordcnt",
                ADR(pldctx->ordcnt), SIZ(int),
                SOLT_INT, &pldctx->ordcnt_len);

            /* Replaced T.kato 03.07.18 New Oracle10i tool
            kit */
            #if 0
            ! OCIBNDPL(pldctx->curp2, pldctx->del_date_bp, errhp, "now",
            ! dctx->del_date, SIZ(OCIDate),
            SOLT_ODT, &pldctx->del_date_len);
            #endif

            #ifdef TSL
            OCIBNDPL(pldctx->curp2, pldctx->del_date_bp, errhp, "now",
            dctx->del_date, SIZ(OCIDate),
            SOLT_ODT, &pldctx->del_date_len);
            #else
            OCIBNDPL(pldctx->curp2, pldctx->del_date_bp, errhp, "now",
            ADR(pldctx->del_date), SIZ(OCIDate),
            SOLT_ODT, &pldctx->del_date_len);
            #endif
            /* Replaced end */

            OCIBNDPL(pldctx->curp2, pldctx->carrier_id_bp, errhp,
            "carrier_id", ADR(o_carrier_id), SIZ(int),

```

```

            SOLT_INT, &pldctx->carrier_id_len);
            OCIBNDPL(pldctx->curp2, pldctx->d_id_bp,
            errhp, "d_id",
                pldctx->del_d_id, SIZ(int), SOLT_INT,
                pldctx->del_d_id_len,
                NDISTS, &pldctx->del_d_id_rcnt);
            OCIBNDPL(pldctx->curp2, pldctx->o_id_bp,
            errhp, "order_id",
                pldctx->del_o_id, SIZ(int), SOLT_INT,
                pldctx->del_o_id_len, NDISTS,
                &pldctx->del_o_id_rcnt);

            /* Replaced T.kato 03.09.09 Oracle10g tool kit */
            #if 0
            /* Replaced T.kato 03.07.18 New Oracle10i tool
            kit */
            ! #if 0
            ! OCIBNDPL(pldctx->curp2, pldctx->sums_bp, errhp, "sums",
            ! pldctx->sums, SIZ(int), SOLT_INT,
            pldctx->sums_len, NDISTS,
            ! &pldctx->sums_rcnt);
            ! #endif
            !
            ! #ifdef TSL
            ! OCIBNDPL(pldctx->curp2, pldctx->sums_bp, errhp, "sums",
            ! pldctx->sums, SIZ(int), SOLT_INT,
            pldctx->sums_len, NDISTS,
            ! &pldctx->sums_rcnt);
            ! #else
            ! OCIBNDPL(pldctx->curp2, pldctx->sums_bp, errhp, "sums",
            ! pldctx->sums, SIZ(float), SOLT_BFLOAT, pldctx->sums_len, NDISTS,
            ! &pldctx->sums_rcnt);
            ! #endif
            /* Replaced end */
            #endif

            #ifdef USE_IEEE_NUMBER
            OCIBNDPL(pldctx->curp2, pldctx->sums_bp, errhp, "sums",
            pldctx->sums, SIZ(float), SOLT_BFLOAT, pldctx->sums_len, NDISTS,
            &pldctx->sums_rcnt);
            #else
            OCIBNDPL(pldctx->curp2, pldctx->sums_bp, errhp, "sums",
            pldctx->sums, SIZ(int), SOLT_INT,
            pldctx->sums_len, NDISTS,
            &pldctx->sums_rcnt);
            #endif
            /* Replaced end */

            OCIBNDPL(pldctx->curp2, pldctx->o_c_id_bp, errhp, "o_c_id",
            pldctx->o_c_id, SIZ(int), SOLT_INT,
            pldctx->o_c_id_len, NDISTS,
            &pldctx->o_c_id_rcnt);
            OCIBNDPL(pldctx->curp2, pldctx->retry_bp,
            errhp, "retry",
            ADR(pldctx->retry), SIZ(int), SOLT_INT);
        }
    }
    else
    {

```

```

dctx = (delctx *) malloc (sizeof(delctx));
memset(dctx,(char)0,sizeof(delctx));
dctx->norow = 0;
actx = (amtctx *) malloc (sizeof(amtctx));
memset(actx,(char)0,sizeof(amtctx));

OCIHandleAlloc(tpcenv, (dvoid **)&dctx-
>curd1, OCI_HTYPE_STMT, 0,
(dvoid**)0);
DISCARD sprintf ((char *) stmbuf, "%s",
SQLTXT1);
DISCARD OCISmtPrepare(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);

/* open third cursor */

DISCARD OCIHandleAlloc(tpcenv, (dvoid
**)&dctx->curd3, OCI_HTYPE_STMT,
0, (dvoid**)0);
DISCARD sprintf ((char *) stmbuf, SQLTXT3);
DISCARD OCISmtPrepare(dctx->curd3,
errhp, stmbuf, strlen((char *)stmbuf),
OCI_NTV_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);

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);

/* open fourth cursor */

DISCARD OCIHandleAlloc(tpcenv, (dvoid
**)&dctx->curd4, OCI_HTYPE_STMT, 0,
(dvoid**)0);
DISCARD sprintf ((char *) stmbuf, SQLTXT4);
DISCARD OCISmtPrepare(dctx->curd4,
errhp, stmbuf, strlen((char *)stmbuf),
OCI_NTV_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);
OCIBNDRAD(dctx->curd4, dctx->olamt_bp,
errhp, ":o_l_amount",
SIZ(int), SQLT_INT,NULL,
actx,no_data,amt_data);

/* open sixth cursor */

DISCARD OCIHandleAlloc(tpcenv, (dvoid
**)&dctx->curd6, OCI_HTYPE_STMT,
0, (dvoid**)0);
DISCARD sprintf ((char *) stmbuf, SQLTXT6);
DISCARD OCISmtPrepare(dctx->curd6,
errhp, stmbuf, strlen((char *)stmbuf),
OCI_NTV_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);
}
return (0);
}

void shiftdata(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];
}
}

int tkvcd (int plsqflag)
{
/*int i, j;
int i;
/*int rpc,rcount,count;
int rpc,rcount;
int invalid;

if (plsqflag)
{

```

```

pldctx->w_id_len = sizeof (int);
pldctx->carrier_id_len = sizeof (int);
for (i = 0; i < NDISTS; i++)
{
pldctx->del_o_id_len[i] = sizeof(int);
del_o_id[i] = 0;
}
pldctx->del_date_len = DEL_DATE_LEN;
DISCARD memcpy(&pldctx-
>del_date,&cr_date,sizeof(OCIDate));

pldctx->retry=0;

DISCARD OCIERROR(errhp,
OCISmtExecute(tpcsvc,pldctx-
>curp2,errhp,1,0,NULLP(CONST OCISnapshot),
NULLP(OCISnapshot),OCI_DEFAULT));
for (i = 0; i < NDISTS; i++)
{
del_o_id[i] = 0;
}
for (i = 0; (unsigned int)i < pldctx-
>del_o_id_rcnt; i++)
del_o_id[pldctx->del_o_id[i] - 1] = pldctx-
>del_o_id[i];
}
else
{

retry:

invalid = 0;

/* initialization for array operations */

for (i = 0; i < NDISTS; i++)
{
dctx->del_o_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->del_o_id_len[i] = SIZ(dctx-
>del_o_id(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->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));
}

memset(actx,(char)0,sizeof(amtctx));

/* array select from new_order and orders
tables */

execstatus=OCISmtExecute(tpcsvc,dctx-
>curd1,errhp,NDISTS,0,
NULLP(CONST
OCISnapshot),NULLP(OCISnapshot),OCI_DEF
AULT);

```

```

    if((execstatus != OCI_SUCCESS) &&
(execstatus != OCI_NO_DATA))
    {
        DISCARD
OCITransRollback(tpcsvc, errhp, OCI_DEFAULT)
;
        errcode = OCIERROR(errhp, execstatus);
        if(errcode == NOT_SERIALIZABLE)
        {
            retries++;
            goto retry;
        }
        else if (errcode == RECOVERR)
        {
            retries++;
            goto retry;
        }
        else if (errcode == SNAPSHOT_TOO_OLD)
        {
            retries++;
            goto retry;
        }
        else
        {
            return -1;
        }
    }
    /* mark districts with no new order */
DISCARD OCIAttrGet(dctx-
>curd1, OCI_HTYPE_STMT, &rcount, NULLP(ub4
),
        OCI_ATTR_ROW_COUNT, errhp);
    rpc = rcount;
    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);
        }
    }

    execstatus=OCIStmtExecute(tpcsvc, dctx-
>curd3, errhp, rpc, 0,
        NULLP(CONST
OCISnapshot), NULLP(OCISnapshot), OCI_DEF
AULT);
    if(execstatus != OCI_SUCCESS)
    {
        DISCARD
OCITransRollback(tpcsvc, errhp, OCI_DEFAULT)
;
        errcode = OCIERROR(errhp, execstatus);
        if(errcode == NOT_SERIALIZABLE)
        {
            retries++;
            goto retry;
        }
        else if (errcode == RECOVERR)
        {
            retries++;
            goto retry;
        }
        else if (errcode == SNAPSHOT_TOO_OLD)
        {
            retries++;
            goto retry;
        }
        else
        {
            return -1;
        }
    }
}

```

```

    }
}

DISCARD OCIAttrGet(dctx-
>curd3, OCI_HTYPE_STMT, &rcount, NULLP(ub4
),
        OCI_ATTR_ROW_COUNT, errhp);

    if (rcount != rpc)
    {
        TpcUserLog (LOG_FILE_INF, "Error in
TPC-C server %d: %d rows selected, %d ords
updated\n",
            proc_no, rpc, rcount);
        DISCARD
OCITransRollback(tpcsvc, errhp, OCI_DEFAULT)
;
        return (-1);
    }

    /* array update of order_line table */
    execstatus=OCIStmtExecute(tpcsvc, dctx-
>curd4, errhp, rpc, 0,
        NULLP(CONST
OCISnapshot), NULLP(OCISnapshot), OCI_DEF
AULT);
    if(execstatus != OCI_SUCCESS)
    {
        DISCARD
OCITransRollback(tpcsvc, errhp, OCI_DEFAULT)
;
        errcode = OCIERROR(errhp, execstatus);
        if(errcode == NOT_SERIALIZABLE)
        {
            retries++;
            goto retry;
        }
        else if (errcode == RECOVERR)
        {
            retries++;
            goto retry;
        }
        else if (errcode == SNAPSHOT_TOO_OLD)
        {
            retries++;
            goto retry;
        }
        else
        {
            return -1;
        }
    }
DISCARD OCIAttrGet(dctx-
>curd4, OCI_HTYPE_STMT, &rcount, NULLP(ub4
),
        OCI_ATTR_ROW_COUNT, errhp);
/* transfer amounts */
for (i=0; i<rpc; i++)
{
    dctx->amt[i]=0;
    if ( actx->ol_amt_rcode[i] == 0)
    {
        dctx->amt[i] = actx->ol_amt[i];
    }
}
#ifdef OLD
    if (rcount > rpc) {
        TpcUserLog
        (LOG_FILE_INF, "Error in TPC-C
server %d: %d ordnrs updated, %d ordl
updated\n",
            proc_no, rpc, rcount);
    }
#endif
}

```

```

    /* array update of customer table */
    execstatus=OCIStmtExecute(tpcsvc, dctx-
>curd6, errhp, rpc, 0,
        NULLP(CONST
OCISnapshot), NULLP(OCISnapshot),
        OCI_COMMIT_ON_SUCCESS |
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 == RECOVERR)
        {
            retries++;
            goto retry;
        }
        else if (errcode == SNAPSHOT_TOO_OLD)
        {
            retries++;
            goto retry;
        }
        else
        {
            return -1;
        }
    }

    DISCARD OCIAttrGet(dctx-
>curd6, OCI_HTYPE_STMT, &rcount, NULLP(ub4
),
        OCI_ATTR_ROW_COUNT, errhp);

    if (rcount != rpc) {
        TpcUserLog(LOG_FILE_INF, "Error in
TPC-C server %d: %d rows selected, %d cust
updated\n",
            proc_no, rpc, rcount);

        DISCARD OCITransRollback(tpcsvc, errhp,
OCI_DEFAULT);
        return (-1);
    }

    /* 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];
    return (0);
}

void tkvcddone (int plsqflag)
{
    if (plsqflag)
    {
        if (pldctx)
        {
            DISCARD OCIHandleFree((dvoid *)dctx-
>curd0, OCI_HTYPE_STMT);
            DISCARD free(pldctx);
        }
    }
}

```



```

}
else
{
  if (dctx)
  {
    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);
    DISCARD free (dctx);
  }
}

.....
svrapl/3tier/plnew.c
.....

#ifdef RCSID
static char *RCSid =
"$Header: tkvcnew.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 "forlinux.h"
#include "log.h"

#ifdef ORA_TPCC
# define ORA_TPCC
# include "tpcc.h"
#endif

#include "GlobalArea.h"
#include "prototype.h"

#define SQLTXT2 "BEGIN
inittpc.init_no(idx1arr); END;"

#define NITEMS 15
#define ROWIDLEN 20
#define OCIROWLEN 20

int tkvcninit ()
{
  /* for warning */
  /* int i;*/

  /* Replaced T.Kato 03.03.19 Repaleced Oracle
  10i tool kit */
  /* text stmbuff[16*1024];*/
  text stmbuff[32*1024];
  /* Replaced end */

  nctx = (newctx *) malloc (sizeof(newctx));
  DISCARD
  memset(nctx,(char)0,sizeof(newctx));
  nctx->w_id_len = sizeof(w_id);
  nctx->d_id_len = sizeof(d_id);
  nctx->c_id_len = sizeof(c_id);
  nctx->o_all_local_len = sizeof(o_all_local);
  nctx->o_ol_cnt_len = sizeof(o_ol_cnt);
  nctx->w_tax_len = 0;
  nctx->d_tax_len = 0;
  nctx->o_id_len = sizeof(o_id);
  nctx->c_discount_len = 0;
  nctx->c_credit_len = 0;
  nctx->c_last_len = 0;
  nctx->retries_len = sizeof(retries);
  nctx->cr_date_len = sizeof(cr_date);

  /* open first cursor */
  DISCARD
  OCIERROR(errhp,OCIHandleAlloc(tpcenv,(dvoi
d**>(&nctx->cur1),
OCI_HTYPE_STMT, 0, (dvoid**)0));
  /* Replaced T.kato 03.03.19 Replaced Oracle
  10i tool kit */
  /* sqlfile("../blocks/tkvcnew.sql",stmbuff);*/
  #if defined(ISO)
  sqlfile("../blocks/tkvcnew_iso.sql",stmbuff);
  #else
  #if defined(ISO7)
  sqlfile("../blocks/tkvcnew_iso7.sql",stmbuff);
  #else
  /* Replaced 04.01.20 TUXEDO Client */
  #if 0
  ! sqlfile("../blocks/tkvcnew.sql",stmbuff);
  #endif
  sqlfile("/home/tpc/blocks/tkvcnew.sql",stmbuff);
  /* Replaced end */
  #endif
  #endif
  /* Replaced end */

  DISCARD
  OCIERROR(errhp,OCIStmtPrepare(nctx->cur1,
errhp, stmbuff,
strlen((char *)stmbuff),
OCI_NTV_SYNTAX, OCI_DEFAULT));

  /* bind variables */

  OCIBNDPL(nctx->cur1, nctx->w_id_bp, errhp,
"w_id",ADR(w_id),SIZ(w_id),
SQLT_INT, &nctx->w_id_len);
  OCIBNDPL(nctx->cur1, nctx->d_id_bp, errhp,
"d_id",ADR(d_id),SIZ(d_id),
SQLT_INT, &nctx->d_id_len);
  OCIBNDPL(nctx->cur1, nctx->c_id_bp, errhp,
"c_id",ADR(c_id),SIZ(c_id),
SQLT_INT, &nctx->c_id_len);
  OCIBNDPL(nctx->cur1, nctx->o_all_local_bp,
errhp, "o_all_local",
ADR(o_all_local),
SIZ(o_all_local),SQLT_INT, &nctx-
>o_all_local_len);
  OCIBNDPL(nctx->cur1, nctx->o_ol_cnt_bp,
errhp, "o_ol_cnt",ADR(o_ol_cnt),
SIZ(o_ol_cnt),SQLT_INT, &nctx-
>o_ol_cnt_len);
  OCIBNDPL(nctx->cur1, nctx->w_tax_bp,
errhp, "w_tax",ADR(w_tax),SIZ(w_tax),
SQLT_FLT, &nctx->w_tax_len);
  OCIBNDPL(nctx->cur1, nctx->d_tax_bp, errhp,
"d_tax",ADR(d_tax),SIZ(d_tax),
SQLT_FLT, &nctx->d_tax_len);
  OCIBNDPL(nctx->cur1, nctx->o_id_bp, errhp,
"o_id",ADR(o_id),SIZ(o_id),
SQLT_INT, &nctx->o_id_len);
  OCIBNDPL(nctx->cur1, nctx->c_discount_bp,
errhp, "c_discount",
ADR(c_discount),
SIZ(c_discount),SQLT_FLT, &nctx-
>c_discount_len);
  OCIBNDPL(nctx->cur1, nctx->c_credit_bp,
errhp, "c_credit",c_credit,
SIZ(c_credit),SQLT_CHR, &nctx-
>c_credit_len);
  OCIBNDPL(nctx->cur1, nctx->c_last_bp,
errhp, "c_last",c_last,SIZ(c_last),
SQLT_STR, &nctx->c_last_len);
  OCIBNDPL(nctx->cur1, nctx->retries_bp,
errhp, "retry",ADR(retries),
SIZ(retries),SQLT_INT, &nctx-
>retries_len);
  OCIBNDPL(nctx->cur1, nctx->cr_date_bp,
errhp, "cr_date",&cr_date,
SIZ(OCIDate), SQLT_ODT, &nctx-
>cr_date_len);

  OCIBNDPLA(nctx->cur1, nctx-
>ol_i_id_bp,errhp,"ol_i_id",nol_i_id,
SIZ(int), SQLT_INT, nctx-
>nol_i_id_len,NITEMS,&nctx->nol_i_count);
  OCIBNDPLA(nctx->cur1, nctx-
>ol_supply_w_id_bp, errhp, "ol_supply_w_id",
nol_supply_w_id,SIZ(int),SQLT_INT,
nctx->nol_supply_w_id_len,
NITEMS, &nctx->nol_s_count);

/* Replaced T.kato 03.09.09 Oracle10g tool kit */
#if 0
! OCIBNDPLA(nctx->cur1, nctx-
>ol_quantity_bp,errhp,"ol_quantity",
! nol_quantity, SIZ(int),SQLT_INT,nctx-
>nol_quantity_len,
! NITEMS,&nctx->nol_q_count);
! OCIBNDPLA(nctx->cur1, nctx-
>i_price_bp,errhp,"i_price",i_price,SIZ(int),
! SQLT_INT, nctx->i_price_len, NITEMS,
&nctx->nol_item_count);
#endif

#ifdef USE_IEEE_NUMBER
OCIBNDPLA(nctx->cur1, nctx-
>ol_quantity_bp,errhp,"ol_quantity",
nol_quantity,
SIZ(float),SQLT_BFLOAT,nctx-
>nol_quantity_len,
NITEMS,&nctx->nol_q_count);

OCIBNDPLA(nctx->cur1, nctx-
>i_price_bp,errhp,"i_price",i_price,SIZ(float),
SQLT_BFLOAT, nctx->i_price_len,
NITEMS, &nctx->nol_item_count);
#else

```

```

OCIBNDPLA(nctx->curm1, nctx-
>o_l_quantity_bp,errhp,":o_l_quantity",
    nol_quantity, SIZ(int),SQLT_INT,nctx-
>nol_quantity_len,
    NITEMS,&nctx->nol_q_count);

OCIBNDPLA(nctx->curm1, nctx-
>i_price_bp,errhp,":i_price",i_price,SIZ(int),
    SQLT_INT, nctx->i_price_len, NITEMS,
&nctx->nol_item_count);
#endif /* USE_IEEE_NUMBER */
/* Replaced end */

OCIBNDPLA(nctx->curm1, nctx-
>i_name_bp,errhp,":i_name",i_name,
    SIZ(i_name[0]),SQLT_STR, nctx-
>i_name_len,NITEMS,
    &nctx->nol_name_count);

/* Replaced T.kato 03.09.09 Oracle10g tool kit */
#if 0
! OCIBNDPLA(nctx->curm1, nctx-
>s_quantity_bp,errhp,":s_quantity",s_quantity,
!     SIZ(int), SQLT_INT,nctx-
>s_quant_len,NITEMS,&nctx->nol_qty_count);
#endif

#ifdef USE_IEEE_NUMBER
OCIBNDPLA(nctx->curm1, nctx-
>s_quantity_bp,errhp,":s_quantity",s_quantity,
    SIZ(float), SQLT_BFLOAT,nctx-
>s_quant_len,NITEMS,&nctx->nol_qty_count);
#else
OCIBNDPLA(nctx->curm1, nctx-
>s_quantity_bp,errhp,":s_quantity",s_quantity,
    SIZ(int), SQLT_INT,nctx-
>s_quant_len,NITEMS,&nctx->nol_qty_count);
#endif /* USE_IEEE_NUMBER */
/* Replaced end */

OCIBNDPLA(nctx->curm1, nctx-
>s_bg_bp,errhp,":brand_generic",brand_generic,
    SIZ(char), SQLT_CHR,nctx-
>s_bg_len,NITEMS,&nctx->nol_bg_count);

/* Replaced T.kato 03.09.09 Oracle10g tool kit */
#if 0
! OCIBNDPLA(nctx->curm1, nctx-
>o_l_amount_bp,errhp,":o_l_amount",nol_amount,
!     SIZ(int),SQLT_INT, nctx-
>nol_amount_len,NITEMS,&nctx-
>nol_am_count);
! OCIBNDPLA(nctx->curm1, nctx-
>s_remote_bp,errhp,":s_remote",nctx-
>s_remote,
!     SIZ(int),SQLT_INT, nctx-
>s_remote_len,NITEMS,&nctx-
>s_remote_count);
#endif

#ifdef USE_IEEE_NUMBER
OCIBNDPLA(nctx->curm1, nctx-
>o_l_amount_bp,errhp,":o_l_amount",nol_amount,
    SIZ(float),SQLT_BFLOAT, nctx-
>nol_amount_len,NITEMS,&nctx-
>nol_am_count);

OCIBNDPLA(nctx->curm1, nctx-
>s_remote_bp,errhp,":s_remote",nctx-
>s_remote,
    SIZ(float),SQLT_BFLOAT, nctx-
>s_remote_len,NITEMS,&nctx-
>s_remote_count);
#else
#endif

```

```

OCIBNDPLA(nctx->curm1, nctx-
>o_l_amount_bp,errhp,":o_l_amount",nol_amount,
    SIZ(int),SQLT_INT, nctx-
>nol_amount_len,NITEMS,&nctx-
>nol_am_count);

OCIBNDPLA(nctx->curm1, nctx-
>s_remote_bp,errhp,":s_remote",nctx-
>s_remote,
    SIZ(int),SQLT_INT, nctx-
>s_remote_len,NITEMS,&nctx-
>s_remote_count);
#endif /* USE_IEEE_NUMBER */
/* Replaced end */

/* open second cursor */
DISCARD
OCIERROR(errhp,OCIHandleAlloc(tpcenv,
(dvoid**)(&nctx->curm2),
    OCI_HTYPE_STMT, 0, (dvoid**)0));
DISCARD sprintf((char *) stmbuf, SQLTXT2);
DISCARD
OCIERROR(errhp,OCIStmtPrepare(nctx->curm2,
errhp, stmbuf,
    strlen((char *)stmbuf),
OCI_NTV_SYNTAX, OCI_DEFAULT));

/* execute second cursor to init newinit
package */
{
    int idx1arr[NITEMS];
    OCIBind *idx1arr_bp;
    ub2 idx1arr_len[NITEMS];
/* for Warning */
/* ub2 idx1arr_rcode[NITEMS];*/

    sb2 idx1arr_ind[NITEMS];
    ub4 idx1arr_count;
    ub2 idx;

    for (idx = 0; idx < NITEMS; idx++) {
        idx1arr[idx] = idx + 1;
        idx1arr_ind[idx] = TRUE;
        idx1arr_len[idx] = sizeof(int);
    }
    idx1arr_count = NITEMS;
    o_l_cnt = NITEMS;

    /* Bind array */
    OCIBNDPLA(nctx->curm2,
idx1arr_bp,errhp,":idx1arr",idx1arr,
        SIZ(int), SQLT_INT, idx1arr_len,
NITEMS,&idx1arr_count);

    DBGLOG("NEW:[1]Start",0);
    execstatus = OCIStmtExecute(tpcsvc,nctx-
>curm2,errhp,1,0,
        NULLP(CONST
OCI_Snapshot),NULLP(OCI_Snapshot),OCI_DEF
AULT);
    DBGLOG("NEW:[1]End >%d",execstatus);
    if(execstatus != OCI_SUCCESS) {

OCITransRollback(tpcsvc,errhp,OCI_DEFAULT)
;
        errcode = OCIERROR(errhp,execstatus);
        return -1;
    }
}

return (0);
}

```

```

int tkvcn ()
{
    int i;
    int rcount;

retry:

    status = 0;          /* number of invalid
items */

    /* get number of order lines, and check if all
are local */

    o_o_l_cnt = NITEMS;
    o_all_local = 1;
    for (i = 0; i < NITEMS; i++) {
        if (nol_i_id[i] == 0) {
            o_o_l_cnt = i;
            break;
        }
        if (nol_supply_w_id[i] != w_id) {

/* Replaced T.kato 03.09.09 Oracle10g tool kit */
#if 0
!     nctx->s_remote[i] = 1;
#endif

#ifdef USE_IEEE_NUMBER
        nctx->s_remote[i] = 1.0;
#else
        nctx->s_remote[i] = 1;
#endif /* USE_IEEE_NUMBER */
/* Replaced end */

        o_all_local = 0;
    }
    else
        nctx->s_remote[i] = 0;
}

    nctx->w_id_len = sizeof(w_id);
    nctx->d_id_len = sizeof(d_id);
    nctx->c_id_len = sizeof(c_id);
    nctx->o_all_local_len = sizeof(o_all_local);
    nctx->o_o_l_cnt_len = sizeof(o_o_l_cnt);
    nctx->w_tax_len = 0;
    nctx->d_tax_len = 0;
    nctx->o_id_len = sizeof(o_id);
    nctx->c_discount_len = 0;
    nctx->c_credit_len = 0;
    nctx->c_last_len = 0;
    nctx->retries_len = sizeof(retries);
    nctx->cr_date_len = sizeof(cr_date);
    /* this is the row count */
    rcount = o_o_l_cnt;
    nctx->nol_i_count = o_o_l_cnt;
    nctx->nol_q_count = o_o_l_cnt;
    nctx->nol_s_count = o_o_l_cnt;
    nctx->s_remote_count = o_o_l_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;

    /* initialization for array operations */
    for (i = 0; i < o_o_l_cnt; i++) {
        nctx->o_l_number[i] = i + 1;
        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_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->s_remote_len[i] = sizeof(int);
nctx->s_quant_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_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_o_id_len[i] = 0;
nctx->ol_number_len[i] = 0;
nctx->ol_dist_info_len[i] = 0;
nctx->s_remote_len[i] = 0;
nctx->s_quant_len[i] = 0;
nctx->i_name_len[i]=0;
nctx->s_bg_len[i] = 0;
}

DBGLOG("NEW:[2]Start",0);
execstatus = OCISmtExecute(tpcsvc,nctx-
>curr1,errhp,1,0,0,0,
OCI_DEFAULT |
OCI_COMMIT_ON_SUCCESS);
DBGLOG("NEW:[2]End >%d",execstatus);

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;
}
/* Deleted T.Kato 02.10.25 */
#if 0
! ) else if (errcode ==
SNAPSHOT_TOO_OLD) {
! retries++;
! goto retry;
#endif
/* Deleted end */
} else {
return -1;
}
}

/* did the txn succeed ? */
if (rcount != o_ol_cnt)
{
status = rcount - o_ol_cnt;
o_ol_cnt = rcount;
}

#ifdef DEBUG
printf("w_id = %d, d_id = %d, c_id
= %d\n",w_id, d_id, c_id);
#endif

return (0);
}

```

```

void tkvcndone ()
{
/* for warning */
/* int i;*/

if (nctx)
{
DISCARD OCISmtFree((dvoid *)nctx-
>curr1,OCI_HTYPE_STMT);
DISCARD OCISmtFree((dvoid *)nctx-
>curr2,OCI_HTYPE_STMT);
free (nctx);
}
}

:-----:
svrapl/3tier/plord.c
:-----:

/* Copyright (c) 2002, Oracle Corporation. All
rights reserved. */

NAME
tkvcordq.c - OCI version using queues of
ORDER STATUS
transaction in TPC-C benchmark.

DESCRIPTION
<short description of facility this file
declares/defines>

EXPORT FUNCTION(S)

INTERNAL FUNCTION(S)
<other external functions defined - one-line
descriptions>

STATIC FUNCTION(S)
<static functions defined - one-line
descriptions>

NOTES
<other useful comments, qualifications, etc.>

MODIFIED (MM/DD/YY)
xnie 06/25/02 - queue open cluster join.
heri 05/07/02 - Fix error in cursor.
heri 02/01/02 - Cleanup, remove indicator
values and return codes.
lwang 07/25/01 - Merged lwang_tpcitrc
lwang 07/23/01 - fix include
lwang 07/23/01 - Creation

*/

#include "forlinux.h"
#include "log.h"
#include "tpcc.h"
#include "GlobalArea.h"
#include "prototype.h"

/*-----:
:-----:
PRIVATE TYPES AND
CONSTANTS
:-----:
:-----*/

```

```

/*-----:
:-----:
STATIC FUNCTION
DECLARATIONS
:-----*/

/* Replaced T.Kato 2004.12.21 New Oracle10g
tool kit */
#if 0
#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 /*+ USE_NL(cust)
INDEX_DESC(ordr iordr2) */ \
! c_id, c_balance, c_first, c_middle,
c_last, \
! o_id, o_entry_d, o_carrier_id,
o_ol_cnt, ordr.rowid \
! 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 DESC, o_d_id
DESC, o_w_id DESC, o_id DESC"
!
#define SQLCUR2 "SELECT /*+ USE_NL(cust)
INDEX_DESC (ordr iordr2) */ \
! c_balance, c_first, c_middle, \
! o_id, o_entry_d, o_carrier_id,
o_ol_cnt, ordr.rowid \
! 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 DESC, o_d_id
DESC, o_w_id DESC , o_id DESC"
!
#define SQLCUR3 "SELECT /*+ ORDERED
USE_NL(ordl) CLUSTER(ordl) */ \
! ol_i_id, ol_supply_w_id, ol_quantity,
ol_amount, ol_delivery_d \
! FROM ordr, ordl \
! WHERE ordr.rowid = :ordr_rowid \
! AND o_id = ol_o_id AND ol_d_id =
o_d_id AND ol_w_id = o_w_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

#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 /*+ USE_NL(cust)
INDEX_DESC(ordr iordr2) */ \
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 DESC, o_d_id
DESC, o_w_id DESC, o_id DESC"

#define SQLCUR2 "SELECT /*+ USE_NL(cust)
INDEX_DESC (ordr iordr2) */ \
        c_balance, c_first, c_middle, c_last, \
        o_id, o_entry_d, o_carrier_id,
o_of_cnt \
        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 DESC, o_d_id
DESC, o_w_id DESC , o_id DESC"

#define SQLCUR3 "SELECT /*+ INDEX(ordl) */
\
        ol_i_id, ol_supply_w_id, ol_quantity,
ol_amount, ol_delivery_d \
        FROM ordl \
        WHERE ol_o_id = :o_id AND ol_d_id
= :d_id AND ol_w_id = :w_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"

/* Replaced end */

int tkvcoin0 ()
{
    int i;
    text stmbuff[SQL_BUF_SIZE];

    octx = (ordctx *) malloc (sizeof(ordctx));
    DISCARD memset(octx, (char)0, sizeof(ordctx));
    octx->cs = 1;
    octx->norow = 0;
    octx->somerows = 10;

    /* Deleted T.Kato 2004.12.21 New Oracle10g
    tool kit */
    #if 0
    ! /* get the rowid handles */
    ! OCIERROR(errhp, OCIDescriptorAlloc((dvoid
    *)tpcenv, (dvoid **)&octx->o_rowid,
    ! (ub4)OCI_DTYPE_ROWID,
    (size_t) 0, (dvoid **)0));
    #endif
    /* Deleted end */

    for(i=0;i<100;i++) {
        DISCARD OCIERROR(errhp,
        OCIDescriptorAlloc(tpcenv,
        (dvoid **)&octx->c_rowid_ptr[i],
        OCI_DTYPE_ROWID, 0, (dvoid **)0));
    }

    DISCARD OCIERROR(errhp,
    OCIHandleAlloc(tpcenv, (dvoid **)&octx-
    >curo0, OCI_HTYPE_STMT, 0, (dvoid **)0));

    /* Deleted T.Kato 2004.12.21 New Oracle10g
    tool kit */
    #if 0
    ! DISCARD OCIERROR(errhp,
    ! OCIHandleAlloc(tpcenv, (dvoid **)&octx-
    >curo0, OCI_HTYPE_STMT, 0, (dvoid **)0));

```

```

#endif
/* Deleted end */

    DISCARD OCIERROR(errhp,
    OCIHandleAlloc(tpcenv, (dvoid **)&octx-
    >curo1, OCI_HTYPE_STMT, 0, (dvoid **)0));
    DISCARD OCIERROR(errhp,
    OCIHandleAlloc(tpcenv, (dvoid **)&octx-
    >curo2, OCI_HTYPE_STMT, 0, (dvoid **)0));
    DISCARD OCIERROR(errhp,
    OCIHandleAlloc(tpcenv, (dvoid **)&octx-
    >curo3, OCI_HTYPE_STMT, 0, (dvoid **)0));
    DISCARD OCIERROR(errhp,
    OCIHandleAlloc(tpcenv, (dvoid **)&octx-
    >curo4, OCI_HTYPE_STMT, 0, (dvoid **)0));

    /* c_id = 0, use find customer by lastname. Get
    an array or rowid's back */
    DISCARD sprintf((char *) stmbuff, SQLCUR0);
    DISCARD OCIERROR(errhp,
    OCIStmtPrepare(octx-
    >curo0, errhp, stmbuff, (ub4)strlen((char *)stmbuff),

    OCI_NTV_SYNTAX, OCI_DEFAULT));
    DISCARD OCIERROR(errhp,
    OCIAttrSet(octx-
    >curo0, OCI_HTYPE_STMT, &octx->norow, 0,

    OCI_ATTR_PREFETCH_ROWS, errhp));
    /* get order/customer info back based on rowid */
    DISCARD sprintf((char *) stmbuff, SQLCUR1);
    DISCARD OCIERROR(errhp,
    OCIStmtPrepare(octx-
    >curo1, errhp, stmbuff, (ub4)strlen((char *)stmbuff),

    OCI_NTV_SYNTAX, OCI_DEFAULT));
    DISCARD OCIERROR(errhp,
    OCIAttrSet(octx-
    >curo1, OCI_HTYPE_STMT, &octx->norow, 0,

    OCI_ATTR_PREFETCH_ROWS, errhp));

    /* c_id == 0, use lastname to find customer */
    DISCARD sprintf((char *) stmbuff, SQLCUR2);
    DISCARD OCIERROR(errhp,
    OCIStmtPrepare(octx-
    >curo2, errhp, stmbuff, (ub4)strlen((char *)stmbuff),

    OCI_NTV_SYNTAX, OCI_DEFAULT));
    DISCARD OCIERROR(errhp,
    OCIAttrSet(octx-
    >curo2, OCI_HTYPE_STMT, &octx->norow, 0,

    OCI_ATTR_PREFETCH_ROWS, errhp));

    DISCARD sprintf((char *) stmbuff, SQLCUR3);
    DISCARD OCIERROR(errhp,
    OCIStmtPrepare(octx-
    >curo3, errhp, stmbuff, (ub4)strlen((char *)stmbuff),

    OCI_NTV_SYNTAX, OCI_DEFAULT));
    DISCARD OCIERROR(errhp,
    OCIAttrSet(octx-
    >curo3, OCI_HTYPE_STMT, &octx->norow, 0,

    OCI_ATTR_PREFETCH_ROWS, errhp));

    DISCARD sprintf((char *) stmbuff, SQLCUR4);
    DISCARD OCIERROR(errhp,
    OCIStmtPrepare(octx-
    >curo4, errhp, stmbuff, (ub4)strlen((char *)stmbuff),

    OCI_NTV_SYNTAX, OCI_DEFAULT));
    DISCARD OCIERROR(errhp,

```

```

    OCIAttrSet(octx-
    >curo4, OCI_HTYPE_STMT, &octx->norow, 0,

    OCI_ATTR_PREFETCH_ROWS, errhp));

    for (i = 0; i < NITEMS; i++) {

        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_len = sizeof(int);
    octx->ol_d_id_len = sizeof(int);
    octx->ol_o_id_len = sizeof(int);

    /* bind variables */

    /* c_id (customer id) is not known */
    OCIBND(octx->curo0, octx-
    >w_id_bp[0], errhp, "w_id", ADR(w_id),
    SIZ(int), SOLT_INT);
    OCIBND(octx->curo0, octx-
    >d_id_bp[0], errhp, "d_id", ADR(d_id),
    SIZ(int), SOLT_INT);
    OCIBND(octx->curo0, octx-
    >c_last_bp[0], errhp, "c_last", c_last,
    SIZ(c_last), SOLT_STR);
    OCIDFNRA(octx->curo0, octx-
    >c_rowid_dp, errhp, 1, octx->c_rowid_ptr,
    SIZ(OCIRowid*), SOLT_RDD, NULL,
    octx->c_rowid_len, NULL);

    OCIBND(octx->curo1, octx-
    >c_rowid_bp, errhp, "cust_rowid", &octx-
    >c_rowid_cust,
    sizeof(octx->c_rowid_ptr[0]), SOLT_RDD);
    OCIDFNRA(octx->curo1, octx-
    >c_id_dp, errhp, 1, ADR(c_id), SIZ(int), SOLT_INT);

    /* Replaced T.kato 03.09.09 Oracle10g tool kit */
    #if 0
    ! OCIDFNRA(octx->curo1, octx-
    >c_balance_dp[0], errhp, 2, ADR(c_balance),
    ! SIZ(double), SOLT_FLT);
    #endif

    #ifdef USE_IEEE_NUMBER
    OCIDFNRA(octx->curo1, octx-
    >c_balance_dp[0], errhp, 2, ADR(c_balance),
    SIZ(double), SOLT_BDOUBLE);
    #else
    OCIDFNRA(octx->curo1, octx-
    >c_balance_dp[0], errhp, 2, ADR(c_balance),
    SIZ(double), SOLT_FLT);
    #endif /* USE_IEEE_NUMBER */
    /* Replaced end */

    OCIDFNRA(octx->curo1, octx-
    >c_first_dp[0], errhp, 3, c_first, SIZ(c_first)-1,
    SOLT_CHAR);
    OCIDFNRA(octx->curo1, octx-
    >c_middle_dp[0], errhp, 4, c_middle,
    SIZ(c_middle)-1, SOLT_AFC);

```

```

OCIDEF(octx->curo1,octx-
>c_last_dp[0],errhp,5,c_last,SIZ(c_last)-1,
SQLT_CHR);
OCIDEF(octx->curo1,octx-
>o_id_dp[0],errhp,6,ADR(o_id),SIZ(int),SQLT_IN
T);
OCIDEF(octx->curo1,octx-
>o_entry_d_dp[0],errhp,7,

&o_entry_d_base,SIZ(OCIDate),SQLT_ODT);
OCIDEF(octx->curo1,octx-
>o_cr_id_dp[0],errhp,8,ADR(o_carrier_id),
SIZ(int),SQLT_INT);
OCIDEF(octx->curo1,octx-
>o_ol_cnt_dp[0],errhp,9,ADR(o_ol_cnt),
SIZ(int),SQLT_INT);

/* Deleted T.Kato 2004.12.21 New Oracle10g
tool kit */
#if 0
! OCIDEF(octx->curo1,octx-
>o_rowid_dp[0],errhp,10,ADR(octx->o_rowid),
! SIZ(OCIRowid*),SQLT_RDD);
#endif
/* deleted end */

/* Bind for third cursor , no-zero customer id */
OCIBND(octx->curo2,octx-
>w_id_bp[1],errhp,:"w_id",ADR(w_id),
SIZ(int),SQLT_INT);
OCIBND(octx->curo2,octx-
>d_id_bp[1],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);

/* Replaced T.kato 03.09.09 Oracle10g tool kit */
#if 0
! OCIDEF(octx->curo2,octx-
>c_balance_dp[1],errhp,1,ADR(c_balance),
! SIZ(double),SQLT_FLT);
#endif

#ifdef USE_IEEE_NUMBER
OCIDEF(octx->curo2,octx-
>c_balance_dp[1],errhp,1,ADR(c_balance),
SIZ(double),SQLT_BDOUBLE);
#else
OCIDEF(octx->curo2,octx-
>c_balance_dp[1],errhp,1,ADR(c_balance),
SIZ(double),SQLT_FLT);
#endif /* USE_IEEE_NUMBER */
/* Replaced end */

OCIDEF(octx->curo2,octx-
>c_first_dp[1],errhp,2,c_first,SIZ(c_first)-1,
SQLT_CHR);
OCIDEF(octx->curo2,octx-
>c_middle_dp[1],errhp,3,c_middle,
SIZ(c_middle)-1,SQLT_AFC);
OCIDEF(octx->curo2,octx-
>c_last_dp[1],errhp,4,c_last,SIZ(c_last)-1,
SQLT_CHR);
OCIDEF(octx->curo2,octx-
>o_id_dp[1],errhp,5,ADR(o_id),SIZ(int),SQLT_IN
T);
OCIDEF(octx->curo2,octx-
>o_entry_d_dp[1],errhp,6, &o_entry_d_base,
SIZ(OCIDate),SQLT_ODT);
OCIDEF(octx->curo2, octx-
>o_cr_id_dp[1],errhp,7,ADR(o_carrier_id),
SIZ(int), SQLT_INT);
OCIDEF(octx->curo2,octx-
>o_ol_cnt_dp[1],errhp,8,ADR(o_ol_cnt),

```

```

SIZ(int),SQLT_INT);

/* Deleted T.Kato 2004.12.21 New Oracle10g
tool kit */
#if 0
! OCIDEF(octx->curo2,octx-
>o_rowid_dp[1],errhp,9,ADR(octx->o_rowid),
! SIZ(OCIRowid*),SQLT_RDD);
#endif
/* Deleted end */

/* Bind for last cursor */

/* Replaced T.Kato 2004.12.21 New Oracle10g
tool kit */
#if 0
! OCIBND(octx->curo3,octx-
>w_id_bp[2],errhp,:"w_id",ADR(w_id),
SIZ(int),SQLT_INT);
! OCIBND(octx->curo3,octx-
>d_id_bp[2],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);
! OCIBND(octx->curo3,octx-
>c_id_bp,errhp,:"c_id",ADR(c_id),
SIZ(int),SQLT_INT);
! */
#endif

OCIBND(octx->curo3,octx-
>w_id_bp[2],errhp,:"w_id",ADR(w_id),
SIZ(int),SQLT_INT);
OCIBND(octx->curo3,octx-
>d_id_bp[2],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);
/* Replaced end */

/* Deleted T.Kato 2004.12.21 New Oracle10g
tool kit */
#if 0
! OCIBND(octx->curo3,octx-
>o_rowid_bp,errhp,:"ordr_rowid",
! &octx->o_rowid,
SIZ(OCIRowid*),SQLT_RDD);
#endif
/* Deleted end */

OCIDFNRA(octx->curo3, octx->ol_i_id_dp,
errhp, 1, ol_i_id,SIZ(int),SQLT_INT,
NULL,octx->ol_i_id_len, NULL);
OCIDFNRA(octx->curo3,octx-
>ol_supply_w_id_dp,errhp,2, ol_supply_w_id,
SIZ(int),SQLT_INT, NULL,
octx->ol_supply_w_id_len, NULL);

/* Replaced T.kato 03.09.09 Oracle10g tool kit */
#if 0
! OCIDFNRA(octx->curo3, octx-
>ol_quantity_dp,errhp,3, ol_quantity,SIZ(int),
! SQLT_INT, NULL,octx->ol_quantity_len,
NULL);
! OCIDFNRA(octx->curo3,octx-
>ol_amount_dp,errhp,4,ol_amount, SIZ(int),
! SQLT_INT,NULL, octx->ol_amount_len,
NULL);
#endif

#ifdef USE_IEEE_NUMBER

```

```

OCIDFNRA(octx->curo3, octx-
>ol_quantity_dp,errhp,3, ol_quantity,SIZ(float),
SQLT_BFLOAT, NULL,octx-
>ol_quantity_len, NULL);
OCIDFNRA(octx->curo3,octx-
>ol_amount_dp,errhp,4,ol_amount, SIZ(float),
SQLT_BFLOAT,NULL, octx-
>ol_amount_len, NULL);
#else
OCIDFNRA(octx->curo3, octx-
>ol_quantity_dp,errhp,3, ol_quantity,SIZ(int),
SQLT_INT, NULL,octx->ol_quantity_len,
NULL);
OCIDFNRA(octx->curo3,octx-
>ol_amount_dp,errhp,4,ol_amount, SIZ(int),
SQLT_INT,NULL, octx->ol_amount_len,
NULL);
#endif /* USE_IEEE_NUMBER */
/* Replaced end */

OCIDFNRA(octx->curo3,octx-
>ol_d_base_dp,errhp,5,ol_d_base,SIZ(OCIDate),
SQLT_ODT, NULL,octx-
>ol_delivery_d_len,NULL);

OCIBND(octx->curo4,octx-
>w_id_bp[3],errhp,:"w_id",ADR(w_id),
SIZ(int),SQLT_INT);
OCIBND(octx->curo4,octx-
>d_id_bp[3],errhp,:"d_id",ADR(d_id),
SIZ(int),SQLT_INT);
OCIBND(octx->curo4,octx-
>c_last_bp[1],errhp,:"c_last",c_last,
SIZ(c_last), SQLT_STR);
OCIDEF(octx->curo4,octx-
>c_count_dp,errhp,1,ADR(octx->rcount),SIZ(int),
SQLT_INT);

return (0);
}

int tkvco ()
{
int i;
int rcount;

#ifdef ISO9
int secondread = 0;
char sdate[30];
ub4 datelen;
sysdate(sdate);
printf("Order Status started at: %s\n", sdate);
#endif

int oci_stat;

/* Deleted T.Kato 2004.12.21 "o_rowid" was
deleted by New Oracle10g tool kit */
#if 0
!!int f_w_id = w_id;
!!int f_d_id = d_id;
!!int f_c_id = c_id;
!!
!!int c2_w_id = -1;
!!int c2_d_id = -1;
!!int c2_c_id = -1;
!!unsigned char b_row_id[512];
!!unsigned char a_row_id[512];
!!
!!ub2 buf_len = sizeof(b_row_id) - 1;
!!

```

```

!!memset(b_row_id, 0x00, sizeof(b_row_id));
!!memset(a_row_id, 0x00, sizeof(a_row_id));
#endif
/* Deleted end */

/* Added T.Kato 2004.12.21 New Oracle10g tool
kit */
#ifdef DEBUG
if (bylastname) tkvc_trace_on();
#endif

#ifdef BLANK_PAD_C_LAST
for (i = strlen(c_last); i < sizeof(c_last)-1; i++)
{
c_last[i] = ' ';
}
c_last[i] = '\0';
#endif
/* Added end */

for (i = 0; i < NITEMS; i++) {
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;
retry:
if (bylastname)
{
/* Replaced T.Kato 2004.12.21 New Oracle tool
kit */
/* cbctx.reexec = FALSE;*/

ordcount++;
cbctx.reexec = FALSE;
errcode = 0;
/*#define STRIP_BLANKS_C_LAST Always
no blanks */
#ifdef STRIP_BLANKS_C_LAST
for (i = strlen(c_last)-1; i >= 0 && (c_last[i] == '
'); i--)
{
c_last[i] = '\0';
}
#endif
/* Replaced end */

DBGLOG("ORD:[1]Start",0);
execstatus=OCIStmtExecute(tpcsvc,octx-
>curo0,errhp,100,0,
NULLP(CONST
OCI_Snapshot),NULLP(OCI_Snapshot),OCI_DEF
AULT);
DBGLOG("ORD:[1]End >%d",execstatus);
/* 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))
{
DISCARD
OCITransCommit(tpcsvc,errhp,OCI_DEFAULT);
retries++;
goto retry;
} else {

```

```

/* Added T.Kato 2004.12.21 New Oracle10g tool
kit */
#ifdef DEBUG
tkvc_trace_off();
#endif
/* Added end */
return -1;
}
}
if (execstatus == OCI_NO_DATA) /* there are
no more rows */
{
/* get rowcount, find middle one */
/* Replaced T.Kato 03.10.14 Add error check */
/* DISCARD OCIAttrGet(octx-
>curo0,OCI_HTYPE_STMT,&rcount,NULL, */
/* OCI_ATTR_ROW_COUNT,errhp); */

oci_stat = OCIAttrGet(octx-
>curo0,OCI_HTYPE_STMT,&rcount,NULL,
OCI_ATTR_ROW_COUNT,errhp);
DISCARD OCIERROR(errhp, oci_stat);

/* Deleted T.Kato 04.06.22 for Linux */
#if 0
! if (oci_stat == OCI_SUCCESS)
! {
! TpcUserLog(LOG_FILE_INF,
"ORDERSTATUS OCI_ATTR_ROW_COUNT
success\n");
! }
#endif
/* Deleted end */

/* Replaced end */

if (rcount < 1)
{
/* Replaced T.Kato 2004.12.21 New Oracle10g
tool kit */
#if 0
! TpcUserLog(LOG_FILE_INF,
"ORDERSTATUS rcount=%d\n",rcount);
! return (-1);
#endif
TpcUserLog(LOG_FILE_INF,
"ORDERSTATUS rcount=%d\n",rcount);
TpcUserLog(LOG_FILE_INF, "
w_id =%d\n",w_id);
TpcUserLog(LOG_FILE_INF, "
d_id =%d\n",d_id);
TpcUserLog(LOG_FILE_INF, "
c_last=%s\n",c_last);
TpcUserLog(LOG_FILE_INF, "
retries=%d\n",retries);
TpcUserLog(LOG_FILE_INF, "
errcode=%d\n",errcode);
TpcUserLog(LOG_FILE_INF, "
execstatus=%d\n",execstatus);
TpcUserLog(LOG_FILE_INF, "
ordcount=%d\n",ordcount);
#endif
#ifdef DEBUG
tkvc_trace_off();
#endif
return -1;
/* Replaced end */

}
octx->cust_idx=(rcount)/2 ;

```

```

}
else
{
/* count the number of rows */
DBGLOG("ORD:[2]Start",0);
execstatus=OCIStmtExecute(tpcsvc,octx-
>curo4,errhp,1,0,
NULLP(CONST
OCI_Snapshot),NULLP(OCI_Snapshot),OCI_DEF
AULT);
DBGLOG("ORD:[2]End >%d",execstatus);
if ((execstatus != OCI_NO_DATA) &&
(execstatus != OCI_SUCCESS))
{
errcode=OCIERROR(errhp, execstatus);
if ((errcode == NOT_SERIALIZABLE) ||
(errcode == RECOVER))
{
DISCARD
OCITransCommit(tpcsvc,errhp,OCI_DEFAULT);
retries++;
goto retry;
} else {
return -1;
}
}
}

/* Replaced T.Kato 2004.12.21 New Oracle10g
tool kit */
#if 0
! if (octx->rcount+1 < 2*10 )
! octx->cust_idx=(octx->rcount+1)/2 ;
! else /* */
! {
! cbctx.reexec = TRUE;
! cbctx.count = (octx->rcount+1)/2 ;
! DBGLOG("ORD:[3]Start",0);
! execstatus=OCIStmtExecute(tpcsvc,octx-
>curo0,errhp,cbctx.count,
! 0,NULLP(CONST
OCI_Snapshot),
! NULLP(OCI_Snapshot),OCI_DEFAULT);
! DBGLOG("ORD:[3]End
>%d",execstatus);
! /* will get OCI_NO_DATA if <100 found */
! if (cbctx.count > 0)
! {
! TpcUserLog (LOG_FILE_INF, "did not
get all rows ");
! return (-1);
! }
! if ((execstatus != OCI_NO_DATA) &&
(execstatus != OCI_SUCCESS))
! {
! errcode=OCIERROR(errhp, execstatus);
! if ((errcode == NOT_SERIALIZABLE) ||
(errcode == RECOVER))
! {
! DISCARD
OCITransCommit(tpcsvc,errhp,OCI_DEFAULT);
! retries++;
! goto retry;
! } else {
! return -1;
! }
! }
! octx->cust_idx=0 ;
! }
#endif

cbctx.reexec = TRUE;
cbctx.count = (octx->rcount+1)/2 ;

```

```

    execstatus=OCISmtExecute(tpcsvc,octx-
>curo0,errhp,cbctx.count,
        0,NULLP(CONST
OCI_Snapshot),
NULLP(OCI_Snapshot),OCI_DEFAULT);

    DISCARD OCIAtrGet(octx-
>curo0,OCI_HTYPE_STMT,&rcount,NULL,
OCI_ATTR_ROW_COUNT,errhp);

    /* will get OCI_NO_DATA if <100 found */
    if (cbctx.count != (unsigned int)rcount)
    {
        TpcUserLog (LOG_ERR, "did not get all
rows ");
        return (-1);
    }

    if ((execstatus != OCI_NO_DATA) &&
(execstatus != OCI_SUCCESS))
    {
        errcode=OCIERROR(errhp, execstatus);
        if((errcode == NOT_SERIALIZABLE) ||
(errcode == RECOVERR))
        {
            DISCARD
OCITransCommit(tpcsvc,errhp,OCI_DEFAULT);
            retries++;
            goto retry;
        } else {
            return -1;
        }
    }
    octx->cust_idx=cbctx.count - 1 ;
/* Replaced end */

}

    octx->c_rowid_cust = octx->c_rowid_ptr[octx-
>cust_idx];
    DBGLOG("ORD:[4]Start",0);
    execstatus=OCISmtExecute(tpcsvc,octx-
>curo1,errhp,1,0,
        NULLP(CONST
OCI_Snapshot),NULLP(OCI_Snapshot),OCI_DEF
AULT);
    DBGLOG("ORD:[4]End >%d",execstatus);
    if (execstatus != OCI_SUCCESS)
    {
        errcode=OCIERROR(errhp,execstatus);
        DISCARD
OCITransCommit(tpcsvc,errhp,OCI_DEFAULT);
        if((errcode == NOT_SERIALIZABLE) ||
(errcode == RECOVERR)
        || (errcode == SNAPSHOT_TOO_OLD))
        {
            retries++;
            goto retry;
        } else {
            return -1;
        }
    }
    else
    {
        DBGLOG("ORD:[5]Start",0);
        execstatus=OCISmtExecute(tpcsvc,octx-
>curo2,errhp,1,0,
        NULLP(CONST
OCI_Snapshot),NULLP(OCI_Snapshot),
OCI_DEFAULT);
        DBGLOG("ORD:[5]End >%d",execstatus);
        if (execstatus != OCI_SUCCESS)

```

```

    {
        errcode=OCIERROR(errhp,execstatus);
        DISCARD
OCITransCommit(tpcsvc,errhp,OCI_DEFAULT);
        if((errcode == NOT_SERIALIZABLE) ||
(errcode == RECOVERR)
        || (errcode == SNAPSHOT_TOO_OLD))
        {
            retries++;
            goto retry;
        }
        else
        {
            return -1;
        }
    }

/* Deleted T.Kato 2004.12.21 "o_rowid" was
deleted by New Oracle10g tool kit */
#ifndef ISO9
    /* Deleted T.Kato 2004.12.21 "o_rowid" was
deleted by New Oracle10g tool kit */
    #if 0
    !c2_w_id = w_id;
    !c2_d_id = d_id;
    !c2_c_id = c_id;
    !!
    !!OCIRowidToChar(octx->o_rowid, b_row_id,
&buf_len, errhp);
    #endif
    /* Deleted end */

#ifdef ISO9
    sysdate (sdate);
    if (!secondread)
        printf ("----- FIRST READ RESULT
(out) %s -----\\n", sdate);
    else
        printf ("----- SECOND READ RESULT
(out) %s -----\\n", sdate);

    printf ("c_id = %d\\n", c_id);
    printf ("c_last = %s\\n", c_last);
    printf ("c_first = %s\\n", c_first);
    printf ("c_middle = %s\\n", c_middle);
    printf ("c_balance = %7.2f\\n",
(float)c_balance/100);
    printf ("o_id = %d\\n", o_id);
    datelen = sizeof(o_entry_d);

OCIERROR(errhp,OCIDateToText(errhp,&o_ent
ry_d_base,(text*)FULLDATE,SIZ(FULLDATE),(t
ext*
)0,0,&datelen,o_entry_d));
    printf ("o_entry_d = %s\\n", o_entry_d);
    printf ("o_carrier_id = %d\\n", o_carrier_id);
    printf ("o_ol_cnt = %d\\n", o_ol_cnt);
    printf ("-----
\\n\\n", sdate);

    if (!secondread) {
        printf ("Sleep before re-read order at: %s\\n",
sdate);
        sleep (30);
        sysdate (sdate);
        printf ("Wake up and reread at: %s\\n",
sdate);
        secondread = 1;
        goto retry;
    }
}
#endif /* ISO9 */
}
    octx->o_l_w_id_len = sizeof(int);
    octx->o_l_d_id_len = sizeof(int);
    octx->o_l_o_id_len = sizeof(int);

    DBGLOG("ORD:[6]Start",0);

```

```

    execstatus = OCISmtExecute(tpcsvc,octx-
>curo3,errhp,o_ol_cnt,0,
        NULLP(CONST
OCI_Snapshot),NULLP(OCI_Snapshot),
OCI_DEFAULT |
OCI_COMMIT_ON_SUCCESS);
    DBGLOG("ORD:[6]End >%d",execstatus);
    if (execstatus != OCI_SUCCESS )
    {
        errcode=OCIERROR(errhp,execstatus);

/* Deleted T.Kato 2004.12.21 "o_rowid" was
deleted by New Oracle10g tool kit */
    #if 0
    !!OCIRowidToChar(octx->o_rowid, a_row_id,
&buf_len, errhp);
    !!TpcUserLog(LOG_FILE_INF, "DBG_LOG
start : w_id=%d d_id=%d c_id=%d\\n", f_w_id,
f_d_id, f_c_id);
    !!TpcUserLog(LOG_FILE_INF, "DBG_LOG
cur2 : w_id=%d d_id=%d c_id=%d\\n", c2_w_id,
c2_d_id, c2_c_id);
    !!TpcUserLog(LOG_FILE_INF, "DBG_LOG
cur2 : row_id=%s\\n", b_row_id);
    !!TpcUserLog(LOG_FILE_INF, "DBG_LOG
error : row_id=%s\\n", a_row_id);
    #endif
    /* Deleted end */

    DISCARD
OCITransCommit(tpcsvc,errhp,OCI_DEFAULT);
    if((errcode == NOT_SERIALIZABLE) ||
(errcode == RECOVERR)
    || (errcode == SNAPSHOT_TOO_OLD))
    {
        retries++;
        goto retry;
    }
    else
    {

/* Added T.Kato 2004.12.21 New Oracle10g tool
kit */
    #ifdef DEBUG
        if (bylastname) tkvc_trace_off();
    #endif
    /* Added end */
        return -1;
    }

}
/* clean up and convert the delivery dates */
for (i = 0; i < o_ol_cnt; i++)
{
    ol_del_len[i]=sizeof(ol_delivery_d[i]);
    DISCARD
OCIERROR(errhp,OCIDateToText(errhp,&o_l_d_
base[i],
        (const
text*)SHORTDATE,(ub1)strlen(SHORTDATE),(t
ext*)0,0,
        &ol_del_len[i], ol_delivery_d[i]);
}
/*
    cvtdmy(ol_d_base[i],ol_delivery_d[i]);
*/
}

/* Added T.Kato 2004.12.21 New Oracle10g tool
kit */
#ifdef DEBUG
    if (bylastname) tkvc_trace_off();
#endif
/* Added end */

```

```

return (0);
}

/* Added T.Kato 2004.12.21 New Oracle10g tool
kit */
#ifdef DEBUG
#define SQLTRCON "alter session set events
'10046 trace name context forever, level 12"
#define SQLTRCOFF "alter session set events
'10046 trace name context off"

/*static trace_on = 0: Moved to Global Area */

tkvc_trace_on()
{
    if (!trace_on)
    {
        char stmbuf[100];
        OCIStmt *curtrc;
        OCIHandleAlloc(tpcenv, (dvoid **)&curtrc,
        OCI_HTYPE_STMT, 0, (dvoid**)0);
        strcpy ((char *) stmbuf, SQLTRCON);
        DISCARD OCIERROR(errhp,
        OCIStmtPrepare(curtrc, errhp, stmbuf,
        strlen((char *)stmbuf),
        OCI_NTV_SYNTAX, OCI_DEFAULT));
        OCIERROR(errhp,
        OCIStmtExecute(tpcenv, curtrc,
        errhp,1,0,0,0,OCI_DEFAULT));
        OCIHandleFree((dvoid *)curtrc,
        OCI_HTYPE_STMT);
        trace_on++;
    }
}

tkvc_trace_off()
{
    if (trace_on)
    {
        char stmbuf[100];
        OCIStmt *curtrc;
        OCIHandleAlloc(tpcenv, (dvoid **)&curtrc,
        OCI_HTYPE_STMT, 0, (dvoid**)0);
        strcpy (stmbuf, SQLTRCOFF);
        DISCARD OCIERROR(errhp,
        OCIStmtPrepare(curtrc, errhp, stmbuf,
        strlen((char *)stmbuf),
        OCI_NTV_SYNTAX, OCI_DEFAULT));
        OCIERROR(errhp,
        OCIStmtExecute(tpcenv, curtrc,
        errhp,1,0,0,0,OCI_DEFAULT));
        OCIHandleFree((dvoid *)curtrc,
        OCI_HTYPE_STMT);
        trace_on = 0;
    }
}
#endif
/* Added end */

void tkvcodone ()
{
    if (octx)
        free (octx);
}

/* end of file tkvcord.c */

```

```

svrapl/3tier/plpay.c
.....

#ifdef RCSID
static char *RCSid =
"$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 "forlinux.h"
#include "log.h"
#include "tpcc.h"
#include "GlobalArea.h"
#include "prototype.h"

#define SQLTXT_INIT "BEGIN inittpc.init_pay;
END;"

int tkvcpin (void)
{
    text stmbuf[SQL_BUF_SIZE];

    pctx = (payctx *)malloc(sizeof(payctx));
    memset(pctx, (char)0, sizeof(payctx));

    /* cursor for init */
    DISCARD
    OCIERROR(errhp, OCIHandleAlloc(tpcenv,
    (dvoid **)&(pctx->curpi),
    OCI_HTYPE_STMT, 0, (dvoid**)0));

    DISCARD
    OCIERROR(errhp, OCIHandleAlloc(tpcenv,
    (dvoid **)&(pctx->curp0),
    OCI_HTYPE_STMT, 0, (dvoid**)0));

    /* build the init statement and execute it */

    sprintf ((char *)stmbuf, SQLTXT_INIT);
    DISCARD
    OCIERROR(errhp, OCIStmtPrepare(pctx->curpi,
    errhp, stmbuf,
    strlen((char *)stmbuf),
    OCI_NTV_SYNTAX, OCI_DEFAULT));
    DBGLOG("PAY:[1]Start", 0);

```

```

DISCARD OCIERROR(errhp,
OCIStmtExecute(tpcenv, pctx->curpi, errhp, 1, 0,
NULLP(CONST
OCI_Snapshot), NULLP(OCI_Snapshot), OCI_DEF
AULT));
DBGLOG("PAY:[1]End ", 0);

/* customer id != 0, go by last name */

/* Replaced 04.01.20 TUXEDO Client */
#if 0
! sqlfile("../blocks/paynz.sql", stmbuf);
#endif
sqlfile("/home/tpc/blocks/paynz.sql", stmbuf);
/* Replaced end */
DISCARD
OCIERROR(errhp, OCIStmtPrepare(pctx->curp0,
errhp, stmbuf,
strlen((char *)stmbuf),
OCI_NTV_SYNTAX, OCI_DEFAULT));

/* customer id == 0, go by last name */

/* Replaced 04.01.20 TUXEDO Client */
#if 0
! sqlfile("../blocks/payz.sql", stmbuf); /* sqlfile
opens $O/bench/.../blocks/... */
#endif
sqlfile("/home/tpc/blocks/payz.sql", stmbuf); /*
sqlfile opens $O/bench/.../blocks/... */
/* Replaced end */
DISCARD
OCIERROR(errhp, OCIStmtPrepare(pctx->curp1,
errhp, stmbuf,
strlen((char *)stmbuf),
OCI_NTV_SYNTAX, OCI_DEFAULT));

pctx->w_id_len = SIZ(w_id);
pctx->d_id_len = SIZ(d_id);
pctx->c_w_id_len = SIZ(c_w_id);
pctx->c_d_id_len = SIZ(c_d_id);
pctx->c_id_len = 0;
pctx->h_amount_len = SIZ(h_amount);
pctx->c_last_len = 0;
pctx->w_street_1_len = 0;
pctx->w_street_2_len = 0;
pctx->w_city_len = 0;
pctx->w_state_len = 0;
pctx->w_zip_len = 0;
pctx->d_street_1_len = 0;
pctx->d_street_2_len = 0;
pctx->d_city_len = 0;
pctx->d_state_len = 0;
pctx->d_zip_len = 0;
pctx->c_first_len = 0;
pctx->c_middle_len = 0;
pctx->c_street_1_len = 0;
pctx->c_street_2_len = 0;
pctx->c_city_len = 0;
pctx->c_state_len = 0;
pctx->c_zip_len = 0;
pctx->c_phone_len = 0;
pctx->c_since_len = 0;
pctx->c_credit_len = 0;
pctx->c_credit_lim_len = 0;
pctx->c_discount_len = 0;
pctx->c_balance_len = sizeof(double);
pctx->c_data_len = 0;
pctx->h_date_len = 0;
pctx->retries_len = SIZ(retries);
pctx->cr_date_len = 7;

/* bind variables */

```



```

OCIBNDPL(pctx->curp0, pctx->w_id_bp[0],
errhp,"w_id",ADR(w_id),SIZ(int),
SQLT_INT, NULL);
OCIBNDPL(pctx->curp0, pctx->d_id_bp[0],
errhp,"d_id",ADR(d_id),SIZ(int),
SQLT_INT, NULL);
OCIBND(pctx->curp0, pctx->c_w_id_bp[0],
errhp,"c_w_id",ADR(c_w_id),SIZ(int),
SQLT_INT);
OCIBND(pctx->curp0, pctx->c_d_id_bp[0],
errhp,"c_d_id",ADR(c_d_id),SIZ(int),
SQLT_INT);
OCIBND(pctx->curp0, pctx->c_id_bp[0],
errhp,"c_id",ADR(c_id),SIZ(int),
SQLT_INT);

/* Replaced T.kato 03.09.09 Oracle10g tool kit */
#if 0
! OCIBNDPL(pctx->curp0, pctx-
>h_amount_bp[0],
errhp,"h_amount",ADR(h_amount),
! SIZ(int),SQLT_INT, &pctx-
>h_amount_len);
#endif

#ifdef USE_IEEE_NUMBER
OCIBNDPL(pctx->curp0, pctx-
>h_amount_bp[0],
errhp,"h_amount",ADR(h_amount),
SIZ(float),SQLT_BFLOAT, &pctx-
>h_amount_len);
#else
OCIBNDPL(pctx->curp0, pctx-
>h_amount_bp[0],
errhp,"h_amount",ADR(h_amount),
SIZ(int),SQLT_INT, &pctx-
>h_amount_len);
#endif /* USE_IEEE_NUMBER */
/* Replaced end */

OCIBNDPL(pctx->curp0, pctx->c_last_bp[0],
errhp,"c_last",c_last,SIZ(c_last),
SQLT_STR, &pctx->c_last_len);
OCIBNDPL(pctx->curp0, pctx-
>w_street_1_bp[0],
errhp,"w_street_1",w_street_1,
SIZ(w_street_1),SQLT_STR, &pctx-
>w_street_1_len);
OCIBNDPL(pctx->curp0, pctx-
>w_street_2_bp[0],
errhp,"w_street_2",w_street_2,
SIZ(w_street_2),SQLT_STR, &pctx-
>w_street_2_len);
OCIBNDPL(pctx->curp0, pctx->w_city_bp[0],
errhp,"w_city",w_city,SIZ(w_city),
SQLT_STR, &pctx->w_city_len);
OCIBNDPL(pctx->curp0, pctx->w_state_bp[0],
errhp,"w_state",w_state,
SIZ(w_state), SQLT_STR, &pctx-
>w_state_len);
OCIBNDPL(pctx->curp0, pctx->w_zip_bp[0],
errhp,"w_zip",w_zip,SIZ(w_zip),
SQLT_STR, &pctx->w_zip_len);
OCIBNDPL(pctx->curp0, pctx-
>d_street_1_bp[0],
errhp,"d_street_1",d_street_1,
SIZ(d_street_1),SQLT_STR, &pctx-
>d_street_1_len);
OCIBNDPL(pctx->curp0, pctx-
>d_street_2_bp[0],
errhp,"d_street_2",d_street_2,
SIZ(d_street_2),SQLT_STR, &pctx-
>d_street_2_len);

```

```

OCIBNDPL(pctx->curp0, pctx->d_city_bp[0],
errhp,"d_city",d_city,SIZ(d_city),
SQLT_STR, &pctx->d_city_len);
OCIBNDPL(pctx->curp0, pctx->d_state_bp[0],
errhp,"d_state",d_state,
SIZ(d_state), SQLT_STR, &pctx-
>d_state_len);
OCIBNDPL(pctx->curp0, pctx->d_zip_bp[0],
errhp,"d_zip",d_zip,SIZ(d_zip),
SQLT_STR, &pctx->d_zip_len);
OCIBNDPL(pctx->curp0, pctx->c_first_bp[0],
errhp,"c_first",c_first,
SIZ(c_first), SQLT_STR, &pctx-
>c_first_len);
OCIBNDPL(pctx->curp0, pctx->c_middle_bp[0],
errhp,"c_middle",c_middle,2,
SQLT_AFC, &pctx->c_middle_len);
OCIBNDPL(pctx->curp0, pctx-
>c_street_1_bp[0],
errhp,"c_street_1",c_street_1,
SIZ(c_street_1),SQLT_STR, &pctx-
>c_street_1_len);
OCIBNDPL(pctx->curp0, pctx-
>c_street_2_bp[0],
errhp,"c_street_2",c_street_2,
SIZ(c_street_2),SQLT_STR, &pctx-
>c_street_2_len);
OCIBNDPL(pctx->curp0, pctx->c_city_bp[0],
errhp,"c_city",c_city,SIZ(c_city),
SQLT_STR, &pctx->c_city_len);
OCIBNDPL(pctx->curp0, pctx->c_state_bp[0],
errhp,"c_state",c_state,
SIZ(c_state), SQLT_STR, &pctx-
>c_state_len);
OCIBNDPL(pctx->curp0, pctx->c_zip_bp[0],
errhp,"c_zip",c_zip,SIZ(c_zip),
SQLT_STR, &pctx->c_zip_len);
OCIBNDPL(pctx->curp0, pctx->c_phone_bp[0],
errhp,"c_phone",c_phone,
SIZ(c_phone), SQLT_STR, &pctx-
>c_phone_len);
OCIBNDPL(pctx->curp0, pctx->c_since_bp[0],
errhp,"c_since",&c_since,
SIZ(OCIDate), SQLT_ODT, &pctx-
>c_since_len);
OCIBNDPL(pctx->curp0, pctx->c_credit_bp[0],
errhp,"c_credit",c_credit,
SIZ(c_credit),SQLT_CHR, &pctx-
>c_credit_len);
OCIBNDPL(pctx->curp0, pctx-
>c_credit_lim_bp[0], errhp,"c_credit_lim",
ADR(c_credit_lim),SIZ(int), SQLT_INT,
&pctx->c_credit_lim_len);
OCIBNDPL(pctx->curp0, pctx-
>c_discount_bp[0], errhp,"c_discount",
ADR(c_discount),SIZ(c_discount),
SQLT_FLT, &pctx->c_discount_len);

/* Replaced T.kato 03.09.09 Oracle10g tool kit */
#if 0
! OCIBNDPL(pctx->curp0, pctx-
>c_balance_bp[0], errhp,"c_balance",
! ADR(c_balance), SIZ(double),SQLT_FLT,
&pctx->c_balance_len);
#endif

#ifdef USE_IEEE_NUMBER
OCIBNDPL(pctx->curp0, pctx-
>c_balance_bp[0], errhp,"c_balance",
ADR(c_balance),
SIZ(double),SQLT_BDOUBLE, &pctx-
>c_balance_len);
#else
OCIBNDPL(pctx->curp0, pctx-
>c_balance_bp[0], errhp,"c_balance",

```

```

ADR(c_balance), SIZ(double),SQLT_FLT,
&pctx->c_balance_len);
#endif /* USE_IEEE_NUMBER */
/* Replaced end */

OCIBNDPL(pctx->curp0, pctx->c_data_bp[0],
errhp,"c_data",c_data,SIZ(c_data),
SQLT_STR, &pctx->c_data_len);
/*
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);
*/
OCIBNDPL(pctx->curp0, pctx->retries_bp[0],
errhp,"retry",ADR(retries),
SIZ(int), SQLT_INT, &pctx->retries_len);
OCIBNDPL(pctx->curp0, pctx->cr_date_bp[0],
errhp,"cr_date",ADR(cr_date),
SIZ(OCIDate),SQLT_ODT, &pctx-
>cr_date_len);

/* ---- Binds for the second cursor */

OCIBNDPL(pctx->curp1, pctx->w_id_bp[1],
errhp,"w_id",ADR(w_id),SIZ(int),
SQLT_INT, &pctx->w_id_len);
OCIBNDPL(pctx->curp1, pctx->d_id_bp[1],
errhp,"d_id",ADR(d_id),SIZ(int),
SQLT_INT, &pctx->d_id_len);
OCIBND(pctx->curp1, pctx->c_w_id_bp[1],
errhp,"c_w_id",ADR(c_w_id),SIZ(int),
SQLT_INT);
OCIBND(pctx->curp1, pctx->c_d_id_bp[1],
errhp,"c_d_id",ADR(c_d_id),SIZ(int),
SQLT_INT);
OCIBNDPL(pctx->curp1, pctx->c_id_bp[1],
errhp,"c_id",ADR(c_id),SIZ(int),
SQLT_INT, &pctx->c_id_len);

/* Replaced T.kato 03.09.09 Oracle10g tool kit */
#if 0
! OCIBNDPL(pctx->curp1, pctx-
>h_amount_bp[1],
errhp,"h_amount",ADR(h_amount),
! SIZ(int),SQLT_INT, &pctx-
>h_amount_len);
#endif

#ifdef USE_IEEE_NUMBER
OCIBNDPL(pctx->curp1, pctx-
>h_amount_bp[1],
errhp,"h_amount",ADR(h_amount),
SIZ(float),SQLT_BFLOAT, &pctx-
>h_amount_len);
#else
OCIBNDPL(pctx->curp1, pctx-
>h_amount_bp[1],
errhp,"h_amount",ADR(h_amount),
SIZ(int),SQLT_INT, &pctx-
>h_amount_len);
#endif /* USE_IEEE_NUMBER */
/* Replaced end */

OCIBND(pctx->curp1, pctx->c_last_bp[1],
errhp,"c_last",c_last,SIZ(c_last),
SQLT_STR);
OCIBNDPL(pctx->curp1, pctx-
>w_street_1_bp[1],
errhp,"w_street_1",w_street_1,
SIZ(w_street_1),SQLT_STR, &pctx-
>w_street_1_len);

```

```

OCIBNDPL(pctx->curp1, pctx-
>w_street_2_bp[1],
errhp,"w_street_2",w_street_2,
    SIZ(w_street_2),SQLT_STR, &pctx-
>w_street_2_len);
OCIBNDPL(pctx->curp1, pctx->w_city_bp[1],
errhp,"w_city",w_city,SIZ(w_city),
    SQLT_STR, &pctx->w_city_len);
OCIBNDPL(pctx->curp1, pctx->w_state_bp[1],
errhp,"w_state",w_state,
    SIZ(w_state), SQLT_STR, &pctx-
>w_state_len);
OCIBNDPL(pctx->curp1, pctx->w_zip_bp[1],
errhp,"w_zip",w_zip,SIZ(w_zip),
    SQLT_STR, &pctx->w_zip_len);
OCIBNDPL(pctx->curp1, pctx-
>d_street_1_bp[1],
errhp,"d_street_1",d_street_1,
    SIZ(d_street_1),SQLT_STR, &pctx-
>d_street_1_len);
OCIBNDPL(pctx->curp1, pctx-
>d_street_2_bp[1],
errhp,"d_street_2",d_street_2,
    SIZ(d_street_2),SQLT_STR, &pctx-
>d_street_2_len);
OCIBNDPL(pctx->curp1, pctx->d_city_bp[1],
errhp,"d_city",d_city,SIZ(d_city),
    SQLT_STR, &pctx->d_city_len);
OCIBNDPL(pctx->curp1, pctx->d_state_bp[1],
errhp,"d_state",d_state,
    SIZ(d_state), SQLT_STR, &pctx-
>d_state_len);
OCIBNDPL(pctx->curp1, pctx->d_zip_bp[1],
errhp,"d_zip",d_zip,SIZ(d_zip),
    SQLT_STR, &pctx->d_zip_len);
OCIBNDPL(pctx->curp1, pctx->c_first_bp[1],
errhp,"c_first",c_first,
    SIZ(c_first), SQLT_STR, &pctx-
>c_first_len);
OCIBNDPL(pctx->curp1, pctx->c_middle_bp[1],
errhp,"c_middle",c_middle,2,
    SQLT_AFC, &pctx->c_middle_len);

OCIBNDPL(pctx->curp1, pctx-
>c_street_1_bp[1],
errhp,"c_street_1",c_street_1,
    SIZ(c_street_1),SQLT_STR, &pctx-
>c_street_1_len);
OCIBNDPL(pctx->curp1, pctx-
>c_street_2_bp[1],
errhp,"c_street_2",c_street_2,
    SIZ(c_street_2),SQLT_STR, &pctx-
>c_street_2_len);
OCIBNDPL(pctx->curp1, pctx->c_city_bp[1],
errhp,"c_city",c_city,
    SIZ(c_city),SQLT_STR, &pctx-
>c_city_len);
OCIBNDPL(pctx->curp1, pctx->c_state_bp[1],
errhp,"c_state",c_state,
    SIZ(c_state), SQLT_STR, &pctx-
>c_state_len);
OCIBNDPL(pctx->curp1, pctx->c_zip_bp[1],
errhp,"c_zip",c_zip,SIZ(c_zip),
    SQLT_STR, &pctx->c_zip_len);
OCIBNDPL(pctx->curp1, pctx->c_phone_bp[1],
errhp,"c_phone",c_phone,
    SIZ(c_phone), SQLT_STR, &pctx-
>c_phone_len);
OCIBNDPL(pctx->curp1, pctx->c_since_bp[1],
errhp,"c_since",&c_since,
    SIZ(OCIDate), SQLT_ODT, &pctx-
>c_since_len);
OCIBNDPL(pctx->curp1, pctx->c_credit_bp[1],
errhp,"c_credit",c_credit,

```

```

    SIZ(c_credit),SQLT_CHR, &pctx-
>c_credit_len);
OCIBNDPL(pctx->curp1, pctx-
>c_credit_lim_bp[1], errhp,"c_credit_lim",
    ADR(c_credit_lim),SIZ(int), SQLT_INT,
&pctx->c_credit_lim_len);
OCIBNDPL(pctx->curp1, pctx-
>c_discount_bp[1], errhp,"c_discount",
    ADR(c_discount),SIZ(c_discount),
SQLT_FLT, &pctx->c_discount_len);

/* Replaced T.kato 03.09.09 Oracle10g tool kit */
#if 0
! OCIBNDPL(pctx->curp1, pctx-
>c_balance_bp[1], errhp,"c_balance",
! ADR(c_balance), SIZ(double),SQLT_FLT,
&pctx->c_balance_len);
#endif

#ifdef USE_IEEE_NUMBER
OCIBNDPL(pctx->curp1, pctx-
>c_balance_bp[1], errhp,"c_balance",
    ADR(c_balance),
SIZ(double),SQLT_BDOUBLE, &pctx-
>c_balance_len);
#else
OCIBNDPL(pctx->curp1, pctx-
>c_balance_bp[1], errhp,"c_balance",
    ADR(c_balance), SIZ(double),SQLT_FLT,
&pctx->c_balance_len);
#endif /* USE_IEEE_NUMBER */
/* Replaced end */

OCIBNDPL(pctx->curp1, pctx->c_data_bp[1],
errhp,"c_data",c_data,SIZ(c_data),
    SQLT_STR, &pctx->c_data_len);

/*
OCIBNDR(pctx->curp1, pctx->h_date_bp1,
errhp,"h_date",h_date,SIZ(h_date),
    SQLT_STR, &pctx->h_date_ind, &pctx-
>h_date_len, &pctx->h_date_rc);
*/
OCIBNDPL(pctx->curp1, pctx->retries_bp[1],
errhp,"retry",ADR(retries),
    SIZ(int), SQLT_INT, &pctx->retries_len);
OCIBNDPL(pctx->curp1, pctx->cr_date_bp[1],
errhp,"cr_date",ADR(cr_date),
    SIZ(OCIDate),SQLT_ODT, &pctx-
>cr_date_len);

return (0);
}

int tkvcp ()
{
retry:

pctx->w_id_len = SIZ(w_id);
pctx->d_id_len = SIZ(d_id);
pctx->c_w_id_len = 0;
pctx->c_d_id_len = 0;
pctx->c_id_len = 0;
pctx->h_amount_len = SIZ(h_amount);
pctx->c_last_len = SIZ(c_last);
pctx->w_street_1_len = 0;
pctx->w_street_2_len = 0;
pctx->w_city_len = 0;
pctx->w_state_len = 0;
pctx->w_zip_len = 0;
pctx->d_street_1_len = 0;
pctx->d_street_2_len = 0;

```

```

pctx->d_city_len = 0;
pctx->d_state_len = 0;
pctx->d_zip_len = 0;
pctx->c_first_len = 0;
pctx->c_middle_len = 0;
pctx->c_street_1_len = 0;
pctx->c_street_2_len = 0;
pctx->c_city_len = 0;
pctx->c_state_len = 0;
pctx->c_zip_len = 0;
pctx->c_phone_len = 0;
pctx->c_since_len = 0;
pctx->c_credit_len = 0;
pctx->c_credit_lim_len = 0;
pctx->c_discount_len = 0;
pctx->c_balance_len = sizeof(double);
pctx->c_data_len = 0;
pctx->h_date_len = 0;
pctx->retries_len = SIZ(retries);
pctx->cr_date_len = 7;

if(bylastname) {
DBGLOG("PAY:[2]Start",0);
execstatus=OCIStmtExecute(tpcsvc,pctx-
>curp1,errhp,1,0,
    NULL(CONST
OCISnapshot),NULL(OCISnapshot),

OCI_DEFAULT|OCI_COMMIT_ON_SUCCESS);
DBGLOG("PAY:[2]End >%d",execstatus);
} else {
DBGLOG("PAY:[3]Start",0);
execstatus=OCIStmtExecute(tpcsvc,pctx-
>curp0,errhp,1,0,
    NULL(CONST
OCISnapshot),NULL(OCISnapshot),

OCI_DEFAULT|OCI_COMMIT_ON_SUCCESS);
DBGLOG("PAY:[3]End >%d",execstatus);
}

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 if (errcode == SNAPSHOT_TOO_OLD)
{
retries++;
goto retry;
} else {
return -1;
}
}
return 0;

void tkvcpdone ()
{
if(pctx) {
free(pctx);
}
}

.....
svrapl/3tier/plsto.c

```

```

.....
#endif 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 "forlinux.h"
#include "log.h"
#include "tpcc.h"
#include "GlobalArea.h"
#include "prototype.h"

#define PLSQLSTO
#define SQLTXT "BEGIN
stocklevel.getstocklevel (:w_id, :d_id, :threshold,
\
:low_stock); END;"
#else
/* Replaced Hayashi 06.02.20 New Oracle10g
tool kit */
#if 0
/* Replaced Hayashi 06.01.12 New Oracle10g
tool kit */
/* Replaced T.Kato 03.07.18 New Oracle10i tool
kit */
/*#define SQLTXT "SELECT count (DISTINCT
s_i_id) \ */
/*#define SQLTXT "SELECT /*+ nocache (stok)
*/ count (DISTINCT s_i_id) \ */
/*#define SQLTXT "SELECT /*+ USE_NL(ordl)
nocache (stok) */ count (DISTINCT s_i_id) \ */
#endif
#define SQLTXT "SELECT /*+ USE_NL(ordl) */
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) \
order by ol_o_id desc"
#endif

int tkvcsinit ()
{
text stmbuf[SQL_BUF_SIZE];

```

```

sctx = (stocx *)malloc(sizeof(stocx));
memset(sctx, (char)0, sizeof(stocx));

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));
#endif 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);

/* Replaced T.kato 03.09.09 Oracle10g tool kit */
#if 0
! OCIBND(sctx->curs, sctx->threshold_bp, errhp,
":threshold", ADR(threshold),
! sizeof(int), SQLT_INT);
#endif

#ifdef USE_IEEE_NUMBER
OCIBND(sctx->curs, sctx->threshold_bp, errhp,
":threshold", ADR(threshold),
sizeof(float), SQLT_BFLOAT);
#else
OCIBND(sctx->curs, sctx->threshold_bp, errhp,
":threshold", ADR(threshold),
sizeof(int), SQLT_INT);
#endif /* USE_IEEE_NUMBER */
/* Replaced end */

#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);
}

int tkvcs ()
{
retry:
DBGLOG("STO:[1]Start", 0);
execstatus= OCIStmtExecute(tpcsvc, sctx-
>curs, errhp, 1, 0, 0, 0,

```

```

OCI_COMMIT_ON_SUCCESS |
OCI_DEFAULT);
DBGLOG("STO:[1]End >%d", execstatus);
if (execstatus != OCI_SUCCESS)
{
errcode=OCIERROR(errhp, execstatus);

OCITransCommit(tpcsvc, errhp, OCI_DEFAULT);
if((errcode == NOT_SERIALIZABLE) ||
(errcode == RECOVERER)
|| (errcode == SNAPSHOT_TOO_OLD))
{
retries++;
goto retry;
} else {
return -1;
}
}

return (0);
}

void tkvcsdone ()
{
if(sctx) free(sctx);
}

.....
svrapl/3tier/tpccpl.c
.....

#endif 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 "forlinux.h"
#include <stdio.h>
#include <sys/types.h>
#include <sys/poll.h>
#include <sys/time.h>
#include <unistd.h>
//#include <time.h>
#include "tpcc.h"
/* Added T.Kato 02.10.23 Ajustment interface for
transaction data organization format*/
#include "tpcc_info.h"
/* Added end */
#include "log.h"
#include "log_level.h"

```

```

#include "GlobalArea.h"
#include "prototype.h"

#define SQLTXT "alter session set
isolation_level = serializable"
#define SQLTXTTRC "alter session set
sql_trace = true"
#define SQLTXTTIM "alter session set
timed_statistics = true"

#ifdef ORA_NT
#undef boolean
#include "dpbcore.h"
#define gettime dpbtimef
#else
extern double gettime ();
#endif

/*
extern char oracle_home[256];
*/

/* NewOrder Binding stuff */

/* vmm313 void ocierror(fname, lineno, errhp,
status) */
int ocierror(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:
TpccUserLog(LOG_FILE_INF, "Module %s
Line %d\n", fname, lineno);
TpccUserLog(LOG_FILE_INF, "Error -
OCI_SUCCESS_WITH_INFO\n");

lstat = OCIErrorGet (errhp, recno++, (text *)
NULL, &errcode, errbuf,
(ub4) sizeof(errbuf),
OCI_HTYPE_ERROR);

TpccUserLog(LOG_FILE_INF, "Error - %s\n",
errbuf);
break;

case OCI_NEED_DATA:
TpccUserLog(LOG_FILE_INF, "Module %s
Line %d\n", fname, lineno);
TpccUserLog(LOG_FILE_INF, "Error -
OCI_NEED_DATA\n");
return (IRRECERR);

case OCI_NO_DATA:
TpccUserLog(LOG_FILE_INF, "Module %s
Line %d\n", fname, lineno);
TpccUserLog(LOG_FILE_INF, "Error -
OCI_NO_DATA\n");
return (IRRECERR);

case OCI_ERROR:
/* Replaced T.Kato 03.09.12 */
#if 0
! lstat = OCIErrorGet (errhp, (ub4) 1,
! (text *) NULL, &errcode, errbuf,

```

```

! (ub4) sizeof(errbuf),
OCI_HTYPE_ERROR);
! if (errcode == NOT_SERIALIZABLE) return
(errcode);
! if (errcode == SNAPSHOT_TOO_OLD) return
(errcode);
#endif

lstat = OCIErrorGet (errhp, (ub4) 1,
(text *) NULL, &errcode, errbuf,
(ub4) sizeof(errbuf),
OCI_HTYPE_ERROR);
if (errcode == NOT_SERIALIZABLE) {
TpccUserLog(LOG_FILE_INF, "Module %s
Line %d\n", fname, lineno);
TpccUserLog(LOG_FILE_INF, "Information
- NOT_SERIALIZABLE (OCI_ERROR)\n");
return (errcode);
}
if (errcode == SNAPSHOT_TOO_OLD) {
TpccUserLog(LOG_FILE_INF, "Module %s
Line %d\n", fname, lineno);
TpccUserLog(LOG_FILE_INF, "Information
- SNAPSHOT_TOO_OLD (OCI_ERROR)\n");
return (errcode);
}

/* Replaced end */
while (lstat != OCI_NO_DATA)
{
TpccUserLog(LOG_FILE_INF, "Module %s
Line %d\n", fname, lineno);
TpccUserLog(LOG_FILE_INF, "Error - %s\n",
errbuf);

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:
TpccUserLog(LOG_FILE_INF, "Module %s
Line %d\n", fname, lineno);
TpccUserLog(LOG_FILE_INF, "Error -
OCI_INVALID_HANDLE\n");
/* Replaced 03.05.15 TPCexit no argument */
// TPCexit(1);
TPCexit();
/* Replaced end */
exit(-1);

case OCI_STILL_EXECUTING:
TpccUserLog(LOG_FILE_INF, "Module %s
Line %d\n", fname, lineno);
TpccUserLog(LOG_FILE_INF, "Error -
OCI_STILL_EXECUTING\n");
return (IRRECERR);

case OCI_CONTINUE:
TpccUserLog(LOG_FILE_INF, "Module %s
Line %d\n", fname, lineno);
TpccUserLog(LOG_FILE_INF, "Error -
OCI_CONTINUE\n");
return (IRRECERR);

default:
TpccUserLog(LOG_FILE_INF, "Module %s
Line %d\n", fname, lineno);
TpccUserLog(LOG_FILE_INF, "Status - %d\n",
status);
return (IRRECERR);

```

```

}
return (RECOVERR);
}

FILE *vopen(char *fnam, char *mode)
{
FILE *fd;

#ifdef DEBUG
! fprintf(stderr, "tkvopen() fnam: %s,
mode: %s\n", fnam, mode);
#endif

fd = fopen((char *)fnam, (char *)mode);
if (!fd) {
TpccUserLog(LOG_FILE_INF, "fopen
on %s failed %d\n", fnam, fd);
exit(-1);
}
return(fd);
}

int sqlfile(char *fnam, text *linebuf)
{
FILE *fd;
int nulpt = 0;
char realfile[512];

#ifdef DEBUG
fprintf(stderr, "sqlfile() fnam: %s,
linebuf: %#x\n", fnam, linebuf);
#endif

/*
sprintf(realfile, "%s/bench/tpc/tpcc/blocks/%s", ora
cle_home, fnam);
*/
sprintf(realfile, "%s", fnam);
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 *orad)
{
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\0",day,month,year);

    return;
}

void cvtdmymhs (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\0",
        day,month,year,hour,min,sec);

    return;
}
#endif

void TPCexit (void)
{
    TpcUserLog(LOG_INF, "Server Apl end
procedure execute (TPCexit)\n");

    if (new_init) {
        tkvcndone();
        new_init = 0;
    }

    if (pay_init) {
        tkvcpdone();
        pay_init = 0;
    }

    if (ord_init) {
        tkvcodone();
        ord_init = 0;
    }

#ifdef DEL_ORA8I
    if (del_init) {
        tkvcddone();
        del_init = 0;
    }
#else
    if (del_init_oci) {
        tkvcddone(0);
        del_init_oci = 0;
    }

    if (del_init_plsql) {
        tkvcddone(1);
        del_init_plsql = 0;
    }
#endif

    if (sto_init) {
        tkvcsdone();
        sto_init = 0;
    }
}

/* Deleted T.Kato 040120 Shutdown can
disconnect server normally without the following
logic for TUXEDO. */
/* But You must be valid the
following logic for COM+. */
#if 0
!
OCIERROR(errhp,OCISessionEnd(tpcsvc,errhp,
tpcusr, OCI_DEFAULT));
! OCIERROR(errhp,OCIServerDetach(tpcsrv,
errhp, OCI_DEFAULT));
#endif
/* Deleted end */

    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);

/* Close Derivery log */
if (lfp) {
    fclose (lfp);
    lfp = NULL;
}
TpcUserLog(LOG_INF, "TPCexit all
finished\n");
}

int TPCinit (int id, char *uid, char *pwd)
{
/* Deleted T.Kato 02.10.24 Deleted derivery log
open
! char filename[40];
Deleted end */

    text stmbuf[100];

/* Added T.Kato 02.10.24 */
    sword rval;
/* Added End */

/* Replaced T.kato 02.10.24 Moved delivery log
open */
#if 0
! proc_no = id;
! sprintf (filename, "tpcc_%d.del", proc_no);
! if ((lfp = fopen (filename, "w")) == NULL) {
#ifndef TUX
! TpcUserLog ("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);
! }
#endif
// Init delevery flag
iflg = 0;
/* replaced end */

```

```

/* Replaced T.Kato 04.03.14 For Tuxedo
process */
#if 0
/* Replaced 03.05.19 For Thread */
#ifndef
!!
OCIInitialize(OCI_DEFAULT|OCI_OBJECT,(dvoi
d *)0,0,0,0);
#endif
!
OCIInitialize(OCI_THREADED|OCI_OBJECT,(d
void *)0,0,0,0);
/* Replaced end */
#endif

OCIInitialize(OCI_DEFAULT|OCI_OBJECT,(dvoi
d *)0,0,0,0);
/* Replaced end */

OCIEnvInit(&tpcenv, OCI_DEFAULT, 0, (dvoid
**));
OCIHandleAlloc((dvoid *)tpcenv, (dvoid
**) &tpcsrv, OCI_HTYPE_SERVER, 0, (dvoid
**));
OCIHandleAlloc((dvoid *)tpcenv, (dvoid
**) &errhp, OCI_HTYPE_ERROR, 0, (dvoid
**));
OCIHandleAlloc((dvoid *)tpcenv, (dvoid
**) &tpscv, OCI_HTYPE_SVCCTX, 0, (dvoid
**));

/* Replaced T.Kato 02.10.24 Retry until
successfully
! OCIAttach(tpcsrv, errhp, (text
*)0,0,OCI_DEFAULT);
*/
for (;;) {
    rval = OCIAttach(tpcsrv, errhp, (text
*)0,0,OCI_DEFAULT);
    if (rval == OCI_SUCCESS || rval ==
OCI_SUCCESS_WITH_INFO)
        break;
    OCIError(errhp, rval);
    sleep(1);
}
/* Replaced end */

OCIAttrSet((dvoid *)tpcsrv,
OCI_HTYPE_SVCCTX, (dvoid *)tpcsrv,
(ub4)0,OCI_ATTR_SERVER, errhp);
OCIHandleAlloc((dvoid *)tpcenv, (dvoid
**) &tpcusr, OCI_HTYPE_SESSION, 0, (dvoid
**));
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(tpcsrv,
errhp, tpcusr, OCI_CRED_RDBMS,
OCI_DEFAULT));

OCIAttrSet(tpcsrv, 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);
OCIStmtPrepare(curi, errhp, stmbuf,
strlen((char *)stmbuf), OCI_NTV_SYNTAX,
OCI_DEFAULT);
DBGLOG("INI:[1]Start",0);
OCIError(errhp,OCIStmtExecute(tpcsrv,
curi, errhp,1,0,0,0,OCI_DEFAULT));
DBGLOG("INI:[1]End ",0);
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);
    OCIStmtPrepare(curi, errhp, stmbuf,
strlen((char *)stmbuf),
OCI_NTV_SYNTAX, OCI_DEFAULT);
    OCIError(errhp, OCIStmtExecute(tpcsrv,
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);
    OCIStmtPrepare(curi, errhp, stmbuf,
strlen((char *)stmbuf),
OCI_NTV_SYNTAX, OCI_DEFAULT);
    DBGLOG("INI:[2]Start",0);
    OCIError(errhp, OCIStmtExecute(tpcsrv,
curi, errhp,1,0,0,0,OCI_DEFAULT));
    DBGLOG("INI:[2]End ",0);
    OCIHandleFree((dvoid *)curi,
OCI_HTYPE_STMT);
}

logon = 1;

OCIError(errhp,OCIDateSysDate(errhp,&cr_d
ate));

if (tkvcninit (0)) { /* new order */
    TPCexit ();
    return (-1);
}
else
    new_init = 1;

if (tkvcpinit (0)) { /* payment */
    TPCexit ();
    return (-1);
}
else
    pay_init = 1;

if (tkvcoint (0)) { /* order status */
    TPCexit ();
    return (-1);
}
else
    ord_init = 1;

#ifdef DEL_ORA8I
if (tkvcdinit (0)) { /* delivery */
    TPCexit ();
    return (-1);
}
else

```

```

    del_init = 1;
#else
if (tkvcdinit (0)) { /* delivery */
    TPCexit ();
    return (-1);
}
else
    del_init_oci = 1;

if (tkvcdinit (1)) { /* delivery */
    TPCexit ();
    return (-1);
}
else
    del_init_plsql = 1;
#endif

if (tkvcsinit (0)) { /* stock level */
    TPCexit ();
    return (-1);
}
else
    sto_init = 1;

return (0);
}

int TPCnew (struct newstruct *str)
{
/* Added T.Kato 02.11.25 */
#ifdef AVOID_DEADLOCK
    static int
    init_value_index[NITEMS]=(0,1,2,3,4,5,6,7,8,9,1
0,11,12,13,14);
#endif
/* Added end */
    int i;

    w_id = str->newin.w_id;
    d_id = str->newin.d_id;
    c_id = str->newin.c_id;

/* Added T.Kato 02.10.24 */
for (i = 0; i < 15; i++) {
    nol_i_id[i] = 0;
    nol_supply_w_id[i] = 0;
    nol_quantity[i] = 0;
}
/* Added end */

for (i = 0; i < 15; i++) {
/* Added T.Kato 02.10.24 */
    if((str->newin.ol_i_id[i] == 0) && (str-
>newin.ol_supply_w_id[i] == 0) && (str-
>newin.ol_quantity[i] == 0))
        break;
/* Added end */
    nol_i_id[i] = str->newin.ol_i_id[i];
    nol_supply_w_id[i] = str-
>newin.ol_supply_w_id[i];
/* Replaced T.kato 03.09.09 Oracle10g tool kit */
/* nol_quantity[i] = str->newin.ol_quantity[i];*/

#ifdef USE_IEEE_NUMBER
    nol_quantity[i] = (float)str-
>newin.ol_quantity[i];
#else
    nol_quantity[i] = str->newin.ol_quantity[i];
#endif
/* Replaced end */
}
}

```

```

retries = 0;

#ifndef AVOID_DEADLOCK

for (i = NITEMS; i > 0; i--) {
    if (no_l_i_id[i-1] > 0) {
        ordl_cnt = i;
        break;
    }
}

/* Replaced T.Kato 02.11.22 */
// for (i = 0; i < NITEMS; i++) indx[i] = i;
memcpy( indx, init_value_index, sizeof(indx) );
/* Replaced End */

q_sort_item(no_l_i_id, str, 0, ordl_cnt-1);

#endif

/*
vgetdate(cr_date); */

OCIERROR(errhp,OCIDateSysDate(errhp,&cr_date));

if ((str->newout.terror = tkvcn ()) != 0) {
    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_credil, 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);
/* Replaced T.Kato 02.11.13 */
#if 0
! str->newout.total_amount = total_amount;
#endif
str->newout.total_amount = 0.0;
/* Replaced end */
for (i = 0; i < o_ol_cnt; i++) {
    strncpy (str->newout.i_name[i], i_name[i],
25);
    str->newout.brand_generic[i] =
brand_generic[i][0];

/* Replaced T.kato 03.09.09 Oracle10g tool kit */
#if 0
! str->newout.s_quantity[i] = s_quantity[i];
! str->newout.i_price[i] = (float)(i_price[i])/100;
! str->newout.ol_amount[i] =
(float)(no_l_amount[i])/100;
#endif

```

```

#ifdef USE_IEEE_NUMBER
str->newout.s_quantity[i] = (int) s_quantity[i];
str->newout.i_price[i] = i_price[i]/100;
str->newout.ol_amount[i] =
no_l_amount[i]/100;
#else
str->newout.s_quantity[i] = s_quantity[i];
str->newout.i_price[i] = (float)(i_price[i])/100;
str->newout.ol_amount[i] =
(float)(no_l_amount[i])/100;
#endif /* USE_IEEE_NUMBER */
/* Replaced end */

/* Added T.Kato 02.11.13 */
str->newout.total_amount += str-
>newout.ol_amount[i];
/* Added end */

}

/* Added T.Kato 03.08.15 */
str->newout.total_amount =
(float)(str->newout.total_amount * (1.0 -
c_discount) * (1.0 + w_tax + d_tax));
/* Added End */
#ifndef AVOID_DEADLOCK
q_sort(indx, str, 0, ordl_cnt-1);
#endif

if (status)
    strcpy (str->newout.status, "Item number is
not valid");
else
    str->newout.status[0] = '0';
str->newout.retry = retries;
return (0);

}

int TPCpay (struct paystruct *str)
{

long double long64bit;

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;

/* Replaced T.kato 03.09.09 Oracle10g tool kit */
#if 0
! h_amount = str->payin.h_amount;
#endif

#ifdef USE_IEEE_NUMBER
h_amount = (float) str->payin.h_amount;
#else
h_amount = str->payin.h_amount;
#endif /* USE_IEEE_NUMBER */
/* Replaced end */

bylastname = str->payin.bylastname;

/* Added T.Kato 03.08.15 */
memset(c_data, 0x00, sizeof(c_data));
/* Added end */

/*
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;
    strcpy (c_last, " ");
}
retries = 0;

if ((str->payout.terror = tkvcn ()) != 0) {
    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,(ub1)strlen(FULLDATE),(text*)
0,0,&hlen,h_date));

/*
cvtdmy(c_since,c_since_d);
*/
sincelen=SIZ(c_since_d);

OCIERROR(errhp,OCIDateToText(errhp,&c_since,
(text*)SHORTDATE,(ub1)strlen(SHORTDATE),(t
ext*)0,0,&sincelen,c_since_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);

```

```

/* Replaced T.Kato 03.08.15 */
/*str->payout.c_credit_lim =
(float)(c_credit_lim)/100;*/

    long64bit = (long double)((c_credit_lim / 100.0
+ 0.005555) * 100.0);
    str->payout.c_credit_lim =
(float)((double)long64bit / 100.0);
/* replaced end */

    str->payout.c_discount = c_discount;
/* Replaced T.Kato 03.08.15 */
/*str->payout.c_balance =
(float)(c_balance)/100;*/
    long64bit = (long double)((c_balance / 100.0 +
0.005555) * 100.0);
    str->payout.c_balance =
(float)((double)long64bit / 100.0);
/* Replaced end */
    strncpy(str->payout.c_data, c_data, 201);
    strncpy(str->payout.h_date, (char*)h_date,
20);
    str->payout.retry = retries;
    return (0);
}

int TPCord (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;
        strcpy(c_last, "");
    }
    retries = 0;

    if ((str->ordout.terror = tkvco ()) != 0) {
        if (str->ordout.terror != RECOVER)
            str->ordout.terror = IRRECERR;
        return (-1);
    }

    datelen = sizeof(o_entry_d);
    OCIERROR(errhp,
    OCIDateToText(errhp,&o_entry_d_base,(text*)F
ULLDATE,SIZ(FULLDATE),(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";

```

```

/* Replaced by TSL -- BEGIN -- 2006.03.17
adjust data on DB. */
/* if ( !strcmp((char*)ol_delivery_d[i], "15-09-
1911") ) */

    if ( !strcmp((char*)ol_delivery_d[i], "01-01-
1811") )
/* Replaced by TSL -- END -- 2006.03.17 adjust
data on DB. */

        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];

/* Replaced T.kato 03.09.09 Oracle10g tool kit */
#if 0
! str->ordout.ol_quantity[i] = ol_quantity[i];
! str->ordout.ol_amount[i] =
(float)(ol_amount[i])/100;
#endif

#ifdef USE_IEEE_NUMBER
    str->ordout.ol_quantity[i] = (int) ol_quantity[i];
    str->ordout.ol_amount[i] = ol_amount[i]/100;
#else
    str->ordout.ol_quantity[i] = ol_quantity[i];
    str->ordout.ol_amount[i] =
(float)(ol_amount[i])/100;
#endif /* USE_IEEE_NUMBER */
/* Replaced end */

    strncpy(str->ordout.ol_delivery_d[i],
(char*)ol_delivery_d[i], 11);
}
str->ordout.retry = retries;
return (0);
}

int TPCdel (struct delstruct *str)
{
/* Replaced T.kato 02.10.24 Change the delivery
log writing method */
#if 0
! double tr_end;
! int i;
#endif

    int i;

/* Replaced T.kato 03.12.22 Convert to linux
time. */
#if 0
! SYSTEMTIME systemTime;
! struct tm times;
#else
    struct timeval times;
    int msec;
#endif
/* Replaced end */

    char filename[40];
    //int svrcnt;
/* Replaced end */

/* Added T.Kato 02.10.24 Open the delivery log
file */
    if (iflg == 0)

```

```

{
    // Execute first delivery transaction
    sprintf (filename,
"/home/tpc/dellog/tpcc_%08d.del", (int)getpid());

    if ((lfp = fopen (filename, "w")) == NULL) {
        TpcUserLog (LOG_FILE_INF,
"DELIVERY: Error in TPC-C server %d: Failed to
open %s\n",
            proc_no, filename);
        return (-1);
    }

    // Set first execution indicator
    iflg = 1;
}
/* Added end */

    w_id = str->delin.w_id;
    o_carrier_id = str->delin.o_carrier_id;
    retries = 0;
/*
    vgetdate(cr_date); */

    OCIERROR(errhp, OCIDateSysDate(errhp, &cr_d
ate));
#ifdef DEL_ORA8I
    if ((str->delout.terror = tkvcd ()) != 0) {
#else
    if ((str->delout.terror = tkvcd (PLSQLFLAG)) !=
0) { // "PLSQLFLAG" are supplied from
Compile option!!
#endif
        if (str->delout.terror == DEL_ERROR)
            return DEL_ERROR;
        if (str->delout.terror != RECOVER)
            str->delout.terror = IRRECERR;
        return (-1);
    }

/* Replaced T.Kato 02.10.24 Changed time
stamp method */
#if 0
! 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);
#endif

/* Replaced T.Kato 03.12.22 Convert to linux
time. */
#if 0
! GetLocalTime(&systemTime);
! times.tm_year = (int)systemTime.wYear -
1900;
! times.tm_mon = (int)systemTime.wMonth - 1;
! times.tm_mday = (int)systemTime.wDay;
! times.tm_hour = (int)systemTime.wHour;
! times.tm_min = (int)systemTime.wMinute;
! times.tm_sec = (int)systemTime.wSecond;
!
! fprintf(lfp, "%09d%03d %09d%03d %d %d", str-
>delin.startsec,
! str->delin.startusec, ((long)mktime
(&times)), (long)systemTime.wMilliseconds, w_id,
o_carrier_id);
/* Replaced end */
#else
/* get system time */
gettimeofday(&times, 0);
msec = times.tv_usec / 1000;

```



```

fprintf(lfp,"%010d%03d %010d%03d %d %d",(int)
)str->delin.startsec,
(int)str->delin.startusec,(int)times.tv_sec,
msec, w_id,o_carrier_id);
#endif
/* Replaced end T.Kato */

for (i = 0; i < 10; i++) {
    fprintf (lfp, " %d %d", i + 1, del_o_id[i]);
    if (del_o_id[i] <= 0) {
        TpcUserLog (LOG_FILE_INF,
"DELIVERY: no new order for w_id: %d,
d_id %d\n",
        w_id, i + 1);
    }
}
fprintf (lfp, " %d\n", retries);
str->delout.terror = NOERR;
str->delout.retry = retries;
return (0);
}

int TPCsto (struct stostruct *str)
{
    w_id = str->stoin.w_id;
    d_id = str->stoin.d_id;

/* Replaced T.kato 03.09.09 Oracle10g tool kit */
#if 0
! threshold = str->stoin.threshold;
#endif

#ifdef USE_IEEE_NUMBER
    threshold = (float) str->stoin.threshold;
#else
    threshold = str->stoin.threshold;
#endif /* USE_IEEE_NUMBER */
/* Replaced end */

    retries = 0;

    if ((str->stoout.terror = tkvcs ()) != 0) {
        if (str->stoout.terror != RECOVERR)
            str->stoout.terror = IRRECERR;
        return (-1);
    }

    str->stoout.terror = NOERR;
    str->stoout.low_stock = low_stock;
    str->stoout.retry = retries;
    return (0);
}

#ifdef AVOID_DEADLOCK

/* Added T.Kato 02.11.22 */
void q_sort_item(int *arr,struct newstruct *str,int
left, int right)
{
    int i, last;

    if(left >= right)
        return;
    swap(str,left,(left+right)/2);
    last = left;
    for(i=left+1;i<=right;i++)
        if(arr[i] < arr[left])
            swap(str,last,i);
    swap(str,left,last);
    q_sort(arr,str,left,last-1);
    q_sort(arr,str,last+1,right);
}

```

```

swap_item(str,left,last);
q_sort_item(arr,str,left,last-1);
q_sort_item(arr,str,last+1,right);
}

void swap_item(struct newstruct *str, int i, int j)
{
    int temp;

/* Added T.kato 03.09.09 Oracle10g tool kit */
#ifdef USE_IEEE_NUMBER
    float temp_float;
#endif
/* Added end */

    temp = indx[i];
    indx[i] = indx[j];
    indx[j] = temp;

    temp = nol_i_id[i];
    nol_i_id[i] = nol_i_id[j];
    nol_i_id[j] = temp;

    temp = nol_supply_w_id[i];
    nol_supply_w_id[i] = nol_supply_w_id[j];
    nol_supply_w_id[j] = temp;

/* Replaced T.kato 03.09.09 Oracle10g tool kit */
#if 0
! temp = nol_quantity[i];
! nol_quantity[i] = nol_quantity[j];
! nol_quantity[j] = temp;
#endif

#ifdef USE_IEEE_NUMBER
    temp_float = nol_quantity[i];
    nol_quantity[i] = nol_quantity[j];
    nol_quantity[j] = temp_float;
#else
    temp = nol_quantity[i];
    nol_quantity[i] = nol_quantity[j];

    nol_quantity[j] = temp;
#endif /* USE_IEEE_NUMBER */
/* Replaced end */

/* Added T.Kato 03.08.15 */
temp = str->newin.ol_quantity[i];
str->newin.ol_quantity[i] = str->
newin.ol_quantity[j];
str->newin.ol_quantity[j] = temp;
/* Added End */
}
/* Added end */

void q_sort(int *arr,struct newstruct *str,int left,
int right)
{
    int i, last;

    if(left >= right)
        return;
    swap(str,left,(left+right)/2);
    last = left;
    for(i=left+1;i<=right;i++)
        if(arr[i] < arr[left])
            swap(str,last,i);
    swap(str,left,last);
    q_sort(arr,str,left,last-1);
    q_sort(arr,str,last+1,right);
}

```

```

void swap(struct newstruct *str, int i, int j)
{
    int temp;
    char tmpstr[25];
    char tmpch;

/* Added T.Kato 02.11.13 */
    float tmpflt;
/* Added end */

    temp = indx[i];
    indx[i] = indx[j];
    indx[j] = temp;

/* Deleted T.Kato 02.11.22 */
#if 0
! temp = nol_i_id[i];
! nol_i_id[i] = nol_i_id[j];
! nol_i_id[j] = temp;
!
! temp = nol_supply_w_id[i];
! nol_supply_w_id[i] = nol_supply_w_id[j];
! nol_supply_w_id[j] = temp;
!
! temp = nol_quantity[i];
! nol_quantity[i] = nol_quantity[j];
! nol_quantity[j] = temp;
#endif
/* Deleted End */

/* Replaced T.Kato 03.03.19 Chenged Oracle
10i tool kit */
#if 0
! strcpy(tmpstr,str->newout.i_name[i]);
! strcpy(str->newout.i_name[i],str->
newout.i_name[j]);
! strcpy(str->newout.i_name[j],tmpstr);
#endif
    strcpy(tmpstr,str->newout.i_name[i],25);
    strcpy(str->newout.i_name[i],str->
newout.i_name[j],25);
    strcpy(str->newout.i_name[j],tmpstr,25);
/* Replaced end */

/* Added T.Kato 03.08.15 */
temp = str->newin.ol_quantity[i];
str->newin.ol_quantity[i] = str->
newin.ol_quantity[j];
str->newin.ol_quantity[j] = temp;
/* Added End */

temp = str->newout.s_quantity[i];
str->newout.s_quantity[i] = str->
newout.s_quantity[j];
str->newout.s_quantity[j] = temp;

tmpch = str->newout.brand_generic[i];
str->newout.brand_generic[i] = str->
newout.brand_generic[j];

str->newout.brand_generic[j] = tmpch;

/* Replaced T.Kato 02.11.13 (int)temp =>
(float)tmpflt */
#if 0
! temp = str->newout.i_price[i];
! str->newout.i_price[i] = str->newout.i_price[j];
! str->newout.i_price[j] = temp;
!
! temp = str->newout.ol_amount[i];
! str->newout.ol_amount[i] = str->
newout.ol_amount[j];
! str->newout.ol_amount[j] = temp;

```

```
#endif

tmpflt = str->newout.i_price[i];
str->newout.i_price[i] = str->newout.i_price[j];
str->newout.i_price[j] = tmpflt;

tmpflt = str->newout.ol_amount[i];
str->newout.ol_amount[i] = str-
>newout.ol_amount[j];
str->newout.ol_amount[j] = tmpflt;
/* Replaced end */

}

#endif
```

Appendix C:

RTE Scripts

```

.....
rte01.conf
.....
#
# rte01.conf :configuration file for TPC-C
Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl089a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 374001
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl089a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 377751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl089a
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 381501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl089b
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 385001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl089b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 385251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl089b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 389001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl089b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 392751
  LOGPATH = /w05

```

```

LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl090a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 616001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl090a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 616501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl090a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 620251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl090a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 624001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl090b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 627001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl090b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 627501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl090b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 631001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl090b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 634501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
WAREHOUSE = 193600
MEASUREMENT = 21600
PAY-MIX = 4302
ORD-MIX = 402
DEL-MIX = 402
STK-MIX = 402
NEW-KEYING = 18010

```

```

PAY-KEYING = 3010
ORD-KEYING = 2010
DEL-KEYING = 2010
STK-KEYING = 2010
NEW-THINK = 12020
PAY-THINK = 12020
ORD-THINK = 10020
DEL-THINK = 5020
STK-THINK = 5020
CONST-CLAST = 111
CONST-CID = 1023
CONST-IID = 8191
THR-PER-PROC = 250
SYNC = 0
ENDVARIABLE
ENDGROUP

.....
rte02.conf
.....
#
# rte02.conf :configuration file for TPC-C
Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl007a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl007a
  SUTPORT = 80
  SUTTERM = 1750
  BASENO = 3751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl007b
  SUTPORT = 80
  SUTTERM = 2000
  BASENO = 5501
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl007b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 7501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl007b
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 242001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl008a
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 242001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl008a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 242251

```

```

LOGPATH = /w03
LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl008a
  SUTPORT = 80
  SUTTERM = 1500
  BASENO = 246001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl008b
  SUTPORT = 80
  SUTTERM = 2250
  BASENO = 247501
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl008b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 249751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl071a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1782001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl071a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1782501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl071a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1786251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl071a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 1790001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl071b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1793001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl071b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1793501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT

```

```

STARTSUT
  SUTHOST = cl071b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1797001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl071b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1800501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
  WAREHOUSE = 193600
  MEASUREMENT = 21600
  PAY-MIX = 4302
  ORD-MIX = 402
  DEL-MIX = 402
  STK-MIX = 402
  NEW-KEYING = 18010
  PAY-KEYING = 3010
  ORD-KEYING = 2010
  DEL-KEYING = 2010
  STK-KEYING = 2010
  NEW-THINK = 12020
  PAY-THINK = 12020
  ORD-THINK = 10020
  DEL-THINK = 5020
  STK-THINK = 5020
  CONST-CLAST = 111
  CONST-CID = 1023
  CONST-IID = 8191
  THR-PER-PROC = 250
  SYNC = 0
ENDVARIABLE
ENDGROUP

.....
rte03.conf
.....

#
#  rte03.conf :configuration file for TPC-C
# Rev3.0
#  Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl009a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 484001
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl009a
  SUTPORT = 80
  SUTTERM = 1750
  BASENO = 487751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl009b
  SUTPORT = 80
  SUTTERM = 2000

```

```

BASENO = 489501
LOGPATH = /w01
LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl009b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 491501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl010a
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 726001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl010a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 726251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl010a
  SUTPORT = 80
  SUTTERM = 1500
  BASENO = 730001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl010b
  SUTPORT = 80
  SUTTERM = 2250
  BASENO = 731501
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl010b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 733751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl011a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 968001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl011a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 968501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl011a
  SUTPORT = 80
  SUTTERM = 1250
  BASENO = 972251
  LOGPATH = /w07
  LOGLEVEL = 0

```

```

ENDSUT
STARTSUT
  SUTHOST = cl011b
  SUTPORT = 80
  SUTTERM = 2500
  BASENO = 973501
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl011b
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 976001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl012a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1210001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl012a
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1210501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl012a
  SUTPORT = 80
  SUTTERM = 1500
  BASENO = 1214001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl012b
  SUTPORT = 80
  SUTTERM = 2000
  BASENO = 1215501
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl012b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1217501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
  WAREHOUSE = 193600
  MEASUREMENT = 21600
  PAY-MIX = 4302
  ORD-MIX = 402
  DEL-MIX = 402
  STK-MIX = 402
  NEW-KEYING = 18010
  PAY-KEYING = 3010
  ORD-KEYING = 2010
  DEL-KEYING = 2010
  STK-KEYING = 2010
  NEW-THINK = 12020
  PAY-THINK = 12020
  ORD-THINK = 10020
  DEL-THINK = 5020
  STK-THINK = 5020

CONST-CLAST = 111
CONST-CID = 1023
CONST-IID = 8191
THR-PER-PROC = 250
SYNC = 0
ENDVARIABLE
ENDGROUP

.....
rte04.conf
.....

#
# rte04.conf :configuration file for TPC-C
Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl013a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1452001
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl013a
  SUTPORT = 80
  SUTTERM = 1750
  BASENO = 1455751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl013a
  SUTPORT = 80
  SUTTERM = 2000
  BASENO = 1457501
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl013b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1459501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl014a
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 1694001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl014a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1694251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl014a
  SUTPORT = 80
  SUTTERM = 1500
  BASENO = 1698001
  LOGPATH = /w04

LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl014b
  SUTPORT = 80
  SUTTERM = 2250
  BASENO = 1699501
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl014b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 1701751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl072a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 88001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl072a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 88501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl072a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 92251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl072a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 96001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl072b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 99001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl072b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 99501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl072b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 103001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT

```

```

SUTHOST = cl072b
SUTPORT = 80
SUTTERM = 3500
BASENO = 106501
LOGPATH = /w11
LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
WAREHOUSE = 193600
MEASUREMENT = 21600
PAY-MIX = 4302
ORD-MIX = 402
DEL-MIX = 402
STK-MIX = 402
NEW-KEYING = 18010
PAY-KEYING = 3010
ORD-KEYING = 2010
DEL-KEYING = 2010
STK-KEYING = 2010
NEW-THINK = 12020
PAY-THINK = 12020
ORD-THINK = 10020
DEL-THINK = 5020
STK-THINK = 5020
CONST-CLAST = 111
CONST-CID = 1023
CONST-IID = 8191
THR-PER-PROC = 250
SYNC = 0
ENDVARIABLE
ENDGROUP

.....
rte05.conf
.....

#
# rte05.conf :configuration file for TPC-C
# Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
STARTRTE
STARTSUT
SUTHOST = cl017a
SUTPORT = 80
SUTTERM = 3750
BASENO = 11001
LOGPATH = /w00
LOGLEVEL = 0
ENDSUT
STARTSUT
SUTHOST = cl017a
SUTPORT = 80
SUTTERM = 1750
BASENO = 14751
LOGPATH = /w01
LOGLEVEL = 0
ENDSUT
STARTSUT
SUTHOST = cl017b
SUTPORT = 80
SUTTERM = 2000
BASENO = 16501
LOGPATH = /w01
LOGLEVEL = 0
ENDSUT
STARTSUT
SUTHOST = cl017b
SUTPORT = 80
SUTTERM = 3500
BASENO = 18501

```

```

LOGPATH = /w02
LOGLEVEL = 0
ENDSUT
STARTSUT
SUTHOST = cl018a
SUTPORT = 80
SUTTERM = 250
BASENO = 253001
LOGPATH = /w02
LOGLEVEL = 0
ENDSUT
STARTSUT
SUTHOST = cl018a
SUTPORT = 80
SUTTERM = 3750
BASENO = 253251
LOGPATH = /w03
LOGLEVEL = 0
ENDSUT
STARTSUT
SUTHOST = cl018a
SUTPORT = 80
SUTTERM = 1500
BASENO = 257001
LOGPATH = /w04
LOGLEVEL = 0
ENDSUT
STARTSUT
SUTHOST = cl018b
SUTPORT = 80
SUTTERM = 2250
BASENO = 258501
LOGPATH = /w04
LOGLEVEL = 0
ENDSUT
STARTSUT
SUTHOST = cl018b
SUTPORT = 80
SUTTERM = 3250
BASENO = 260751
LOGPATH = /w05
LOGLEVEL = 0
ENDSUT
STARTSUT
SUTHOST = cl019a
SUTPORT = 80
SUTTERM = 500
BASENO = 495001
LOGPATH = /w05
LOGLEVEL = 0
ENDSUT
STARTSUT
SUTHOST = cl019a
SUTPORT = 80
SUTTERM = 3750
BASENO = 495501
LOGPATH = /w06
LOGLEVEL = 0
ENDSUT
STARTSUT
SUTHOST = cl019a
SUTPORT = 80
SUTTERM = 1250
BASENO = 499251
LOGPATH = /w07
LOGLEVEL = 0
ENDSUT
STARTSUT
SUTHOST = cl019b
SUTPORT = 80
SUTTERM = 2500
BASENO = 500501
LOGPATH = /w07
LOGLEVEL = 0
ENDSUT

```

```

STARTSUT
SUTHOST = cl019b
SUTPORT = 80
SUTTERM = 3000
BASENO = 503001
LOGPATH = /w08
LOGLEVEL = 0
ENDSUT
STARTSUT
SUTHOST = cl020a
SUTPORT = 80
SUTTERM = 500
BASENO = 737001
LOGPATH = /w08
LOGLEVEL = 0
ENDSUT
STARTSUT
SUTHOST = cl020a
SUTPORT = 80
SUTTERM = 3500
BASENO = 737501
LOGPATH = /w09
LOGLEVEL = 0
ENDSUT
STARTSUT
SUTHOST = cl020a
SUTPORT = 80
SUTTERM = 1500
BASENO = 741001
LOGPATH = /w10
LOGLEVEL = 0
ENDSUT
STARTSUT
SUTHOST = cl020b
SUTPORT = 80
SUTTERM = 2000
BASENO = 742501
LOGPATH = /w10
LOGLEVEL = 0
ENDSUT
STARTSUT
SUTHOST = cl020b
SUTPORT = 80
SUTTERM = 3500
BASENO = 744501
LOGPATH = /w11
LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
WAREHOUSE = 193600
MEASUREMENT = 21600
PAY-MIX = 4302
ORD-MIX = 402
DEL-MIX = 402
STK-MIX = 402
NEW-KEYING = 18010
PAY-KEYING = 3010
ORD-KEYING = 2010
DEL-KEYING = 2010
STK-KEYING = 2010
NEW-THINK = 12020
PAY-THINK = 12020
ORD-THINK = 10020
DEL-THINK = 5020
STK-THINK = 5020
CONST-CLAST = 111
CONST-CID = 1023
CONST-IID = 8191
THR-PER-PROC = 250
SYNC = 0
ENDVARIABLE
ENDGROUP

```

```

.....
rte06.conf
.....
#
# rte06.conf :configuration file for TPC-C
Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl021a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 979001
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl021a
  SUTPORT = 80
  SUTTERM = 1750
  BASENO = 982751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl021b
  SUTPORT = 80
  SUTTERM = 2000
  BASENO = 984501
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl021b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 986501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl022a
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 1221001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl022a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1221251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl022a
  SUTPORT = 80
  SUTTERM = 1500
  BASENO = 1225001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl022b
  SUTPORT = 80
  SUTTERM = 2250
  BASENO = 1226501
  LOGPATH = /w04
  LOGLEVEL = 0

```

```

ENDSUT
STARTSUT
  SUTHOST = cl022b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 1228751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl095a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 330001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl095a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 330501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl095a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 334251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl095a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 338001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl095b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 341001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl095b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 341501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl095b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 345001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl095b
  SUTPORT = 80
  SUTTERM = 80
  BASENO = 348501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE

```

```

WAREHOUSE = 193600
MEASUREMENT = 21600
PAY-MIX = 4302
ORD-MIX = 402
DEL-MIX = 402
STK-MIX = 402
NEW-KEYING = 18010
PAY-KEYING = 3010
ORD-KEYING = 2010
DEL-KEYING = 2010
STK-KEYING = 2010
NEW-THINK = 12020
PAY-THINK = 12020
ORD-THINK = 10020
DEL-THINK = 5020
STK-THINK = 5020
CONST-CLAST = 111
CONST-CID = 1023
CONST-IID = 8191
THR-PER-PROC = 250
SYNC = 0
ENDVARIABLE
ENDGROUP

.....
rte07.conf
.....
#
# rte07.conf :configuration file for TPC-C
Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl025a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1463001
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl025a
  SUTPORT = 80
  SUTTERM = 1750
  BASENO = 1466751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl025b
  SUTPORT = 80
  SUTTERM = 2000
  BASENO = 1468501
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl025b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1470501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl026a
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 1705001
  LOGPATH = /w02

```

```

LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl026a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1705251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl026a
  SUTPORT = 80
  SUTTERM = 1500
  BASENO = 1709001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl026b
  SUTPORT = 80
  SUTTERM = 2250
  BASENO = 1710501
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl026b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 1712751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl027a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1232001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl027a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1232501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl027a
  SUTPORT = 80
  SUTTERM = 1250
  BASENO = 1236251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl027b
  SUTPORT = 80
  SUTTERM = 2500
  BASENO = 1237501
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl027b
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 1240001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT

```

```

SUTHOST = cl028a
SUTPORT = 80
SUTTERM = 500
BASENO = 1474001
LOGPATH = /w08
LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl028a
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1474501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl028a
  SUTPORT = 80
  SUTTERM = 1500
  BASENO = 1478001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl028b
  SUTPORT = 80
  SUTTERM = 2000
  BASENO = 1479501
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl028b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1481501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
  WAREHOUSE = 193600
  MEASUREMENT = 21600
  PAY-MIX = 4302
  ORD-MIX = 402
  DEL-MIX = 402
  STK-MIX = 402
  NEW-KEYING = 18010
  PAY-KEYING = 3010
  ORD-KEYING = 2010
  DEL-KEYING = 2010
  STK-KEYING = 2010
  NEW-THINK = 12020
  PAY-THINK = 12020
  ORD-THINK = 10020
  DEL-THINK = 5020
  STK-THINK = 5020
  CONST-CLAST = 111
  CONST-CID = 1023
  CONST-IID = 8191
  THR-PER-PROC = 250
  SYNC = 0
ENDVARIABLE
ENDGROUP

.....
rte08.conf
.....

#
# rte08.conf : configuration file for TPC-C
# Rev3.0
# Author : mkdef - Auto Configurator for R3-
#

```

```

STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl029a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1716001
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl029a
  SUTPORT = 80
  SUTTERM = 1750
  BASENO = 1719751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl029b
  SUTPORT = 80
  SUTTERM = 2000
  BASENO = 1721501
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl029b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1723501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl030a
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 1243001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl030a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1243251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl030a
  SUTPORT = 80
  SUTTERM = 1500
  BASENO = 1247001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl030b
  SUTPORT = 80
  SUTTERM = 2250
  BASENO = 1248501
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl030b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 1250751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT

```



```

STARTSUT
  SUTHOST = cl074a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 572001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl074a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 572501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl074a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 576251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl074a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 580001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl074b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 583001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl074b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 583501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl074b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 587001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl074b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 590501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
  WAREHOUSE = 193600
  MEASUREMENT = 21600
  PAY-MIX = 4302
  ORD-MIX = 402
  DEL-MIX = 402
  STK-MIX = 402
  NEW-KEYING = 18010
  PAY-KEYING = 3010
  ORD-KEYING = 2010

```

```

DEL-KEYING = 2010
STK-KEYING = 2010
NEW-THINK = 12020
PAY-THINK = 12020
ORD-THINK = 10020
DEL-THINK = 5020
STK-THINK = 5020
CONST-CLAST = 111
CONST-CID = 1023
CONST-IID = 8191
THR-PER-PROC = 250
SYNC = 0
ENDVARIABLE
ENDGROUP

.....
rte09.conf
.....

#
# rte09.conf :configuration file for TPC-C
Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
  STARTRTE
    STARTSUT
      SUTHOST = cl065a
      SUTPORT = 80
      SUTTERM = 3750
      BASENO = 550001
      LOGPATH = /w00
      LOGLEVEL = 0
    ENDSUT
    STARTSUT
      SUTHOST = cl065a
      SUTPORT = 80
      SUTTERM = 3750
      BASENO = 553751
      LOGPATH = /w01
      LOGLEVEL = 0
    ENDSUT
    STARTSUT
      SUTHOST = cl065a
      SUTPORT = 80
      SUTTERM = 3500
      BASENO = 557501
      LOGPATH = /w02
      LOGLEVEL = 0
    ENDSUT
    STARTSUT
      SUTHOST = cl065b
      SUTPORT = 80
      SUTTERM = 250
      BASENO = 561001
      LOGPATH = /w02
      LOGLEVEL = 0
    ENDSUT
    STARTSUT
      SUTHOST = cl065b
      SUTPORT = 80
      SUTTERM = 3750
      BASENO = 561251
      LOGPATH = /w03
      LOGLEVEL = 0
    ENDSUT
    STARTSUT
      SUTHOST = cl065b
      SUTPORT = 80
      SUTTERM = 3750
      BASENO = 565001
      LOGPATH = /w04
      LOGLEVEL = 0

```

```

ENDSUT
STARTSUT
  SUTHOST = cl065b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 568751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl066a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 792001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl066a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 792501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl066a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 796251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl066a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 800001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl066b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 803001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl066b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 803501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl066b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 807001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl066b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 810501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE

```

```

WAREHOUSE = 193600
MEASUREMENT = 21600
PAY-MIX = 4302
ORD-MIX = 402
DEL-MIX = 402
STK-MIX = 402
NEW-KEYING = 18010
PAY-KEYING = 3010
ORD-KEYING = 2010
DEL-KEYING = 2010
STK-KEYING = 2010
NEW-THINK = 12020
PAY-THINK = 12020
ORD-THINK = 10020
DEL-THINK = 5020
STK-THINK = 5020
CONST-CLAST = 111
CONST-CID = 1023
CONST-IID = 8191
THR-PER-PROC = 250
SYNC = 0
ENDVARIABLE
ENDGROUP

.....
rte10.conf
.....

#
# rte10.conf :configuration file for TPC-C
Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl067a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1034001
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl067a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1037751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl067a
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1041501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl067b
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 1045001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl067b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1045251
  LOGPATH = /w03

```

```

LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl067b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1049001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl067b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 1052751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl068a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1298001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl068a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1298501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl068a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1302251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl068a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 1306001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl068b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1309001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl068b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1309501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl068b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1313001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT

```

```

SUTHOST = cl068b
SUTPORT = 80
SUTTERM = 3500
BASENO = 1316501
LOGPATH = /w11
LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
WAREHOUSE = 193600
MEASUREMENT = 21600
PAY-MIX = 4302
ORD-MIX = 402
DEL-MIX = 402
STK-MIX = 402
NEW-KEYING = 18010
PAY-KEYING = 3010
ORD-KEYING = 2010
DEL-KEYING = 2010
STK-KEYING = 2010
NEW-THINK = 12020
PAY-THINK = 12020
ORD-THINK = 10020
DEL-THINK = 5020
STK-THINK = 5020
CONST-CLAST = 111
CONST-CID = 1023
CONST-IID = 8191
THR-PER-PROC = 250
SYNC = 0
ENDVARIABLE
ENDGROUP

.....
rte11.conf
.....

#
# rte11.conf :configuration file for TPC-C
Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl034a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 22001
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl034a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 25751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl034a
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 29501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl034b
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 33001

```

```

LOGPATH = /w02
LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl034b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 33251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl034b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 37001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl034b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 40751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl035a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 264001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl035a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 264501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl035a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 268251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl035a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 272001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl035b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 275001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl035b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 275501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT

```

```

STARTSUT
  SUTHOST = cl035b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 279001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl035b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 282501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
  WAREHOUSE = 193600
  MEASUREMENT = 21600
  PAY-MIX = 4302
  ORD-MIX = 402
  DEL-MIX = 402
  STK-MIX = 402
  NEW-KEYING = 18010
  PAY-KEYING = 3010
  ORD-KEYING = 2010
  DEL-KEYING = 2010
  STK-KEYING = 2010
  NEW-THINK = 12020
  PAY-THINK = 12020
  ORD-THINK = 10020
  DEL-THINK = 5020
  STK-THINK = 5020
  CONST-CLAST = 111
  CONST-CID = 1023
  CONST-IID = 8191
  THR-PER-PROC = 250
  SYNC = 0
ENDVARIABLE
ENDGROUP

.....
rte12.conf
.....

#
# rte12.conf :configuration file for TPC-C
# Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
  STARTRTE
    STARTSUT
      SUTHOST = cl036a
      SUTPORT = 80
      SUTTERM = 3750
      BASENO = 506001
      LOGPATH = /w00
      LOGLEVEL = 0
    ENSUT
  STARTSUT
    SUTHOST = cl036a
    SUTPORT = 80
    SUTTERM = 3750
    BASENO = 509751
    LOGPATH = /w01
    LOGLEVEL = 0
  ENSUT
  STARTSUT
    SUTHOST = cl036a
    SUTPORT = 80
    SUTTERM = 3500

```

```

  BASENO = 513501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl036b
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 517001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl036b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 517251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl036b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 521001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl036b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 524751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl075a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 814001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl075a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 814501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl075a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 818251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl075a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 822001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl075b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 825001
  LOGPATH = /w08
  LOGLEVEL = 0

```

```

ENDSUT
STARTSUT
  SUTHOST = cl075b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 825501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl075b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 829001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl075b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 832501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
  WAREHOUSE = 193600
  MEASUREMENT = 21600
  PAY-MIX = 4302
  ORD-MIX = 402
  DEL-MIX = 402
  STK-MIX = 402
  NEW-KEYING = 18010
  PAY-KEYING = 3010
  ORD-KEYING = 2010
  DEL-KEYING = 2010
  STK-KEYING = 2010
  NEW-THINK = 12020
  PAY-THINK = 12020
  ORD-THINK = 10020
  DEL-THINK = 5020
  STK-THINK = 5020
  CONST-CLAST = 111
  CONST-CID = 1023
  CONST-IID = 8191
  THR-PER-PROC = 250
  SYNC = 0
ENDVARIABLE
ENDGROUP

.....
rte13.conf
.....

#
# rte13.conf :configuration file for TPC-C
# Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl037a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 748001
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl037a
  SUTPORT = 80

```

```

  SUTTERM = 3750
  BASENO = 751751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl037a
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 755501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl037b
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 759001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl037b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 759251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl037b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 763001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl037b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 766751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl038a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 990001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl038a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 990501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl038a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 994251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl038a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 998001
  LOGPATH = /w08

```

```

  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl038b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1001001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl038b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1001501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl038b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1005001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl038b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1008501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
  WAREHOUSE = 193600
  MEASUREMENT = 21600
  PAY-MIX = 4302
  ORD-MIX = 402
  DEL-MIX = 402
  STK-MIX = 402
  NEW-KEYING = 18010
  PAY-KEYING = 3010
  ORD-KEYING = 2010
  DEL-KEYING = 2010
  STK-KEYING = 2010
  NEW-THINK = 12020
  PAY-THINK = 12020
  ORD-THINK = 10020
  DEL-THINK = 5020
  STK-THINK = 5020
  CONST-CLAST = 111
  CONST-CID = 1023
  CONST-IID = 8191
  THR-PER-PROC = 250
  SYNC = 0
ENDVARIABLE
ENDGROUP

.....
rte14.conf
.....

#
# rte14.conf :configuration file for TPC-C
# Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl039a

```

```

SUTPORT = 80
SUTTERM = 3750
BASENO = 1826001
LOGPATH = /w00
LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl039a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1829751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl039a
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1833501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl039b
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 1837001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl039b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1837251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl039b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1841001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl039b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 1844751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl040a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1485001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl040a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1485501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl040a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1489251

```

```

LOGPATH = /w07
LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl040a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 1493001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl040b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1496001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl040b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1496501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl040b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1500001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl040b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1503501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
  WAREHOUSE = 193600
  MEASUREMENT = 21600
  PAY-MIX = 4302
  ORD-MIX = 402
  DEL-MIX = 402
  STK-MIX = 402
  NEW-KEYING = 18010
  PAY-KEYING = 3010
  ORD-KEYING = 2010
  DEL-KEYING = 2010
  STK-KEYING = 2010
  NEW-THINK = 12020
  PAY-THINK = 12020
  ORD-THINK = 10020
  DEL-THINK = 5020
  STK-THINK = 5020
  CONST-CLAST = 111
  CONST-CID = 1023
  CONST-IID = 8191
  THR-PER-PROC = 250
  SYNC = 0
ENDVARIABLE
ENDGROUP

.....
rte15.conf
.....

#

```

```

# rte15.conf :configuration file for TPC-C
Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
  STARTRTE
    STARTSUT
      SUTHOST = cl041a
      SUTPORT = 80
      SUTTERM = 3750
      BASENO = 1727001
      LOGPATH = /w00
      LOGLEVEL = 0
    ENSUT
    STARTSUT
      SUTHOST = cl041a
      SUTPORT = 80
      SUTTERM = 3750
      BASENO = 1730751
      LOGPATH = /w01
      LOGLEVEL = 0
    ENSUT
    STARTSUT
      SUTHOST = cl041a
      SUTPORT = 80
      SUTTERM = 3500
      BASENO = 1734501
      LOGPATH = /w02
      LOGLEVEL = 0
    ENSUT
    STARTSUT
      SUTHOST = cl041b
      SUTPORT = 80
      SUTTERM = 250
      BASENO = 1738001
      LOGPATH = /w02
      LOGLEVEL = 0
    ENSUT
    STARTSUT
      SUTHOST = cl041b
      SUTPORT = 80
      SUTTERM = 3750
      BASENO = 1738251
      LOGPATH = /w03
      LOGLEVEL = 0
    ENSUT
    STARTSUT
      SUTHOST = cl041b
      SUTPORT = 80
      SUTTERM = 80
      SUTTERM = 3750
      BASENO = 1742001
      LOGPATH = /w04
      LOGLEVEL = 0
    ENSUT
    STARTSUT
      SUTHOST = cl041b
      SUTPORT = 80
      SUTTERM = 3250
      BASENO = 1745751
      LOGPATH = /w05
      LOGLEVEL = 0
    ENSUT
    STARTSUT
      SUTHOST = cl042a
      SUTPORT = 80
      SUTTERM = 500
      BASENO = 44001
      LOGPATH = /w05
      LOGLEVEL = 0
    ENSUT
    STARTSUT
      SUTHOST = cl042a
      SUTPORT = 80
      SUTTERM = 3750

```

```

BASENO = 44501
LOGPATH = /w06
LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl042a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 48251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl042a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 52001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl042b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 55001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl042b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 55501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl042b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 59001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl042b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 62501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
WAREHOUSE = 193600
MEASUREMENT = 21600
PAY-MIX = 4302
ORD-MIX = 402
DEL-MIX = 402
STK-MIX = 402
NEW-KEYING = 18010
PAY-KEYING = 3010
ORD-KEYING = 2010
DEL-KEYING = 2010
STK-KEYING = 2010
NEW-THINK = 12020
PAY-THINK = 12020
ORD-THINK = 10020
DEL-THINK = 5020
STK-THINK = 5020
CONST-CLAST = 111
CONST-CID = 1023
CONST-IID = 8191
THR-PER-PROC = 250
SYNC = 0

```

```

ENDVARIABLE
ENDGROUP
.....
rte16.conf
.....
#
# rte16.conf :configuration file for TPC-C
Rev3.0
# Author : mkdef -Auto Configurator for R3-
#
STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl043a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1518001
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl043a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1521751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl043a
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1525501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl043a
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 1529001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl043b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1529251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl043b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1533001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl043b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 1536751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl044a
  SUTPORT = 80

```

```

SUTTERM = 500
BASENO = 528001
LOGPATH = /w05
LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl044a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 528501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl044a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 532251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl044a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 536001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl044b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 539001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl044b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 539501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl044b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 543001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl044b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 546501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
WAREHOUSE = 193600
MEASUREMENT = 21600
PAY-MIX = 4302
ORD-MIX = 402
DEL-MIX = 402
STK-MIX = 402
NEW-KEYING = 18010
PAY-KEYING = 3010
ORD-KEYING = 2010
DEL-KEYING = 2010
STK-KEYING = 2010
NEW-THINK = 12020

```

```

PAY-THINK = 12020
ORD-THINK = 10020
DEL-THINK = 5020
STK-THINK = 5020
CONST-CLAST = 111
CONST-CID = 1023
CONST-IID = 8191
THR-PER-PROC = 250
SYNC = 0
ENDVARIABLE
ENDGROUP

.....
rte17.conf
.....

#
# rte17.conf :configuration file for TPC-C
Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl045a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 770001
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl045a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 773751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl045a
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 777501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl045b
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 781001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl045b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 781251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl045b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 785001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl045b

```

```

SUTPORT = 80
SUTTERM = 3250
BASENO = 788751
LOGPATH = /w05
LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl046a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1012001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl046a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1012501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl046a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1016251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl046a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 1020001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl046b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1023001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl046b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1023501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl046b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1027001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl046b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1030501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
WAREHOUSE = 193600
MEASUREMENT = 21600
PAY-MIX = 4302

```

```

ORD-MIX = 402
DEL-MIX = 402
STK-MIX = 402
NEW-KEYING = 18010
PAY-KEYING = 3010
ORD-KEYING = 2010
DEL-KEYING = 2010
STK-KEYING = 2010
NEW-THINK = 12020
PAY-THINK = 12020
ORD-THINK = 10020
DEL-THINK = 5020
STK-THINK = 5020
CONST-CLAST = 111
CONST-CID = 1023
CONST-IID = 8191
THR-PER-PROC = 250
SYNC = 0
ENDVARIABLE
ENDGROUP

.....
rte18.conf
.....

#
# rte18.conf :configuration file for TPC-C
Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl047a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1276001
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl047a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1279751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl047a
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1283501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl047b
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 1287001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl047b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1287251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT

```

```

SUTHOST = cl047b
SUTPORT = 80
SUTTERM = 3750
BASENO = 1291001
LOGPATH = /w04
LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl047b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 1294751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl076a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1056001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl076a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1056501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl076a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1060251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl076a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 1064001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl076b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1067001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl076b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1067501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl076b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1071001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl076b
  SUTPORT = 80
  SUTTERM = 3500

```

```

BASENO = 1074501
LOGPATH = /w11
LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
  WAREHOUSE = 193600
  MEASUREMENT = 21600
  PAY-MIX = 4302
  ORD-MIX = 402
  DEL-MIX = 402
  STK-MIX = 402
  NEW-KEYING = 18010
  PAY-KEYING = 3010
  ORD-KEYING = 2010
  DEL-KEYING = 2010
  STK-KEYING = 2010
  NEW-THINK = 12020
  PAY-THINK = 12020
  ORD-THINK = 10020
  DEL-THINK = 5020
  STK-THINK = 5020
  CONST-CLAST = 111
  CONST-CID = 1023
  CONST-IID = 8191
  THR-PER-PROC = 250
  SYNC = 0
ENDVARIABLE
ENDGROUP

.....
rte19.conf
.....

#
# rte19.conf :configuration file for TPC-C
# Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
  STARTRTE
    STARTSUT
      SUTHOST = cl051a
      SUTPORT = 80
      SUTTERM = 3750
      BASENO = 1507001
      LOGPATH = /w00
      LOGLEVEL = 0
    ENDSUT
    STARTSUT
      SUTHOST = cl051a
      SUTPORT = 80
      SUTTERM = 1750
      BASENO = 1510751
      LOGPATH = /w01
      LOGLEVEL = 0
    ENDSUT
    STARTSUT
      SUTHOST = cl051b
      SUTPORT = 80
      SUTTERM = 2000
      BASENO = 1512501
      LOGPATH = /w01
      LOGLEVEL = 0
    ENDSUT
    STARTSUT
      SUTHOST = cl051b
      SUTPORT = 80
      SUTTERM = 3500
      BASENO = 1514501
      LOGPATH = /w02
      LOGLEVEL = 0
    ENDSUT

```

```

STARTSUT
  SUTHOST = cl052a
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 1749001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl052a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1749251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl052a
  SUTPORT = 80
  SUTTERM = 1500
  BASENO = 1753001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl052b
  SUTPORT = 80
  SUTTERM = 2250
  BASENO = 1754501
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl052b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 1756751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl077a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1320001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl077a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1320501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl077a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1324251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl077a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 1328001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl077b
  SUTPORT = 80

```



```

SUTTERM = 500
BASENO = 1331001
LOGPATH = /w08
LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl077b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1331501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl077b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1335001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl077b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1338501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
  WAREHOUSE = 193600
  MEASUREMENT = 21600
  PAY-MIX = 4302
  ORD-MIX = 402
  DEL-MIX = 402
  STK-MIX = 402
  NEW-KEYING = 18010
  PAY-KEYING = 3010
  ORD-KEYING = 2010
  DEL-KEYING = 2010
  STK-KEYING = 2010
  NEW-THINK = 12020
  PAY-THINK = 12020
  ORD-THINK = 10020
  DEL-THINK = 5020
  STK-THINK = 5020
  CONST-CLAST = 111
  CONST-CID = 1023
  CONST-IID = 8191
  THR-PER-PROC = 250
  SYNC = 0
ENDVARIABLE
ENDGROUP

.....:
rte20.conf
.....:

#
# rte20.conf :configuration file for TPC-C
# Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl078a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1562001
  LOGPATH = /w00
  LOGLEVEL = 0

```

```

ENDSUT
STARTSUT
  SUTHOST = cl078a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1565751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl078a
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1569501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl078b
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 1573001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl078b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1573251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl078b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1577001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl078b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 1580751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl079a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1804001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl079a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1804501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl079a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1808251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl079a

```

```

SUTPORT = 80
SUTTERM = 3000
BASENO = 1812001
LOGPATH = /w08
LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl079b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1815001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl079b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1815501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl079b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1819001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl079b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1822501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
  WAREHOUSE = 193600
  MEASUREMENT = 21600
  PAY-MIX = 4302
  ORD-MIX = 402
  DEL-MIX = 402
  STK-MIX = 402
  NEW-KEYING = 18010
  PAY-KEYING = 3010
  ORD-KEYING = 2010
  DEL-KEYING = 2010
  STK-KEYING = 2010
  NEW-THINK = 12020
  PAY-THINK = 12020
  ORD-THINK = 10020
  DEL-THINK = 5020
  STK-THINK = 5020
  CONST-CLAST = 111
  CONST-CID = 1023
  CONST-IID = 8191
  THR-PER-PROC = 250
  SYNC = 0
ENDVARIABLE
ENDGROUP

.....:
rte21.conf
.....:

#
# rte21.conf :configuration file for TPC-C
# Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

```

```

STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl105a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1364001
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl105a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1367751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl105a
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1371501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl105b
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 1375001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl105b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1375251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl105b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1379001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl105b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 1382751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl106a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1606001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl106a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1606501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT

```

```

  SUTHOST = cl106a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1610251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl106a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 1614001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl106b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1617001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl106b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1617501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl106b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1621001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl106b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1624501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
  WAREHOUSE = 193600
  MEASUREMENT = 21600
  PAY-MIX = 4302
  ORD-MIX = 402
  DEL-MIX = 402
  STK-MIX = 402
  NEW-KEYING = 18010
  PAY-KEYING = 3010
  ORD-KEYING = 2010
  DEL-KEYING = 2010
  STK-KEYING = 2010
  NEW-THINK = 12020
  PAY-THINK = 12020
  ORD-THINK = 10020
  DEL-THINK = 5020
  STK-THINK = 5020
  CONST-CLAST = 111
  CONST-CID = 1023
  CONST-IID = 8191
  THR-PER-PROC = 250
  SYNC = 0
ENDVARIABLE
ENDGROUP

```

```

rte23.conf
#
# rte23.conf :configuration file for TPC-C
# Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl080a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 110001
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl080a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 113751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl080a
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 117501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl080b
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 121001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl080b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 121251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl080b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 125001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl080b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 128751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl108a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1848001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT

```

```

STARTSUT
  SUTHOST = cl108a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1848501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl108a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1852251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl108a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 1856001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl108b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1859001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl108b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1859501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl108b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1863001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl108b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1866501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
  WAREHOUSE = 193600
  MEASUREMENT = 21600
  PAY-MIX = 4302
  ORD-MIX = 402
  DEL-MIX = 402
  STK-MIX = 402
  NEW-KEYING = 18010
  PAY-KEYING = 3010
  ORD-KEYING = 2010
  DEL-KEYING = 2010
  STK-KEYING = 2010
  NEW-THINK = 12020
  PAY-THINK = 12020
  ORD-THINK = 10020
  DEL-THINK = 5020
  STK-THINK = 5020
  CONST-CLAST = 111

CONST-CID = 1023
CONST-IID = 8191
THR-PER-PROC = 250
SYNC = 0
ENDVARIABLE
ENDGROUP

.....
rte24.conf
.....

#
# rte24.conf :configuration file for TPC-C
Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
STARTRTE
  STARTSUT
    SUTHOST = cl109a
    SUTPORT = 80
    SUTTERM = 3750
    BASENO = 154001
    LOGPATH = /w00
    LOGLEVEL = 0
  ENDSUT
  STARTSUT
    SUTHOST = cl109a
    SUTPORT = 80
    SUTTERM = 3750
    BASENO = 157751
    LOGPATH = /w01
    LOGLEVEL = 0
  ENDSUT
  STARTSUT
    SUTHOST = cl109a
    SUTPORT = 80
    SUTTERM = 3500
    BASENO = 161501
    LOGPATH = /w02
    LOGLEVEL = 0
  ENDSUT
  STARTSUT
    SUTHOST = cl109b
    SUTPORT = 80
    SUTTERM = 250
    BASENO = 165001
    LOGPATH = /w02
    LOGLEVEL = 0
  ENDSUT
  STARTSUT
    SUTHOST = cl109b
    SUTPORT = 80
    SUTTERM = 3750
    BASENO = 165251
    LOGPATH = /w03
    LOGLEVEL = 0
  ENDSUT
  STARTSUT
    SUTHOST = cl109b
    SUTPORT = 80
    SUTTERM = 3750
    BASENO = 169001
    LOGPATH = /w04
    LOGLEVEL = 0
  ENDSUT
  STARTSUT
    SUTHOST = cl109b
    SUTPORT = 80
    SUTTERM = 3250
    BASENO = 172751
    LOGPATH = /w05
    LOGLEVEL = 0
  ENDSUT

ENDSUT
STARTSUT
  SUTHOST = cl110a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 396001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl110a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 396501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl110a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 400251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl110a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 404001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl110b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 407001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl110b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 407501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl110b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 411001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl110b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 414501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
  WAREHOUSE = 193600
  MEASUREMENT = 21600
  PAY-MIX = 4302
  ORD-MIX = 402
  DEL-MIX = 402
  STK-MIX = 402
  NEW-KEYING = 18010
  PAY-KEYING = 3010

```

```

ORD-KEYING = 2010
DEL-KEYING = 2010
STK-KEYING = 2010
NEW-THINK = 12020
PAY-THINK = 12020
ORD-THINK = 10020
DEL-THINK = 5020
STK-THINK = 5020
CONST-CLAST = 111
CONST-CID = 1023
CONST-IID = 8191
THR-PER-PROC = 250
SYNC = 0
ENDVARIABLE
ENDGROUP

```

```

.....
rte26.conf
.....

```

```

#
# rte26.conf :configuration file for TPC-C
Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

```

```

STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl061a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 286001
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl061a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 289751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl061a
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 293501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl061b
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 297001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl061b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 297251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl061b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 301001
  LOGPATH = /w04

```

```

LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl061b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 304751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl062a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1760001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl062a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1760501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl062a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1764251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl062a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 1768001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl062b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1771001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl062b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1771501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl062b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1775001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl062b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1778501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE

```

```

STARTVARIABLE
WAREHOUSE = 193600
MEASUREMENT = 21600
PAY-MIX = 4302
ORD-MIX = 402
DEL-MIX = 402
STK-MIX = 402
NEW-KEYING = 18010
PAY-KEYING = 3010
ORD-KEYING = 2010
DEL-KEYING = 2010
STK-KEYING = 2010
NEW-THINK = 12020
PAY-THINK = 12020
ORD-THINK = 10020
DEL-THINK = 5020
STK-THINK = 5020
CONST-CLAST = 111
CONST-CID = 1023
CONST-IID = 8191
THR-PER-PROC = 250
SYNC = 0
ENDVARIABLE
ENDGROUP

```

```

.....
rte27.conf
.....

```

```

#
# rte27.conf :configuration file for TPC-C
Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

```

```

STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl063a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 66001
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl063a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 69751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl063a
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 73501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl063b
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 77001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl063b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 77251

```

```

LOGPATH = /w03
LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl063b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 81001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl063b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 84751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl064a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 308001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl064a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 308501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl064a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 312251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl064a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 316001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl064b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 319001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl064b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 319501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl064b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 323001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT

```

```

STARTSUT
  SUTHOST = cl064b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 326501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
  WAREHOUSE = 193600
  MEASUREMENT = 21600
  PAY-MIX = 4302
  ORD-MIX = 402
  DEL-MIX = 402
  STK-MIX = 402
  NEW-KEYING = 18010
  PAY-KEYING = 3010
  ORD-KEYING = 2010
  DEL-KEYING = 2010
  STK-KEYING = 2010
  NEW-THINK = 12020
  PAY-THINK = 12020
  ORD-THINK = 10020
  DEL-THINK = 5020
  STK-THINK = 5020
  CONST-CLAST = 111
  CONST-CID = 1023
  CONST-IID = 8191
  THR-PER-PROC = 250
  SYNC = 0
ENDVARIABLE
ENDGROUP

```

```

.....:
rte29.conf
.....:

#
# rte29.conf :configuration file for TPC-C
Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

```

```

STARTGROUP
  STARTRTE
    STARTSUT
      SUTHOST = cl081a
      SUTPORT = 80
      SUTTERM = 3750
      BASENO = 352001
      LOGPATH = /w00
      LOGLEVEL = 0
    ENSUT
    STARTSUT
      SUTHOST = cl081a
      SUTPORT = 80
      SUTTERM = 3750
      BASENO = 355751
      LOGPATH = /w01
      LOGLEVEL = 0
    ENSUT
    STARTSUT
      SUTHOST = cl081a
      SUTPORT = 80
      SUTTERM = 3500
      BASENO = 359501
      LOGPATH = /w02
      LOGLEVEL = 0
    ENSUT
    STARTSUT
      SUTHOST = cl081b
      SUTPORT = 80
      SUTTERM = 250

```

```

  BASENO = 363001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl081b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 363251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl081b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 367001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl081b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 370751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl082a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 594001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl082a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 594501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl082a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 598251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl082a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 602001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl082b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 605001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl082b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 605501
  LOGPATH = /w09
  LOGLEVEL = 0

```

```

ENDSUT
STARTSUT
  SUTHOST = cl082b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 609001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl082b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 612501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
  WAREHOUSE = 193600
  MEASUREMENT = 21600
  PAY-MIX = 4302
  ORD-MIX = 402
  DEL-MIX = 402
  STK-MIX = 402
  NEW-KEYING = 18010
  PAY-KEYING = 3010
  ORD-KEYING = 2010
  DEL-KEYING = 2010
  STK-KEYING = 2010
  NEW-THINK = 12020
  PAY-THINK = 12020
  ORD-THINK = 10020
  DEL-THINK = 5020
  STK-THINK = 5020
  CONST-CLAST = 111
  CONST-CID = 1023
  CONST-IID = 8191
  THR-PER-PROC = 250
  SYNC = 0
ENDVARIABLE
ENDGROUP

.....
rte30.conf
.....

#
# rte30.conf :configuration file for TPC-C
Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl083a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 836001
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl083a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 839751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl083a
  SUTPORT = 80

```

```

SUTTERM = 3500
BASENO = 843501
LOGPATH = /w02
LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl083b
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 847001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl083b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 847251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl083b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 851001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl083b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 854751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl084a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1078001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl084a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1078501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl084a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1082251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl084a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 1086001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl084b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1089001
  LOGPATH = /w08

```

```

LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl084b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1089501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl084b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1093001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl084b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1096501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
  WAREHOUSE = 193600
  MEASUREMENT = 21600
  PAY-MIX = 4302
  ORD-MIX = 402
  DEL-MIX = 402
  STK-MIX = 402
  NEW-KEYING = 18010
  PAY-KEYING = 3010
  ORD-KEYING = 2010
  DEL-KEYING = 2010
  STK-KEYING = 2010
  NEW-THINK = 12020
  PAY-THINK = 12020
  ORD-THINK = 10020
  DEL-THINK = 5020
  STK-THINK = 5020
  CONST-CLAST = 111
  CONST-CID = 1023
  CONST-IID = 8191
  THR-PER-PROC = 250
  SYNC = 0
ENDVARIABLE
ENDGROUP

.....
rte31.conf
.....

#
# rte31.conf :configuration file for TPC-C
Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl085a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1342001
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl085a

```

```

SUTPORT = 80
SUTTERM = 3750
BASENO = 1345751
LOGPATH = /w01
LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl085a
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1349501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl085b
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 1353001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl085b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1353251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl085b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1357001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl085b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 1360751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl086a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1584001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl086a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1584501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl086a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1588251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl086a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 1592001

```

```

LOGPATH = /w08
LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl086b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1595001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl086b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1595501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl086b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1599001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl086b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1602501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
  WAREHOUSE = 193600
  MEASUREMENT = 21600
  PAY-MIX = 4302
  ORD-MIX = 402
  DEL-MIX = 402
  STK-MIX = 402
  NEW-KEYING = 18010
  PAY-KEYING = 3010
  ORD-KEYING = 2010
  DEL-KEYING = 2010
  STK-KEYING = 2010
  NEW-THINK = 12020
  PAY-THINK = 12020
  ORD-THINK = 10020
  DEL-THINK = 5020
  STK-THINK = 5020
  CONST-CLAST = 111
  CONST-CID = 1023
  CONST-IID = 8191
  THR-PER-PROC = 250
  SYNC = 0
ENDVARIABLE
ENDGROUP
.....
rte32.conf
.....
#
# rte32.conf :configuration file for TPC-C
# Rev3.0
# Author : mkdef -Auto Configurator for R3-
#
STARTGROUP
STARTRTE
STARTSUT

```

```

SUTHOST = cl087a
SUTPORT = 80
SUTTERM = 3750
BASENO = 1254001
LOGPATH = /w00
LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl087a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1257751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl087a
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1261501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl087b
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 1265001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl087b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1265251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl087b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1269001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl087b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 1272751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl088a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 132001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl088a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 132501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl088a
  SUTPORT = 80
  SUTTERM = 3750

```

```

BASENO = 136251
LOGPATH = /w07
LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl088a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 140001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl088b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 143001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl088b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 143501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl088b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 147001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl088b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 150501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
  WAREHOUSE = 193600
  MEASUREMENT = 21600
  PAY-MIX = 4302
  ORD-MIX = 402
  DEL-MIX = 402
  STK-MIX = 402
  NEW-KEYING = 18010
  PAY-KEYING = 3010
  ORD-KEYING = 2010
  DEL-KEYING = 2010
  STK-KEYING = 2010
  NEW-THINK = 12020
  PAY-THINK = 12020
  ORD-THINK = 10020
  DEL-THINK = 5020
  STK-THINK = 5020
  CONST-CLAST = 111
  CONST-CID = 1023
  CONST-IID = 8191
  THR-PER-PROC = 250
  SYNC = 0
ENDVARIABLE
ENDGROUP

```

```

.....
rte34.conf
.....

```

```

#
# rte34.conf :configuration file for TPC-C
# Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
  STARTRTE
    STARTSUT
      SUTHOST = cl111a
      SUTPORT = 80
      SUTTERM = 3750
      BASENO = 638001
      LOGPATH = /w00
      LOGLEVEL = 0
    ENDSUT
    STARTSUT
      SUTHOST = cl111a
      SUTPORT = 80
      SUTTERM = 3750
      BASENO = 641751
      LOGPATH = /w01
      LOGLEVEL = 0
    ENDSUT
    STARTSUT
      SUTHOST = cl111a
      SUTPORT = 80
      SUTTERM = 3500
      BASENO = 645501
      LOGPATH = /w02
      LOGLEVEL = 0
    ENDSUT
    STARTSUT
      SUTHOST = cl111b
      SUTPORT = 80
      SUTTERM = 250
      BASENO = 649001
      LOGPATH = /w02
      LOGLEVEL = 0
    ENDSUT
    STARTSUT
      SUTHOST = cl111b
      SUTPORT = 80
      SUTTERM = 3750
      BASENO = 649251
      LOGPATH = /w03
      LOGLEVEL = 0
    ENDSUT
    STARTSUT
      SUTHOST = cl111b
      SUTPORT = 80
      SUTTERM = 3750
      BASENO = 653001
      LOGPATH = /w04
      LOGLEVEL = 0
    ENDSUT
    STARTSUT
      SUTHOST = cl111b
      SUTPORT = 80
      SUTTERM = 3250
      BASENO = 656751
      LOGPATH = /w05
      LOGLEVEL = 0
    ENDSUT
    STARTSUT
      SUTHOST = cl112a
      SUTPORT = 80
      SUTTERM = 500
      BASENO = 880001
      LOGPATH = /w05
      LOGLEVEL = 0
    ENDSUT
    STARTSUT
      SUTHOST = cl112a
      SUTPORT = 80

```

```

      SUTTERM = 3750
      BASENO = 880501
      LOGPATH = /w06
      LOGLEVEL = 0
    ENDSUT
    STARTSUT
      SUTHOST = cl112a
      SUTPORT = 80
      SUTTERM = 3750
      BASENO = 884251
      LOGPATH = /w07
      LOGLEVEL = 0
    ENDSUT
    STARTSUT
      SUTHOST = cl112a
      SUTPORT = 80
      SUTTERM = 3000
      BASENO = 888001
      LOGPATH = /w08
      LOGLEVEL = 0
    ENDSUT
    STARTSUT
      SUTHOST = cl112b
      SUTPORT = 80
      SUTTERM = 500
      BASENO = 891001
      LOGPATH = /w08
      LOGLEVEL = 0
    ENDSUT
    STARTSUT
      SUTHOST = cl112b
      SUTPORT = 80
      SUTTERM = 3500
      BASENO = 891501
      LOGPATH = /w09
      LOGLEVEL = 0
    ENDSUT
    STARTSUT
      SUTHOST = cl112b
      SUTPORT = 80
      SUTTERM = 3500
      BASENO = 895001
      LOGPATH = /w10
      LOGLEVEL = 0
    ENDSUT
    STARTSUT
      SUTHOST = cl112b
      SUTPORT = 80
      SUTTERM = 3500
      BASENO = 898501
      LOGPATH = /w11
      LOGLEVEL = 0
    ENDSUT
    STARTVARIABLE
      WAREHOUSE = 193600
      MEASUREMENT = 21600
      PAY-MIX = 4302
      ORD-MIX = 402
      DEL-MIX = 402
      STK-MIX = 402
      NEW-KEYING = 18010
      PAY-KEYING = 3010
      ORD-KEYING = 2010
      DEL-KEYING = 2010
      STK-KEYING = 2010
      NEW-THINK = 12020
      PAY-THINK = 12020
      ORD-THINK = 10020
      DEL-THINK = 5020
      STK-THINK = 5020
      CONST-CLAST = 111
      CONST-CID = 1023
      CONST-IID = 8191
      THR-PER-PROC = 250

```



```

SYNC = 0
ENDVARIABLE
ENDGROUP

.....
rte35.conf
.....

#
# rte35.conf :configuration file for TPC-C
Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl113a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1122001
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl113a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1125751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl113a
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1129501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl113b
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 1133001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl113b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1133251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl113b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1137001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl113b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 1140751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl114a

```

```

SUTPORT = 80
SUTTERM = 500
BASENO = 1386001
LOGPATH = /w05
LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl114a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1386501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl114a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1390251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl114a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 1394001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl114b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1397001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl114b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1397501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl114b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1401001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl114b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1404501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
  WAREHOUSE = 193600
  MEASUREMENT = 21600
  PAY-MIX = 4302
  ORD-MIX = 402
  DEL-MIX = 402
  STK-MIX = 402
  NEW-KEYING = 18010
  PAY-KEYING = 3010
  ORD-KEYING = 2010
  DEL-KEYING = 2010
  STK-KEYING = 2010

```

```

NEW-THINK = 12020
PAY-THINK = 12020
ORD-THINK = 10020
DEL-THINK = 5020
STK-THINK = 5020
CONST-CLAST = 111
CONST-CID = 1023
CONST-IID = 8191
THR-PER-PROC = 250
SYNC = 0
ENDVARIABLE
ENDGROUP

.....
rte36.conf
.....

#
# rte36.conf :configuration file for TPC-C
Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl115a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1628001
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl115a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1631751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl115a
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1635501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl115b
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 1639001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl115b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1639251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl115b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1643001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT

```

```

SUTHOST = cl115b
SUTPORT = 80
SUTTERM = 3250
BASENO = 1646751
LOGPATH = /w05
LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl116a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1870001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl116a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1870501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl116a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1874251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl116a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 1878001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl116b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1881001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl116b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1881501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl116b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1885001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl116b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1888501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
WAREHOUSE = 193600
MEASUREMENT = 21600

```

```

PAY-MIX = 4302
ORD-MIX = 402
DEL-MIX = 402
STK-MIX = 402
NEW-KEYING = 18010
PAY-KEYING = 3010
ORD-KEYING = 2010
DEL-KEYING = 2010
STK-KEYING = 2010
NEW-THINK = 12020
PAY-THINK = 12020
ORD-THINK = 10020
DEL-THINK = 5020
STK-THINK = 5020
CONST-CLAST = 111
CONST-CID = 1023
CONST-IID = 8191
THR-PER-PROC = 250
SYNC = 0
ENDVARIABLE
ENDGROUP

.....
rte37.conf
.....

#
# rte37.conf :configuration file for TPC-C
Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
STARTRTE
  STARTSUT
    SUTHOST = cl117a
    SUTPORT = 80
    SUTTERM = 3750
    BASENO = 176001
    LOGPATH = /w00
    LOGLEVEL = 0
  ENDSUT
  STARTSUT
    SUTHOST = cl117a
    SUTPORT = 80
    SUTTERM = 3750
    BASENO = 179751
    LOGPATH = /w01
    LOGLEVEL = 0
  ENDSUT
  STARTSUT
    SUTHOST = cl117a
    SUTPORT = 80
    SUTTERM = 3500
    BASENO = 183501
    LOGPATH = /w02
    LOGLEVEL = 0
  ENDSUT
  STARTSUT
    SUTHOST = cl117b
    SUTPORT = 80
    SUTTERM = 250
    BASENO = 187001
    LOGPATH = /w02
    LOGLEVEL = 0
  ENDSUT
  STARTSUT
    SUTHOST = cl117b
    SUTPORT = 80
    SUTTERM = 3750
    BASENO = 187251
    LOGPATH = /w03
    LOGLEVEL = 0
  ENDSUT

```

```

STARTSUT
  SUTHOST = cl117b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 191001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl117b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 194751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl118a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 418001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl118a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 418501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl118a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 422251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl118a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 426001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl118b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 429001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl118b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 429501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl118b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 433001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl118b
  SUTPORT = 80

```

```

SUTTERM = 3500
BASENO = 436501
LOGPATH = /w11
LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
WAREHOUSE = 193600
MEASUREMENT = 21600
PAY-MIX = 4302
ORD-MIX = 402
DEL-MIX = 402
STK-MIX = 402
NEW-KEYING = 18010
PAY-KEYING = 3010
ORD-KEYING = 2010
DEL-KEYING = 2010
STK-KEYING = 2010
NEW-THINK = 12020
PAY-THINK = 12020
ORD-THINK = 10020
DEL-THINK = 5020
STK-THINK = 5020
CONST-CLAST = 111
CONST-CID = 1023
CONST-IID = 8191
THR-PER-PROC = 250
SYNC = 0
ENDVARIABLE
ENDGROUP

.....
rte38.conf
.....

#
# rte38.conf :configuration file for TPC-C
Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl091a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 858001
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl091a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 861751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl091a
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 865501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl091b
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 869001
  LOGPATH = /w02
  LOGLEVEL = 0

```

```

ENDSUT
STARTSUT
  SUTHOST = cl091b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 869251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl091b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 873001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl091b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 876751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl119a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 660001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl119a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 660501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl119a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 664251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl119a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 668001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl119b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 671001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl119b
  SUTPORT = 80
  SUTTERM = 80
  BASENO = 671501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl119b

```

```

SUTPORT = 80
SUTTERM = 3500
BASENO = 675001
LOGPATH = /w10
LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl119b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 678501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
WAREHOUSE = 193600
MEASUREMENT = 21600
PAY-MIX = 4302
ORD-MIX = 402
DEL-MIX = 402
STK-MIX = 402
NEW-KEYING = 18010
PAY-KEYING = 3010
ORD-KEYING = 2010
DEL-KEYING = 2010
STK-KEYING = 2010
NEW-THINK = 12020
PAY-THINK = 12020
ORD-THINK = 10020
DEL-THINK = 5020
STK-THINK = 5020
CONST-CLAST = 111
CONST-CID = 1023
CONST-IID = 8191
THR-PER-PROC = 250
SYNC = 0
ENDVARIABLE
ENDGROUP

.....
rte39.conf
.....

#
# rte39.conf :configuration file for TPC-C
Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl121a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 902001
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl121a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 905751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl121a
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 909501
  LOGPATH = /w02

```

```

LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl121b
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 913001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl121b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 913251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl121b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 917001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl121b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 920751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl122a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1144001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl122a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1144501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl122a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1148251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl122a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 1152001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl122b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1155001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT

```

```

SUTHOST = cl122b
SUTPORT = 80
SUTTERM = 3500
BASENO = 1155501
LOGPATH = /w09
LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl122b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1159001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl122b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1162501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
  WAREHOUSE = 193600
  MEASUREMENT = 21600
  PAY-MIX = 4302
  ORD-MIX = 402
  DEL-MIX = 402
  STK-MIX = 402
  NEW-KEYING = 18010
  PAY-KEYING = 3010
  ORD-KEYING = 2010
  DEL-KEYING = 2010
  STK-KEYING = 2010
  NEW-THINK = 12020
  PAY-THINK = 12020
  ORD-THINK = 10020
  DEL-THINK = 5020
  STK-THINK = 5020
  CONST-CLAST = 111
  CONST-CID = 1023
  CONST-IID = 8191
  THR-PER-PROC = 250
  SYNC = 0
ENDVARIABLE
ENDGROUP

.....
rte40.conf
.....

#
# rte40.conf :configuration file for TPC-C
Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl092a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1100001
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl092a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1103751

```

```

LOGPATH = /w01
LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl092a
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1107501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl092b
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 1111001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl092b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1111251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl092b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1115001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl092b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 1118751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl124a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1408001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl124a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1408501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl124a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1412251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl124a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 1416001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT

```

```

STARTSUT
  SUTHOST = cl124b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1419001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl124b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1419501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl124b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1423001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl124b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1426501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
  WAREHOUSE = 193600
  MEASUREMENT = 21600
  PAY-MIX = 4302
  ORD-MIX = 402
  DEL-MIX = 402
  STK-MIX = 402
  NEW-KEYING = 18010
  PAY-KEYING = 3010
  ORD-KEYING = 2010
  DEL-KEYING = 2010
  STK-KEYING = 2010
  NEW-THINK = 12020
  PAY-THINK = 12020
  ORD-THINK = 10020
  DEL-THINK = 5020
  STK-THINK = 5020
  CONST-CLAST = 111
  CONST-CID = 1023
  CONST-IID = 8191
  THR-PER-PROC = 250
  SYNC = 0
ENDVARIABLE
ENDGROUP

.....:
rte41.conf
.....:

#
# rte41.conf :configuration file for TPC-C
Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl125a
  SUTPORT = 80
  SUTTERM = 3750

```

```

  BASENO = 1650001
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl125a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1653751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl125a
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1657501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl125b
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 1661001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl125b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1661251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl125b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1665001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl125b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 1668751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl126a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1892001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl126a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1892501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl126a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1896251
  LOGPATH = /w07
  LOGLEVEL = 0

```

```

ENDSUT
STARTSUT
  SUTHOST = cl126a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 1900001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl126b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1903001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl126b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1903501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl126b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1907001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl126b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1910501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
  WAREHOUSE = 193600
  MEASUREMENT = 21600
  PAY-MIX = 4302
  ORD-MIX = 402
  DEL-MIX = 402
  STK-MIX = 402
  NEW-KEYING = 18010
  PAY-KEYING = 3010
  ORD-KEYING = 2010
  DEL-KEYING = 2010
  STK-KEYING = 2010
  NEW-THINK = 12020
  PAY-THINK = 12020
  ORD-THINK = 10020
  DEL-THINK = 5020
  STK-THINK = 5020
  CONST-CLAST = 111
  CONST-CID = 1023
  CONST-IID = 8191
  THR-PER-PROC = 250
  SYNC = 0
ENDVARIABLE
ENDGROUP

.....:
rte42.conf
.....:

#
# rte42.conf :configuration file for TPC-C
Rev3.0

```

```
# Author : mkdef -Auto Configurator for R3-
#
```

```
STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl127a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 198001
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl127a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 201751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl127a
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 205501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl127b
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 209001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl127b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 209251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl127b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 213001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl127b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 216751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl128a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 440001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl128a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 440501
  LOGPATH = /w06
```

```
LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl128a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 444251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl128a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 448001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl128b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 451001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl128b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 451501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl128b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 455001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl128b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 458501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
  WAREHOUSE = 193600
  MEASUREMENT = 21600
  PAY-MIX = 4302
  ORD-MIX = 402
  DEL-MIX = 402
  STK-MIX = 402
  NEW-KEYING = 18010
  PAY-KEYING = 3010
  ORD-KEYING = 2010
  DEL-KEYING = 2010
  STK-KEYING = 2010
  NEW-THINK = 12020
  PAY-THINK = 12020
  ORD-THINK = 10020
  DEL-THINK = 5020
  STK-THINK = 5020
  CONST-CLAST = 111
  CONST-CID = 1023
  CONST-IID = 8191
  THR-PER-PROC = 250
  SYNC = 0
ENDVARIABLE
ENDGROUP
```

```
.....
rte43.conf
.....
#
# rte43.conf :configuration file for TPC-C
Rev3.0
# Author : mkdef -Auto Configurator for R3-
#
STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl129a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 682001
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl129a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 685751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl129a
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 689501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl129b
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 693001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl129b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 693251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl129b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 697001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl129b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 700751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl130a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 924001
```

```

LOGPATH = /w05
LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl130a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 924501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl130a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 928251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl130a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 932001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl130b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 935001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl130b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 935501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl130b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 939001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl130b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 942501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
  WAREHOUSE = 193600
  MEASUREMENT = 21600
  PAY-MIX = 4302
  ORD-MIX = 402
  DEL-MIX = 402
  STK-MIX = 402
  NEW-KEYING = 18010
  PAY-KEYING = 3010
  ORD-KEYING = 2010
  DEL-KEYING = 2010
  STK-KEYING = 2010
  NEW-THINK = 12020
  PAY-THINK = 12020
  ORD-THINK = 10020

DEL-THINK = 5020
STK-THINK = 5020
CONST-CLAST = 111
CONST-CID = 1023
CONST-IID = 8191
THR-PER-PROC = 250
SYNC = 0
ENDVARIABLE
ENDGROUP

.....:
rte44.conf
.....:

#
# rte44.conf :configuration file for TPC-C
Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl131a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1166001
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl131a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1169751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl131a
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1173501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl131b
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 1177001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl131b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1177251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl131b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1181001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl131b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 1184751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl132a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1430001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl132a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1430501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl132a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1434251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl132a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 1438001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl132b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1441001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl132b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1441501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl132b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1445001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl132b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1448501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
  WAREHOUSE = 193600
  MEASUREMENT = 21600
  PAY-MIX = 4302
  ORD-MIX = 402
  DEL-MIX = 402

```

```

STK-MIX = 402
NEW-KEYING = 18010
PAY-KEYING = 3010
ORD-KEYING = 2010
DEL-KEYING = 2010
STK-KEYING = 2010
NEW-THINK = 12020
PAY-THINK = 12020
ORD-THINK = 10020
DEL-THINK = 5020
STK-THINK = 5020
CONST-CLAST = 111
CONST-CID = 1023
CONST-IID = 8191
THR-PER-PROC = 250
SYNC = 0
ENDVARIABLE
ENDGROUP

```

```

.....
rte45.conf
.....

```

```

#
# rte45.conf :configuration file for TPC-C
Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

```

```

STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl133a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1672001
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl133a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1675751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl133a
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1679501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl133b
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 1683001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl133b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1683251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl133b
  SUTPORT = 80

```

```

SUTTERM = 3750
BASENO = 1687001
LOGPATH = /w04
LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl133b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 1690751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl134a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1914001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl134a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1914501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl134a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1918251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl134a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 1922001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl134b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1925001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl134b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1925501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl134b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1929001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl134b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1932501
  LOGPATH = /w11

```

```

LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
  WAREHOUSE = 193600
  MEASUREMENT = 21600
  PAY-MIX = 4302
  ORD-MIX = 402
  DEL-MIX = 402
  STK-MIX = 402
  NEW-KEYING = 18010
  PAY-KEYING = 3010
  ORD-KEYING = 2010
  DEL-KEYING = 2010
  STK-KEYING = 2010
  NEW-THINK = 12020
  PAY-THINK = 12020
  ORD-THINK = 10020
  DEL-THINK = 5020
  STK-THINK = 5020
  CONST-CLAST = 111
  CONST-CID = 1023
  CONST-IID = 8191
  THR-PER-PROC = 250
  SYNC = 0
ENDVARIABLE
ENDGROUP

```

```

.....
rte46.conf
.....

```

```

#
# rte46.conf :configuration file for TPC-C
Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

```

```

STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl135a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 220001
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl135a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 223751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl135a
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 227501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl135b
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 231001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl135b

```



```

SUTPORT = 80
SUTTERM = 3750
BASENO = 231251
LOGPATH = /w03
LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl135b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 235001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl135b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 238751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl136a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 462001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl136a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 462501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl136a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 466251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl136a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 470001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl136b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 473001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl136b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 473501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl136b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 477001

```

```

LOGPATH = /w10
LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl136b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 480501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
  WAREHOUSE = 193600
  MEASUREMENT = 21600
  PAY-MIX = 4302
  ORD-MIX = 402
  DEL-MIX = 402
  STK-MIX = 402
  NEW-KEYING = 18010
  PAY-KEYING = 3010
  ORD-KEYING = 2010
  DEL-KEYING = 2010
  STK-KEYING = 2010
  NEW-THINK = 12020
  PAY-THINK = 12020
  ORD-THINK = 10020
  DEL-THINK = 5020
  STK-THINK = 5020
  CONST-CLAST = 111
  CONST-CID = 1023
  CONST-IID = 8191
  THR-PER-PROC = 250
  SYNC = 0
ENDVARIABLE
ENDGROUP

.....
rte47.conf
.....

#
# rte47.conf :configuration file for TPC-C
# Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

STARTGROUP
  STARTRTE
    STARTSUT
      SUTHOST = cl137a
      SUTPORT = 80
      SUTTERM = 3750
      BASENO = 704001
      LOGPATH = /w00
      LOGLEVEL = 0
    ENSUT
    STARTSUT
      SUTHOST = cl137a
      SUTPORT = 80
      SUTTERM = 3750
      BASENO = 707751
      LOGPATH = /w01
      LOGLEVEL = 0
    ENSUT
    STARTSUT
      SUTHOST = cl137a
      SUTPORT = 80
      SUTTERM = 3500
      BASENO = 711501
      LOGPATH = /w02
      LOGLEVEL = 0
    ENSUT
    STARTSUT

```

```

  SUTHOST = cl137b
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 715001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl137b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 715251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl137b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 719001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl137b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 722751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl138a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 946001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl138a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 946501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl138a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 950251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl138a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 954001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl138b
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 957001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl138b
  SUTPORT = 80
  SUTTERM = 3500

```

```

BASENO = 957501
LOGPATH = /w09
LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl138b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 961001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl138b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 964501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
  WAREHOUSE = 193600
  MEASUREMENT = 21600
  PAY-MIX = 4302
  ORD-MIX = 402
  DEL-MIX = 402
  STK-MIX = 402
  NEW-KEYING = 18010
  PAY-KEYING = 3010
  ORD-KEYING = 2010
  DEL-KEYING = 2010
  STK-KEYING = 2010
  NEW-THINK = 12020
  PAY-THINK = 12020
  ORD-THINK = 10020
  DEL-THINK = 5020
  STK-THINK = 5020
  CONST-CLAST = 111
  CONST-CID = 1023
  CONST-IID = 8191
  THR-PER-PROC = 250
  SYNC = 0
ENDVARIABLE
ENDGROUP

```

```

.....
rte48.conf
.....

```

```

#
# rte48.conf :configuration file for TPC-C
# Rev3.0
# Author : mkdef -Auto Configurator for R3-
#

```

```

STARTGROUP
STARTRTE
STARTSUT
  SUTHOST = cl069a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1540001
  LOGPATH = /w00
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl069a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1543751
  LOGPATH = /w01
  LOGLEVEL = 0
ENDSUT

```

```

STARTSUT
  SUTHOST = cl069a
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1547501
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl069b
  SUTPORT = 80
  SUTTERM = 250
  BASENO = 1551001
  LOGPATH = /w02
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl069b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1551251
  LOGPATH = /w03
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl069b
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1555001
  LOGPATH = /w04
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl069b
  SUTPORT = 80
  SUTTERM = 3250
  BASENO = 1558751
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl139a
  SUTPORT = 80
  SUTTERM = 500
  BASENO = 1188001
  LOGPATH = /w05
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl139a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1188501
  LOGPATH = /w06
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl139a
  SUTPORT = 80
  SUTTERM = 3750
  BASENO = 1192251
  LOGPATH = /w07
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl139a
  SUTPORT = 80
  SUTTERM = 3000
  BASENO = 1196001
  LOGPATH = /w08
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl139b
  SUTPORT = 80

```

```

SUTTERM = 500
BASENO = 1199001
LOGPATH = /w08
LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl139b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1199501
  LOGPATH = /w09
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl139b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1203001
  LOGPATH = /w10
  LOGLEVEL = 0
ENDSUT
STARTSUT
  SUTHOST = cl139b
  SUTPORT = 80
  SUTTERM = 3500
  BASENO = 1206501
  LOGPATH = /w11
  LOGLEVEL = 0
ENDSUT
ENDRTE
STARTVARIABLE
  WAREHOUSE = 193600
  MEASUREMENT = 21600
  PAY-MIX = 4302
  ORD-MIX = 402
  DEL-MIX = 402
  STK-MIX = 402
  NEW-KEYING = 18010
  PAY-KEYING = 3010
  ORD-KEYING = 2010
  DEL-KEYING = 2010
  STK-KEYING = 2010
  NEW-THINK = 12020
  PAY-THINK = 12020
  ORD-THINK = 10020
  DEL-THINK = 5020
  STK-THINK = 5020
  CONST-CLAST = 111
  CONST-CID = 1023
  CONST-IID = 8191
  THR-PER-PROC = 250
  SYNC = 0
ENDVARIABLE
ENDGROUP

```

Appendix D: System Tunables

=====
(PRIMEQUEST configuration)
=====

[OS tunables]

[/etc/modprobe.conf]

```
alias eth0 tg3
alias eth1 e100
alias eth2 tg3
alias eth3 tg3
alias eth4 tg3
alias eth5 tg3
alias scsi_hostadapter mptbase
alias eth6 tg3
alias eth7 tg3
alias eth8 tg3
alias eth10 tg3
alias eth12 tg3
alias eth16 tg3
alias eth20 tg3
alias eth24 tg3
alias eth28 tg3
alias eth32 tg3
alias scsi_hostadapter1 mptscsih
options lpfc lpfc_lun_queue_depth=12
alias scsi_hostadapter2 lpfc
alias usb-controller ehci-hcd
alias usb-controller1 uhci-hcd
alias eth35 e1000
alias eth36 e1000
```

[/etc/sysctl.conf]

```
# Kernel sysctl configuration file for Red Hat
Linux
#
# For binary values, 0 is disabled, 1 is enabled.
See sysctl(8) and
# sysctl.conf(5) for more details.
```

```
# Controls IP packet forwarding
net.ipv4.ip_forward = 0
```

```
# Controls source route verification
net.ipv4.conf.default.rp_filter = 1
```

```
# Do not accept source routing
net.ipv4.conf.default.accept_source_route = 0
```

```
# Controls the System Request debugging
functionality of the kernel
kernel.sysrq = 0
```

```
# Controls whether core dumps will append the
PID to the core filename.
# Useful for debugging multi-threaded
applications.
kernel.core_uses_pid = 1
kernel.sem = 100 100000 120 512
```

```
kernel.shmmax = 0x4000000000
kernel.shmall = 0x200000000
fs.aio-max-nr = 5242880
```

```
vm.nr_hugepages = 8036
```

[/etc/security/limits.conf]

```
#
#Each line describes a limit for a user in the
form:
#
#<domain> <type> <item> <value>
#
#Where:
#<domain> can be:
# - an user name
# - a group name, with @group syntax
# - the wildcard *, for default entry
# - the wildcard %, can be also used
with %group syntax,
# for maxlogin limit
#
#<type> can have the two values:
# - "soft" for enforcing the soft limits
# - "hard" for enforcing hard limits
#
#<item> can be one of the following:
# - core - limits the core file size (KB)
# - data - max data size (KB)
# - fsize - maximum filesize (KB)
# - memlock - max locked-in-memory
address space (KB)
# - nfile - max number of open files
# - rss - max resident set size (KB)
# - stack - max stack size (KB)
# - cpu - max CPU time (MIN)
# - nproc - max number of processes
# - as - address space limit
# - maxlogins - max number of logins for this
user
# - priority - the priority to run user process
with
# - locks - max number of file locks the user
can hold
#
#<domain> <type> <item> <value>
#
#* soft core 0
#* hard rss 10000
#@student hard nproc 20
#@faculty soft nproc 20
#@faculty hard nproc 50
#ftp hard nproc 0
#@student - maxlogins 4
```

```
oracle soft memlock 2147483648
oracle hard memlock 2147483648
oracle soft nfile 4096
oracle hard nfile 65536
oracle soft nproc 4095
oracle hard nproc 16384
```

```
# End of file
```

[/etc/eliilo.conf]

```
prompt
timeout=20
default=tpc-c
relocatable

image=vmlinux-2.6.9-42.EL.oralgessmp
label=tpc-c
initrd=initrd-2.6.9-
42.EL.oralgessmp.lpfc.img
read-only
append="rhgb root=/dev/sda2
log_buf_len=2M console=ttyS0,19200n8r
```

```
elevator=noop ide=nodma
ihash_entries=1000000 dhash_entries=1000000
rhash_entries=500000 thash_entries=100
000 ro"
```

[Database tunables]

```
-----
p_run.ora
-----
aq_tm_processes = 0
compatible = 10.1.0.0.0
control_files =
(/ora_dev/control_001,/ora_dev/control_002)
cursor_space_for_time = true
db_16k_cache_size = 511000M
db_4k_cache_size = 2048M
db_8k_cache_size = 3072M
db_block_checking = false
db_block_checksum = false
db_block_size = 2048
db_cache_size = 21360M
db_file_multiblock_read_count = 32
db_files = 3806
db_keep_cache_size = 1193216M
db_name = tpcc
db_recycle_cache_size = 275200M
disk_asynch_io = true
dml_locks = 500
fast_start_mttr_target = 0
java_pool_size = 0
log_buffer = 33554432
log_checkpoint_interval = 0
log_checkpoint_timeout = 1740
log_checkpoints_to_alert = true
parallel_max_servers = 16
plsql_optimize_level = 2
processes = 3100
query_rewrite_enabled = false
remote_login_passwordfile = shared
replication_dependency_tracking = false
sessions = 4600
shared_pool_size = 50176M
statistics_level = basic
timed_statistics = false
trace_enabled = false
transactions = 5000
transactions_per_rollback_segment = 1
undo_management = auto
undo_retention = 1
undo_tablespace = undo_1
```

[PRIMERGY tunables]

=====
(cl001 configuration)
=====

[OS tunables]

```
-----
limits.conf
-----
```

[/etc/security/limits.conf]

```
#
#Each line describes a limit for a user in the
form:
#
#<domain> <type> <item> <value>
#
#Where:
#<domain> can be:
# - an user name
# - a group name, with @group syntax
```

```

# - the wildcard *, for default entry
#
#<type> can have the two values:
# - "soft" for enforcing the soft limits
# - "hard" for enforcing hard limits
#
#<item> can be one of the following:
# - core - limits the core file size (KB)
# - data - max data size (KB)
# - fsize - maximum filesize (KB)
# - memlock - max locked-in-memory
address space (KB)
# - nofile - max number of open files
# - rss - max resident set size (KB)
# - stack - max stack size (KB)
# - cpu - max CPU time (MIN)
# - nproc - max number of processes
# - as - address space limit
# - maxlogins - max number of logins for this
user
# - priority - the priority to run user process
with
# - locks - max number of file locks the user
can hold
#
#<domain> <type> <item> <value>
#
#*      soft core      0
#*      hard  rss      10000
#@student hard nproc   20
#@faculty soft nproc   20
#@faculty hard nproc   50
#ftp    hard nproc    0
#@student -   maxlogins 4
tpc - nofile 30000
tpc - nproc 30000

# End of file

.....
sysctl.conf
.....

# Kernel sysctl configuration file for Red Hat
Linux
#
# For binary values, 0 is disabled, 1 is enabled.
See sysctl(8) and
# sysctl.conf(5) for more details.

# Controls IP packet forwarding
net.ipv4.ip_forward = 0

# Controls source route verification
net.ipv4.conf.default.rp_filter = 1

# Controls the System Request debugging
functionality of the kernel
kernel.sysrq = 0

# Controls whether core dumps will append the
PID to the core filename.
# Useful for debugging multi-threaded
applications.
kernel.core_uses_pid = 1

# Change filedescriptor
fs.file-max = 30000

# Change Message queue
kernel.msgmni = 30000
kernel.msgmnb = 1536000

```

```

# Change Max process
kernel.threads-max = 30000

# Change Semaphore
kernel.sem = 3000 384000 32 128

# Change TCP/IP backlog
net.ipv4.tcp_max_syn_backlog = 4096

[HTTP server tunables]
-----
.....
apache_cl_start.sh
.....

#!/bin/sh
export
LD_LIBRARY_PATH=$ORACLE_HOME/srvm/li
b:$ORACLE_HOME/lib64:$ORACLE_HOME/lib:
/usr/lib:$ORACLE_HOME/rdbms/lib:$ORACLE_
HOME/network/lib:$TUXDIR/lib

ulimit -u 30000
ulimit -s 1536

#/sbin/swapon -a

# For 3tier tune
SVRAPL= ps -e | grep tpcsfmlw | awk '{print $1}'
/usr/bin/renice -20 -p ${SVRAPL}

#rm -f /home/tpc/sar.tmp
/home/tpc/sar.`hostname`
#/usr/lib/sa/sadc 5 > /home/tpc/sar.tmp &
# For 3tier tune

apachectl start

.....
httpd.conf
.....

#
# Based upon the NCSA server configuration
files originally by Rob McCool.
#
# This is the main Apache server configuration
file. It contains the
# configuration directives that give the server its
instructions.
# See <URL:http://httpd.apache.org/docs-2.0/>
for detailed information about
# the directives.
#
# Do NOT simply read the instructions in here
without understanding
# what they do. They're here only as hints or
reminders. If you are unsure
# consult the online docs. You have been
warned.
#
# The configuration directives are grouped into
three basic sections:
# 1. Directives that control the operation of the
Apache server process as a
# whole (the 'global environment').
# 2. Directives that define the parameters of the
'main' or 'default' server,
# which responds to requests that aren't
handled by a virtual host.

```

```

# These directives also provide default values
for the settings
# of all virtual hosts.
# 3. Settings for virtual hosts, which allow Web
requests to be sent to
# different IP addresses or hostnames and
have them handled by the
# same Apache server process.
#
# Configuration and logfile names: If the
filenames you specify for many
# of the server's control files begin with "/" (or
"drive:/" for Win32), the
# server will use that explicit path. If the
filenames do "not" begin
# with "/", the value of ServerRoot is prepended -
- so "logs/foo.log"
# with ServerRoot set to "/etc/httpd" will be
interpreted by the
# server as "/etc/httpd/logs/foo.log".
#
### Section 1: Global Environment
#
# The directives in this section affect the overall
operation of Apache,
# such as the number of concurrent requests it
can handle or where it
# can find its configuration files.
#
#
# Don't give away too much information about all
the subcomponents
# we are running. Comment out this line if you
don't mind remote sites
# finding out what major optional modules you
are running
#ServerTokens OS
ServerTokens Productly

#
# ServerRoot: The top of the directory tree under
which the server's
# configuration, error, and log files are kept.
#
# NOTE! If you intend to place this on an NFS
(or otherwise network)
# mounted filesystem then please read the
LockFile documentation
# (available at
<URL:http://httpd.apache.org/docs-
2.0/mod/core.html#lockfile>);
# you will save yourself a lot of trouble.
#
# Do NOT add a slash at the end of the directory
path.
#
ServerRoot "/etc/httpd"

#
# ScoreBoardFile: File used to store internal
server process information.
# If unspecified (the default), the scoreboard will
be stored in an
# anonymous shared memory segment, and will
be unavailable to third-party
# applications.
# If specified, ensure that no two invocations of
Apache share the same
# scoreboard file. The scoreboard file MUST BE
STORED ON A LOCAL DISK.
#
#ScoreBoardFile run/httpd.scoreboard

```

```
#
# PidFile: The file in which the server should
# record its process
# identification number when it starts.
#
PidFile run/httpd.pid

#
# Timeout: The number of seconds before
# receives and sends time out.
#
#Timeout 300
Timeout 999

#
# KeepAlive: Whether or not to allow persistent
# connections (more than
# one request per connection). Set to "Off" to
# deactivate.
#
#KeepAlive Off
KeepAlive On

#
# MaxKeepAliveRequests: The maximum
# number of requests to allow
# during a persistent connection. Set to 0 to
# allow an unlimited amount.
# We recommend you leave this number high,
# for maximum performance.
#
#MaxKeepAliveRequests 100
MaxKeepAliveRequests 0

#
# KeepAliveTimeout: Number of seconds to wait
# for the next request from the
# same client on the same connection.
#
#KeepAliveTimeout 15
KeepAliveTimeout 999

##
## Server-Pool Size Regulation (MPM specific)
##

# prefork MPM
# StartServers: number of server processes to
# start
# MinSpareServers: minimum number of server
# processes which are kept spare
# MaxSpareServers: maximum number of server
# processes which are kept spare
# MaxClients: maximum number of server
# processes allowed to start
# MaxRequestsPerChild: maximum number of
# requests a server process serves
<IfModule prefork.c>
StartServers 8
MinSpareServers 5
MaxSpareServers 20
MaxClients 150
MaxRequestsPerChild 1000
</IfModule>

# worker MPM
# StartServers: initial number of server
# processes to start
# MaxClients: maximum number of simultaneous
# client connections
# MinSpareThreads: minimum number of worker
# threads which are kept spare
# MaxSpareThreads: maximum number of
# worker threads which are kept spare
```

```
# ThreadsPerChild: constant number of worker
# threads in each server process
# MaxRequestsPerChild: maximum number of
# requests a server process serves
<IfModule worker.c>

StartServers 23
ServerLimit 23
ThreadLimit 500
MaxClients 11500
MinSpareThreads 1
MaxSpareThreads 11500
ThreadsPerChild 500
MaxRequestsPerChild 0

#
#
# To reduce memory usage in the worker MPM,
# the thread guard page
#
# To reduce memory usage in the worker MPM,
# the thread guard page
# can be disabled, at the expense of some
# protection against stack
# overflow.
#
#ThreadGuardArea off

</IfModule>

#
# Listen: Allows you to bind Apache to specific
# IP addresses and/or
# ports, in addition to the default. See also the
# <VirtualHost>
# directive.
#
# Change this to Listen on specific IP addresses
# as shown below to
# prevent Apache from glomming onto all bound
# IP addresses (0.0.0.0)
# e.g. "Listen 12.34.56.78:80"
#
# To allow connections to IPv6 addresses add
# "Listen [::]:80"
#
Listen 0.0.0.0:80

#
# Dynamic Shared Object (DSO) Support
#

# To be able to use the functionality of a module
# which was built as a DSO you
# have to place corresponding 'LoadModule'
# lines at this location so the
# directives contained in it are actually available
#_before_ they are used.
# Statically compiled modules (those listed by
# `httpd -l') do not need
# to be loaded here.
#
# Example:
# LoadModule foo_module modules/mod_foo.so
#
LoadModule tpmpl_module
modules/mod_tpmpl.so
LoadModule access_module
modules/mod_access.so
LoadModule status_module
modules/mod_status.so
LoadModule alias_module
modules/mod_alias.so
LoadModule cgi_module modules/mod_cgi.so
```

```
#
# Load config files from the config directory
# "/etc/httpd/conf.d".
#
#Include conf.d/*.conf

#
# ExtendedStatus controls whether Apache will
# generate "full" status
# information (ExtendedStatus On) or just basic
# information (ExtendedStatus
# Off) when the "server-status" handler is called.
# The default is Off.
#
#ExtendedStatus On

### Section 2: 'Main' server configuration
#
# The directives in this section set up the values
# used by the 'main'
# server, which responds to any requests that
# aren't handled by a
# <VirtualHost> definition. These values also
# provide defaults for
# any <VirtualHost> containers you may define
# later in the file.
#
# All of these directives may appear inside
# <VirtualHost> containers,
# in which case these default settings will be
# overridden for the
# virtual host being defined.
#

#
# If you wish httpd to run as a different user or
# group, you must run
# httpd as root initially and it will switch.
#
# User/Group: The name (or #number) of the
# user/group to run httpd as.
# . On SCO (ODT 3) use "User nouser" and
# "Group nogroup".
# . On HP-UX you may not be able to use
# shared memory as nobody, and the
# suggested workaround is to create a user
# www and use that user.
# NOTE that some kernels refuse to
# setgid(Group) or semctl(IPC_SET)
# when the value of (unsigned)Group is above
# 60000;
# don't use Group #-1 on these systems!
#
#User apache
#Group apache
User tpc
Group tpc

#
# ServerAdmin: Your address, where problems
# with the server should be
# e-mailed. This address appears on some
# server-generated pages, such
# as error documents. e.g. admin@your-
# domain.com
#
ServerAdmin root@localhost

#
# ServerName gives the name and port that the
# server uses to identify itself.
# This can often be determined automatically,
# but we recommend you specify
```

```
# it explicitly to prevent problems during startup.
#
# If this is not set to valid DNS name for your
host, server-generated
# redirections will not work. See also the
UseCanonicalName directive.
#
# If your host doesn't have a registered DNS
name, enter its IP address here.
# You will have to access it by its address
anyway, and this will make
# redirections work in a sensible way.
#
#ServerName new.host.name:80
ServerName tpccserver:80

#
# UseCanonicalName: Determines how Apache
constructs self-referencing
# URLs and the SERVER_NAME and
SERVER_PORT variables.
# When set "Off", Apache will use the Hostname
and Port supplied
# by the client. When set "On", Apache will use
the value of the
# ServerName directive.
#
UseCanonicalName Off

#
# DocumentRoot: The directory out of which you
will serve your
# documents. By default, all requests are taken
from this directory, but
# symbolic links and aliases may be used to
point to other locations.
#
#DocumentRoot "/var/www/html"

#
# Each directory to which Apache has access
can be configured with respect
# to which services and features are allowed
and/or disabled in that
# directory (and its subdirectories).
#
# First, we configure the "default" to be a very
restrictive set of
# features.
#
#<Directory />
# Options FollowSymLinks
# AllowOverride None
#</Directory>

#
# Note that from this point forward you must
specifically allow
# particular features to be enabled - so if
something's not working as
# you might expect, make sure that you have
specifically enabled it
# below.
#

#
# UserDir: The name of the directory that is
appended onto a user's home
# directory if a ~user request is received.
#
# The path to the end user account 'public_html'
directory must be
# accessible to the webserver userid. This
usually means that ~userid
```

```
# must have permissions of 711,
~userid/public_html must have permissions
# of 755, and documents contained therein must
be world-readable.
# Otherwise, the client will only receive a "403
Forbidden" message.
#
# See also:
http://httpd.apache.org/docs/misc/FAQ.html#forb
idden
#
#<IfModule mod_userdir.c>
#
# UserDir is disabled by default since it can
confirm the presence
# of a username on the system (depending on
home directory
# permissions).
#
# UserDir disable

#
# To enable requests to ~-user/ to serve the
user's public_html
# directory, remove the "UserDir disable" line
above, and uncomment
# the following line instead:
#
#UserDir public_html

#</IfModule>

#
# Control access to UserDir directories. The
following is an example
# for a site where these directories are restricted
to read-only.
#
#<Directory /home/"*/public_html>
# AllowOverride FileInfo AuthConfig Limit
# Options MultiViews Indexes
SymLinksIfOwnerMatch IncludesNoExec
# <Limit GET POST OPTIONS>
# Order allow,deny
# Allow from all
# </Limit>
# <LimitExcept GET POST OPTIONS>
# Order deny,allow
# Deny from all
# </LimitExcept>
#</Directory>

#
# DirectoryIndex: sets the file that Apache will
serve if a directory
# is requested.
#
# The index.html.var file (a type-map) is used to
deliver content-
# negotiated documents. The MultiViews Option
can be used for the
# same purpose, but it is much slower.
#

#
# AccessFileName: The name of the file to look
for in each directory
# for additional configuration directives. See
also the AllowOverride
# directive.
#
AccessFileName .htaccess

#
```

```
# The following lines prevent .htaccess
and .htpasswd files from being
# viewed by Web clients.
#

#
# TypesConfig describes where the mime.types
file (or equivalent) is
# to be found.
#

#
# DefaultType is the default MIME type the
server will use for a document
# if it cannot otherwise determine one, such as
from filename extensions.
# If your server contains mostly text or HTML
documents, "text/plain" is
# a good value. If most of your content is binary,
such as applications
# or images, you may want to use
"application/octet-stream" instead to
# keep browsers from trying to display binary
files as though they are
# text.
#
DefaultType text/plain

#
# The mod_mime_magic module allows the
server to use various hints from the
# contents of the file itself to determine its type.
The MIMEMagicFile
# directive tells the module where the hint
definitions are located.
#
#<IfModule mod_mime_magic.c>
## MIMEMagicFile /usr/share/magic.mime
# MIMEMagicFile conf/magic
#</IfModule>

#
# HostnameLookups: Log the names of clients
or just their IP addresses
# e.g., www.apache.org (on) or 204.62.129.132
(off).
# The default is off because it'd be overall better
for the net if people
# had to knowingly turn this feature on, since
enabling it means that
# each client request will result in AT LEAST one
lookup request to the
# nameserver.
#
HostnameLookups Off

#
# EnableMMAP: Control whether memory-
mapping is used to deliver
# files (assuming that the underlying OS
supports it).
# The default is on; turn this off if you serve from
NFS-mounted
# filesystems. On some systems, turning it off
(regardless of
# filesystem) can improve performance; for
details, please see
# http://httpd.apache.org/docs-
2.0/mod/core.html#enablenmap
#
#EnableMMAP off

#
# EnableSendfile: Control whether the sendfile
kernel support is
```

```
# used to deliver files (assuming that the OS
supports it).
# The default is on; turn this off if you serve from
NFS-mounted
# filesystems. Please see
# http://httpd.apache.org/docs-
2.0/mod/core.html#enablesendfile
#
#EnableSendfile off
#
#
# ErrorLog: The location of the error log file.
# If you do not specify an ErrorLog directive
within a <VirtualHost>
# container, error messages relating to that
virtual host will be
# logged here. If you *do* define an error logfile
for a <VirtualHost>
# container, that host's errors will be logged
there and not here.
#
ErrorLog logs/error_log
#
# LogLevel: Control the number of messages
logged to the error_log.
# Possible values include: debug, info, notice,
warn, error, crit,
# alert, emerg.
#
LogLevel warn
#
# The following directives define some format
nicknames for use with
# a CustomLog directive (see below).
#
#
# The location and format of the access logfile
(Common Logfile Format).
# If you do not define any access logfiles within
a <VirtualHost>
# container, they will be logged here.
Contrariwise, if you *do*
# define per-<VirtualHost> access logfiles,
transactions will be
# logged therein and *not* in this file.
#
# CustomLog logs/access_log common
#CustomLog logs/access_log combined
#
# If you would like to have agent and referer
logfiles, uncomment the
# following directives.
#
#CustomLog logs/referer_log referer
#CustomLog logs/agent_log agent
#
# If you prefer a single logfile with access, agent,
and referer information
# (Combined Logfile Format) you can use the
following directive.
#
#CustomLog logs/access_log combined
#
# Optionally add a line containing the server
version and virtual host
# name to server-generated pages (error
documents, FTP directory listings,
```

```
# mod_status and mod_info output etc., but not
CGI generated documents).
# Set to "EMail" to also include a mailto: link to
the ServerAdmin.
# Set to one of: On | Off | EMail
#
#ServerSignature On
ServerSignature Off
#
# Aliases: Add here as many aliases as you
need (with no limit). The format is
# Alias fakename realname
#
# Note that if you include a trailing / on fakename
then the server will
# require it to be present in the URL. So "/icons"
isn't aliased in this
# example, only "/icons/". If the fakename is
slash-terminated, then the
# realname must also be slash terminated, and if
the fakename omits the
# trailing slash, the realname must also omit it.
#
# We include the /icons/ alias for FancyIndexed
directory listings. If you
# do not use FancyIndexing, you may comment
this out.
#
#
# This should be changed to the
ServerRoot/manual/. The alias provides
# the manual, even if you choose to move your
DocumentRoot. You may comment
# this out if you do not care for the
documentation.
#
#<IfModule mod_dav_fs.c>
# # Location of the WebDAV lock database.
# DAVLockDB /var/lib/dav/lockdb
#</IfModule>
#
# ScriptAlias: This controls which directories
contain server scripts.
# ScriptAliases are essentially the same as
Aliases, except that
# documents in the realname directory are
treated as applications and
# run by the server when requested rather than
as documents sent to the client.
# The same rules about trailing "/" apply to
ScriptAlias directives as to
# Alias.
#
#ScriptAlias /cgi-bin/ "/var/www/cgi-bin/"
ScriptAlias /cgi-bin/ "/home/tpc/tool/bin/"
#
# "/var/www/cgi-bin" should be changed to
whatever your ScriptAliased
# CGI directory exists, if you have that
configured.
#
<Directory "/var/www/cgi-bin">
    AllowOverride None
    Options None
    Order allow,deny
    Allow from all
</Directory>
#
# Redirect allows you to tell clients about
documents which used to exist in
```

```
# your server's namespace, but do not anymore.
This allows you to tell the
# clients where to look for the relocated
document.
# Example:
# Redirect permanent /foo
http://www.example.com/bar
#
# Directives controlling the display of server-
generated directory listings.
#
#
# FancyIndexing is whether you want fancy
directory indexing or standard.
# VersionSort is whether files containing version
numbers should be
# compared in the natural way, so that `apache-
1.3.9.tar' is placed before
# `apache-1.3.12.tar'.
#
#
# AddIcon* directives tell the server which icon
to show for different
# files or filename extensions. These are only
displayed for
# FancyIndexed directories.
#
#
# DefaultIcon is which icon to show for files
which do not have an icon
# explicitly set.
#
#
# AddDescription allows you to place a short
description after a file in
# server-generated indexes. These are only
displayed for FancyIndexed
# directories.
# Format: AddDescription "description" filename
#
#AddDescription "GZIP compressed
document" .gz
#AddDescription "tar archive" .tar
#AddDescription "GZIP compressed tar
archive" .tgz
#
# ReadmeName is the name of the README file
the server will look for by
# default, and append to directory listings.
#
# HeaderName is the name of a file which
should be prepended to
# directory indexes.
#
#
# IndexIgnore is a set of filenames which
directory indexing should ignore
# and not include in the listing. Shell-style
wildcarding is permitted.
#
#
# AddEncoding allows you to have certain
browsers (Mosaic/X 2.1+) uncompress
# information on the fly. Note: Not all browsers
support this.
# Despite the name similarity, the following Add*
directives have nothing
```

```
# to do with the FancyIndexing customization
directives above.
#
#
# DefaultLanguage and AddLanguage allows
you to specify the language of
# a document. You can then use content
negotiation to give a browser a
# file in a language the user can understand.
#
# Specify a default language. This means that all
data
# going out without a specific language tag (see
below) will
# be marked with this one. You probably do NOT
want to set
# this unless you are sure it is correct for all
cases.
#
# * It is generally better to not mark a page as
# * being a certain language than marking it with
the wrong
# * language!
#
# DefaultLanguage nl
#
# Note 1: The suffix does not have to be the
same as the language
# keyword --- those with documents in Polish
(whose net-standard
# language code is pl) may wish to use
"AddLanguage pl .po" to
# avoid the ambiguity with the common suffix for
perl scripts.
#
# Note 2: The example entries below illustrate
that in some cases
# the two character 'Language' abbreviation is
not identical to
# the two character 'Country' code for its country,
# E.g. 'Danmark/dk' versus 'Danish/da'.
#
# Note 3: In the case of 'ltz' we violate the RFC
by using a three char
# specifier. There is 'work in progress' to fix this
and get
# the reference data for rfc1766 cleaned up.
#
# Danish (da) - Dutch (nl) - English (en) -
Estonian (et)
# French (fr) - German (de) - Greek-Modern (el)
# Italian (it) - Norwegian (no) - Norwegian
Nynorsk (nn) - Korean (ko)
# Portugese (pt) - Luxembourggeois* (ltz)
# Spanish (es) - Swedish (sv) - Catalan (ca) -
Czech(cs)
# Polish (pl) - Brazilian Portuguese (pt-br) -
Japanese (ja)
# Russian (ru) - Croatian (hr)
#
#
# LanguagePriority allows you to give
precedence to some languages
# in case of a tie during content negotiation.
#
# Just list the languages in decreasing order of
preference. We have
# more or less alphabetized them here. You
probably want to change this.
#
#
```

```
# ForceLanguagePriority allows you to serve a
result page rather than
# MULTIPLE CHOICES (Prefer) [in case of a tie]
or NOT ACCEPTABLE (Fallback)
# [in case no accepted languages matched the
available variants]
#
#
# Specify a default charset for all pages sent out.
This is
# always a good idea and opens the door for
future internationalisation
# of your web site, should you ever want it.
Specifying it as
# a default does little harm; as the standard
dictates that a page
# is in iso-8859-1 (latin1) unless specified
otherwise i.e. you
# are merely stating the obvious. There are also
some security
# reasons in browsers, related to javascript and
URL parsing
# which encourage you to always set a default
char set.
#
AddDefaultCharset UTF-8
#
# Commonly used filename extensions to
character sets. You probably
# want to avoid clashes with the language
extensions, unless you
# are good at carefully testing your setup after
each change.
# See
http://www.iana.org/assignments/character-sets
for the
# official list of charset names and their
respective RFCs
#
#
# AddType allows you to add to or override the
MIME configuration
# file mime.types for specific file types.
#
#
# AddHandler allows you to map certain file
extensions to "handlers":
# actions unrelated to filetype. These can be
either built into the server
# or added with the Action directive (see below)
#
# To use CGI scripts outside of ScriptAliased
directories:
# (You will also need to add "ExecCGI" to the
"Options" directive.)
#
#AddHandler cgi-script .cgi
#
#
# For files that include their own HTTP headers:
#
#AddHandler send-as-is asis
#
#
# For server-parsed imagemap files:
#
#
#
# For type maps (negotiated resources):
# (This is enabled by default to allow the Apache
"It Worked" page
```

```
# to be distributed in multiple languages.)
#
# Filters allow you to process content before it is
sent to the client.
#
# To parse .shtml files for server-side includes
(SS):
# (You will also need to add "Includes" to the
"Options" directive.)
#
#
# Action lets you define media types that will
execute a script whenever
# a matching file is called. This eliminates the
need for repeated URL
# pathnames for oft-used CGI file processors.
# Format: Action media/type /cgi-script/location
# Format: Action handler-name /cgi-
script/location
#
#
# Customizable error responses come in three
flavors:
# 1) plain text 2) local redirects 3) external
redirects
#
# Some examples:
#ErrorDocument 500 "The server made a boo
boo."
#ErrorDocument 404 /missing.html
#ErrorDocument 404 "/cgi-
bin/missing_handler.pl"
#ErrorDocument 402
http://www.example.com/subscription_info.html
#
#
# Putting this all together, we can
Internationalize error responses.
#
# We use Alias to redirect any
/error/HTTP_<error>.html.var response to
# our collection of by-error message multi-
language collections. We use
# includes to substitute the appropriate text.
#
# You can modify the messages' appearance
without changing any of the
# default HTTP_<error>.html.var files by adding
the line:
#
# Alias /error/include/ "your/include/path/"
#
# which allows you to create your own set of files
by starting with the
# /var/www/error/include/ files and
# copying them to /your/include/path/, even on a
per-VirtualHost basis.
#
Alias /error/ "/var/www/error/"
#
# ErrorDocument 400
/error/HTTP_BAD_REQUEST.html.var
# ErrorDocument 401
/error/HTTP_UNAUTHORIZED.html.var
# ErrorDocument 403
/error/HTTP_FORBIDDEN.html.var
# ErrorDocument 404
/error/HTTP_NOT_FOUND.html.var
```



```
# ErrorDocument 405
/error/HTTP_METHOD_NOT_ALLOWED.html.var
ar
# ErrorDocument 408
/error/HTTP_REQUEST_TIME_OUT.html.var
# ErrorDocument 410
/error/HTTP_GONE.html.var
# ErrorDocument 411
/error/HTTP_LENGTH_REQUIRED.html.var
# ErrorDocument 412
/error/HTTP_PRECONDITION_FAILED.html.var
# ErrorDocument 413
/error/HTTP_REQUEST_ENTITY_TOO_LARGE.html.var
# ErrorDocument 414
/error/HTTP_REQUEST_URI_TOO_LARGE.html.var
# ErrorDocument 415
/error/HTTP_SERVICE_UNAVAILABLE.html.var
# ErrorDocument 500
/error/HTTP_INTERNAL_SERVER_ERROR.html.var
# ErrorDocument 501
/error/HTTP_NOT_IMPLEMENTED.html.var
# ErrorDocument 502
/error/HTTP_BAD_GATEWAY.html.var
# ErrorDocument 503
/error/HTTP_SERVICE_UNAVAILABLE.html.var
# ErrorDocument 506
/error/HTTP_VARIANT_ALSO_VARIES.html.var

#
# The following directives modify normal HTTP
response behavior to
# handle known problems with browser
implementations.
#

#
# The following directive disables redirects on
non-GET requests for
# a directory that does not include the trailing
slash. This fixes a
# problem with Microsoft WebFolders which
does not appropriately handle
# redirects for folders with DAV methods.
# Same deal with Apple's DAV filesystem and
Gnome VFS support for DAV.
#
# Allow server status reports, with the URL of
http://servername/server-status
# Change the ".your-domain.com" to match your
domain to enable.
#
<Location /server-status>
    SetHandler server-status
    Order deny,allow
    Deny from all
    Allow from 192.168.
</Location>

#
# Allow remote server configuration reports, with
the URL of
# http://servername/server-info (requires that
mod_info.c be loaded).
# Change the ".example.com" to match your
domain to enable.
#
#<Location /server-info>
# SetHandler server-info
# Order deny,allow
# Deny from all
# Allow from .example.com
```

```
</Location>

#
# Proxy Server directives. Uncomment the
following lines to
# enable the proxy server:
#
#<IfModule mod_proxy.c>
#ProxyRequests On
#
#<Proxy *>
# Order deny,allow
# Deny from all
# Allow from .example.com
#</Proxy>

#
# Enable/disable the handling of HTTP/1.1 "Via:"
headers.
# ("Full" adds the server version; "Block"
removes all outgoing Via: headers)
# Set to one of: Off | On | Full | Block
#
#ProxyVia On

#
# To enable a cache of proxied content,
uncomment the following lines.
# See http://httpd.apache.org/docs-
2.0/mod/mod_cache.html for more details.
#
#<IfModule mod_disk_cache.c>
# CacheEnable disk /
# CacheRoot "/var/cache/mod_proxy"
#</IfModule>
#

#</IfModule>
# End of proxy directives.

### Section 3: Virtual Hosts
#
# VirtualHost: If you want to maintain multiple
domains/hostnames on your
# machine you can setup VirtualHost containers
for them. Most configurations
# use only name-based virtual hosts so the
server doesn't need to worry about
# IP addresses. This is indicated by the asterisks
in the directives below.
#
# Please see the documentation at
# <URL:http://httpd.apache.org/docs-
2.0/vhosts/>
# for further details before you try to setup virtual
hosts.
#
# You may use the command line option '-S' to
verify your virtual host
# configuration.

#
# Use name-based virtual hosting.
#
#NameVirtualHost *:80

#
# VirtualHost example:
# Almost any Apache directive may go into a
VirtualHost container.
# The first VirtualHost section is used for
requests without a known
# server name.
#
#<VirtualHost *>
```

```
# ServerAdmin webmaster@dummy-
host.example.com
# DocumentRoot /www/docs/dummy-
host.example.com
# ServerName dummy-host.example.com
# ErrorLog logs/dummy-host.example.com-
error_log
# CustomLog logs/dummy-host.example.com-
access_log common
#</VirtualHost>

#
# For TPAPL
#
<Location /tpapl>
    SetHandler tpapl
    TpaPlConf /home/tpc/conf/tpapl.conf
</Location>
```

[Front-end application tunables]

```
.....
tpapl.conf
.....

[TPAPL_INFO]
Term_Base="1"
NumWarehouses="193600"
MaxUsers="1936000"
MaxTermOfClient="11000"
CONTROL_Flag="0"
LogPath="/home/tpc/log/userlog.log"

[SVRAPL_INFO]
LogPath="/home/tpc/log/DBDepend_Userlog.log"
```

```
.....
tnsnames.ora
.....

#
# Filename: Tnsnames.ora
#
extproc_connection_data =
(DESCRIPTION =
(AADDRESS = (PROTOCOL = IPC)(KEY =
tpcc))
(SDU=14600)
(CONNECT_DATA = (SERVICE_NAME =
tpcc))
)

tpcc =
(DESCRIPTION =
(AADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
(SDU=14600)
(CONNECT_DATA = (SERVICE_NAME =
tpcc))
)
```

[TP monitor tunables]

```
.....
ubbcnfig
.....
```

```
#
# ubbconfig : TUXEDO configuration file
# (WAREHOUSE BINED)
#

*RESOURCES
IPCKEY      211940
MASTER     SITE1
UID         500
GID         500
PERM        0660
MAXACCESSERS 1000
MAXSERVERS  100
MAXSERVICES 100
MAXGTT      0
MODEL       SHM
LDBAL       Y
OPTIONS     NO_AA,NO_XA

*MACHINES
cl007 LMID=SITE1
  APPDIR="/home/tpc/bin"
  TUXCONFIG="/home/tpc/conf/tuxconfig"
  TUXDIR="/usr/local/BEA/tuxedo8.1"
  ULOGPFX="/home/tpc/log/tuxedo.log"
  SICACHEENTRIESMAX="0"

*GROUPS
group1 LMID=SITE1 GRPNO=1

*SERVERS
DEFAULT: RESTART=Y MAXGEN=5
REPLYQ=N ROPERM=0660

tpccfmlw SRVGRP=group1 RQADDR=ware01
SRVID=1 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware02
SRVID=2 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware03
SRVID=3 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware04
SRVID=4 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware05
SRVID=5 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware06
SRVID=6 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware07
SRVID=7 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware08
SRVID=8 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware09
SRVID=9 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware10
SRVID=10 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"

*SERVICES
"OPSTUXSERVER" TRANTIME=0
SRVGRP=group1

*ROUTING

=====
=====
```

```
(configuration difference between cl007 and
cl008)
=====
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="1"
---
> Term_Base="242001"

.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))

[TP monitor tunables]
-----

.....
ubbconfig
.....

20c20
< cl007 LMID=SITE1
---
> cl008 LMID=SITE1

=====
=====
(configuration difference between cl007 and
cl009)
=====
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="1"
---
> Term_Base="726001"

.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_d)(Port= 1524))

[TP monitor tunables]
-----

.....
ubbconfig
.....

20c20
< cl007 LMID=SITE1
---
> cl010 LMID=SITE1

=====
=====
(configuration difference between cl007 and
cl011)
=====
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_c)(Port= 1523))
```

```
[TP monitor tunables]
-----

.....
ubbconfig
.....

20c20
< cl007 LMID=SITE1
---
> cl009 LMID=SITE1

=====
=====
(configuration difference between cl007 and
cl010)
=====
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="1"
---
> Term_Base="726001"

.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_d)(Port= 1524))

[TP monitor tunables]
-----

.....
ubbconfig
.....

20c20
< cl007 LMID=SITE1
---
> cl010 LMID=SITE1

=====
=====
(configuration difference between cl007 and
cl011)
=====
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....
```

```
2c2
< Term_Base="1"
...
> Term_Base="968001"
```

```
.....:
tnsnames.ora
.....:
```

```
13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
...
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_e)(Port= 1525))
```

[TP monitor tunables]

```
.....:
ubbcnfig
.....:
```

```
20c20
< cl007 LMID=SITE1
...
> cl011 LMID=SITE1
```

```
=====
=====
(configuration difference between cl007 and
cl012)
=====
=====
```

[Front-end application tunables]

```
.....:
tpapl.conf
.....:
```

```
2c2
< Term_Base="1"
...
> Term_Base="1210001"
```

```
.....:
tnsnames.ora
.....:
```

```
13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
...
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_f)(Port= 1526))
```

[TP monitor tunables]

```
.....:
ubbcnfig
.....:
```

```
20c20
< cl007 LMID=SITE1
...
> cl013 LMID=SITE1
```

```
> cl012 LMID=SITE1
```

```
=====
=====
(configuration difference between cl007 and
cl013)
=====
=====
```

[Front-end application tunables]

```
.....:
tpapl.conf
.....:
```

```
2c2
< Term_Base="1"
...
> Term_Base="1452001"
```

```
.....:
tnsnames.ora
.....:
```

```
13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
...
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_g)(Port= 1527))
```

[TP monitor tunables]

```
.....:
ubbcnfig
.....:
```

```
20c20
< cl007 LMID=SITE1
...
> cl013 LMID=SITE1
```

```
=====
=====
(configuration difference between cl007 and
cl014)
=====
=====
```

[Front-end application tunables]

```
.....:
tpapl.conf
.....:
```

```
2c2
< Term_Base="1"
...
> Term_Base="1694001"
```

```
.....:
tnsnames.ora
.....:
```

```
13c13
```

```
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
...
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_h)(Port= 1528))
```

[TP monitor tunables]

```
.....:
ubbcnfig
.....:
```

```
20c20
< cl007 LMID=SITE1
...
> cl014 LMID=SITE1
```

```
=====
=====
(configuration difference between cl007 and
cl017)
=====
=====
```

[Front-end application tunables]

```
.....:
tpapl.conf
.....:
```

```
2c2
< Term_Base="1"
...
> Term_Base="11001"
```

```
.....:
tnsnames.ora
.....:
```

[TP monitor tunables]

```
.....:
ubbcnfig
.....:
```

```
20c20
< cl007 LMID=SITE1
...
> cl017 LMID=SITE1
```

```
=====
=====
(configuration difference between cl007 and
cl018)
=====
=====
```

[Front-end application tunables]

```
.....:
tpapl.conf
.....:
```

```
2c2
< Term_Base="1"
...
> Term_Base="253001"
```

```
.....
tnsnames.ora
.....
```

```
13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
...
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
```

[TP monitor tunables]

```
.....
ubbcfg
.....
```

```
20c20
< cl007 LMID=SITE1
...
> cl018 LMID=SITE1
```

```
=====
=====
(configuration difference between cl007 and
cl019)
=====
=====
```

[Front-end application tunables]

```
.....
tpapl.conf
.....
```

```
2c2
< Term_Base="1"
...
> Term_Base="495001"
```

```
.....
tnsnames.ora
.....
```

```
13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
...
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_c)(Port= 1523))
```

[TP monitor tunables]

```
.....
ubbcfg
.....
```

```
20c20
< cl007 LMID=SITE1
...
> cl019 LMID=SITE1
```

```
=====
=====
(configuration difference between cl007 and
cl020)
=====
=====
```

[Front-end application tunables]

```
.....
tpapl.conf
.....
```

```
2c2
< Term_Base="1"
...
> Term_Base="737001"
```

```
.....
tnsnames.ora
.....
```

```
13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
...
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_d)(Port= 1524))
```

[TP monitor tunables]

```
.....
ubbcfg
.....
```

```
20c20
< cl007 LMID=SITE1
...
> cl020 LMID=SITE1
```

```
=====
=====
(configuration difference between cl007 and
cl021)
=====
=====
```

[Front-end application tunables]

```
.....
tpapl.conf
.....
```

```
2c2
< Term_Base="1"
...
> Term_Base="979001"
```

```
.....
tnsnames.ora
.....
```

```
13c13
```

```
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
...
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_e)(Port= 1525))
```

[TP monitor tunables]

```
.....
ubbcfg
.....
```

```
20c20
< cl007 LMID=SITE1
...
> cl021 LMID=SITE1
```

```
=====
=====
(configuration difference between cl007 and
cl022)
=====
=====
```

[Front-end application tunables]

```
.....
tpapl.conf
.....
```

```
2c2
< Term_Base="1"
...
> Term_Base="1221001"
```

```
.....
tnsnames.ora
.....
```

```
13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
...
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_f)(Port= 1526))
```

[TP monitor tunables]

```
.....
ubbcfg
.....
```

```
20c20
< cl007 LMID=SITE1
...
> cl022 LMID=SITE1
```

```
=====
=====
(configuration difference between cl007 and
cl025)
=====
=====
```

[Front-end application tunables]

```

-----
:
:
: tpapl.conf
:
:
2c2
< Term_Base="1"
---
> Term_Base="1463001"

:
:
: tnsnames.ora
:
:

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_g)(Port= 1527))

[TP monitor tunables]
-----

:
:
: ubbconfig
:
:

20c20
< cl007 LMID=SITE1
---
> cl025 LMID=SITE1

=====
(configuration difference between cl007 and
cl026)
=====

[Front-end application tunables]
-----

:
:
: tpapl.conf
:
:

2c2
< Term_Base="1"
---
> Term_Base="1705001"

:
:
: tnsnames.ora
:
:

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_h)(Port= 1528))

[TP monitor tunables]
-----

:
:
: ubbconfig

```

```

:
:
:
20c20
< cl007 LMID=SITE1
---
> cl026 LMID=SITE1

=====
(configuration difference between cl007 and
cl027)
=====

[Front-end application tunables]
-----

:
:
: tpapl.conf
:
:

2c2
< Term_Base="1"
---
> Term_Base="1232001"

:
:
: tnsnames.ora
:
:

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_f)(Port= 1526))

[TP monitor tunables]
-----

:
:
: ubbconfig
:
:

20c20
< cl007 LMID=SITE1
---
> cl027 LMID=SITE1

=====
(configuration difference between cl007 and
cl028)
=====

[Front-end application tunables]
-----

:
:
: tpapl.conf
:
:

2c2
< Term_Base="1"
---
> Term_Base="1474001"

```

```

:
:
: tnsnames.ora
:
:

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_g)(Port= 1527))

[TP monitor tunables]
-----

:
:
: ubbconfig
:
:

20c20
< cl007 LMID=SITE1
---
> cl028 LMID=SITE1

=====
(configuration difference between cl007 and
cl029)
=====

[Front-end application tunables]
-----

:
:
: tpapl.conf
:
:

2c2
< Term_Base="1"
---
> Term_Base="1716001"

:
:
: tnsnames.ora
:
:

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_h)(Port= 1528))

[TP monitor tunables]
-----

:
:
: ubbconfig
:
:

20c20
< cl007 LMID=SITE1
---
> cl029 LMID=SITE1

=====
(configuration difference between cl007 and
cl030)

```

```

=====
[Front-end application tunables]
-----
.....
tpapl.conf
.....

2c2
< Term_Base="1"
---
> Term_Base="1243001"

.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_f)(Port= 1526))

[TP monitor tunables]
-----
.....
ubbbconfig
.....

20c20
< cl007 LMID=SITE1
---
> cl030 LMID=SITE1

=====
(configuration difference between cl007 and
cl051)
=====

[Front-end application tunables]
-----
.....
tpapl.conf
.....

2c2
< Term_Base="1"
---
> Term_Base="1749001"

.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_h)(Port= 1528))

[TP monitor tunables]
-----
.....
ubbbconfig
.....

20c20
< cl007 LMID=SITE1
---
> cl052 LMID=SITE1

=====
(cl034 configuration)
=====

[Front-end application tunables]
-----
.....
tpapl.conf
.....

2c2
< Term_Base="1"
---
> Term_Base="1507001"

.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_g)(Port= 1527))

```

```

[TP monitor tunables]
-----
.....
ubbbconfig
.....

20c20
< cl007 LMID=SITE1
---
> cl051 LMID=SITE1

=====
(configuration difference between cl007 and
cl052)
=====

[Front-end application tunables]
-----
.....
tpapl.conf
.....

2c2
< Term_Base="1"
---
> Term_Base="1749001"

.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_h)(Port= 1528))

[TP monitor tunables]
-----
.....
ubbbconfig
.....

20c20
< cl007 LMID=SITE1
---
> cl052 LMID=SITE1

=====
(cl034 configuration)
=====

[OS tunables]
-----
.....
chkconfig
.....

sendmail 0:off 1:off 2:off 3:off 4:off
5:off 6:off

```

```

xinetd 0:off 1:off 2:off 3:on 4:on
5:on 6:off
named 0:off 1:off 2:off 3:off 4:off
5:off 6:off
smartd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
syslog 0:off 1:off 2:on 3:on 4:on
5:on 6:off
radiusd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
rwhod 0:off 1:off 2:off 3:off 4:off 5:off
6:off
mdmonitor 0:off 1:off 2:off 3:off 4:off
5:off 6:off
ypbind 0:off 1:off 2:off 3:off 4:off 5:off
6:off
nscd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
isdn 0:off 1:off 2:off 3:off 4:off 5:off
6:off
arptables_jf 0:off 1:off 2:off 3:off 4:off
5:off 6:off
lisa 0:off 1:off 2:off 3:off 4:off 5:off
6:off
rusersd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
dhcp6s 0:off 1:off 2:off 3:off 4:off
5:off 6:off
cyrus-imapd 0:off 1:off 2:off 3:off 4:off
5:off 6:off
winbind 0:off 1:off 2:off 3:off 4:off 5:off
6:off
vncserver 0:off 1:off 2:off 3:off 4:off
5:off 6:off
amd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
gpm 0:off 1:off 2:off 3:off 4:off 5:off
6:off
apmd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
bgpd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
readahead 0:off 1:off 2:off 3:off 4:off
5:on 6:off
ypxfrd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
mysqld 0:off 1:off 2:off 3:off 4:off
5:off 6:off
mailman 0:off 1:off 2:off 3:off 4:off
5:off 6:off
rpcgssd 0:off 1:off 2:off 3:off 4:off
5:off 6:off
innd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
pcmcia 0:off 1:off 2:off 3:off 4:off
5:off 6:off
mdmpd 0:off 1:off 2:off 3:off 4:off
5:off 6:off
autofs 0:off 1:off 2:off 3:off 4:off 5:off
6:off
rawdevices 0:off 1:off 2:off 3:on 4:on
5:on 6:off
ip6tables 0:off 1:off 2:off 3:off 4:off
5:off 6:off
nfs 0:off 1:off 2:off 3:off 4:off 5:off
6:off
bluetooth 0:off 1:off 2:off 3:off 4:off
5:off 6:off
netdump-server 0:off 1:off 2:off 3:off 4:off
5:off 6:off
ripngd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
iptables 0:off 1:off 2:on 3:on 4:on
5:on 6:off

```

```

NetworkManager 0:off 1:off 2:off 3:off 4:off
5:off 6:off
rpcsvcgssd 0:off 1:off 2:off 3:off 4:off
5:off 6:off
dhcrelay 0:off 1:off 2:off 3:off 4:off
5:off 6:off
bootparamd 0:off 1:off 2:off 3:off 4:off
5:off 6:off
squid 0:off 1:off 2:off 3:off 4:off 5:off
6:off
diskdump 0:off 1:off 2:off 3:off 4:off
5:off 6:off
haldaemon 0:off 1:off 2:off 3:off 4:off
5:off 6:off
cups 0:off 1:off 2:off 3:off 4:off 5:off
6:off
yppasswdd 0:off 1:off 2:off 3:off 4:off
5:off 6:off
saslauthd 0:off 1:off 2:off 3:off 4:off
5:off 6:off
netplugd 0:off 1:off 2:off 3:off 4:off
5:off 6:off
snmptrapd 0:off 1:off 2:off 3:off 4:off
5:off 6:off
canna 0:off 1:off 2:off 3:off 4:off 5:off
6:off
readahead_early 0:off 1:off 2:off 3:off 4:off
5:on 6:off
kprop 0:off 1:off 2:off 3:off 4:off 5:off
6:off
ripd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
irqbalance 0:off 1:off 2:off 3:on 4:on
5:on 6:off
messagebus 0:off 1:off 2:off 3:off 4:off
5:off 6:off
kudzu 0:off 1:off 2:off 3:off 4:off 5:off
6:off
ldap 0:off 1:off 2:off 3:off 4:off 5:off
6:off
microcode_ctl 0:off 1:off 2:off 3:off 4:off
5:off 6:off
network 0:off 1:off 2:on 3:on 4:on
5:on 6:off
rstatd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
dhcpcd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
portmap 0:off 1:off 2:off 3:off 4:off
5:off 6:off
lm_sensors 0:off 1:off 2:off 3:off 4:off
5:off 6:off
atd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
ntpd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
krb524 0:off 1:off 2:off 3:off 4:off 5:off
6:off
smb 0:off 1:off 2:off 3:off 4:off 5:off
6:off
httpd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
rpcidmapd 0:off 1:off 2:off 3:off 4:off
5:off 6:off
krb5kdc 0:off 1:off 2:off 3:off 4:off
5:off 6:off
anacron 0:off 1:off 2:off 3:off 4:off
5:off 6:off
ospf6d 0:off 1:off 2:off 3:off 4:off 5:off
6:off
cpuspeed 0:off 1:on 2:on 3:on 4:on
5:on 6:off
nfslock 0:off 1:off 2:off 3:off 4:off 5:off
6:off

```

```

dc_client 0:off 1:off 2:off 3:off 4:off 5:off
6:off
dovecot 0:off 1:off 2:off 3:off 4:off
5:off 6:off
sshd 0:off 1:off 2:on 3:on 4:on
5:on 6:off
psacct 0:off 1:off 2:off 3:off 4:off 5:off
6:off
hpoj 0:off 1:off 2:off 3:off 4:off 5:off
6:off
radvd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
ypserv 0:off 1:off 2:off 3:off 4:off 5:off
6:off
iim 0:off 1:off 2:off 3:off 4:off 5:off
6:off
netdump 0:off 1:off 2:off 3:off 4:off
5:off 6:off
ospfd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
cups-config-daemon 0:off 1:off 2:off 3:off
4:off 5:off 6:off
snmpd 0:off 1:off 2:off 3:off 4:off
5:off 6:off
acpid 0:off 1:off 2:off 3:off 4:off 5:off
6:off
dc_server 0:off 1:off 2:off 3:off 4:off
5:off 6:off
sysstat 0:off 1:on 2:off 3:off 4:off 5:off
6:off
kadmin 0:off 1:off 2:off 3:off 4:off
5:off 6:off
xfs 0:off 1:off 2:off 3:off 4:off 5:off
6:off
arpwatch 0:off 1:off 2:off 3:off 4:off
5:off 6:off
netfs 0:off 1:off 2:off 3:off 4:off 5:off
6:off
spamassassin 0:off 1:off 2:off 3:off 4:off
5:off 6:off
FreeWnn 0:off 1:off 2:off 3:off 4:off
5:off 6:off
lux 0:off 1:off 2:off 3:off 4:off 5:off
6:off
crond 0:off 1:off 2:on 3:on 4:on
5:on 6:off
vsftpd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
rhnsd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
irda 0:off 1:off 2:off 3:off 4:off 5:off
6:off
postgresql 0:off 1:off 2:off 3:off 4:off
5:off 6:off
zebra 0:off 1:off 2:off 3:off 4:off 5:off
6:off
xinetd based services:
talk: off
daytime: off
kshell: off
amandaidx: off
amanda: off
krb5-telnet: off
auth: on
telnet: on
finger: off
gssftp: off
amidxtape: off
dbskkd-cdb: off
ntalk: off
ktalk: off
rsync: off
time-udp: off
echo: off
echo-udp: off

```

```

chargen-udp: off
eklogin: off
klogin: off
rsh: on
cups-lpd: off
time: off
reexec: off
daytime-udp: off
rlogin: on
chargen: off
swat: off
tftp: off

.....
limits.conf
.....

# /etc/security/limits.conf
#
#Each line describes a limit for a user in the
form:
#
#<domain> <type> <item> <value>
#
#Where:
#<domain> can be:
# - an user name
# - a group name, with @group syntax
# - the wildcard *, for default entry
#
#<type> can have the two values:
# - "soft" for enforcing the soft limits
# - "hard" for enforcing hard limits
#
#<item> can be one of the following:
# - core - limits the core file size (KB)
# - data - max data size (KB)
# - fsize - maximum filesize (KB)
# - memlock - max locked-in-memory
address space (KB)
# - nofile - max number of open files
# - rss - max resident set size (KB)
# - stack - max stack size (KB)
# - cpu - max CPU time (MIN)
# - nproc - max number of processes
# - as - address space limit
# - maxlogins - max number of logins for this
user
# - priority - the priority to run user process
with
# - locks - max number of file locks the user
can hold
#
#<domain> <type> <item> <value>
#
#* soft core 0
#* hard rss 10000
#@student hard nproc 20
#@faculty soft nproc 20
#@faculty hard nproc 50
#ftp hard nproc 0
#@student - maxlogins 4
tpc - nofile 30000
tpc - nproc 30000

# End of file

.....
sysctl.conf
.....

```

```

# Kernel sysctl configuration file for Red Hat
Linux
#
# For binary values, 0 is disabled, 1 is enabled.
See sysctl(8) and
# sysctl.conf(5) for more details.

# Controls IP packet forwarding
net.ipv4.ip_forward = 0

# Controls source route verification
net.ipv4.conf.default.rp_filter = 1

# Controls the System Request debugging
functionality of the kernel
kernel.sysrq = 0

# Controls whether core dumps will append the
PID to the core filename.
# Useful for debugging multi-threaded
applications.
kernel.core_uses_pid = 1

# Change filedescriptor
fs.file-max = 30000

# Change Message queue
kernel.msgmni = 30000
kernel.msgmnb = 1536000

# Change Max process
kernel.threads-max = 30000

# Change Semaphore
kernel.sem = 3000 384000 32 128

# Change TCP/IP backlog
net.ipv4.tcp_max_syn_backlog = 4096

[HTTP server tunables]
-----

.....
apache_cl_start.sh
.....

#/bin/sh
export
LD_LIBRARY_PATH=$ORACLE_HOME/srvmlib:$ORACLE_HOME/lib64:$ORACLE_HOME/lib:
/usr/lib:$ORACLE_HOME/rdbms/lib:$ORACLE_
HOME/network/lib:$TUXDIR/lib

ulimit -u 30000
ulimit -s 1536

#/sbin/swapon -a

# For 3tier tune
SVRAPL=`ps -e | grep tpccfmlw | awk '{print $1}'
/usr/bin/renice -20 -p ${SVRAPL}`

#rm -f /home/tpc/sar.tmp
/home/tpc/sar.`hostname`
#/usr/lib/sa/sadc 5 > /home/tpc/sar.tmp &
# For 3tier tune

apachectl start

.....
httpd.conf

```

```

.....
#
# Based upon the NCSA server configuration
files originally by Rob McCool.
#
# This is the main Apache server configuration
file. It contains the
# configuration directives that give the server its
instructions.
# See <URL:http://httpd.apache.org/docs-2.0/>
for detailed information about
# the directives.
#
# Do NOT simply read the instructions in here
without understanding
# what they do. They're here only as hints or
reminders. If you are unsure
# consult the online docs. You have been
warned.
#
# The configuration directives are grouped into
three basic sections:
# 1. Directives that control the operation of the
Apache server process as a
# whole (the 'global environment').
# 2. Directives that define the parameters of the
'main' or 'default' server,
# which responds to requests that aren't
handled by a virtual host.
# These directives also provide default values
for the settings
# of all virtual hosts.
# 3. Settings for virtual hosts, which allow Web
requests to be sent to
# different IP addresses or hostnames and
have them handled by the
# same Apache server process.
#
# Configuration and logfile names: If the
filenames you specify for many
# of the server's control files begin with "/" (or
"drive:/" for Win32), the
# server will use that explicit path. If the
filenames do "not" begin
# with "/", the value of ServerRoot is prepended -
- so "logs/foo.log"
# with ServerRoot set to "/etc/httpd" will be
interpreted by the
# server as "/etc/httpd/logs/foo.log".
#
### Section 1: Global Environment
#
# The directives in this section affect the overall
operation of Apache,
# such as the number of concurrent requests it
can handle or where it
# can find its configuration files.
#
# Don't give away too much information about all
the subcomponents
# we are running. Comment out this line if you
don't mind remote sites
# finding out what major optional modules you
are running
#ServerTokens OS
ServerTokens Producty

#
# ServerRoot: The top of the directory tree under
which the server's
# configuration, error, and log files are kept.

```

```

#
# NOTE! If you intend to place this on an NFS
(or otherwise network)
# mounted filesystem then please read the
LockFile documentation
# (available at
<URL:http://httpd.apache.org/docs-
2.0/mod/core.html#lockfile>);
# you will save yourself a lot of trouble.
#
# Do NOT add a slash at the end of the directory
path.
#
ServerRoot "/etc/httpd"

#
# ScoreBoardFile: File used to store internal
server process information.
# If unspecified (the default), the scoreboard will
be stored in an
# anonymous shared memory segment, and will
be unavailable to third-party
# applications.
# If specified, ensure that no two invocations of
Apache share the same
# scoreboard file. The scoreboard file MUST BE
STORED ON A LOCAL DISK.
#
#ScoreBoardFile run/httpd.scoreboard

#
# PidFile: The file in which the server should
record its process
# identification number when it starts.
#
PidFile run/httpd.pid

#
# Timeout: The number of seconds before
receives and sends time out.
#
#Timeout 300
Timeout 999

#
# KeepAlive: Whether or not to allow persistent
connections (more than
# one request per connection). Set to "Off" to
deactivate.
#
#KeepAlive Off
KeepAlive On

#
# MaxKeepAliveRequests: The maximum
number of requests to allow
# during a persistent connection. Set to 0 to
allow an unlimited amount.
# We recommend you leave this number high,
for maximum performance.
#
#MaxKeepAliveRequests 100
MaxKeepAliveRequests 0

#
# KeepAliveTimeout: Number of seconds to wait
for the next request from the
# same client on the same connection.
#
#KeepAliveTimeout 15
KeepAliveTimeout 999

##
## Server-Pool Size Regulation (MPM specific)

```



```

##
# prefork MPM
# StartServers: number of server processes to
start
# MinSpareServers: minimum number of server
processes which are kept spare
# MaxSpareServers: maximum number of server
processes which are kept spare
# MaxClients: maximum number of server
processes allowed to start
# MaxRequestsPerChild: maximum number of
requests a server process serves
<IfModule prefork.c>
StartServers      8
MinSpareServers   5
MaxSpareServers  20
MaxClients       150
MaxRequestsPerChild 1000
</IfModule>

# worker MPM
# StartServers: initial number of server
processes to start
# MaxClients: maximum number of simultaneous
client connections
# MinSpareThreads: minimum number of worker
threads which are kept spare
# MaxSpareThreads: maximum number of
worker threads which are kept spare
# ThreadsPerChild: constant number of worker
threads in each server process
# MaxRequestsPerChild: maximum number of
requests a server process serves
<IfModule worker.c>

StartServers      45
ServerLimit       45
ThreadLimit       500
MaxClients        22500
MinSpareThreads   1
MaxSpareThreads  22500
ThreadsPerChild   500
MaxRequestsPerChild 0

#
#
# To reduce memory usage in the worker MPM,
the thread guard page
#
# To reduce memory usage in the worker MPM,
the thread guard page
# can be disabled, at the expense of some
protection against stack
# overflow.
#
#ThreadGuardArea off

</IfModule>

#
# Listen: Allows you to bind Apache to specific
IP addresses and/or
# ports, in addition to the default. See also the
<VirtualHost>
# directive.
#
# Change this to Listen on specific IP addresses
as shown below to
# prevent Apache from glomming onto all bound
IP addresses (0.0.0.0)
# e.g. "Listen 12.34.56.78:80"
#
# To allow connections to IPv6 addresses add
"Listen [::]:80"

#
Listen 0.0.0.0:80

#
# Dynamic Shared Object (DSO) Support
#

# To be able to use the functionality of a module
which was built as a DSO you
# have to place corresponding 'LoadModule'
lines at this location so the
# directives contained in it are actually available
_before_ they are used.
# Statically compiled modules (those listed by
`httpd -l') do not need
# to be loaded here.
#
# Example:
# LoadModule foo_module modules/mod_foo.so
#
LoadModule tpmpl_module
modules/mod_tpmpl.so
LoadModule access_module
modules/mod_access.so
LoadModule status_module
modules/mod_status.so
LoadModule alias_module
modules/mod_alias.so
LoadModule cgi_module modules/mod_cgi.so

#
# Load config files from the config directory
"/etc/httpd/conf.d".
#
#Include conf.d/*.conf

#
# ExtendedStatus controls whether Apache will
generate "full" status
# information (ExtendedStatus On) or just basic
information (ExtendedStatus
# Off) when the "server-status" handler is called.
The default is Off.
#
#ExtendedStatus On

### Section 2: 'Main' server configuration
#
# The directives in this section set up the values
used by the 'main'
# server, which responds to any requests that
aren't handled by a
# <VirtualHost> definition. These values also
provide defaults for
# any <VirtualHost> containers you may define
later in the file.
#
# All of these directives may appear inside
<VirtualHost> containers,
# in which case these default settings will be
overridden for the
# virtual host being defined.
#
#
# If you wish httpd to run as a different user or
group, you must run
# httpd as root initially and it will switch.
#
# User/Group: The name (or #number) of the
user/group to run httpd as.
# . On SCO (ODT 3) use "User nouser" and
"Group nogroup".

# . On HP-UX you may not be able to use
shared memory as nobody, and the
# suggested workaround is to create a user
www and use that user.
# NOTE that some kernels refuse to
setgid(Group) or semctl(IPC_SET)
# when the value of (unsigned)Group is above
60000;
# don't use Group #-1 on these systems!
#
#User apache
#Group apache
User tpc
Group tpc

#
# ServerAdmin: Your address, where problems
with the server should be
# e-mailed. This address appears on some
server-generated pages, such
# as error documents. e.g. admin@your-
domain.com
#
ServerAdmin root@localhost

#
# ServerName gives the name and port that the
server uses to identify itself.
# This can often be determined automatically,
but we recommend you specify
# it explicitly to prevent problems during startup.
#
# If this is not set to valid DNS name for your
host, server-generated
# redirections will not work. See also the
UseCanonicalName directive.
#
# If your host doesn't have a registered DNS
name, enter its IP address here.
# You will have to access it by its address
anyway, and this will make
# redirections work in a sensible way.
#
#ServerName new.host.name:80
ServerName tpcserver:80

#
# UseCanonicalName: Determines how Apache
constructs self-referencing
# URLs and the SERVER_NAME and
SERVER_PORT variables.
# When set "Off", Apache will use the Hostname
and Port supplied
# by the client. When set "On", Apache will use
the value of the
# ServerName directive.
#
UseCanonicalName Off

#
# DocumentRoot: The directory out of which you
will serve your
# documents. By default, all requests are taken
from this directory, but
# symbolic links and aliases may be used to
point to other locations.
#
#DocumentRoot "/var/www/html"

#
# Each directory to which Apache has access
can be configured with respect
# to which services and features are allowed
and/or disabled in that

```

```

# directory (and its subdirectories).
#
# First, we configure the "default" to be a very
restrictive set of
# features.
#
#<Directory />
# Options FollowSymLinks
# AllowOverride None
#</Directory>

#
# Note that from this point forward you must
specifically allow
# particular features to be enabled - so if
something's not working as
# you might expect, make sure that you have
specifically enabled it
# below.
#
#
# UserDir: The name of the directory that is
appended onto a user's home
# directory if a ~user request is received.
#
# The path to the end user account 'public_html'
directory must be
# accessible to the webserver userid. This
usually means that ~userid
# must have permissions of 711,
~userid/public_html must have permissions
# of 755, and documents contained therein must
be world-readable.
# Otherwise, the client will only receive a "403
Forbidden" message.
#
# See also:
http://httpd.apache.org/docs/misc/FAQ.html#forb
idden
#
#<IfModule mod_userdir.c>
#
# UserDir is disabled by default since it can
confirm the presence
# of a username on the system (depending on
home directory
# permissions).
#
# UserDir disable

#
# To enable requests to /~user/ to serve the
user's public_html
# directory, remove the "UserDir disable" line
above, and uncomment
# the following line instead:
#
#UserDir public_html

#</IfModule>

#
# Control access to UserDir directories. The
following is an example
# for a site where these directories are restricted
to read-only.
#
#<Directory /home/*>public_html>
# AllowOverride FileInfo AuthConfig Limit
# Options MultiViews Indexes
SymLinksIfOwnerMatch IncludesNoExec
# <Limit GET POST OPTIONS>
# Order allow,deny
# Allow from all

```

```

# </Limit>
# <LimitExcept GET POST OPTIONS>
# Order deny,allow
# Deny from all
# </LimitExcept>
#</Directory>

#
# DirectoryIndex: sets the file that Apache will
serve if a directory
# is requested.
#
# The index.html.var file (a type-map) is used to
deliver content-
# negotiated documents. The MultiViews Option
can be used for the
# same purpose, but it is much slower.
#
#
# AccessFileName: The name of the file to look
for in each directory
# for additional configuration directives. See
also the AllowOverride
# directive.
#
AccessFileName .htaccess

#
# The following lines prevent .htaccess
and .htpasswd files from being
# viewed by Web clients.
#
#
# TypesConfig describes where the mime.types
file (or equivalent) is
# to be found.
#
#
# DefaultType is the default MIME type the
server will use for a document
# if it cannot otherwise determine one, such as
from filename extensions.
# If your server contains mostly text or HTML
documents, "text/plain" is
# a good value. If most of your content is binary,
such as applications
# or images, you may want to use
"application/octet-stream" instead to
# keep browsers from trying to display binary
files as though they are
# text.
#
DefaultType text/plain

#
# The mod_mime_magic module allows the
server to use various hints from the
# contents of the file itself to determine its type.
The MIMEMagicFile
# directive tells the module where the hint
definitions are located.
#
#<IfModule mod_mime_magic.c>
## MIMEMagicFile /usr/share/magic.mime
# MIMEMagicFile conf/magic
#</IfModule>

#
# HostnameLookups: Log the names of clients
or just their IP addresses
# e.g., www.apache.org (on) or 204.62.129.132
(off).

```

```

# The default is off because it'd be overall better
for the net if people
# had to knowingly turn this feature on, since
enabling it means that
# each client request will result in AT LEAST one
lookup request to the
# nameserver.
#
HostnameLookups Off

#
# EnableMMAP: Control whether memory-
mapping is used to deliver
# files (assuming that the underlying OS
supports it).
# The default is on; turn this off if you serve from
NFS-mounted
# filesystems. On some systems, turning it off
(regardless of
# filesystem) can improve performance; for
details, please see
# http://httpd.apache.org/docs-
2.0/mod/core.html#enablmmmap
#
#EnableMMAP off

#
# EnableSendfile: Control whether the sendfile
kernel support is
# used to deliver files (assuming that the OS
supports it).
# The default is on; turn this off if you serve from
NFS-mounted
# filesystems. Please see
# http://httpd.apache.org/docs-
2.0/mod/core.html#enablesendfile
#
#EnableSendfile off
#

#
# ErrorLog: The location of the error log file.
# If you do not specify an ErrorLog directive
within a <VirtualHost>
# container, error messages relating to that
virtual host will be
# logged here. If you *do* define an error logfile
for a <VirtualHost>
# container, that host's errors will be logged
there and not here.
#
ErrorLog logs/error_log

#
# LogLevel: Control the number of messages
logged to the error_log.
# Possible values include: debug, info, notice,
warn, error, crit,
# alert, emerg.
#
LogLevel warn

#
# The following directives define some format
nicknames for use with
# a CustomLog directive (see below).
#
#
# The location and format of the access logfile
(Common Logfile Format).
# If you do not define any access logfiles within
a <VirtualHost>
# container, they will be logged here.
Contrariwise, if you *do*

```

```

# define per-<VirtualHost> access logfiles,
transactions will be
# logged therein and *not* in this file.
#
# CustomLog logs/access_log common
#CustomLog logs/access_log combined

#
# If you would like to have agent and referer
logfiles, uncomment the
# following directives.
#
#CustomLog logs/referer_log referer
#CustomLog logs/agent_log agent

#
# If you prefer a single logfile with access, agent,
and referer information
# (Combined Logfile Format) you can use the
following directive.
#
#CustomLog logs/access_log combined

#
# Optionally add a line containing the server
version and virtual host
# name to server-generated pages (error
documents, FTP directory listings,
# mod_status and mod_info output etc., but not
CGI generated documents).
# Set to "EMail" to also include a mailto: link to
the ServerAdmin.
# Set to one of: On | Off | EMail
#
#ServerSignature On
ServerSignature Off

#
# Aliases: Add here as many aliases as you
need (with no limit). The format is
# Alias fakename realname
#
# Note that if you include a trailing / on fakename
then the server will
# require it to be present in the URL. So "/icons"
isn't aliased in this
# example, only "/icons/". If the fakename is
slash-terminated, then the
# realname must also be slash terminated, and if
the fakename omits the
# trailing slash, the realname must also omit it.
#
# We include the /icons/ alias for FancyIndexed
directory listings. If you
# do not use FancyIndexing, you may comment
this out.
#

#
# This should be changed to the
ServerRoot/manual/. The alias provides
# the manual, even if you choose to move your
DocumentRoot. You may comment
# this out if you do not care for the
documentation.
#
#<IfModule mod_dav_fs.c>
# # Location of the WebDAV lock database.
# DAVLockDB /var/lib/dav/lockdb
#</IfModule>

#
# ScriptAlias: This controls which directories
contain server scripts.

```

```

# ScriptAliases are essentially the same as
Aliases, except that
# documents in the realname directory are
treated as applications and
# run by the server when requested rather than
as documents sent to the client.
# The same rules about trailing "/" apply to
ScriptAlias directives as to
# Alias.
#
#ScriptAlias /cgi-bin/ "/var/www/cgi-bin/"
ScriptAlias /cgi-bin/ "/home/tpc/tool/bin/"

#
# "/var/www/cgi-bin" should be changed to
whatever your ScriptAliased
# CGI directory exists, if you have that
configured.
#
<Directory "/var/www/cgi-bin">
    AllowOverride None
    Options None
    Order allow,deny
    Allow from all
</Directory>

#
# Redirect allows you to tell clients about
documents which used to exist in
# your server's namespace, but do not anymore.
This allows you to tell the
# clients where to look for the relocated
document.
# Example:
# Redirect permanent /foo
http://www.example.com/bar

#
# Directives controlling the display of server-
generated directory listings.
#
#
# FancyIndexing is whether you want fancy
directory indexing or standard.
# VersionSort is whether files containing version
numbers should be
# compared in the natural way, so that `apache-
1.3.9.tar' is placed before
# `apache-1.3.12.tar'.
#
#
# AddIcon* directives tell the server which icon
to show for different
# files or filename extensions. These are only
displayed for
# FancyIndexed directories.
#
#
# DefaultIcon is which icon to show for files
which do not have an icon
# explicitly set.
#
#
# AddDescription allows you to place a short
description after a file in
# server-generated indexes. These are only
displayed for FancyIndexed
# directories.
# Format: AddDescription "description" filename
#

```

```

#AddDescription "GZIP compressed
document" .gz
#AddDescription "tar archive" .tar
#AddDescription "GZIP compressed tar
archive" .tgz

#
# ReadmeName is the name of the README file
the server will look for by
# default, and append to directory listings.
#
# HeaderName is the name of a file which
should be prepended to
# directory indexes.

#
# IndexIgnore is a set of filenames which
directory indexing should ignore
# and not include in the listing. Shell-style
wildcarding is permitted.
#
#
# AddEncoding allows you to have certain
browsers (Mosaic/X 2.1+) uncompress
# information on the fly. Note: Not all browsers
support this.
# Despite the name similarity, the following Add*
directives have nothing
# to do with the FancyIndexing customization
directives above.
#
#
# DefaultLanguage and AddLanguage allows
you to specify the language of
# a document. You can then use content
negotiation to give a browser a
# file in a language the user can understand.
#
# Specify a default language. This means that all
data
# going out without a specific language tag (see
below) will
# be marked with this one. You probably do NOT
want to set
# this unless you are sure it is correct for all
cases.
#
# * It is generally better to not mark a page as
# * being a certain language than marking it with
the wrong
# * language!
#
# DefaultLanguage nl
#
# Note 1: The suffix does not have to be the
same as the language
# keyword --- those with documents in Polish
(whose net-standard
# language code is pl) may wish to use
"AddLanguage pl .po" to
# avoid the ambiguity with the common suffix for
perl scripts.
#
# Note 2: The example entries below illustrate
that in some cases
# the two character 'Language' abbreviation is
not identical to
# the two character 'Country' code for its country,
# E.g. 'Danmark/dk' versus 'Danish/da'.
#
# Note 3: In the case of 'ltz' we violate the RFC
by using a three char

```

```
# specifier. There is 'work in progress' to fix this
and get
# the reference data for rfc1766 cleaned up.
#
# Danish (da) - Dutch (nl) - English (en) -
Estonian (et)
# French (fr) - German (de) - Greek-Modern (el)
# Italian (it) - Norwegian (no) - Norwegian
Nynorsk (nn) - Korean (ko)
# Portugese (pt) - Luxembourgish* (ltz)
# Spanish (es) - Swedish (sv) - Catalan (ca) -
Czech(cs)
# Polish (pl) - Brazilian Portuguese (pt-br) -
Japanese (ja)
# Russian (ru) - Croatian (hr)
#
#
# LanguagePriority allows you to give
precedence to some languages
# in case of a tie during content negotiation.
#
# Just list the languages in decreasing order of
preference. We have
# more or less alphabetized them here. You
probably want to change this.
#
#
# ForceLanguagePriority allows you to serve a
result page rather than
# MULTIPLE CHOICES (Prefer) [in case of a tie]
or NOT ACCEPTABLE (Fallback)
# [in case no accepted languages matched the
available variants]
#
#
# Specify a default charset for all pages sent out.
This is
# always a good idea and opens the door for
future internationalisation
# of your web site, should you ever want it.
Specifying it as
# a default does little harm; as the standard
dictates that a page
# is in iso-8859-1 (latin1) unless specified
otherwise i.e. you
# are merely stating the obvious. There are also
some security
# reasons in browsers, related to javascript and
URL parsing
# which encourage you to always set a default
charset.
#
AddDefaultCharset UTF-8
#
#
# Commonly used filename extensions to
character sets. You probably
# want to avoid clashes with the language
extensions, unless you
# are good at carefully testing your setup after
each change.
# See
http://www.iana.org/assignments/character-sets
for the
# official list of charset names and their
respective RFCs
#
#
# AddType allows you to add to or override the
MIME configuration
# file mime.types for specific file types.
```

```
#
#
# AddHandler allows you to map certain file
extensions to "handlers":
# actions unrelated to filetype. These can be
either built into the server
# or added with the Action directive (see below)
#
# To use CGI scripts outside of ScriptAliased
directories:
# (You will also need to add "ExecCGI" to the
"Options" directive.)
#
AddHandler cgi-script .cgi
#
#
# For files that include their own HTTP headers:
#
AddHandler send-as-is asis
#
#
# For server-parsed imagemap files:
#
#
# For type maps (negotiated resources):
# (This is enabled by default to allow the Apache
"It Worked" page
# to be distributed in multiple languages.)
#
# Filters allow you to process content before it is
sent to the client.
#
# To parse .shtml files for server-side includes
(SSIs):
# (You will also need to add "Includes" to the
"Options" directive.)
#
#
# Action lets you define media types that will
execute a script whenever
# a matching file is called. This eliminates the
need for repeated URL
# pathnames for oft-used CGI file processors.
# Format: Action media/type /cgi-script/location
# Format: Action handler-name /cgi-
script/location
#
#
# Customizable error responses come in three
flavors:
# 1) plain text 2) local redirects 3) external
redirects
#
# Some examples:
#ErrorDocument 500 "The server made a boo
boo."
#ErrorDocument 404 /missing.html
#ErrorDocument 404 "/cgi-
bin/missing_handler.pl"
#ErrorDocument 402
http://www.example.com/subscription_info.html
#
#
# Putting this all together, we can
Internationalize error responses.
#
# We use Alias to redirect any
/error/HTTP_<error>.html.var response to
```

```
# our collection of by-error message multi-
language collections. We use
# includes to substitute the appropriate text.
#
# You can modify the messages' appearance
without changing any of the
# default HTTP_<error>.html.var files by adding
the line;
#
# Alias /error/include/ "/your/include/path/"
#
# which allows you to create your own set of files
by starting with the
# /var/www/error/include/ files and
# copying them to /your/include/path/, even on a
per-VirtualHost basis.
#
Alias /error/ "/var/www/error/"
#
# ErrorDocument 400
/error/HTTP_BAD_REQUEST.html.var
# ErrorDocument 401
/error/HTTP_UNAUTHORIZED.html.var
# ErrorDocument 403
/error/HTTP_FORBIDDEN.html.var
# ErrorDocument 404
/error/HTTP_NOT_FOUND.html.var
# ErrorDocument 405
/error/HTTP_METHOD_NOT_ALLOWED.html.v
ar
# ErrorDocument 408
/error/HTTP_REQUEST_TIME_OUT.html.var
# ErrorDocument 410
/error/HTTP_GONE.html.var
# ErrorDocument 411
/error/HTTP_LENGTH_REQUIRED.html.var
# ErrorDocument 412
/error/HTTP_PRECONDITION_FAILED.html.var
# ErrorDocument 413
/error/HTTP_REQUEST_ENTITY_TOO_LARGE
.html.var
# ErrorDocument 414
/error/HTTP_REQUEST_URI_TOO_LARGE.htm
l.var
# ErrorDocument 415
/error/HTTP_SERVICE_UNAVAILABLE.html.var
# ErrorDocument 500
/error/HTTP_INTERNAL_SERVER_ERROR.htm
l.var
# ErrorDocument 501
/error/HTTP_NOT_IMPLEMENTED.html.var
# ErrorDocument 502
/error/HTTP_BAD_GATEWAY.html.var
# ErrorDocument 503
/error/HTTP_SERVICE_UNAVAILABLE.html.var
# ErrorDocument 506
/error/HTTP_VARIANT_ALSO_VARIES.html.var
#
# The following directives modify normal HTTP
response behavior to
# handle known problems with browser
implementations.
#
#
# The following directive disables redirects on
non-GET requests for
# a directory that does not include the trailing
slash. This fixes a
# problem with Microsoft WebFolders which
does not appropriately handle
```

```
# redirects for folders with DAV methods.
# Same deal with Apple's DAV filesystem and
# Gnome VFS support for DAV.
#
# Allow server status reports, with the URL of
# http://servername/server-status
# Change the ".your-domain.com" to match your
# domain to enable.
#
<Location /server-status>
    SetHandler server-status
    Order deny,allow
    Deny from all
    Allow from 192.168.
</Location>

#
# Allow remote server configuration reports, with
# the URL of
# http://servername/server-info (requires that
# mod_info.c be loaded).
# Change the ".example.com" to match your
# domain to enable.
#
#<Location /server-info>
#   SetHandler server-info
#   Order deny,allow
#   Deny from all
#   Allow from .example.com
#</Location>

#
# Proxy Server directives. Uncomment the
# following lines to
# enable the proxy server:
#
#<IfModule mod_proxy.c>
#ProxyRequests On
#
#<Proxy *>
#   Order deny,allow
#   Deny from all
#   Allow from .example.com
#</Proxy>

#
# Enable/disable the handling of HTTP/1.1 "Via:"
# headers.
# ("Full" adds the server version; "Block"
# removes all outgoing Via: headers)
# Set to one of: Off | On | Full | Block
#
#ProxyVia On

#
# To enable a cache of proxied content,
# uncomment the following lines.
# See http://httpd.apache.org/docs-
# 2.0/mod/mod_cache.html for more details.
#
#<IfModule mod_disk_cache.c>
#   CacheEnable disk /
#   CacheRoot "/var/cache/mod_proxy"
#</IfModule>
#

#</IfModule>
# End of proxy directives.

### Section 3: Virtual Hosts
#
# VirtualHost: If you want to maintain multiple
# domains/hostnames on your
# machine you can setup VirtualHost containers
# for them. Most configurations
```

```
# use only name-based virtual hosts so the
# server doesn't need to worry about
# IP addresses. This is indicated by the asterisks
# in the directives below.
#
# Please see the documentation at
# <URL:http://httpd.apache.org/docs-
# 2.0/vhosts/>
# for further details before you try to setup virtual
# hosts.
#
# You may use the command line option '-S' to
# verify your virtual host
# configuration.

#
# Use name-based virtual hosting.
#
#NameVirtualHost *:80

#
# VirtualHost example:
# Almost any Apache directive may go into a
# VirtualHost container.
# The first VirtualHost section is used for
# requests without a known
# server name.
#
#<VirtualHost *>
#   ServerAdmin webmaster@dummy-
#   host.example.com
#   DocumentRoot /www/docs/dummy-
#   host.example.com
#   ServerName dummy-host.example.com
#   ErrorLog logs/dummy-host.example.com-
#   error_log
#   CustomLog logs/dummy-host.example.com-
#   access_log common
#</VirtualHost>

#
# For TPAPL
#
<Location /tpapl>
    SetHandler tpapl
    TPAPlConf /home/tpc/conf/tpapl.conf
</Location>

[Front-end application tunables]
-----
:
:
:
tpapl.conf
:
:

[TPAPL_INFO]
Term_Base="22001"
NumWarehouses="193600"
MaxUsers="1936000"
MaxTerm of Client="22000"
CONTROL_Flag="0"
LogPath="/home/tpc/log/userlog.log"

[SVRAPL_INFO]
LogPath="/home/tpc/log/DBDepend_Userlog.log"

:
:
:
tnsnames.ora
:
:
:
```

```
#
# Filename: Tnsnames.ora
#
# extproc_connection_data =
#   (DESCRIPTION =
#     (ADDRESS = (PROTOCOL = IPC)(KEY =
# tpcc))
#     (SDU=14600)
#     (CONNECT_DATA = (SERVICE_NAME =
# tpcc))
#   )
#
# tpcc =
#   (DESCRIPTION =
#     (ADDRESS = (PROTOCOL = TCP)(Host=
# pqtpc_a)(Port= 1521))
#     (SDU=14600)
#     (CONNECT_DATA = (SERVICE_NAME =
# tpcc))
#   )

[TP monitor tunables]
-----
:
:
:
ubbcfg
:
:

#
# ubbcfg : TUXEDO configuration file
# (WAREHOUSE BINED)
#

*RESOURCES
IPCKEY      211940
MASTER     SITE1
UID         500
GID         500
PERM       0660
MAXACCESSERS 1000
MAXSERVERS 100
MAXSERVICES 100
MAXGTT      0
MODEL       SHM
LDBAL       Y
OPTIONS     NO_AA,NO_XA

*MACHINES
c1034 LMID=SITE1
      APPDIR="/home/tpc/bin"
      TUXCONFIG="/home/tpc/conf/tuxconfig"
      TUXDIR="/usr/local/BEA/tuxedo8.1"
      ULOGPFX="/home/tpc/log/tuxedo.log"
      SICACHEENTRIESMAX="0"

*GROUPS
group1 LMID=SITE1 GRPNO=1

*SERVERS
DEFAULT:  RESTART=Y MAXGEN=5
REPLYQ=N RQPERM=0660

tpccfmlw SRVGRP=group1 ROADDR=ware01
SRVID=1 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 ROADDR=ware02
SRVID=2 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 ROADDR=ware03
SRVID=3 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 ROADDR=ware04
SRVID=4 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
```

```
tpccfmlw SRVGRP=group1 ROADDR=ware05
SRVID=5 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 ROADDR=ware06
SRVID=6 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 ROADDR=ware07
SRVID=7 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 ROADDR=ware08
SRVID=8 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 ROADDR=ware09
SRVID=9 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 ROADDR=ware10
SRVID=10 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 ROADDR=ware11
SRVID=11 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 ROADDR=ware12
SRVID=12 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 ROADDR=ware13
SRVID=13 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 ROADDR=ware14
SRVID=14 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 ROADDR=ware15
SRVID=15 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 ROADDR=ware16
SRVID=16 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 ROADDR=ware17
SRVID=17 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 ROADDR=ware18
SRVID=18 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 ROADDR=ware19
SRVID=19 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 ROADDR=ware20
SRVID=20 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
```

```
*SERVICES
"OPSTUXSERVER" TRANTIME=0
SRVGRP=group1
```

```
*ROUTING
```

```
=====
(configuration difference between cl034 and
cl035)
=====
```

```
[Front-end application tunables]
-----
```

```
.....
tpapl.conf
.....
```

```
2c2
< Term_Base="22001"
---
> Term_Base="264001"
```

```
.....
tnsnames.ora
.....
```

```
13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
```

```
[TP monitor tunables]
-----
```

```
.....
ubbcnfig
.....
```

```
20c20
< cl034 LMID=SITE1
---
> cl035 LMID=SITE1
```

```
=====
=====
(configuration difference between cl034 and
cl036)
=====
```

```
[Front-end application tunables]
-----
```

```
.....
tpapl.conf
.....
```

```
2c2
< Term_Base="22001"
---
> Term_Base="506001"
```

```
.....
tnsnames.ora
.....
```

```
13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_c)(Port= 1523))
```

```
[TP monitor tunables]
-----
```

```
.....
ubbcnfig
.....
```

```
20c20
< cl034 LMID=SITE1
---
> cl036 LMID=SITE1
```

```
=====
=====
```

```
(configuration difference between cl034 and
cl037)
=====
```

```
[Front-end application tunables]
-----
```

```
.....
tpapl.conf
.....
```

```
2c2
< Term_Base="22001"
---
> Term_Base="748001"
```

```
.....
tnsnames.ora
.....
```

```
13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_d)(Port= 1524))
```

```
[TP monitor tunables]
-----
```

```
.....
ubbcnfig
.....
```

```
20c20
< cl034 LMID=SITE1
---
> cl037 LMID=SITE1
```

```
=====
=====
(configuration difference between cl034 and
cl038)
=====
```

```
[Front-end application tunables]
-----
```

```
.....
tpapl.conf
.....
```

```
2c2
< Term_Base="22001"
---
> Term_Base="990001"
```

```
.....
tnsnames.ora
.....
```

```
13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_e)(Port= 1525))
```

[TP monitor tunables]

```
.....
ubbcnfig
.....
```

```
20c20
< cl034 LMID=SITE1
...
> cl038 LMID=SITE1
```

```
=====
(configuration difference between cl034 and
cl039)
=====
```

[Front-end application tunables]

```
.....
tpapl.conf
.....
```

```
2c2
< Term_Base="22001"
...
> Term_Base="1826001"
```

```
.....
tnsnames.ora
.....
```

```
13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
...
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_h)(Port= 1528))
```

[TP monitor tunables]

```
.....
ubbcnfig
.....
```

```
20c20
< cl034 LMID=SITE1
...
> cl039 LMID=SITE1
```

```
=====
(configuration difference between cl034 and
cl040)
=====
```

[Front-end application tunables]

```
.....
tpapl.conf
.....
```

```
2c2
< Term_Base="22001"
...
> Term_Base="1485001"
```

```
.....
tnsnames.ora
.....
```

```
13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
...
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_g)(Port= 1527))
```

[TP monitor tunables]

```
.....
ubbcnfig
.....
```

```
20c20
< cl034 LMID=SITE1
...
> cl040 LMID=SITE1
```

```
=====
(configuration difference between cl034 and
cl041)
=====
```

[Front-end application tunables]

```
.....
tpapl.conf
.....
```

```
2c2
< Term_Base="22001"
...
> Term_Base="1727001"
```

```
.....
tnsnames.ora
.....
```

```
13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
...
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_h)(Port= 1528))
```

[TP monitor tunables]

```
.....
ubbcnfig
.....
```

```
20c20
< cl034 LMID=SITE1
...
> cl041 LMID=SITE1
```

```
> cl041 LMID=SITE1
```

```
=====
(configuration difference between cl034 and
cl042)
=====
```

[Front-end application tunables]

```
.....
tpapl.conf
.....
```

```
2c2
< Term_Base="22001"
...
> Term_Base="44001"
```

```
.....
tnsnames.ora
.....
```

[TP monitor tunables]

```
.....
ubbcnfig
.....
```

```
20c20
< cl034 LMID=SITE1
...
> cl042 LMID=SITE1
```

```
=====
(configuration difference between cl034 and
cl043)
=====
```

[Front-end application tunables]

```
.....
tpapl.conf
.....
```

```
2c2
< Term_Base="22001"
...
> Term_Base="1518001"
```

```
.....
tnsnames.ora
.....
```

```
13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
...
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_g)(Port= 1527))
```

[TP monitor tunables]

```

-----
.....
ubbcnfig
.....
20c20
< cl034 LMID=SITE1
---
> cl043 LMID=SITE1

```

```

=====
(configuration difference between cl034 and
cl044)
=====

```

[Front-end application tunables]

```

-----
.....
tpapl.conf
.....
2c2
< Term_Base="22001"
---
> Term_Base="528001"

```

```

.....
tnsnames.ora
.....
13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_c)(Port= 1523))

```

[TP monitor tunables]

```

-----
.....
ubbcnfig
.....
20c20
< cl034 LMID=SITE1
---
> cl044 LMID=SITE1

```

```

=====
(configuration difference between cl034 and
cl045)
=====

```

[Front-end application tunables]

```

-----
.....
tpapl.conf
.....

```

```

2c2
< Term_Base="22001"
---
> Term_Base="770001"

```

```

.....
tnsnames.ora
.....

```

```

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_d)(Port= 1524))

```

[TP monitor tunables]

```

-----
.....
ubbcnfig
.....
20c20
< cl034 LMID=SITE1
---
> cl045 LMID=SITE1

```

```

=====
(configuration difference between cl034 and
cl046)
=====

```

[Front-end application tunables]

```

-----
.....
tpapl.conf
.....
2c2
< Term_Base="22001"
---
> Term_Base="1012001"

```

```

.....
tnsnames.ora
.....
13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_e)(Port= 1525))

```

[TP monitor tunables]

```

-----
.....
ubbcnfig
.....
20c20
< cl034 LMID=SITE1
---
> cl046 LMID=SITE1

```

```

=====
(configuration difference between cl034 and
cl047)
=====

```

[Front-end application tunables]

```

-----
.....
tpapl.conf
.....
2c2
< Term_Base="22001"
---
> Term_Base="1276001"

```

```

.....
tnsnames.ora
.....
13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_f)(Port= 1526))

```

[TP monitor tunables]

```

-----
.....
ubbcnfig
.....
20c20
< cl034 LMID=SITE1
---
> cl047 LMID=SITE1

```

```

=====
(cl061 configuration)
=====

```

[OS tunables]

```

-----
.....
chkcnfig
.....
lisa      0:off 1:off 2:off 3:off 4:off 5:off
6:off
isdn      0:off 1:off 2:off 3:off 4:off 5:off
6:off
cups      0:off 1:off 2:off 3:off 4:off 5:off
6:off
xinetd    0:off 1:off 2:off 3:on  4:on
5:on  6:off
postgres  0:off 1:off 2:off 3:off 4:off
5:off 6:off
squid     0:off 1:off 2:off 3:off 4:off 5:off
6:off

```


ospfd	0:off 1:off 2:off 3:off 4:off 5:off 6:off	kprop	0:off 1:off 2:off 3:off 4:off 5:off 6:off	ripngd	0:off 1:off 2:off 3:off 4:off 5:off 6:off
psacct	0:off 1:off 2:off 3:off 4:off 5:off 6:off	ip6tables	0:off 1:off 2:off 3:off 4:off 5:off 6:off	ypbind	0:off 1:off 2:off 3:off 4:off 5:off 6:off
mdmmonitor	0:off 1:off 2:off 3:off 4:off 5:off 6:off	cyrus-imapd	0:off 1:off 2:off 3:off 4:off 5:off 6:off	netdump	0:off 1:off 2:off 3:off 4:off 5:off 6:off
iptables	0:off 1:off 2:on 3:on 4:on 5:on 6:off	rstatd	0:off 1:off 2:off 3:off 4:off 5:off 6:off	ntpd	0:off 1:off 2:off 3:off 4:off 5:off 6:off
network	0:off 1:off 2:on 3:on 4:on 5:on 6:off	nfs	0:off 1:off 2:off 3:off 4:off 5:off 6:off	crond	0:off 1:off 2:on 3:on 4:on 5:on 6:off
arptables_jf	0:off 1:off 2:off 3:off 4:off 5:off 6:off	cups-config-daemon	0:off 1:off 2:off 3:off 4:off 5:off 6:off	dhcpc6s	0:off 1:off 2:off 3:off 4:off 5:off 6:off
dhcpcd	0:off 1:off 2:off 3:off 4:off 5:off 6:off	haldaemon	0:off 1:off 2:off 3:off 4:off 5:off 6:off	smb	0:off 1:off 2:off 3:off 4:off 5:off 6:off
messagebus	0:off 1:off 2:off 3:off 4:off 5:off 6:off	krb5kdc	0:off 1:off 2:off 3:off 4:off 5:off 6:off	canna	0:off 1:off 2:off 3:off 4:off 5:off 6:off
ripd	0:off 1:off 2:off 3:off 4:off 5:off 6:off	netfs	0:off 1:off 2:off 3:off 4:off 5:off 6:off	amd	0:off 1:off 2:off 3:off 4:off 5:off 6:off
dc_client	0:off 1:off 2:off 3:off 4:off 5:off 6:off	ospfd	0:off 1:off 2:off 3:off 4:off 5:off 6:off	rawdevices	0:off 1:off 2:off 3:on 4:on 5:on 6:off
ypserv	0:off 1:off 2:off 3:off 4:off 5:off 6:off	irda	0:off 1:off 2:off 3:off 4:off 5:off 6:off	rpcsvcgssd	0:off 1:off 2:off 3:off 4:off 5:off 6:off
rusersd	0:off 1:off 2:off 3:off 4:off 5:off 6:off	spamassassin	0:off 1:off 2:off 3:off 4:off 5:off 6:off	xfx	0:off 1:off 2:off 3:off 4:off 5:off 6:off
sendmail	0:off 1:off 2:off 3:off 4:off 5:off 6:off	pcmcia	0:off 1:off 2:off 3:off 4:off 5:off 6:off	radvd	0:off 1:off 2:off 3:off 4:off 5:off 6:off
winbind	0:off 1:off 2:off 3:off 4:off 5:off 6:off	irqbalance	0:off 1:off 2:off 3:on 4:on 5:on 6:off	ldap	0:off 1:off 2:off 3:off 4:off 5:off 6:off
sysstat	0:off 1:on 2:off 3:off 4:off 5:off 6:off	FreeWnn	0:off 1:off 2:off 3:off 4:off 5:off 6:off	krb524	0:off 1:off 2:off 3:off 4:off 5:off 6:off
vsftpd	0:off 1:off 2:off 3:off 4:off 5:off 6:off	iiim	0:off 1:off 2:off 3:off 4:off 5:off 6:off	readahead	0:off 1:off 2:off 3:off 4:off 5:on 6:off
kudzu	0:off 1:off 2:off 3:off 4:off 5:off 6:off	bootparamd	0:off 1:off 2:off 3:off 4:off 5:off 6:off	mdmmpd	0:off 1:off 2:off 3:off 4:off 5:off 6:off
rwhod	0:off 1:off 2:off 3:off 4:off 5:off 6:off	nfslock	0:off 1:off 2:off 3:off 4:off 5:off 6:off	yppasswdd	0:off 1:off 2:off 3:off 4:off 5:off 6:off
portmap	0:off 1:off 2:off 3:off 4:off 5:off 6:off	hpoj	0:off 1:off 2:off 3:off 4:off 5:off 6:off	NetworkManager	0:off 1:off 2:off 3:off 4:off 5:off 6:off
smartd	0:off 1:off 2:off 3:off 4:off 5:off 6:off	rhnsd	0:off 1:off 2:off 3:off 4:off 5:off 6:off	rpcidmapd	0:off 1:off 2:off 3:off 4:off 5:off 6:off
dhcrelay	0:off 1:off 2:off 3:off 4:off 5:off 6:off	zebra	0:off 1:off 2:off 3:off 4:off 5:off 6:off	arpwatch	0:off 1:off 2:off 3:off 4:off 5:off 6:off
mailman	0:off 1:off 2:off 3:off 4:off 5:off 6:off	httpd	0:off 1:off 2:off 3:off 4:off 5:off 6:off	apmd	0:off 1:off 2:off 3:off 4:off 5:off 6:off
mysqld	0:off 1:off 2:off 3:off 4:off 5:off 6:off	dovecot	0:off 1:off 2:off 3:off 4:off 5:off 6:off	microcode_ctl	0:off 1:off 2:off 3:off 4:off 5:off 6:off
anacron	0:off 1:off 2:off 3:off 4:off 5:off 6:off	syslog	0:off 1:off 2:on 3:on 4:on 5:on 6:off	xinetd based services:	
netdump-server	0:off 1:off 2:off 3:off 4:off 5:off 6:off	bgpd	0:off 1:off 2:off 3:off 4:off 5:off 6:off	amandaix:	off
atd	0:off 1:off 2:off 3:off 4:off 5:off 6:off	bluetooth	0:off 1:off 2:off 3:off 4:off 5:off 6:off	talk:	off
nscd	0:off 1:off 2:off 3:off 4:off 5:off 6:off	netplugd	0:off 1:off 2:off 3:off 4:off 5:off 6:off	rsync:	off
kadmin	0:off 1:off 2:off 3:off 4:off 5:off 6:off	rpcgssd	0:off 1:off 2:off 3:off 4:off 5:off 6:off	cups-lpd:	off
sshd	0:off 1:off 2:on 3:on 4:on 5:on 6:off	saslauthd	0:off 1:off 2:off 3:off 4:off 5:off 6:off	time-udp:	off
gpm	0:off 1:off 2:off 3:off 4:off 5:off 6:off	vncserver	0:off 1:off 2:off 3:off 4:off 5:off 6:off	krb5-telnet:	off
innd	0:off 1:off 2:off 3:off 4:off 5:off 6:off	radiusd	0:off 1:off 2:off 3:off 4:off 5:off 6:off	echo:	off
cpuspeed	0:off 1:on 2:on 3:on 4:on 5:on 6:off	diskdump	0:off 1:off 2:off 3:off 4:off 5:off 6:off	dbskkd-cdb:	off
named	0:off 1:off 2:off 3:off 4:off 5:off 6:off	ypxfrd	0:off 1:off 2:off 3:off 4:off 5:off 6:off	swat:	off
snmptrapd	0:off 1:off 2:off 3:off 4:off 5:off 6:off	autofs	0:off 1:off 2:off 3:off 4:off 5:off 6:off	auth:	on
dc_server	0:off 1:off 2:off 3:off 4:off 5:off 6:off	snmpd	0:off 1:off 2:off 3:off 4:off 5:off 6:off	klogin:	off
acpid	0:off 1:off 2:off 3:off 4:off 5:off 6:off	readahead_early	0:off 1:off 2:off 3:off 4:off 5:on 6:off	gssftp:	off
lm_sensors	0:off 1:off 2:off 3:off 4:off 5:off 6:off	tux	0:off 1:off 2:off 3:off 4:off 5:off 6:off	rsh:	on
				kshell:	off
				telnet:	on
				daytime-udp:	off
				chargen-udp:	off
				amidxtape:	off
				tftp:	off
				rlogin:	on
				finger:	off
				daytime:	off
				eklogin:	off
				ntalk:	off
				time:	off
				ktalk:	off
				rexec:	off
				amanda:	off

```

echo-udp: off
chargen: off

.....
limits.conf
.....

# /etc/security/limits.conf
#
# Each line describes a limit for a user in the
# form:
#
# <domain> <type> <item> <value>
#
# Where:
# <domain> can be:
# - an user name
# - a group name, with @group syntax
# - the wildcard *, for default entry
#
# <type> can have the two values:
# - "soft" for enforcing the soft limits
# - "hard" for enforcing hard limits
#
# <item> can be one of the following:
# - core - limits the core file size (KB)
# - data - max data size (KB)
# - fsize - maximum filesize (KB)
# - memlock - max locked-in-memory
address space (KB)
# - nfile - max number of open files
# - rss - max resident set size (KB)
# - stack - max stack size (KB)
# - cpu - max CPU time (MIN)
# - nproc - max number of processes
# - as - address space limit
# - maxlogins - max number of logins for this
user
# - priority - the priority to run user process
with
# - locks - max number of file locks the user
can hold
#
# <domain> <type> <item> <value>
#
#* soft core 0
#* hard rss 10000
#@student hard nproc 20
#@faculty soft nproc 20
#@faculty hard nproc 50
#ftp hard nproc 0
#@student - maxlogins 4
tpc - nfile 30000
tpc - nproc 30000

# End of file

.....
sysctl.conf
.....

# Kernel sysctl configuration file for Red Hat
Linux
#
# For binary values, 0 is disabled, 1 is enabled.
See sysctl(8) and
# sysctl.conf(5) for more details.

# Controls IP packet forwarding
net.ipv4.ip_forward = 0

# Controls source route verification

```

```

net.ipv4.conf.default.rp_filter = 1

# Controls the System Request debugging
functionality of the kernel
kernel.sysrq = 0

# Controls whether core dumps will append the
PID to the core filename.
# Useful for debugging multi-threaded
applications.
kernel.core_uses_pid = 1

# Change filedescriptor
fs.file-max = 30000

# Change Message queue
kernel.msgmni = 30000
kernel.msgmnb = 1536000

# Change Max process
kernel.threads-max = 30000

# Change Semaphore
kernel.sem = 3000 384000 32 128

# Change TCP/IP backlog
net.ipv4.tcp_max_syn_backlog = 4096

[HTTP server tunables]
-----
.....
apache_cl_start.sh
.....

#!/bin/sh
export
LD_LIBRARY_PATH=${ORACLE_HOME}/srvm/lib:
${ORACLE_HOME}/lib64:${ORACLE_HOME}/lib:
/usr/lib:${ORACLE_HOME}/rdbsms/lib:${ORACLE_
HOME}/network/lib:$TUXDIR/lib

ulimit -u 30000
ulimit -s 1536

# /sbin/swapoff -a

# For 3tier tune
SVRAPL= ps -e | grep tpccfmlw | awk '{print $1}'
/usr/bin/renice -20 -p ${SVRAPL}

#rm -f /home/tpc/sar.tmp
/home/tpc/sar: hostname`
#/usr/lib/sa/sadc 5 > /home/tpc/sar.tmp &
# For 3tier tune

apachectl start

.....
httpd.conf
.....

#
# Based upon the NCSA server configuration
files originally by Rob McCool.
#
# This is the main Apache server configuration
file. It contains the
# configuration directives that give the server its
instructions.

```

```

# See <URL:http://httpd.apache.org/docs-2.0/>
for detailed information about
# the directives.
#
# Do NOT simply read the instructions in here
without understanding
# what they do. They're here only as hints or
reminders. If you are unsure
# consult the online docs. You have been
warned.
#
# The configuration directives are grouped into
three basic sections:
# 1. Directives that control the operation of the
Apache server process as a
# whole (the 'global environment').
# 2. Directives that define the parameters of the
'main' or 'default' server,
# which responds to requests that aren't
handled by a virtual host.
# These directives also provide default values
for the settings
# of all virtual hosts.
# 3. Settings for virtual hosts, which allow Web
requests to be sent to
# different IP addresses or hostnames and
have them handled by the
# same Apache server process.
#
# Configuration and logfile names: If the
filenames you specify for many
# of the server's control files begin with "/" (or
"drive:/" for Win32), the
# server will use that explicit path. If the
filenames do *not* begin
# with "/", the value of ServerRoot is prepended -
- so "logs/foo.log"
# with ServerRoot set to "/etc/httpd" will be
interpreted by the
# server as "/etc/httpd/logs/foo.log".
#
### Section 1: Global Environment
#
# The directives in this section affect the overall
operation of Apache,
# such as the number of concurrent requests it
can handle or where it
# can find its configuration files.
#
#
# Don't give away too much information about all
the subcomponents
# we are running. Comment out this line if you
don't mind remote sites
# finding out what major optional modules you
are running
#ServerTokens OS
ServerTokens Productly

#
# ServerRoot: The top of the directory tree under
which the server's
# configuration, error, and log files are kept.
#
# NOTE! If you intend to place this on an NFS
(or otherwise network)
# mounted filesystem then please read the
LockFile documentation
# (available at
<URL:http://httpd.apache.org/docs-
2.0/mod/core.html#lockfile>);
# you will save yourself a lot of trouble.
#

```

```
# Do NOT add a slash at the end of the directory
path.
#
ServerRoot "/etc/httpd"

#
# ScoreBoardFile: File used to store internal
server process information.
# If unspecified (the default), the scoreboard will
be stored in an
# anonymous shared memory segment, and will
be unavailable to third-party
# applications.
# If specified, ensure that no two invocations of
Apache share the same
# scoreboard file. The scoreboard file MUST BE
STORED ON A LOCAL DISK.
#
#ScoreBoardFile run/httpd.scoreboard

#
# PidFile: The file in which the server should
record its process
# identification number when it starts.
#
PidFile run/httpd.pid

#
# Timeout: The number of seconds before
receives and sends time out.
#
#Timeout 300
Timeout 999

#
# KeepAlive: Whether or not to allow persistent
connections (more than
# one request per connection). Set to "Off" to
deactivate.
#
#KeepAlive Off
KeepAlive On

#
# MaxKeepAliveRequests: The maximum
number of requests to allow
# during a persistent connection. Set to 0 to
allow an unlimited amount.
# We recommend you leave this number high,
for maximum performance.
#
#MaxKeepAliveRequests 100
MaxKeepAliveRequests 0

#
# KeepAliveTimeout: Number of seconds to wait
for the next request from the
# same client on the same connection.
#
#KeepAliveTimeout 15
KeepAliveTimeout 999

##
## Server-Pool Size Regulation (MPM specific)
##

# prefork MPM
# StartServers: number of server processes to
start
# MinSpareServers: minimum number of server
processes which are kept spare
# MaxSpareServers: maximum number of server
processes which are kept spare
```

```
# MaxClients: maximum number of server
processes allowed to start
# MaxRequestsPerChild: maximum number of
requests a server process serves
<IfModule prefork.c>
StartServers 8
MinSpareServers 5
MaxSpareServers 20
MaxClients 150
MaxRequestsPerChild 1000
</IfModule>

# worker MPM
# StartServers: initial number of server
processes to start
# MaxClients: maximum number of simultaneous
client connections
# MinSpareThreads: minimum number of worker
threads which are kept spare
# MaxSpareThreads: maximum number of
worker threads which are kept spare
# ThreadsPerChild: constant number of worker
threads in each server process
# MaxRequestsPerChild: maximum number of
requests a server process serves
<IfModule worker.c>

StartServers 45
ServerLimit 45
ThreadLimit 500
MaxClients 22500
MinSpareThreads 1
MaxSpareThreads 22500
ThreadsPerChild 500
MaxRequestsPerChild 0

#
#
# To reduce memory usage in the worker MPM,
the thread guard page
#
# To reduce memory usage in the worker MPM,
the thread guard page
# can be disabled, at the expense of some
protection against stack
# overflow.
#
#ThreadGuardArea off

</IfModule>

#
# Listen: Allows you to bind Apache to specific
IP addresses and/or
# ports, in addition to the default. See also the
<VirtualHost>
# directive.
#
# Change this to Listen on specific IP addresses
as shown below to
# prevent Apache from glomming onto all bound
IP addresses (0.0.0.0)
# e.g. "Listen 12.34.56.78:80"
#
# To allow connections to IPv6 addresses add
"Listen [::]:80"
#
Listen 0.0.0.0:80

#
# Dynamic Shared Object (DSO) Support
#
```

```
# To be able to use the functionality of a module
which was built as a DSO you
# have to place corresponding 'LoadModule'
lines at this location so the
# directives contained in it are actually available
_before_ they are used.
# Statically compiled modules (those listed by
`httpd -l') do not need
# to be loaded here.
#
# Example:
# LoadModule foo_module modules/mod_foo.so
#
LoadModule tpmpl_module
modules/mod_tpmpl.so
LoadModule access_module
modules/mod_access.so
LoadModule status_module
modules/mod_status.so
LoadModule alias_module
modules/mod_alias.so
LoadModule cgi_module modules/mod_cgi.so

#
# Load config files from the config directory
"/etc/httpd/conf.d".
#
#Include conf.d/*.conf

#
# ExtendedStatus controls whether Apache will
generate "full" status
# information (ExtendedStatus On) or just basic
information (ExtendedStatus
# Off) when the "server-status" handler is called.
The default is Off.
#
#ExtendedStatus On

### Section 2: 'Main' server configuration
#
# The directives in this section set up the values
used by the 'main'
# server, which responds to any requests that
aren't handled by a
# <VirtualHost> definition. These values also
provide defaults for
# any <VirtualHost> containers you may define
later in the file.
#
# All of these directives may appear inside
<VirtualHost> containers,
# in which case these default settings will be
overridden for the
# virtual host being defined.
#
#
# If you wish httpd to run as a different user or
group, you must run
# httpd as root initially and it will switch.
#
# User/Group: The name (or #number) of the
user/group to run httpd as.
# . On SCO (ODT 3) use "User nouser" and
"Group nogroup".
# . On HP/UX you may not be able to use
shared memory as nobody, and the
# suggested workaround is to create a user
www and use that user.
# NOTE that some kernels refuse to
setgid(Group) or semctl(IPC_SET)
# when the value of (unsigned)Group is above
60000;
# don't use Group #-1 on these systems!
```

```
#
#User apache
#Group apache
User tpc
Group tpc

#
# ServerAdmin: Your address, where problems
with the server should be
# e-mailed. This address appears on some
server-generated pages, such
# as error documents. e.g. admin@your-
domain.com
#
ServerAdmin root@localhost

#
# ServerName gives the name and port that the
server uses to identify itself.
# This can often be determined automatically,
but we recommend you specify
# it explicitly to prevent problems during startup.
#
# If this is not set to valid DNS name for your
host, server-generated
# redirections will not work. See also the
UseCanonicalName directive.
#
# If your host doesn't have a registered DNS
name, enter its IP address here.
# You will have to access it by its address
anyway, and this will make
# redirections work in a sensible way.
#
#ServerName new.host.name:80
ServerName tpccserver:80

#
# UseCanonicalName: Determines how Apache
constructs self-referencing
# URLs and the SERVER_NAME and
SERVER_PORT variables.
# When set "Off", Apache will use the Hostname
and Port supplied
# by the client. When set "On", Apache will use
the value of the
# ServerName directive.
#
UseCanonicalName Off

#
# DocumentRoot: The directory out of which you
will serve your
# documents. By default, all requests are taken
from this directory, but
# symbolic links and aliases may be used to
point to other locations.
#
#DocumentRoot "/var/www/html"

#
# Each directory to which Apache has access
can be configured with respect
# to which services and features are allowed
and/or disabled in that
# directory (and its subdirectories).
#
# First, we configure the "default" to be a very
restrictive set of
# features.
#
#<Directory />
# Options FollowSymLinks
# AllowOverride None
```

```
#</Directory>

#
# Note that from this point forward you must
specifically allow
# particular features to be enabled - so if
something's not working as
# you might expect, make sure that you have
specifically enabled it
# below.
#
#
# UserDir: The name of the directory that is
appended onto a user's home
# directory if a ~user request is received.
#
# The path to the end user account 'public_html'
directory must be
# accessible to the webserver userid. This
usually means that ~userid
# must have permissions of 711,
~userid/public_html must have permissions
# of 755, and documents contained therein must
be world-readable.
# Otherwise, the client will only receive a "403
Forbidden" message.
#
# See also:
http://httpd.apache.org/docs/misc/FAQ.html#forb
idden
#
#<IfModule mod_userdir.c>
#
# UserDir is disabled by default since it can
confirm the presence
# of a username on the system (depending on
home directory
# permissions).
#
# UserDir disable

#
# To enable requests to ~/user/ to serve the
user's public_html
# directory, remove the "UserDir disable" line
above, and uncomment
# the following line instead:
#
#UserDir public_html

#</IfModule>

#
# Control access to UserDir directories. The
following is an example
# for a site where these directories are restricted
to read-only.
#
#<Directory /home/*public_html>
# AllowOverride FileInfo AuthConfig Limit
# Options MultiViews Indexes
SymLinksIfOwnerMatch IncludesNoExec
# <Limit GET POST OPTIONS>
# Order allow,deny
# Allow from all
# </Limit>
# <LimitExcept GET POST OPTIONS>
# Order deny,allow
# Deny from all
# </LimitExcept>
#</Directory>

#
```

```
# DirectoryIndex: sets the file that Apache will
serve if a directory
# is requested.
#
# The index.html.var file (a type-map) is used to
deliver content-
# negotiated documents. The MultiViews Option
can be used for the
# same purpose, but it is much slower.
#
#
# AccessFileName: The name of the file to look
for in each directory
# for additional configuration directives. See
also the AllowOverride
# directive.
#
AccessFileName .htaccess

#
# The following lines prevent .htaccess
and .htpasswd files from being
# viewed by Web clients.
#
#
# TypesConfig describes where the mime.types
file (or equivalent) is
# to be found.
#
#
# DefaultType is the default MIME type the
server will use for a document
# if it cannot otherwise determine one, such as
from filename extensions.
# If your server contains mostly text or HTML
documents, "text/plain" is
# a good value. If most of your content is binary,
such as applications
# or images, you may want to use
"application/octet-stream" instead to
# keep browsers from trying to display binary
files as though they are
# text.
#
DefaultType text/plain

#
# The mod_mime_magic module allows the
server to use various hints from the
# contents of the file itself to determine its type.
The MIMEMagicFile
# directive tells the module where the hint
definitions are located.
#
#<IfModule mod_mime_magic.c>
## MIMEMagicFile /usr/share/magic.mime
# MIMEMagicFile conf/magic
#</IfModule>

#
# HostnameLookups: Log the names of clients
or just their IP addresses
# e.g., www.apache.org (on) or 204.62.129.132
(off).
# The default is off because it'd be overall better
for the net if people
# had to knowingly turn this feature on, since
enabling it means that
# each client request will result in AT LEAST one
lookup request to the
# nameserver.
#
```

```

HostnameLookups Off

#
# EnableMMAP: Control whether memory-
mapping is used to deliver
# files (assuming that the underlying OS
supports it).
# The default is on; turn this off if you serve from
NFS-mounted
# filesystems. On some systems, turning it off
(regardless of
# filesystem) can improve performance; for
details, please see
# http://httpd.apache.org/docs-
2.0/mod/core.html#enablemmap
#
#EnableMMAP off

#
# EnableSendfile: Control whether the sendfile
kernel support is
# used to deliver files (assuming that the OS
supports it).
# The default is on; turn this off if you serve from
NFS-mounted
# filesystems. Please see
# http://httpd.apache.org/docs-
2.0/mod/core.html#enablesendfile
#
#EnableSendfile off
#

#
# ErrorLog: The location of the error log file.
# If you do not specify an ErrorLog directive
within a <VirtualHost>
# container, error messages relating to that
virtual host will be
# logged here. If you *do* define an error logfile
for a <VirtualHost>
# container, that host's errors will be logged
there and not here.
#
ErrorLog logs/error_log

#
# LogLevel: Control the number of messages
logged to the error_log.
# Possible values include: debug, info, notice,
warn, error, crit,
# alert, emerg.
#
LogLevel warn

#
# The following directives define some format
nicknames for use with
# a CustomLog directive (see below).
#

#
# The location and format of the access logfile
(Common Logfile Format).
# If you do not define any access logfiles within
a <VirtualHost>
# container, they will be logged here.
Contrariwise, if you *do*
# define per-<VirtualHost> access logfiles,
transactions will be
# logged therein and *not* in this file.
#
# CustomLog logs/access_log common
# CustomLog logs/access_log combined

#

# If you would like to have agent and referer
logfiles, uncomment the
# following directives.
#
#CustomLog logs/referer_log referer
#CustomLog logs/agent_log agent

#
# If you prefer a single logfile with access, agent,
and referer information
# (Combined Logfile Format) you can use the
following directive.
#
#CustomLog logs/access_log combined

#
# Optionally add a line containing the server
version and virtual host
# name to server-generated pages (error
documents, FTP directory listings,
# mod_status and mod_info output etc., but not
CGI generated documents).
# Set to "EMail" to also include a mailto: link to
the ServerAdmin.
# Set to one of: On | Off | EMail
#
#ServerSignature On
ServerSignature Off

#
# Aliases: Add here as many aliases as you
need (with no limit). The format is
# Alias fakename realname
#
# Note that if you include a trailing / on fakename
then the server will
# require it to be present in the URL. So "/icons"
isn't aliased in this
# example, only "/icons/". If the fakename is
slash-terminated, then the
# realname must also be slash terminated, and if
the fakename omits the
# trailing slash, the realname must also omit it.
#
# We include the /icons/ alias for FancyIndexed
directory listings. If you
# do not use FancyIndexing, you may comment
this out.
#

#
# This should be changed to the
ServerRoot/manual/. The alias provides
# the manual, even if you choose to move your
DocumentRoot. You may comment
# this out if you do not care for the
documentation.
#
#<IfModule mod_dav_fs.c>
# # Location of the WebDAV lock database.
# DAVLockDB /var/lib/dav/lockdb
#</IfModule>

#
# ScriptAlias: This controls which directories
contain server scripts.
# ScriptAliases are essentially the same as
Aliases, except that
# documents in the realname directory are
treated as applications and
# run by the server when requested rather than
as documents sent to the client.
# The same rules about trailing "/" apply to
ScriptAlias directives as to
# Alias.

#
#ScriptAlias /cgi-bin/ "/var/www/cgi-bin/"
ScriptAlias /cgi-bin/ "/home/tpc/tool/bin/"

#
# "/var/www/cgi-bin" should be changed to
whatever your ScriptAliased
# CGI directory exists, if you have that
configured.
#
<Directory "/var/www/cgi-bin">
    AllowOverride None
    Options None
    Order allow,deny
    Allow from all
</Directory>

#
# Redirect allows you to tell clients about
documents which used to exist in
# your server's namespace, but do not anymore.
This allows you to tell the
# clients where to look for the relocated
document.
# Example:
# Redirect permanent /foo
http://www.example.com/bar

#
# Directives controlling the display of server-
generated directory listings.
#

#
# FancyIndexing is whether you want fancy
directory indexing or standard.
# VersionSort is whether files containing version
numbers should be
# compared in the natural way, so that `apache-
1.3.9.tar' is placed before
# `apache-1.3.12.tar'.
#

#
# AddIcon* directives tell the server which icon
to show for different
# files or filename extensions. These are only
displayed for
# FancyIndexed directories.
#

#
# DefaultIcon is which icon to show for files
which do not have an icon
# explicitly set.
#

#
# AddDescription allows you to place a short
description after a file in
# server-generated indexes. These are only
displayed for FancyIndexed
# directories.
# Format: AddDescription "description" filename
#
#AddDescription "GZIP compressed
document" .gz
#AddDescription "tar archive" .tar
#AddDescription "GZIP compressed tar
archive" .tgz

#
# ReadmeName is the name of the README file
the server will look for by

```

```
# default, and append to directory listings.
#
# HeaderName is the name of a file which
should be prepended to
# directory indexes.

#
# IndexIgnore is a set of filenames which
directory indexing should ignore
# and not include in the listing. Shell-style
wildcarding is permitted.
#
#
# AddEncoding allows you to have certain
browsers (Mosaic/X 2.1+) uncompress
# information on the fly. Note: Not all browsers
support this.
# Despite the name similarity, the following Add*
directives have nothing
# to do with the FancyIndexing customization
directives above.
#
#
# DefaultLanguage and AddLanguage allows
you to specify the language of
# a document. You can then use content
negotiation to give a browser a
# file in a language the user can understand.
#
# Specify a default language. This means that all
data
# going out without a specific language tag (see
below) will
# be marked with this one. You probably do NOT
want to set
# this unless you are sure it is correct for all
cases.
#
# * It is generally better to not mark a page as
# * being a certain language than marking it with
the wrong
# * language!
#
# DefaultLanguage nl
#
# Note 1: The suffix does not have to be the
same as the language
# keyword --- those with documents in Polish
(whose net-standard
# language code is pl) may wish to use
"AddLanguage pl .po" to
# avoid the ambiguity with the common suffix for
perl scripts.
#
# Note 2: The example entries below illustrate
that in some cases
# the two character 'Language' abbreviation is
not identical to
# the two character 'Country' code for its country,
# E.g. 'Danmark/dk' versus 'Danish/da'.
#
# Note 3: In the case of 'ltz' we violate the RFC
by using a three char
# specifier. There is 'work in progress' to fix this
and get
# the reference data for rfc1766 cleaned up.
#
# Danish (da) - Dutch (nl) - English (en) -
Estonian (et)
# French (fr) - German (de) - Greek-Modern (el)
# Italian (it) - Norwegian (no) - Norwegian
Nynorsk (nn) - Korean (ko)
# Portugese (pt) - Luxembourgish* (ltz)
```

```
# Spanish (es) - Swedish (sv) - Catalan (ca) -
Czech(cs)
# Polish (pl) - Brazilian Portuguese (pt-br) -
Japanese (ja)
# Russian (ru) - Croatian (hr)
#
#
# LanguagePriority allows you to give
precedence to some languages
# in case of a tie during content negotiation.
#
# Just list the languages in decreasing order of
preference. We have
# more or less alphabetized them here. You
probably want to change this.
#
#
# ForceLanguagePriority allows you to serve a
result page rather than
# MULTIPLE CHOICES (Prefer) [in case of a tie]
or NOT ACCEPTABLE (Fallback)
# [in case no accepted languages matched the
available variants]
#
#
# Specify a default charset for all pages sent out.
This is
# always a good idea and opens the door for
future internationalisation
# of your web site, should you ever want it.
Specifying it as
# a default does little harm; as the standard
dictates that a page
# is in iso-8859-1 (latin1) unless specified
otherwise i.e. you
# are merely stating the obvious. There are also
some security
# reasons in browsers, related to javascript and
URL parsing
# which encourage you to always set a default
charset.
#
AddDefaultCharset UTF-8
#
# Commonly used filename extensions to
character sets. You probably
# want to avoid clashes with the language
extensions, unless you
# are good at carefully testing your setup after
each change.
# See
http://www.iana.org/assignments/character-sets
for the
# official list of charset names and their
respective RFCs
#
#
# AddType allows you to add to or override the
MIME configuration
# file mime.types for specific file types.
#
#
# AddHandler allows you to map certain file
extensions to "handlers":
# actions unrelated to filetype. These can be
either built into the server
# or added with the Action directive (see below)
#
```

```
# To use CGI scripts outside of ScriptAliased
directories:
# (You will also need to add "ExecCGI" to the
"Options" directive.)
#
AddHandler cgi-script .cgi
#
#
# For files that include their own HTTP headers:
#
AddHandler send-as-is asis
#
#
# For server-parsed imagemap files:
#
#
# For type maps (negotiated resources):
# (This is enabled by default to allow the Apache
"It Worked" page
# to be distributed in multiple languages.)
#
# Filters allow you to process content before it is
sent to the client.
#
# To parse .shtml files for server-side includes
(SSI):
# (You will also need to add "Includes" to the
"Options" directive.)
#
#
# Action lets you define media types that will
execute a script whenever
# a matching file is called. This eliminates the
need for repeated URL
# pathnames for oft-used CGI file processors.
# Format: Action media/type /cgi-script/location
# Format: Action handler-name /cgi-
script/location
#
#
# Customizable error responses come in three
flavors:
# 1) plain text 2) local redirects 3) external
redirects
#
# Some examples:
#ErrorDocument 500 "The server made a boo
boo."
#ErrorDocument 404 /missing.html
#ErrorDocument 404 "/cgi-
bin/missing_handler.pl"
#ErrorDocument 402
http://www.example.com/subscription_info.html
#
#
# Putting this all together, we can
Internationalize error responses.
#
# We use Alias to redirect any
/error/HTTP_<error>.html.var response to
# our collection of by-error message multi-
language collections. We use
# includes to substitute the appropriate text.
#
# You can modify the messages' appearance
without changing any of the
# default HTTP_<error>.html.var files by adding
the line;
#
# Alias /error/include/ "your/include/path/"
```

```
#
# which allows you to create your own set of files
# by starting with the
# /var/www/error/include/ files and
# copying them to /your/include/path/, even on a
# per-VirtualHost basis.
#

Alias /error/ "/var/www/error/"

# ErrorDocument 400
/error/HTTP_BAD_REQUEST.html.var
# ErrorDocument 401
/error/HTTP_UNAUTHORIZED.html.var
# ErrorDocument 403
/error/HTTP_FORBIDDEN.html.var
# ErrorDocument 404
/error/HTTP_NOT_FOUND.html.var
# ErrorDocument 405
/error/HTTP_METHOD_NOT_ALLOWED.html.v
ar
# ErrorDocument 408
/error/HTTP_REQUEST_TIME_OUT.html.var
# ErrorDocument 410
/error/HTTP_GONE.html.var
# ErrorDocument 411
/error/HTTP_LENGTH_REQUIRED.html.var
# ErrorDocument 412
/error/HTTP_PRECONDITION_FAILED.html.var
# ErrorDocument 413
/error/HTTP_REQUEST_ENTITY_TOO_LARGE
.html.var
# ErrorDocument 414
/error/HTTP_REQUEST_URI_TOO_LARGE.htm
l.var
# ErrorDocument 415
/error/HTTP_SERVICE_UNAVAILABLE.html.var
# ErrorDocument 500
/error/HTTP_INTERNAL_SERVER_ERROR.htm
l.var
# ErrorDocument 501
/error/HTTP_NOT_IMPLEMENTED.html.var
# ErrorDocument 502
/error/HTTP_BAD_GATEWAY.html.var
# ErrorDocument 503
/error/HTTP_SERVICE_UNAVAILABLE.html.var
# ErrorDocument 506
/error/HTTP_VARIANT_ALSO_VARIES.html.var

#
# The following directives modify normal HTTP
# response behavior to
# handle known problems with browser
# implementations.
#

#
# The following directive disables redirects on
# non-GET requests for
# a directory that does not include the trailing
# slash. This fixes a
# problem with Microsoft WebFolders which
# does not appropriately handle
# redirects for folders with DAV methods.
# Same deal with Apple's DAV filesystem and
# Gnome VFS support for DAV.
#
# Allow server status reports, with the URL of
# http://servername/server-status
# Change the ".example.com" to match your
# domain to enable.
#
<Location /server-status>
```

```
SetHandler server-status
Order deny,allow
Deny from all
Allow from 192.168.
</Location>

#
# Allow remote server configuration reports, with
# the URL of
# http://servername/server-info (requires that
# mod_info.c be loaded).
# Change the ".example.com" to match your
# domain to enable.
#
<Location /server-info>
# SetHandler server-info
# Order deny,allow
# Deny from all
# Allow from .example.com
#</Location>

#
# Proxy Server directives. Uncomment the
# following lines to
# enable the proxy server:
#
#<IfModule mod_proxy.c>
#ProxyRequests On
#
#<Proxy *>
# Order deny,allow
# Deny from all
# Allow from .example.com
#</Proxy>

#
# Enable/disable the handling of HTTP/1.1 "Via:"
# headers.
# ("Full" adds the server version; "Block"
# removes all outgoing Via: headers)
# Set to one of: Off | On | Full | Block
#
#ProxyVia On

#
# To enable a cache of proxied content,
# uncomment the following lines.
# See http://httpd.apache.org/docs-
# 2.0/mod/mod_cache.html for more details.
#
#<IfModule mod_disk_cache.c>
# CacheEnable disk /
# CacheRoot "/var/cache/mod_proxy"
#</IfModule>
#
#</IfModule>
# End of proxy directives.

### Section 3: Virtual Hosts
#
# VirtualHost: If you want to maintain multiple
# domains/hostnames on your
# machine you can setup VirtualHost containers
# for them. Most configurations
# use only name-based virtual hosts so the
# server doesn't need to worry about
# IP addresses. This is indicated by the asterisks
# in the directives below.
#
# Please see the documentation at
# <URL:http://httpd.apache.org/docs-
# 2.0/vhosts/>
# for further details before you try to setup virtual
# hosts.
```

```
#
# You may use the command line option '-S' to
# verify your virtual host
# configuration.

#
# Use name-based virtual hosting.
#
#NameVirtualHost *:80

#
# VirtualHost example:
# Almost any Apache directive may go into a
# VirtualHost container.
# The first VirtualHost section is used for
# requests without a known
# server name.
#
#<VirtualHost *>
# ServerAdmin webmaster@dummy-
# host.example.com
# DocumentRoot /www/docs/dummy-
# host.example.com
# ServerName dummy-host.example.com
# ErrorLog logs/dummy-host.example.com-
# error_log
# CustomLog logs/dummy-host.example.com-
# access_log common
#</VirtualHost>

#
# For TPAPL
#
<Location /tpapl>
SetHandler tpapl
TpAplConf /home/tpc/conf/tpapl.conf
</Location>

[Front-end application tunables]
-----

.....
tpapl.conf
.....

[TPAPL_INFO]
Term_Base="286001"
NumWarehouses="193600"
MaxUsers="1936000"
MaxTerm of Client="22000"
CONTROL_Flag="0"
LogPath="/home/tpc/log/userlog.log"

[SVRAPL_INFO]
LogPath="/home/tpc/log/DBDepend_Userlog.log"
"

.....
tnsnames.ora
.....

#
# Filename: Tnsnames.ora
#
extproc_connection_data =
(DESCRIPTION =
(AADDRESS = (PROTOCOL = IPC)(KEY =
tpcc))
(SDU=14600)
(CONNECT_DATA = (SERVICE_NAME =
tpcc))
```

```

)
tpcc =
(DESCRIPTION =
  (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
  (SDU=14600)
  (CONNECT_DATA = (SERVICE_NAME =
tpcc))
)

[TP monitor tunables]
-----
:
:
ubbcnfig
:
:
#
# ubbcnfig : TUXEDO configuration file
(WAREHOUSE BINED)
#

*RESOURCES
IPCKEY      211940
MASTER     SITE1
UID         500
GID         500
PERM        0660
MAXACCESSERS 1000
MAXSERVERS 100
MAXSERVICES 100
MAXGTT      0
MODEL       SHM
LDBAL       Y
OPTIONS     NO_AA,NO_XA

*MACHINES
cl061 LMID=SITE1
  APPDIR="/home/tpc/bin"
  TUXCONFIG="/home/tpc/conf/tuxconfig"
  TUXDIR="/usr/local/BEA/tuxedo8.1"
  ULOGPFX="/home/tpc/log/tuxedo.log"
  SICACHEENTRIESMAX="0"

*GROUPS
group1 LMID=SITE1 GRPNO=1

*SERVERS
DEFAULT: RESTART=Y MAXGEN=5
REPLYQ=N RQPERM=0660

tpccfmlw SRVGRP=group1 RQADDR=ware01
SRVID=1 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware02
SRVID=2 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware03
SRVID=3 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware04
SRVID=4 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware05
SRVID=5 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware06
SRVID=6 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware07
SRVID=7 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"

```

```

tpccfmlw SRVGRP=group1 RQADDR=ware08
SRVID=8 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware09
SRVID=9 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware10
SRVID=10 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware11
SRVID=11 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware12
SRVID=12 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware13
SRVID=13 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware14
SRVID=14 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware15
SRVID=15 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware16
SRVID=16 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware17
SRVID=17 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware18
SRVID=18 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware19
SRVID=19 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware20
SRVID=20 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"

*SERVICES
"OPSTUXSERVER" TRANTIME=0
SRVGRP=group1

*ROUTING

=====
(configuration difference between cl061 and
cl062)
=====

[Front-end application tunables]
-----
:
:
tpapl.conf
:
:
2c2
< Term_Base="286001"
---
> Term_Base="1760001"

:
:
tnsnames.ora
:
:
13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))

```

```

---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_h)(Port= 1528))

[TP monitor tunables]
-----
:
:
ubbcnfig
:
:
20c20
< cl061 LMID=SITE1
---
> cl062 LMID=SITE1

=====
(configuration difference between cl061 and
cl063)
=====

[Front-end application tunables]
-----
:
:
tpapl.conf
:
:
2c2
< Term_Base="286001"
---
> Term_Base="66001"

:
:
tnsnames.ora
:
:
13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))

[TP monitor tunables]
-----
:
:
ubbcnfig
:
:
20c20
< cl061 LMID=SITE1
---
> cl063 LMID=SITE1

=====
(configuration difference between cl061 and
cl064)
=====

[Front-end application tunables]
-----
:
:
tpapl.conf
:
:
2c2
< Term_Base="286001"
---
> Term_Base="1760001"

:
:
tnsnames.ora
:
:
13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))

```



```

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="308001"

.....

tnsnames.ora
.....

[TP monitor tunables]
-----

.....

ubbconfig
.....

20c20
< cl061 LMID=SITE1
---
> cl064 LMID=SITE1

=====
(configuration difference between cl061 and
cl065)
=====

[Front-end application tunables]
-----

.....

tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="550001"

.....

tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_c)(Port= 1523))

[TP monitor tunables]
-----

.....

ubbconfig
.....

20c20
< cl061 LMID=SITE1
---
> cl065 LMID=SITE1

```

```

=====
(configuration difference between cl061 and
cl066)
=====

[Front-end application tunables]
-----

.....

tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="792001"

.....

tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_d)(Port= 1524))

[TP monitor tunables]
-----

.....

ubbconfig
.....

20c20
< cl061 LMID=SITE1
---
> cl066 LMID=SITE1

=====
(configuration difference between cl061 and
cl067)
=====

[Front-end application tunables]
-----

.....

tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="1034001"

.....

tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---

```

```

> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_e)(Port= 1525))

[TP monitor tunables]
-----

.....

ubbconfig
.....

20c20
< cl061 LMID=SITE1
---
> cl067 LMID=SITE1

=====
(configuration difference between cl061 and
cl068)
=====

[Front-end application tunables]
-----

.....

tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="1298001"

.....

tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_f)(Port= 1526))

[TP monitor tunables]
-----

.....

ubbconfig
.....

20c20
< cl061 LMID=SITE1
---
> cl068 LMID=SITE1

=====
(configuration difference between cl061 and
cl071)
=====

[Front-end application tunables]
-----

.....

```

```

tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="1782001"

.....

tnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_h)(Port= 1528))

[TP monitor tunables]
-----

.....

ubbconfig
.....

20c20
< cl061 LMID=SITE1
---
> cl071 LMID=SITE1

=====
(configuration difference between cl061 and
cl072)
=====

[Front-end application tunables]
-----

.....

tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="88001"

.....

tnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))

[TP monitor tunables]
-----

.....

ubbconfig
.....

20c20

```

```

< cl061 LMID=SITE1
---
> cl072 LMID=SITE1

=====
(configuration difference between cl061 and
cl074)
=====

[Front-end application tunables]
-----

.....

tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="572001"

.....

tnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_c)(Port= 1523))

[TP monitor tunables]
-----

.....

ubbconfig
.....

20c20
< cl061 LMID=SITE1
---
> cl074 LMID=SITE1

=====
(configuration difference between cl061 and
cl075)
=====

[Front-end application tunables]
-----

.....

tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="814001"

.....

tnames.ora
.....

```

```

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_d)(Port= 1524))

[TP monitor tunables]
-----

.....

ubbconfig
.....

20c20
< cl061 LMID=SITE1
---
> cl075 LMID=SITE1

=====
(configuration difference between cl061 and
cl076)
=====

[Front-end application tunables]
-----

.....

tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="1056001"

.....

tnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_e)(Port= 1525))

[TP monitor tunables]
-----

.....

ubbconfig
.....

20c20
< cl061 LMID=SITE1
---
> cl076 LMID=SITE1

=====
(configuration difference between cl061 and
cl077)
=====

```

[Front-end application tunables]

```

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
...
> Term_Base="1320001"

```

```

.....
tnsnames.ora
.....

```

```

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
...
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_f)(Port= 1526))

```

[TP monitor tunables]

```

.....
ubbcnfig
.....

```

```

20c20
< cl061 LMID=SITE1
...
> cl077 LMID=SITE1

```

```

=====
(configuration difference between cl061 and
cl078)
=====

```

[Front-end application tunables]

```

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
...
> Term_Base="1562001"

```

```

.....
tnsnames.ora
.....

```

```

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
...
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_g)(Port= 1527))

```

[TP monitor tunables]

```

.....
ubbcnfig
.....

```

```

20c20
< cl061 LMID=SITE1
...
> cl078 LMID=SITE1

```

```

=====
(configuration difference between cl061 and
cl079)
=====

```

[Front-end application tunables]

```

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
...
> Term_Base="1804001"

```

```

.....
tnsnames.ora
.....

```

```

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
...
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_h)(Port= 1528))

```

[TP monitor tunables]

```

.....
ubbcnfig
.....

```

```

20c20
< cl061 LMID=SITE1
...
> cl079 LMID=SITE1

```

```

=====
(configuration difference between cl061 and
cl080)
=====

```

[Front-end application tunables]

```

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
...
> Term_Base="110001"

```

```

.....
tnsnames.ora
.....

```

```

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
...
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))

```

[TP monitor tunables]

```

.....
ubbcnfig
.....

```

```

20c20
< cl061 LMID=SITE1
...
> cl080 LMID=SITE1

```

```

=====
(configuration difference between cl061 and
cl081)
=====

```

[Front-end application tunables]

```

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
...
> Term_Base="352001"

```

```

.....
tnsnames.ora
.....

```

[TP monitor tunables]

```

.....
ubbcnfig
.....

```

```

20c20
< cl061 LMID=SITE1
...
> cl081 LMID=SITE1

```

```

=====
(configuration difference between cl061 and
cl082)
=====

```

```
[Front-end application tunables]
-----
.....
.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="594001"

.....
.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_c)(Port= 1523))

[TP monitor tunables]
-----
.....
.....
ubbcfg
.....

20c20
< cl061 LMID=SITE1
---
> cl082 LMID=SITE1

=====
(configuration difference between cl061 and
cl083)
=====

[Front-end application tunables]
-----
.....
.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="836001"

.....
.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_d)(Port= 1524))

[TP monitor tunables]
-----
.....
.....
```

```
ubbcfg
-----
.....
.....
20c20
< cl061 LMID=SITE1
---
> cl083 LMID=SITE1

=====
(configuration difference between cl061 and
cl084)
=====

[Front-end application tunables]
-----
.....
.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="1078001"

.....
.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_e)(Port= 1525))

[TP monitor tunables]
-----
.....
.....
ubbcfg
.....

20c20
< cl061 LMID=SITE1
---
> cl084 LMID=SITE1

=====
(configuration difference between cl061 and
cl085)
=====

[Front-end application tunables]
-----
.....
.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="1342001"
```

```
.....
.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_f)(Port= 1526))

[TP monitor tunables]
-----
.....
.....
ubbcfg
.....

20c20
< cl061 LMID=SITE1
---
> cl085 LMID=SITE1

=====
(configuration difference between cl061 and
cl086)
=====

[Front-end application tunables]
-----
.....
.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="1584001"

.....
.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_g)(Port= 1527))

[TP monitor tunables]
-----
.....
.....
ubbcfg
.....

20c20
< cl061 LMID=SITE1
---
> cl086 LMID=SITE1

=====
=====
```

```

(configuration difference between cl061 and
cl087)
=====
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="1254001"

.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_f)(Port= 1526))

[TP monitor tunables]
-----

.....
ubbcfg
.....

20c20
< cl061 LMID=SITE1
---
> cl087 LMID=SITE1

=====
=====
(configuration difference between cl061 and
cl088)
=====
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="132001"

.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))

```

```

[TP monitor tunables]
-----

.....
ubbcfg
.....

20c20
< cl061 LMID=SITE1
---
> cl088 LMID=SITE1

=====
=====
(configuration difference between cl061 and
cl089)
=====
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="374001"

.....
tnsnames.ora
.....

[TP monitor tunables]
-----

.....
ubbcfg
.....

20c20
< cl061 LMID=SITE1
---
> cl089 LMID=SITE1

=====
=====
(configuration difference between cl061 and
cl090)
=====
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="616001"

```

```

.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_c)(Port= 1523))

[TP monitor tunables]
-----

.....
ubbcfg
.....

20c20
< cl061 LMID=SITE1
---
> cl090 LMID=SITE1

=====
=====
(configuration difference between cl061 and
cl091)
=====
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="858001"

.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_d)(Port= 1524))

[TP monitor tunables]
-----

.....
ubbcfg
.....

20c20
< cl061 LMID=SITE1
---
> cl091 LMID=SITE1

=====
=====

```

```
(configuration difference between cl061 and
cl092)
=====
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="1100001"

.....
tnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_e)(Port= 1525))

[TP monitor tunables]
-----

.....
ubbconfig
.....

20c20
< cl061 LMID=SITE1
---
> cl092 LMID=SITE1

=====
=====
(configuration difference between cl061 and
cl095)
=====
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="330001"

.....
tnames.ora
.....

[TP monitor tunables]
-----

.....
```

```
ubbconfig
.....

20c20
< cl061 LMID=SITE1
---
> cl095 LMID=SITE1

=====
=====
(configuration difference between cl061 and
cl105)
=====
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="1364001"

.....
tnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_f)(Port= 1526))

[TP monitor tunables]
-----

.....
ubbconfig
.....

20c20
< cl061 LMID=SITE1
---
> cl105 LMID=SITE1

=====
=====
(configuration difference between cl061 and
cl106)
=====
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="1606001"
```

```
.....
tnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_g)(Port= 1527))

[TP monitor tunables]
-----

.....
ubbconfig
.....

20c20
< cl061 LMID=SITE1
---
> cl106 LMID=SITE1

=====
=====
(configuration difference between cl061 and
cl108)
=====
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="1848001"

.....
tnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_h)(Port= 1528))

[TP monitor tunables]
-----

.....
ubbconfig
.....

20c20
< cl061 LMID=SITE1
---
> cl108 LMID=SITE1

=====
=====
```

```
(configuration difference between cl061 and
cl109)
=====
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="154001"

.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))

[TP monitor tunables]
-----

.....
ubbcfg
.....

20c20
< cl061 LMID=SITE1
---
> cl109 LMID=SITE1

=====
=====
(configuration difference between cl061 and
cl110)
=====
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="396001"

.....
tnsnames.ora
.....

[TP monitor tunables]
-----

.....
```

```
ubbcfg
.....

20c20
< cl061 LMID=SITE1
---
> cl110 LMID=SITE1

=====
=====
(configuration difference between cl061 and
cl111)
=====
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="638001"

.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_c)(Port= 1523))

[TP monitor tunables]
-----

.....
ubbcfg
.....

20c20
< cl061 LMID=SITE1
---
> cl111 LMID=SITE1

=====
=====
(configuration difference between cl061 and
cl112)
=====
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="880001"
```

```
.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_d)(Port= 1524))

[TP monitor tunables]
-----

.....
ubbcfg
.....

20c20
< cl061 LMID=SITE1
---
> cl112 LMID=SITE1

=====
=====
(configuration difference between cl061 and
cl113)
=====
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="1122001"

.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_e)(Port= 1525))

[TP monitor tunables]
-----

.....
ubbcfg
.....

20c20
< cl061 LMID=SITE1
---
> cl113 LMID=SITE1

=====
=====
```

```
(configuration difference between cl061 and
cl114)
=====
```

```
[Front-end application tunables]
-----
```

```
.....
tpapl.conf
.....
```

```
2c2
< Term_Base="286001"
---
> Term_Base="1386001"
```

```
.....
tnsnames.ora
.....
```

```
13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_f)(Port= 1526))
```

```
[TP monitor tunables]
-----
```

```
.....
ubbcfg
.....
```

```
20c20
< cl061 LMID=SITE1
---
> cl114 LMID=SITE1
```

```
=====
(configuration difference between cl061 and
cl115)
=====
```

```
[Front-end application tunables]
-----
```

```
.....
tpapl.conf
.....
```

```
2c2
< Term_Base="286001"
---
> Term_Base="1628001"
```

```
.....
tnsnames.ora
.....
```

```
13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_g)(Port= 1527))
```

```
[TP monitor tunables]
-----
```

```
.....
ubbcfg
.....
```

```
20c20
< cl061 LMID=SITE1
---
> cl115 LMID=SITE1
```

```
=====
(configuration difference between cl061 and
cl116)
=====
```

```
[Front-end application tunables]
-----
```

```
.....
tpapl.conf
.....
```

```
2c2
< Term_Base="286001"
---
> Term_Base="1870001"
```

```
.....
tnsnames.ora
.....
```

```
13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_h)(Port= 1528))
```

```
[TP monitor tunables]
-----
```

```
.....
ubbcfg
.....
```

```
20c20
< cl061 LMID=SITE1
---
> cl116 LMID=SITE1
```

```
=====
(configuration difference between cl061 and
cl117)
=====
```

```
[Front-end application tunables]
-----
```

```
.....
tpapl.conf
.....
```

```
2c2
< Term_Base="286001"
---
> Term_Base="176001"
```

```
.....
tnsnames.ora
.....
```

```
13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))
```

```
[TP monitor tunables]
-----
```

```
.....
ubbcfg
.....
```

```
20c20
< cl061 LMID=SITE1
---
> cl117 LMID=SITE1
```

```
=====
(configuration difference between cl061 and
cl118)
=====
```

```
[Front-end application tunables]
-----
```

```
.....
tpapl.conf
.....
```

```
2c2
< Term_Base="286001"
---
> Term_Base="418001"
```

```
.....
tnsnames.ora
.....
```

```
[TP monitor tunables]
-----
```

```
.....
ubbcfg
.....
```

```
20c20
< cl061 LMID=SITE1
---
> cl118 LMID=SITE1
```

```
=====
tpapl.conf
=====
```



```
(configuration difference between cl061 and
cl119)
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="660001"

.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_c)(Port= 1523))

[TP monitor tunables]
-----

.....
ubbcnfig
.....

20c20
< cl061 LMID=SITE1
---
> cl119 LMID=SITE1

=====
(configuration difference between cl061 and
cl121)
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="902001"

.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_d)(Port= 1524))
```

```
[TP monitor tunables]
-----

.....
ubbcnfig
.....

20c20
< cl061 LMID=SITE1
---
> cl121 LMID=SITE1

=====
(configuration difference between cl061 and
cl122)
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="1144001"

.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_e)(Port= 1525))

[TP monitor tunables]
-----

.....
ubbcnfig
.....

20c20
< cl061 LMID=SITE1
---
> cl122 LMID=SITE1

=====
(configuration difference between cl061 and
cl124)
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....
```

```
2c2
< Term_Base="286001"
---
> Term_Base="1408001"

.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_f)(Port= 1526))

[TP monitor tunables]
-----

.....
ubbcnfig
.....

20c20
< cl061 LMID=SITE1
---
> cl124 LMID=SITE1

=====
(configuration difference between cl061 and
cl125)
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="1650001"

.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_g)(Port= 1527))

[TP monitor tunables]
-----

.....
ubbcnfig
.....

20c20
< cl061 LMID=SITE1
---
```

```

> cl125 LMID=SITE1

=====
(configuration difference between cl061 and
cl126)
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="1892001"

.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_h)(Port= 1528))

[TP monitor tunables]
-----

.....
ubbconfig
.....

20c20
< cl061 LMID=SITE1
---
> cl126 LMID=SITE1

=====
(configuration difference between cl061 and
cl127)
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="198001"

.....
tnsnames.ora
.....

13c13

```

```

< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))

[TP monitor tunables]
-----

.....
ubbconfig
.....

20c20
< cl061 LMID=SITE1
---
> cl127 LMID=SITE1

=====
(configuration difference between cl061 and
cl128)
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="440001"

.....
tnsnames.ora
.....

[TP monitor tunables]
-----

.....
ubbconfig
.....

20c20
< cl061 LMID=SITE1
---
> cl128 LMID=SITE1

=====
(configuration difference between cl061 and
cl129)
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

```

```

2c2
< Term_Base="286001"
---
> Term_Base="682001"

.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_c)(Port= 1523))

[TP monitor tunables]
-----

.....
ubbconfig
.....

20c20
< cl061 LMID=SITE1
---
> cl129 LMID=SITE1

=====
(configuration difference between cl061 and
cl130)
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="924001"

.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_d)(Port= 1524))

[TP monitor tunables]
-----

.....
ubbconfig
.....

20c20
< cl061 LMID=SITE1
---
> cl130 LMID=SITE1

```

```

=====
=====
(configuration difference between cl061 and
cl131)
=====
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="1166001"

.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_e)(Port= 1525))

[TP monitor tunables]
-----

.....
ubbcfg
.....

20c20
< cl061 LMID=SITE1
---
> cl131 LMID=SITE1

=====
=====
(configuration difference between cl061 and
cl132)
=====
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="1430001"

.....
tnsnames.ora
.....

13c13

```

```

< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_f)(Port= 1526))

[TP monitor tunables]
-----

.....
ubbcfg
.....

20c20
< cl061 LMID=SITE1
---
> cl132 LMID=SITE1

=====
=====
(configuration difference between cl061 and
cl133)
=====
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="1672001"

.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_g)(Port= 1527))

[TP monitor tunables]
-----

.....
ubbcfg
.....

20c20
< cl061 LMID=SITE1
---
> cl133 LMID=SITE1

=====
=====
(configuration difference between cl061 and
cl134)
=====
=====

[Front-end application tunables]

```

```

-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="1914001"

.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_h)(Port= 1528))

[TP monitor tunables]
-----

.....
ubbcfg
.....

20c20
< cl061 LMID=SITE1
---
> cl134 LMID=SITE1

=====
=====
(configuration difference between cl061 and
cl135)
=====
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="220001"

.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_a)(Port= 1521))

[TP monitor tunables]
-----

.....
ubbcfg

```

```

.....
20c20
< cl061 LMID=SITE1
---
> cl135 LMID=SITE1

=====
(configuration difference between cl061 and
cl136)
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="462001"

.....
tnsnames.ora
.....

[TP monitor tunables]
-----

.....
ubbconfig
.....

20c20
< cl061 LMID=SITE1
---
> cl136 LMID=SITE1

=====
(configuration difference between cl061 and
cl137)
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="704001"

.....
tnsnames.ora
.....

13c13

```

```

< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_c)(Port= 1523))

[TP monitor tunables]
-----

.....
ubbconfig
.....

20c20
< cl061 LMID=SITE1
---
> cl137 LMID=SITE1

=====
(configuration difference between cl061 and
cl138)
=====

[Front-end application tunables]
-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="946001"

.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_d)(Port= 1524))

[TP monitor tunables]
-----

.....
ubbconfig
.....

20c20
< cl061 LMID=SITE1
---
> cl138 LMID=SITE1

=====
(configuration difference between cl061 and
cl139)
=====

[Front-end application tunables]

```

```

-----

.....
tpapl.conf
.....

2c2
< Term_Base="286001"
---
> Term_Base="1188001"

.....
tnsnames.ora
.....

13c13
< (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_b)(Port= 1522))
---
> (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_e)(Port= 1525))

[TP monitor tunables]
-----

.....
ubbconfig
.....

20c20
< cl061 LMID=SITE1
---
> cl139 LMID=SITE1

=====
(cl069 configuration)
=====

[OS tunables]
-----

.....
chkconfig
.....

ypxfrd      0:off 1:off 2:off 3:off 4:off 5:off
6:off
saslauthd   0:off 1:off 2:off 3:off 4:off
5:off 6:off
snmptrapd   0:off 1:off 2:off 3:off 4:off
5:off 6:off
squid        0:off 1:off 2:off 3:off 4:off 5:off
6:off
zebra        0:off 1:off 2:off 3:off 4:off 5:off
6:off
dovecot     0:off 1:off 2:off 3:off 4:off
5:off 6:off
bgpd        0:off 1:off 2:off 3:off 4:off 5:off
6:off
ibmasm      0:off 1:off 2:off 3:off 4:off
5:off 6:off
ldap        0:off 1:off 2:off 3:off 4:off 5:off
6:off
spamassassin 0:off 1:off 2:off 3:off 4:off
5:off 6:off
readahead_early 0:off 1:off 2:off 3:off 4:off
5:on 6:off
irqbalance  0:off 1:off 2:off 3:on 4:on
5:on 6:off

```

```

winbind 0:off 1:off 2:off 3:off 4:off 5:off
6:off
messagebus 0:off 1:off 2:off 3:off 4:off
5:off 6:off
mdmmonitor 0:off 1:off 2:off 3:off 4:off
5:off 6:off
cron 0:off 1:off 2:on 3:on 4:on
5:on 6:off
nfslock 0:off 1:off 2:off 3:off 4:off 5:off
6:off
pcmcia 0:off 1:off 2:off 3:off 4:off
5:off 6:off
arpwatch 0:off 1:off 2:off 3:off 4:off
5:off 6:off
rpcidmapd 0:off 1:off 2:off 3:off 4:off
5:off 6:off
xinetd 0:off 1:off 2:off 3:on 4:on
5:on 6:off
multipathd 0:off 1:off 2:off 3:off 4:off
5:off 6:off
rarpd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
xfs 0:off 1:off 2:off 3:off 4:off 5:off
6:off
ospf6d 0:off 1:off 2:off 3:off 4:off 5:off
6:off
iptables 0:off 1:off 2:on 3:on 4:on
5:on 6:off
ypbind 0:off 1:off 2:off 3:off 4:off 5:off
6:off
ntpd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
haldaemon 0:off 1:off 2:off 3:off 4:off
5:off 6:off
snmpd 0:off 1:off 2:off 3:off 4:off
5:off 6:off
atd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
NetworkManager 0:off 1:off 2:off 3:off 4:off
5:off 6:off
irda 0:off 1:off 2:off 3:off 4:off 5:off
6:off
krb5kdc 0:off 1:off 2:off 3:off 4:off
5:off 6:off
yppasswdd 0:off 1:off 2:off 3:off 4:off
5:off 6:off
gpm 0:off 1:off 2:off 3:off 4:off 5:off
6:off
rusersd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
canna 0:off 1:off 2:off 3:off 4:off 5:off
6:off
mysqld 0:off 1:off 2:off 3:off 4:off
5:off 6:off
sshd 0:off 1:off 2:on 3:on 4:on
5:on 6:off
kadmin 0:off 1:off 2:off 3:off 4:off
5:off 6:off
postgresql 0:off 1:off 2:off 3:off 4:off
5:off 6:off
acpid 0:off 1:off 2:off 3:off 4:off 5:off
6:off
hpoj 0:off 1:off 2:off 3:off 4:off 5:off
6:off
apmd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
dhcrelay 0:off 1:off 2:off 3:off 4:off
5:off 6:off
isdn 0:off 1:off 2:off 3:off 4:off 5:off
6:off
diskdump 0:off 1:off 2:off 3:off 4:off
5:off 6:off
rwhod 0:off 1:off 2:off 3:off 4:off 5:off
6:off

```

```

kprop 0:off 1:off 2:off 3:off 4:off 5:off
6:off
cyrus-imapd 0:off 1:off 2:off 3:off 4:off
5:off 6:off
amd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
krb524 0:off 1:off 2:off 3:off 4:off 5:off
6:off
arptables_jf 0:off 1:off 2:off 3:off 4:off
5:off 6:off
microcode_ctl 0:off 1:off 2:off 3:off 4:off
5:off 6:off
psacct 0:off 1:off 2:off 3:off 4:off 5:off
6:off
ip6tables 0:off 1:off 2:off 3:off 4:off
5:off 6:off
cpuspeed 0:off 1:on 2:on 3:on 4:on
5:on 6:off
autofs 0:off 1:off 2:off 3:off 4:off 5:off
6:off
portmap 0:off 1:off 2:off 3:off 4:off
5:off 6:off
readahead 0:off 1:off 2:off 3:off 4:off
5:on 6:off
dc_client 0:off 1:off 2:off 3:off 4:off 5:off
6:off
cups 0:off 1:off 2:off 3:off 4:off 5:off
6:off
tog-pegasus 0:off 1:off 2:off 3:off 4:off
5:off 6:off
netfs 0:off 1:off 2:off 3:off 4:off 5:off
6:off
ripd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
vsftpd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
cups-config-daemon 0:off 1:off 2:off 3:off
4:off 5:off 6:off
dc_server 0:off 1:off 2:off 3:off 4:off
5:off 6:off
netplugd 0:off 1:off 2:off 3:off 4:off
5:off 6:off
anacron 0:off 1:off 2:off 3:off 4:off
5:off 6:off
bootparamd 0:off 1:off 2:off 3:off 4:off
5:off 6:off
mailman 0:off 1:off 2:off 3:off 4:off
5:off 6:off
sendmail 0:off 1:off 2:off 3:off 4:off
5:off 6:off
iscsi 0:off 1:off 2:off 3:off 4:off 5:off
6:off
rawdevices 0:off 1:off 2:off 3:on 4:on
5:on 6:off
network 0:off 1:off 2:on 3:on 4:on
5:on 6:off
smartd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
sysstat 0:off 1:on 2:off 3:off 4:off 5:off
6:off
dhcpcd 0:off 1:off 2:off 3:off 4:off
5:off 6:off
auditd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
rstatd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
bluetooth 0:off 1:off 2:off 3:off 4:off
5:off 6:off
iiim 0:off 1:off 2:off 3:off 4:off 5:off
6:off
radiusd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
odddjobd 0:off 1:off 2:off 3:off 4:off
5:off 6:off

```

```

ypserv 0:off 1:off 2:off 3:off 4:off 5:off
6:off
nfs 0:off 1:off 2:off 3:off 4:off 5:off
6:off
vncserver 0:off 1:off 2:off 3:off 4:off
5:off 6:off
mdm 0:off 1:off 2:off 3:off 4:off
5:off 6:off
rhnssd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
named 0:off 1:off 2:off 3:off 4:off
5:off 6:off
FreeWnn 0:off 1:off 2:off 3:off 4:off
5:off 6:off
netdump-server 0:off 1:off 2:off 3:off 4:off
5:off 6:off
kudzu 0:off 1:off 2:off 3:off 4:off 5:off
6:off
ipmi 0:off 1:off 2:off 3:off 4:off 5:off
6:off
lisa 0:off 1:off 2:off 3:off 4:off 5:off
6:off
smb 0:off 1:off 2:off 3:off 4:off 5:off
6:off
lux 0:off 1:off 2:off 3:off 4:off 5:off
6:off
netdump 0:off 1:off 2:off 3:off 4:off
5:off 6:off
httpd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
ospfd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
dhcpcd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
radvd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
innd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
openibd 0:off 1:off 2:off 3:off 4:off
5:off 6:off
lm_sensors 0:off 1:off 2:off 3:off 4:off
5:off 6:off
rpcgssd 0:off 1:off 2:off 3:off 4:off
5:off 6:off
ripngd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
nscd 0:off 1:off 2:off 3:off 4:off 5:off
6:off
syslog 0:off 1:off 2:on 3:on 4:on
5:on 6:off
xinetd based services:
  ktalk: off
  chargen-udp: off
  krb5-telnet: off
  amandaidx: off
  time: off
  klogin: off
  rlogin: on
  cups-lpd: off
  amidxtape: off
  dbsskd-cdb: off
  talk: off
  auth: on
  time-udp: off
  kshell: off
  chargen: off
  daytime: off
  amanda: off
  eklogin: off
  echo: off
  rsync: off
  daytime-udp: off
  gssftp: off
  swat: off
  finger: off

```

```

ftp: off
ntalk: off
telnet: on
rsh: on
echo-udp: off
rexec: off

.....
limits.conf
.....

# /etc/security/limits.conf
#
#Each line describes a limit for a user in the
form:
#
#<domain> <type> <item> <value>
#
#Where:
#<domain> can be:
# - an user name
# - a group name, with @group syntax
# - the wildcard *, for default entry
# - the wildcard %, can be also used
with %group syntax,
# for maxlogin limit
#
#<type> can have the two values:
# - "soft" for enforcing the soft limits
# - "hard" for enforcing hard limits
#
#<item> can be one of the following:
# - core - limits the core file size (KB)
# - data - max data size (KB)
# - fsize - maximum filesize (KB)
# - memlock - max locked-in-memory
address space (KB)
# - nfile - max number of open files
# - rss - max resident set size (KB)
# - stack - max stack size (KB)
# - cpu - max CPU time (MIN)
# - nproc - max number of processes
# - as - address space limit
# - maxlogins - max number of logins for this
user
# - maxsyslogins - max number of logins on
the system
# - priority - the priority to run user process
with
# - locks - max number of file locks the user
can hold
# - sigpending - max number of pending
signals
# - msgqueue - max memory used by
POSIX message queues (bytes)
#
#<domain> <type> <item> <value>
#

#* soft core 0
#* hard rss 10000
#@student hard nproc 20
#@faculty soft nproc 20
#@faculty hard nproc 50
#ftp hard nproc 0
#@student - maxlogins 4
tpc - nfile 30000
tpc - nproc 30000

# End of file

.....
sysctl.conf

```

```

.....
# Kernel sysctl configuration file for Red Hat
Linux
#
# For binary values, 0 is disabled, 1 is enabled.
See sysctl(8) and
# sysctl.conf(5) for more details.

# Controls IP packet forwarding
net.ipv4.ip_forward = 0

# Controls source route verification
net.ipv4.conf.default.rp_filter = 1

# Controls the System Request debugging
functionality of the kernel
kernel.sysrq = 0

# Controls whether core dumps will append the
PID to the core filename.
# Useful for debugging multi-threaded
applications.
kernel.core_uses_pid = 1

# Change filedescriptor
fs.file-max = 30000

# Change Message queue
kernel.msgmni = 30000
kernel.msgmnb = 1536000

# Change Max process
kernel.threads-max = 30000

# Change Semaphore
kernel.sem = 3000 384000 32 128

# Change TCP/IP backlog
net.ipv4.tcp_max_syn_backlog = 4096

[HTTP server tunables]
.....

.....
apache_cl_start.sh
.....

#!/bin/sh
export
LD_LIBRARY_PATH=${ORACLE_HOME}/srvm/li
b:${ORACLE_HOME}/lib64:${ORACLE_HOME}/lib:
/usr/lib:${ORACLE_HOME}/rdbsms/lib:${ORACLE_
HOME}/network/lib:$TUXDIR/lib

ulimit -u 30000
ulimit -s 1536

# /sbin/swapoff -a

# For 3tier tune
SVRAPL='ps -e | grep tpcfmlw | awk '{print $1}'
'/usr/bin/renice -20 -p ${SVRAPL}'

#rm -f /home/tpc/sar.tmp
/home/tpc/sar: hostname`
#/usr/lib/sa/sadc 5 > /home/tpc/sar.tmp &
# For 3tier tune

apachectl start

```

```

.....
httpd.conf
.....

#
# Based upon the NCSA server configuration
files originally by Rob McCool.
#
# This is the main Apache server configuration
file. It contains the
# configuration directives that give the server its
instructions.
# See <URL:http://httpd.apache.org/docs-2.0/>
for detailed information about
# the directives.
#
# Do NOT simply read the instructions in here
without understanding
# what they do. They're here only as hints or
reminders. If you are unsure
# consult the online docs. You have been
warned.
#
# The configuration directives are grouped into
three basic sections:
# 1. Directives that control the operation of the
Apache server process as a
# whole (the 'global environment').
# 2. Directives that define the parameters of the
'main' or 'default' server,
# which responds to requests that aren't
handled by a virtual host.
# These directives also provide default values
for the settings
# of all virtual hosts.
# 3. Settings for virtual hosts, which allow Web
requests to be sent to
# different IP addresses or hostnames and
have them handled by the
# same Apache server process.
#
# Configuration and logfile names: If the
filenames you specify for many
# of the server's control files begin with "/" (or
"drive:/" for Win32), the
# server will use that explicit path. If the
filenames do *not* begin
# with "/", the value of ServerRoot is prepended -
- so "logs/foo.log"
# with ServerRoot set to "/etc/httpd" will be
interpreted by the
# server as "/etc/httpd/logs/foo.log".
#

### Section 1: Global Environment
#
# The directives in this section affect the overall
operation of Apache,
# such as the number of concurrent requests it
can handle or where it
# can find its configuration files.
#

#
# Don't give away too much information about all
the subcomponents
# we are running. Comment out this line if you
don't mind remote sites
# finding out what major optional modules you
are running
#ServerTokens OS
ServerTokens Productly

#

```

```
# ServerRoot: The top of the directory tree under
which the server's
# configuration, error, and log files are kept.
#
# NOTE! If you intend to place this on an NFS
(or otherwise network)
# mounted filesystem then please read the
LockFile documentation
# (available at
<URL:http://httpd.apache.org/docs-
2.0/mod/core.html#lockfile>);
# you will save yourself a lot of trouble.
#
# Do NOT add a slash at the end of the directory
path.
#
ServerRoot "/etc/httpd"

#
# ScoreBoardFile: File used to store internal
server process information.
# If unspecified (the default), the scoreboard will
be stored in an
# anonymous shared memory segment, and will
be unavailable to third-party
# applications.
# If specified, ensure that no two invocations of
Apache share the same
# scoreboard file. The scoreboard file MUST BE
STORED ON A LOCAL DISK.
#
#ScoreBoardFile run/httpd.scoreboard

#
# PidFile: The file in which the server should
record its process
# identification number when it starts.
#
PidFile run/httpd.pid

#
# Timeout: The number of seconds before
receives and sends time out.
#
#Timeout 300
Timeout 999

#
# KeepAlive: Whether or not to allow persistent
connections (more than
# one request per connection). Set to "Off" to
deactivate.
#
#KeepAlive Off
KeepAlive On

#
# MaxKeepAliveRequests: The maximum
number of requests to allow
# during a persistent connection. Set to 0 to
allow an unlimited amount.
# We recommend you leave this number high,
for maximum performance.
#
#MaxKeepAliveRequests 100
MaxKeepAliveRequests 0

#
# KeepAliveTimeout: Number of seconds to wait
for the next request from the
# same client on the same connection.
#
#KeepAliveTimeout 15
KeepAliveTimeout 999
```

```
##
## Server-Pool Size Regulation (MPM specific)
##

# prefork MPM
# StartServers: number of server processes to
start
# MinSpareServers: minimum number of server
processes which are kept spare
# MaxSpareServers: maximum number of server
processes which are kept spare
# MaxClients: maximum number of server
processes allowed to start
# MaxRequestsPerChild: maximum number of
requests a server process serves
<IfModule prefork.c>
StartServers 8
MinSpareServers 5
MaxSpareServers 20
MaxClients 150
MaxRequestsPerChild 1000
</IfModule>

# worker MPM
# StartServers: initial number of server
processes to start
# MaxClients: maximum number of simultaneous
client connections
# MinSpareThreads: minimum number of worker
threads which are kept spare
# MaxSpareThreads: maximum number of
worker threads which are kept spare
# ThreadsPerChild: constant number of worker
threads in each server process
# MaxRequestsPerChild: maximum number of
requests a server process serves
<IfModule worker.c>

StartServers 45
ServerLimit 45
ThreadLimit 500
MaxClients 22500
MinSpareThreads 1
MaxSpareThreads 22500
ThreadsPerChild 500
MaxRequestsPerChild 0

#
#
# To reduce memory usage in the worker MPM,
the thread guard page
#
# To reduce memory usage in the worker MPM,
the thread guard page
# can be disabled, at the expense of some
protection against stack
# overflow.
#
#ThreadGuardArea off

</IfModule>

#
# Listen: Allows you to bind Apache to specific
IP addresses and/or
# ports, in addition to the default. See also the
<VirtualHost>
# directive.
#
# Change this to Listen on specific IP addresses
as shown below to
# prevent Apache from glomming onto all bound
IP addresses (0.0.0.0)
# e.g. "Listen 12.34.56.78:80"
```

```
#
# To allow connections to IPv6 addresses add
"Listen [::]:80"
#
Listen 0.0.0.0:80

#
# Dynamic Shared Object (DSO) Support
#

# To be able to use the functionality of a module
which was built as a DSO you
# have to place corresponding 'LoadModule'
lines at this location so the
# directives contained in it are actually available
_before_ they are used.
# Statically compiled modules (those listed by
`httpd -l`) do not need
# to be loaded here.
#
# Example:
# LoadModule foo_module modules/mod_foo.so
#
LoadModule tpapl_module
modules/mod_tpapl.so
LoadModule access_module
modules/mod_access.so
LoadModule status_module
modules/mod_status.so
LoadModule alias_module
modules/mod_alias.so
LoadModule cgi_module modules/mod_cgi.so

#
# Load config files from the config directory
"/etc/httpd/conf.d".
#
#Include conf.d/*.conf

#
# ExtendedStatus controls whether Apache will
generate "full" status
# information (ExtendedStatus On) or just basic
information (ExtendedStatus
# Off) when the "server-status" handler is called.
The default is Off.
#
#ExtendedStatus On

### Section 2: 'Main' server configuration
#
# The directives in this section set up the values
used by the 'main'
# server, which responds to any requests that
aren't handled by a
# <VirtualHost> definition. These values also
provide defaults for
# any <VirtualHost> containers you may define
later in the file.
#
# All of these directives may appear inside
<VirtualHost> containers,
# in which case these default settings will be
overridden for the
# virtual host being defined.
#
#
# If you wish httpd to run as a different user or
group, you must run
# httpd as root initially and it will switch.
#
# User/Group: The name (or #number) of the
user/group to run httpd as.
```

```

# . On SCO (ODT 3) use "User nouser" and
"Group nogroup".
# . On HP/UX you may not be able to use
shared memory as nobody, and the
# suggested workaround is to create a user
www and use that user.
# NOTE that some kernels refuse to
setgid(Group) or semctl(IPC_SET)
# when the value of (unsigned)Group is above
60000;
# don't use Group #-1 on these systems!
#
#User apache
#Group apache
User tpc
Group tpc

#
# ServerAdmin: Your address, where problems
with the server should be
# e-mailed. This address appears on some
server-generated pages, such
# as error documents. e.g. admin@your-
domain.com
#
ServerAdmin root@localhost

#
# ServerName gives the name and port that the
server uses to identify itself.
# This can often be determined automatically,
but we recommend you specify
# it explicitly to prevent problems during startup.
#
# If this is not set to valid DNS name for your
host, server-generated
# redirections will not work. See also the
UseCanonicalName directive.
#
# If your host doesn't have a registered DNS
name, enter its IP address here.
# You will have to access it by its address
anyway, and this will make
# redirections work in a sensible way.
#
#ServerName new.host.name:80
ServerName tpccserver:80

#
# UseCanonicalName: Determines how Apache
constructs self-referencing
# URLs and the SERVER_NAME and
SERVER_PORT variables.
# When set "Off", Apache will use the Hostname
and Port supplied
# by the client. When set "On", Apache will use
the value of the
# ServerName directive.
#
UseCanonicalName Off

#
# DocumentRoot: The directory out of which you
will serve your
# documents. By default, all requests are taken
from this directory, but
# symbolic links and aliases may be used to
point to other locations.
#
#DocumentRoot "/var/www/html"

#
# Each directory to which Apache has access
can be configured with respect

```

```

# to which services and features are allowed
and/or disabled in that
# directory (and its subdirectories).
#
# First, we configure the "default" to be a very
restrictive set of
# features.
#
#<Directory />
# Options FollowSymLinks
# AllowOverride None
#</Directory>

#
# Note that from this point forward you must
specifically allow
# particular features to be enabled - so if
something's not working as
# you might expect, make sure that you have
specifically enabled it
# below.
#

#
# UserDir: The name of the directory that is
appended onto a user's home
# directory if a -user request is received.
#
# The path to the end user account 'public_html'
directory must be
# accessible to the webserver userid. This
usually means that ~userid
# must have permissions of 711,
~userid/public_html must have permissions
# of 755, and documents contained therein must
be world-readable.
# Otherwise, the client will only receive a "403
Forbidden" message.
#
# See also:
http://httpd.apache.org/docs/misc/FAQ.html#forbidden
#
#<IfModule mod_userdir.c>
#
# UserDir is disabled by default since it can
confirm the presence
# of a username on the system (depending on
home directory
# permissions).
#
# UserDir disable

#
# To enable requests to /-user/ to serve the
user's public_html
# directory, remove the "UserDir disable" line
above, and uncomment
# the following line instead:
#
#UserDir public_html

#</IfModule>

#
# Control access to UserDir directories. The
following is an example
# for a site where these directories are restricted
to read-only.
#
#<Directory /home/*public_html>
# AllowOverride FileInfo AuthConfig Limit
# Options MultiViews Indexes
SymLinksIfOwnerMatch IncludesNoExec
# <Limit GET POST OPTIONS>

```

```

# Order allow,deny
# Allow from all
# </Limit>
# <LimitExcept GET POST OPTIONS>
# Order deny,allow
# Deny from all
# </LimitExcept>
#</Directory>

#
# DirectoryIndex: sets the file that Apache will
serve if a directory
# is requested.
#
# The index.html.var file (a type-map) is used to
deliver content-
# negotiated documents. The MultiViews Option
can be used for the
# same purpose, but it is much slower.
#

#
# AccessFileName: The name of the file to look
for in each directory
# for additional configuration directives. See
also the AllowOverride
# directive.
#
AccessFileName .htaccess

#
# The following lines prevent .htaccess
and .htpasswd files from being
# viewed by Web clients.
#

#
# TypesConfig describes where the mime.types
file (or equivalent) is
# to be found.
#

#
# DefaultType is the default MIME type the
server will use for a document
# if it cannot otherwise determine one, such as
from filename extensions.
# If your server contains mostly text or HTML
documents, "text/plain" is
# a good value. If most of your content is binary,
such as applications
# or images, you may want to use
"application/octet-stream" instead to
# keep browsers from trying to display binary
files as though they are
# text.
#
DefaultType text/plain

#
# The mod_mime_magic module allows the
server to use various hints from the
# contents of the file itself to determine its type.
The MIMEMagicFile
# directive tells the module where the hint
definitions are located.
#
#<IfModule mod_mime_magic.c>
## MIMEMagicFile /usr/share/magic.mime
# MIMEMagicFile conf/magic
#</IfModule>

#
# HostnameLookups: Log the names of clients
or just their IP addresses

```



```
# e.g., www.apache.org (on) or 204.62.129.132
(off).
# The default is off because it'd be overall better
for the net if people
# had to knowingly turn this feature on, since
enabling it means that
# each client request will result in AT LEAST one
lookup request to the
# nameserver.
#
HostnameLookups Off

#
# EnableMMAP: Control whether memory-
mapping is used to deliver
# files (assuming that the underlying OS
supports it).
# The default is on; turn this off if you serve from
NFS-mounted
# filesystems. On some systems, turning it off
(regardless of
# filesystem) can improve performance; for
details, please see
# http://httpd.apache.org/docs-
2.0/mod/core.html#enablemmap
#
#EnableMMAP off

#
# EnableSendfile: Control whether the sendfile
kernel support is
# used to deliver files (assuming that the OS
supports it).
# The default is on; turn this off if you serve from
NFS-mounted
# filesystems. Please see
# http://httpd.apache.org/docs-
2.0/mod/core.html#enablesendfile
#
#EnableSendfile off
#

#
# ErrorLog: The location of the error log file.
# If you do not specify an ErrorLog directive
within a <VirtualHost>
# container, error messages relating to that
virtual host will be
# logged here. If you *do* define an error logfile
for a <VirtualHost>
# container, that host's errors will be logged
there and not here.
#
ErrorLog logs/error_log

#
# LogLevel: Control the number of messages
logged to the error_log.
# Possible values include: debug, info, notice,
warn, error, crit,
# alert, emerg.
#
LogLevel warn

#
# The following directives define some format
nicknames for use with
# a CustomLog directive (see below).
#

#
# The location and format of the access logfile
(Common Logfile Format).
# If you do not define any access logfiles within
a <VirtualHost>
```

```
# container, they will be logged here.
Contrariwise, if you *do*
# define per-<VirtualHost> access logfiles,
transactions will be
# logged therein and *not* in this file.
#
# CustomLog logs/access_log common
#CustomLog logs/access_log combined

#
# If you would like to have agent and referer
logfile, uncomment the
# following directives.
#
#CustomLog logs/referer_log referer
#CustomLog logs/agent_log agent

#
# If you prefer a single logfile with access, agent,
and referer information
# (Combined Logfile Format) you can use the
following directive.
#
#CustomLog logs/access_log combined

#
# Optionally add a line containing the server
version and virtual host
# name to server-generated pages (error
documents, FTP directory listings,
# mod_status and mod_info output etc., but not
CGI generated documents).
# Set to "EMail" to also include a mailto: link to
the ServerAdmin.
# Set to one of: On | Off | EMail
#
#ServerSignature On
ServerSignature Off

#
# Aliases: Add here as many aliases as you
need (with no limit). The format is
# Alias fakename realname
#
# Note that if you include a trailing / on fakename
then the server will
# require it to be present in the URL. So "/icons"
isn't aliased in this
# example, only "/icons/". If the fakename is
slash-terminated, then the
# realname must also be slash terminated, and if
the fakename omits the
# trailing slash, the realname must also omit it.
#
# We include the /icons/ alias for FancyIndexed
directory listings. If you
# do not use FancyIndexing, you may comment
this out.
#
#
# This should be changed to the
ServerRoot/manual/. The alias provides
# the manual, even if you choose to move your
DocumentRoot. You may comment
# this out if you do not care for the
documentation.
#
#<IfModule mod_dav_fs.c>
# # Location of the WebDAV lock database.
# DAVLockDB /var/lib/dav/lockdb
#</IfModule>

#
```

```
# ScriptAlias: This controls which directories
contain server scripts.
# ScriptAliases are essentially the same as
Aliases, except that
# documents in the realname directory are
treated as applications and
# run by the server when requested rather than
as documents sent to the client.
# The same rules about trailing "/" apply to
ScriptAlias directives as to
# Alias.
#
#ScriptAlias /cgi-bin/ "/var/www/cgi-bin/"
ScriptAlias /cgi-bin/ "/home/tpc/tool/bin/"

#
# "/var/www/cgi-bin" should be changed to
whatever your ScriptAliased
# CGI directory exists, if you have that
configured.
#
<Directory "/var/www/cgi-bin">
    AllowOverride None
    Options None
    Order allow,deny
    Allow from all
</Directory>

#
# Redirect allows you to tell clients about
documents which used to exist in
# your server's namespace, but do not anymore.
This allows you to tell the
# clients where to look for the relocated
document.
# Example:
# Redirect permanent /foo
http://www.example.com/bar

#
# Directives controlling the display of server-
generated directory listings.
#

#
# FancyIndexing is whether you want fancy
directory indexing or standard.
# VersionSort is whether files containing version
numbers should be
# compared in the natural way, so that `apache-
1.3.9.tar' is placed before
# `apache-1.3.12.tar'.
#

#
# AddIcon* directives tell the server which icon
to show for different
# files or filename extensions. These are only
displayed for
# FancyIndexed directories.
#

#
# DefaultIcon is which icon to show for files
which do not have an icon
# explicitly set.
#

#
# AddDescription allows you to place a short
description after a file in
# server-generated indexes. These are only
displayed for FancyIndexed
# directories.
```

```
# Format: AddDescription "description" filename
#
#AddDescription "GZIP compressed
document" .gz
#AddDescription "tar archive" .tar
#AddDescription "GZIP compressed tar
archive" .tgz

#
# ReadmeName is the name of the README file
the server will look for by
# default, and append to directory listings.
#
# HeaderName is the name of a file which
should be prepended to
# directory indexes.

#
# IndexIgnore is a set of filenames which
directory indexing should ignore
# and not include in the listing. Shell-style
wildcarding is permitted.
#

#
# AddEncoding allows you to have certain
browsers (Mosaic/X 2.1+) uncompress
# information on the fly. Note: Not all browsers
support this.
# Despite the name similarity, the following Add*
directives have nothing
# to do with the FancyIndexing customization
directives above.
#

#
# DefaultLanguage and AddLanguage allows
you to specify the language of
# a document. You can then use content
negotiation to give a browser a
# file in a language the user can understand.
#
# Specify a default language. This means that all
data
# going out without a specific language tag (see
below) will
# be marked with this one. You probably do NOT
want to set
# this unless you are sure it is correct for all
cases.
#
# * It is generally better to not mark a page as
# * being a certain language than marking it with
the wrong
# * language!
#
# DefaultLanguage nl
#
# Note 1: The suffix does not have to be the
same as the language
# keyword --- those with documents in Polish
(whose net-standard
# language code is pl) may wish to use
"AddLanguage pl .po" to
# avoid the ambiguity with the common suffix for
perl scripts.
#
# Note 2: The example entries below illustrate
that in some cases
# the two character 'Language' abbreviation is
not identical to
# the two character 'Country' code for its country,
# E.g. 'Danmark/dk' versus 'Danish/da'.
#
```

```
# Note 3: In the case of 'ltz' we violate the RFC
by using a three char
# specifier. There is 'work in progress' to fix this
and get
# the reference data for rfc1766 cleaned up.
#
# Danish (da) - Dutch (nl) - English (en) -
Estonian (et)
# French (fr) - German (de) - Greek-Modern (el)
# Italian (it) - Norwegian (no) - Norwegian
Nynorsk (nn) - Korean (ko)
# Portugese (pt) - Luxembourggeois* (ltz)
# Spanish (es) - Swedish (sv) - Catalan (ca) -
Czech(cs)
# Polish (pl) - Brazilian Portuguese (pt-br) -
Japanese (ja)
# Russian (ru) - Croatian (hr)
#

#
# LanguagePriority allows you to give
precedence to some languages
# in case of a tie during content negotiation.
#
# Just list the languages in decreasing order of
preference. We have
# more or less alphabetized them here. You
probably want to change this.
#

#
# ForceLanguagePriority allows you to serve a
result page rather than
# MULTIPLE CHOICES (Prefer) [in case of a tie]
or NOT ACCEPTABLE (Fallback)
# [in case no accepted languages matched the
available variants]
#

#
# Specify a default charset for all pages sent out.
This is
# always a good idea and opens the door for
future internationalisation
# of your web site, should you ever want it.
Specifying it as
# a default does little harm; as the standard
dictates that a page
# is in iso-8859-1 (latin1) unless specified
otherwise i.e. you
# are merely stating the obvious. There are also
some security
# reasons in browsers, related to javascript and
URL parsing
# which encourage you to always set a default
char set.
#
AddDefaultCharset UTF-8

#
# Commonly used filename extensions to
character sets. You probably
# want to avoid clashes with the language
extensions, unless you
# are good at carefully testing your setup after
each change.
# See
http://www.iana.org/assignments/character-sets
for the
# official list of charset names and their
respective RFCs
#

#
```

```
# AddType allows you to add to or override the
MIME configuration
# file mime.types for specific file types.
#

#
# AddHandler allows you to map certain file
extensions to "handlers":
# actions unrelated to filetype. These can be
either built into the server
# or added with the Action directive (see below)
#
# To use CGI scripts outside of ScriptAliased
directories:
# (You will also need to add "ExecCGI" to the
"Options" directive.)
#
#AddHandler cgi-script .cgi

#
# For files that include their own HTTP headers:
#
#AddHandler send-as-is asis

#
# For server-parsed imagemap files:
#

#
# For type maps (negotiated resources):
# (This is enabled by default to allow the Apache
"It Worked" page
# to be distributed in multiple languages.)
#

# Filters allow you to process content before it is
sent to the client.
#
# To parse .shtml files for server-side includes
(SSl):
# (You will also need to add "Includes" to the
"Options" directive.)
#

#
# Action lets you define media types that will
execute a script whenever
# a matching file is called. This eliminates the
need for repeated URL
# pathnames for oft-used CGI file processors.
# Format: Action media/type /cgi-script/location
# Format: Action handler-name /cgi-
script/location
#

#
# Customizable error responses come in three
flavors:
# 1) plain text 2) local redirects 3) external
redirects
#
# Some examples:
#ErrorDocument 500 "The server made a boo
boo."
#ErrorDocument 404 /missing.html
#ErrorDocument 404 "/cgi-
bin/missing_handler.pl"
#ErrorDocument 402
http://www.example.com/subscription_info.html
#

#
# Putting this all together, we can
Internationalize error responses.
#
```

```
# We use Alias to redirect any
/error/HTTP_<error>.html.var response to
# our collection of by-error message multi-
language collections. We use
# includes to substitute the appropriate text.
#
# You can modify the messages' appearance
without changing any of the
# default HTTP_<error>.html.var files by adding
the line;
#
# Alias /error/include/ "/your/include/path/"
#
# which allows you to create your own set of files
by starting with the
# /var/www/error/include/ files and
# copying them to /your/include/path/, even on a
per-VirtualHost basis.
#
Alias /error/ "/var/www/error/"

# ErrorDocument 400
/error/HTTP_BAD_REQUEST.html.var
# ErrorDocument 401
/error/HTTP_UNAUTHORIZED.html.var
# ErrorDocument 403
/error/HTTP_FORBIDDEN.html.var
# ErrorDocument 404
/error/HTTP_NOT_FOUND.html.var
# ErrorDocument 405
/error/HTTP_METHOD_NOT_ALLOWED.html.v
ar
# ErrorDocument 408
/error/HTTP_REQUEST_TIME_OUT.html.var
# ErrorDocument 410
/error/HTTP_GONE.html.var
# ErrorDocument 411
/error/HTTP_LENGTH_REQUIRED.html.var
# ErrorDocument 412
/error/HTTP_PRECONDITION_FAILED.html.var
# ErrorDocument 413
/error/HTTP_REQUEST_ENTITY_TOO_LARGE
.html.var
# ErrorDocument 414
/error/HTTP_REQUEST_URI_TOO_LARGE.htm
l.var
# ErrorDocument 415
/error/HTTP_SERVICE_UNAVAILABLE.html.var
# ErrorDocument 500
/error/HTTP_INTERNAL_SERVER_ERROR.htm
l.var
# ErrorDocument 501
/error/HTTP_NOT_IMPLEMENTED.html.var
# ErrorDocument 502
/error/HTTP_BAD_GATEWAY.html.var
# ErrorDocument 503
/error/HTTP_SERVICE_UNAVAILABLE.html.var
# ErrorDocument 506
/error/HTTP_VARIANT_ALSO_VARIES.html.var

#
# The following directives modify normal HTTP
response behavior to
# handle known problems with browser
implementations.
#
#
# The following directive disables redirects on
non-GET requests for
# a directory that does not include the trailing
slash. This fixes a
```

```
# problem with Microsoft WebFolders which
does not appropriately handle
# redirects for folders with DAV methods.
# Same deal with Apple's DAV filesystem and
Gnome VFS support for DAV.
#
# Allow server status reports, with the URL of
http://servername/server-status
# Change the ".your-domain.com" to match your
domain to enable.
#
<Location /server-status>
    SetHandler server-status
    Order deny,allow
    Deny from all
    Allow from 192.168.
</Location>

#
# Allow remote server configuration reports, with
the URL of
# http://servername/server-info (requires that
mod_info.c be loaded).
# Change the ".example.com" to match your
domain to enable.
#
<Location /server-info>
    SetHandler server-info
    Order deny,allow
    Deny from all
    Allow from .example.com
</Location>

#
# Proxy Server directives. Uncomment the
following lines to
# enable the proxy server:
#
#<IfModule mod_proxy.c>
#ProxyRequests On
#
#<Proxy *>
# Order deny,allow
# Deny from all
# Allow from .example.com
#</Proxy>

#
# Enable/disable the handling of HTTP/1.1 "Via:"
headers.
# ("Full" adds the server version; "Block"
removes all outgoing Via: headers)
# Set to one of: Off | On | Full | Block
#
#ProxyVia On

#
# To enable a cache of proxied content,
uncomment the following lines.
# See http://httpd.apache.org/docs-
2.0/mod/mod_cache.html for more details.
#
<IfModule mod_disk_cache.c>
# CacheEnable disk /
# CacheRoot "/var/cache/mod_proxy"
#</IfModule>
#
#</IfModule>
# End of proxy directives.

### Section 3: Virtual Hosts
#
# VirtualHost: If you want to maintain multiple
domains/hostnames on your
```

```
# machine you can setup VirtualHost containers
for them. Most configurations
# use only name-based virtual hosts so the
server doesn't need to worry about
# IP addresses. This is indicated by the asterisks
in the directives below.
#
# Please see the documentation at
# <URL:http://httpd.apache.org/docs-
2.0/vhosts/>
# for further details before you try to setup virtual
hosts.
#
# You may use the command line option '-S' to
verify your virtual host
# configuration.

#
# Use name-based virtual hosting.
#
#NameVirtualHost *:80

#
# VirtualHost example:
# Almost any Apache directive may go into a
VirtualHost container.
# The first VirtualHost section is used for
requests without a known
# server name.
#
#<VirtualHost *>
# ServerAdmin webmaster@dummy-
host.example.com
# DocumentRoot /www/docs/dummy-
host.example.com
# ServerName dummy-host.example.com
# ErrorLog logs/dummy-host.example.com-
error_log
# CustomLog logs/dummy-host.example.com-
access_log common
#</VirtualHost>

#
# For TPAPL
#
<Location /tpapl>
    SetHandler tpapl
    TpaPlConf /home/tpc/conf/tpapl.conf
</Location>

[Front-end application tunables]
-----
.....
tpapl.conf
.....

[TPAPL_INFO]
Term_Base="1540001"
NumWarehouses="193600"
MaxUsers="1936000"
MaxTerm of Client="22000"
CONTROL_Flag="0"
LogPath="/home/tpc/log/userlog.log"

[SVRAPL_INFO]
LogPath="/home/tpc/log/DBDepend_Userlog.log"
"
.....
tnsnames.ora
```

```

.....

#
# Filename: Tnsnames.ora
#
extproc_connection_data =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = IPC)(KEY =
tpcc))
    (SDU=14600)
    (CONNECT_DATA = (SERVICE_NAME =
tpcc)
  )

tpcc =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL= TCP)(Host=
pqtpc_g)(Port= 1527))
    (SDU=14600)
    (CONNECT_DATA = (SERVICE_NAME =
tpcc)
  )

[TP monitor tunables]
.....

.....
ubconfig
.....

#
# ubconfig : TUXEDO configuration file
(WAREHOUSE BINED)
#

*RESOURCES
IPCKEY    211940
MASTER   SITE1
UID       500
GID       500
PERM      0660
MAXACCESSERS 1000
MAXSERVERS 100
MAXSERVICES 100
MAXGTT    0
MODEL     SHM
LDBAL     Y
OPTIONS   NO_AA,NO_XA

*MACHINES
cl069 LMID=SITE1
  APPDIR="/home/tpc/bin"
  TUXCONFIG="/home/tpc/conf/tuxconfig"
  TUXDIR="/usr/local/BEA/tuxedo8.1"
  ULOGPFX="/home/tpc/log/tuxedo.log"
  SICACHEENTRIESMAX="0"

*GROUPS
group1 LMID=SITE1 GRPNO=1

*SERVERS
DEFAULT: RESTART=Y MAXGEN=5
REPLYQ=N ROPERM=0660

tpccfmlw SRVGRP=group1 RQADDR=ware01
SRVID=1 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware02
SRVID=2 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware03
SRVID=3 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"

```

```

tpccfmlw SRVGRP=group1 RQADDR=ware04
SRVID=4 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware05
SRVID=5 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware06
SRVID=6 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware07
SRVID=7 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware08
SRVID=8 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware09
SRVID=9 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware10
SRVID=10 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware11
SRVID=11 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware12
SRVID=12 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware13
SRVID=13 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware14
SRVID=14 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware15
SRVID=15 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware16
SRVID=16 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware17
SRVID=17 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware18
SRVID=18 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware19
SRVID=19 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"
tpccfmlw SRVGRP=group1 RQADDR=ware20
SRVID=20 REPLYQ=N CLOPT="-s
OPSTUXSERVER:OPSTUXSERVER"

```

```

*SERVICES
"OPSTUXSERVER" TRANTIME=0
SRVGRP=group1

```

```

*ROUTING

```

Appendix E: Database Creation Code

```

.....
App_E/Makefile.linux
.....

#=====
# Copyright (c) 1996 Oracle Corp,
# Redwood Shores, CA |
# OPEN SYSTEMS PERFORMANCE
# GROUP |
# All Rights Reserved
# |
#=====
# FILENAME
# Makefile
# DESCRIPTION
# Makefile for lib for batch driver, load
# program and tx testing.
#=====
#
# Programs:
#
# dpblibunix.o
#

all: compile dpblibunix.o

#include
$(ORACLE_HOME)/bench/buildtools/prefix.mk
L_SYM=-l
#include
$(ORACLE_HOME)/rdbms/lib/env_rdbms.mk
REMOVE=rm
#CC=/opt/SunProd/SUNWspr06.1/bin/./WS6U1/
bin/cc
CC=/usr/bin/gcc

TARGETS=compile cleanup

TPCBIN=.
INCLUDE=$(L_SYM).
$(L_SYM)$(ORACLE_HOME)/rdbms/demo \
$(L_SYM)$(ORACLE_HOME)/rdbms/public \
$(L_SYM)$(ORACLE_HOME)/rdbms/include \
$(L_SYM)$(ORACLE_HOME)/plsql/public \
$(L_SYM)$(ORACLE_HOME)/network/public
ITUX=$(L_SYM)$(ROOTDIR)/include

MEMBS=
OBJJS=gettime.o dpbproc.o dpbwait.o dpbpchk.o
dpbtimef.o

CFLAGS=

files:

compile: $(OBJJS)
@-$(DOTARGETS)

```

```

cleanup:
$(REMOVE) $(OBJJS) dpblibunix.o

dpbtimef.o: dpbtimef.c
$(CC) $(CFLAGS) -DORA_PC $(INCLUDE) -
c dpbtimef.c

dpbproc.o: dpbproc.c
$(CC) $(CFLAGS) -DORA_AUX $(INCLUDE)
-c dpbproc.c

dpbwait.o: dpbwait.c
$(CC) $(CFLAGS) -DORA_AUX $(INCLUDE)
-c dpbwait.c

dpbpchk.o: dpbpchk.c
$(CC) $(CFLAGS) -DORA_AUX $(INCLUDE)
-c dpbpchk.c

gettime.o: gettime.c
$(CC) $(CFLAGS) $(INCLUDE) -c gettime.c

trigger.o: trigger.c

dpblibunix.o: $(OBJJS)
$(LD) -r -o @$ $(OBJJS)

c_trans_lux.o: $(CTRANTUX_OBJJS)
$(LD) -r -o @$ $(CTRANTUX_OBJJS)

.....
App_E/addfile.sh
.....

#!/bin/sh
# $1 = tablespace name
# $2 = filename
# $3 = size
# $4 = temporary ts (1) or not (0)
# global variable $tpcc_listfiles, does not
# execute sql

if expr x$tpcc_listfiles = xt > /dev/null; then
echo $2 $3 >> $tpcc_bench/files.dat
exit 0
fi

if expr $4 = 1 > /dev/null; then
altersql="alter tablespace $1 add tempfile '$2'
size $3 reuse;"
else
altersql="alter tablespace $1 add datafile '$2'
size $3 reuse autoextend on;"
fi

$tpcc_sqlplus $tpcc_user_pass <<!
spool addfile_$.log
set echo on
$altersql
set echo off
spool off
exit ;
!

.....
App_E/addts.sh
.....

#!/bin/sh
# $1 = tablespace name
# $2 = filename
# $3 = size

```

```

# $4 = uniform size
# $5 = block size
# $6 = temporary ts (1) or not (0)
# $7 = bitmapped manage (t) or not (f) or (d) for
# dictionary
# global variable $tpcc_listfiles, does not
# execute sql
# drop tablespace $1 including contents;

if expr x$tpcc_listfiles = xt > /dev/null; then
echo $2 $3 >> $tpcc_bench/files.dat
exit 0
fi

if expr $5 = auto > /dev/null; then
bssql=
else
bssql="blocksize $5"
fi

if expr $6 = 1 > /dev/null; then
createsql="create temporary tablespace $1
size $3 size $3 reuse extent management
local uniform size $4;"
else
if expr x$7 = xt > /dev/null; then
createsql="create tablespace $1 datafile '$2'
size $3 reuse extent management local uniform
size $4 segment space management auto
$bssql nologging ;"
else
if expr x$7 = xd > /dev/null; then
createsql="create tablespace $1 datafile '$2'
size $3 reuse extent management dictionary
nologging $bssql;"
else
createsql="create tablespace $1 datafile '$2'
size $3 reuse extent management local uniform
size $4 segment space management manual
$bssql nologging ;"
fi
fi
fi

$tpcc_sqlplus $tpcc_user_pass <<!
spool createts_$.log
set echo on
$createsql
set echo off
spool off
exit ;
!

.....
App_E/analyze.sh
.....

#!/bin/sh
$tpcc_sqlplus $tpcc_user_pass
@${tpcc_sql_dir}/analyze > $tpcc_log_dir/junk
2>&1

if test $? -ne 0
then
exit 1;
else
exit 0;
fi

.....
App_E/analyze.sql
.....

```

```

spool analyze.log;
set echo on;

connect tpcc/tpcc

execute dbms_stats.GATHER_TABLE_STATS
(OWNNAME=>'TPCC', -
      TABNAME=>'STOK', -
      PARTNAME=>NULL, -

ESTIMATE_PERCENT=>1, -

BLOCK_SAMPLE=>TRUE, -
      METHOD_OPT=>'FOR
ALL COLUMNS SIZE 1', -
      DEGREE=>10, -

GRANULARITY=>'DEFAULT', -
      CASCADE=>TRUE);

execute dbms_stats.GATHER_TABLE_STATS
(OWNNAME=>'TPCC', -
      TABNAME=>'CUST', -
      PARTNAME=>NULL, -

ESTIMATE_PERCENT=>1, -

BLOCK_SAMPLE=>TRUE, -
      METHOD_OPT=>'FOR
ALL COLUMNS SIZE 1', -
      DEGREE=>10, -

GRANULARITY=>'DEFAULT', -
      CASCADE=>TRUE);

execute dbms_stats.GATHER_TABLE_STATS
(OWNNAME=>'TPCC', -
      TABNAME=>'ORDL', -
      PARTNAME=>NULL, -

ESTIMATE_PERCENT=>1, -

BLOCK_SAMPLE=>TRUE, -
      METHOD_OPT=>'FOR
ALL COLUMNS SIZE 1', -
      DEGREE=>10, -

GRANULARITY=>'DEFAULT', -
      CASCADE=>TRUE);

execute dbms_stats.GATHER_TABLE_STATS
(OWNNAME=>'TPCC', -
      TABNAME=>'NORD', -
      PARTNAME=>NULL, -

ESTIMATE_PERCENT=>1, -

BLOCK_SAMPLE=>TRUE, -
      METHOD_OPT=>'FOR
ALL COLUMNS SIZE 1', -
      DEGREE=>10, -

GRANULARITY=>'DEFAULT', -
      CASCADE=>TRUE);

execute
dbms_stats.GATHER_TABLE_STATS(OWNNAM
E=>'TPCC', -
      TABNAME=>'HIST', -
      PARTNAME=>NULL, -

ESTIMATE_PERCENT=>1, -

BLOCK_SAMPLE=>TRUE, -

METHOD_OPT=>'FOR
ALL COLUMNS SIZE 1', -
      DEGREE=>10, -

GRANULARITY=>'DEFAULT', -
      CASCADE=>TRUE);

execute
dbms_stats.GATHER_TABLE_STATS(OWNNAM
E=>'TPCC', -
      TABNAME=>'DIST', -
      PARTNAME=>NULL, -

ESTIMATE_PERCENT=>1, -

BLOCK_SAMPLE=>TRUE, -
      METHOD_OPT=>'FOR
ALL COLUMNS SIZE 1', -
      DEGREE=>10, -

GRANULARITY=>'DEFAULT', -
      CASCADE=>TRUE);

execute
dbms_stats.GATHER_TABLE_STATS(OWNNAM
E=>'TPCC', -
      TABNAME=>'ITEM', -
      PARTNAME=>NULL, -

ESTIMATE_PERCENT=>10, -

BLOCK_SAMPLE=>TRUE, -
      METHOD_OPT=>'FOR
ALL COLUMNS SIZE 1', -
      DEGREE=>1, -

GRANULARITY=>'DEFAULT', -
      CASCADE=>TRUE);

execute
dbms_stats.GATHER_TABLE_STATS(OWNNAM
E=>'TPCC', -
      TABNAME=>'WARE', -
      PARTNAME=>NULL, -

ESTIMATE_PERCENT=>10, -

BLOCK_SAMPLE=>TRUE, -
      METHOD_OPT=>'FOR
ALL COLUMNS SIZE 1', -
      DEGREE=>10, -

GRANULARITY=>'DEFAULT', -
      CASCADE=>TRUE);

set echo off;
spool off;

exit sql.sqlcode;

.....
App_E/assigntemp.sh
.....

#!/bin/sh

echo Assigning temporary tablespace to user
tpcc...
$tpcc_sqlplus $tpcc_dba_user_pass
@$tpcc_sql_dir/assigntemp > junk 2>&1
if test $? -ne 0

then
exit 1;
else
exit 0;
fi

.....
App_E/assigntemp.sql
.....

spool assigntemp.log;

set echo on;

alter user tpcc temporary tablespace temp_0;

set echo off;
spool off;

exit ;

.....
App_E/bcexpr.sh
.....

#!/bin/sh
# send command line to bc
echo "$*" | bc

.....
App_E/createdb.sql
.....

spool createdb.log

set echo on

shutdown abort

startup pfile=p_create.ora nomount
create database tpcc
controlfile reuse
maxinstances 1
datafile
'ora_dev/system_1' size 400M reuse
logfile 'ora_dev/log_1_1' size 8192M reuse,
'ora_dev/log_1_2' size 8192M reuse
sysaux datafile 'ora_dev/tpccaux' size 120M
reuse ;

create undo tablespace undo_1 datafile
'ora_dev/roll1' size 8096M reuse blocksize 8K;

set echo off
exit sql.sqlcode
logfile 'ora_dev/log_1_1' size 376523M reuse,
'ora_dev/log_1_2' size 376523M reuse

.....
App_E/createindex_icust1.sql
.....

set timing on
set sqlblanklines on
spool createindex_icust1.log ;
set echo on ;
drop index icust1 ;
create unique index icust1 on cust ( c_w_id

```

```

, c_d_id
, c_id )
pctfree 1 initrans 3
storage ( buffer_pool default )
parallel 64
compute statistics
tablespace icust1_0 ;
set echo off
spool off
exit sql.sqlcode;

.....
App_E/createindex_icust2.sql
.....

set timing on
set sqlblanklines on
spool createindex_icust2.log ;
set echo on ;
drop index icust2 ;
create unique index icust2 on cust ( c_last

, c_w_id
, c_d_id
, c_first
, c_id )
pctfree 1 initrans 3
storage ( buffer_pool default )
parallel 64
compute statistics
tablespace icust2_0 ;
set echo off
spool off
exit sql.sqlcode;

.....
App_E/createindex_idist.sql
.....

set timing on
set sqlblanklines on
spool createindex_idist.log ;
set echo on ;
drop index idist ;
create unique index idist on dist ( d_w_id
, d_id )
pctfree 5 initrans 3
storage ( buffer_pool default )
parallel 1
compute statistics
tablespace idist_0 ;
set echo off
spool off
exit sql.sqlcode;

.....
App_E/createindex_iitem.sql
.....

set timing on
set sqlblanklines on
spool createindex_iitem.log ;
set echo on ;
drop index iitem ;
create unique index iitem on item ( i_id )
pctfree 5 initrans 4
storage ( buffer_pool default )

compute statistics
tablespace iitem_0 ;
set echo off
spool off
exit sql.sqlcode;

.....
App_E/createindex_inord.sql
.....

set timing on
exit 0;

.....
App_E/createindex_iord1.sql
.....

set timing on
exit 0;

.....
App_E/createindex_iord1.sql
.....

set timing on
exit 0;

.....
App_E/createindex_iord2.sql
.....

set timing on
set sqlblanklines on
spool createindex_iord2.log ;
set echo on ;
drop index iordr2 ;
create unique index iordr2 on ordr (
o_w_id
, o_c_id
, o_d_id
, o_id )
global partition by range (o_w_id) (
partition iordr2_1 values less than (26040) ,
partition iordr2_2 values less than (52080) ,
partition iordr2_3 values less than (78120) ,
partition iordr2_4 values less than (104160) ,
partition iordr2_5 values less than (130200) ,
partition iordr2_6 values less than (156240) ,
partition iordr2_7 values less than (182280) ,
partition iordr2_8 values less than (MAXVALUE)
)
pctfree 25 initrans 4
storage ( buffer_pool default )
parallel 256
compute statistics
tablespace iordr2_0 ;
alter index iordr2 noparallel;
set echo off
spool off
exit sql.sqlcode;

.....
App_E/createindex_istok.sql
.....

set timing on
set sqlblanklines on
spool createindex_istok.log ;
set echo on ;
drop index istok ;
create unique index istok on stok ( s_i_id
, s_w_id )
pctfree 1 initrans 3
storage ( buffer_pool default )
parallel 512

compute statistics
tablespace istok_0 ;
set echo off
spool off
exit sql.sqlcode;

.....
App_E/createindex_iware.sql
.....

set timing on
set sqlblanklines on
spool createindex_iware.log ;
set echo on ;
drop index iware ;
create unique index iware on ware ( w_id )
pctfree 1 initrans 3
storage ( buffer_pool default )
parallel 1
compute statistics
tablespace iware_0 ;
set echo off
spool off
exit sql.sqlcode;

.....
App_E/createmisc.sh
.....

#!/bin/sh

$tpcc_sqlplus $tpcc_sqlplus_args << !
$tpcc_internal_connect

spool createmisc.log
set echo on;
alter user tpcc temporary tablespace system;
grant execute on dbms_lock to public;
grant execute on dbms_pipe to public;
grant select on v_$parameter to public;

REM
REM begin plsql_mon.sql
REM

connect tpcc/tpcc;
set echo on;
CREATE OR REPLACE PACKAGE
plsql_mon_pack
IS
PROCEDURE print
(
info VARCHAR2
);
END;
/
show errors;

CREATE OR REPLACE PACKAGE BODY
plsql_mon_pack
IS
PROCEDURE print
(
info VARCHAR2
)
IS
s NUMBER;
BEGIN
dbms_pipe.pack_message (info);
s := dbms_pipe.send_message
(plsql_mon);
IF (s <> 0) THEN

```

```

        raise_application_error (-20000, 'Error:' ||
to_char(s) ||
        ' sending on pipe');
    END IF;
END;
END;
/
show errors;

set echo off;

REM
REM end plsql_mon.sql
REM

REM
REM begin cre_tab.sql
REM

connect tpcc/tpcc;
set echo on;

drop table temp_o1;
drop table temp_no;
drop table temp_o2;
drop table temp_ol;
drop table tpcc_audit_tab;

create table temp_o1 (
    o_w_id integer,
    o_d_id integer,
    o_o_id integer);

create table temp_no (
    no_w_id integer,
    no_d_id integer,
    no_o_id integer);

create table temp_o2 (
    o_w_id integer,
    o_d_id integer,
    o_count integer);

create table temp_ol (
    ol_w_id integer,
    ol_d_id integer,
    ol_count integer);

create table tpcc_audit_tab (starttime date);

delete from tpcc_audit_tab;

set echo off;

REM
REM end cre_tab.sql
REM

REM
REM begin views.sql
REM

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;

REM
REM end views.sql
REM

REM
REM begin 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;

REM
REM end dml.sql
REM

REM
REM begin extent.sql
REM

$$SYS_CONNECTION_STRING

@$tpcc_sql_dir/extent

@$tpcc_sql_dir/freeext

exit sql.sqlcode;

!
```

```

.....
App_E/createspacestats.sql
.....

@space_init
@space_get 12 10
@space_rpt
spool off
exit sql.sqlcode;

.....
App_E/createstats.sh
.....

#!/bin/sh

cstat=c_stat
if test $tpcc_np -gt 1 ; then
    cstat=c_stat_rac
fi

$tpcc_sqlplus $tpcc_sqlplus_args << !
$tpcc_internal_connect

REM
REM create tablespace for statspack user sp
begin
REM

spool createstats.log

set echo on
drop tablespace sp_0 including contents;
create tablespace sp_0 datafile
'$(tpcc_disks_location)sp_0' size
$tpcc_statspack_size reuse autoextend on
extent management local uniform size 1M
nologging ;
spool off

REM
REM create tablespace for statspack user sp
end
REM

REM
REM begin now call spcreate to create
statspack sp package
REM

$tpcc_internal_connect

define default_tablespace='sp_0'

define temporary_tablespace='temp_0'

@$ORACLE_HOME/rdbms/admin/spcreate
perfstat

REM note that the last thing (after spcreate) is
the perfstat password.
REM since we're not worried about security,
perfstat will do.

REM
REM tpcc stat table for NT, it is not working so I
comment it out
REM shui.lau@oracle.com it is better to use
perfmom

```



```

REM
@${tpcc_sql_dir}/cs_tpcc
@${tpcc_sql_dir}/cs_cpu
@${tpcc_sql_dir}/cs_os
@${tpcc_sql_dir}/cs_proc
@${tpcc_sql_dir}/cs_thread

REM
REM tpcc result table for unix and NT
REM

@${tpcc_sql_dir}/$(cstat)
@${tpcc_sql_dir}/pst_c

!

.....
App_E/createtabledistsql.sh
.....

#!/bin/sh
cd $tpcc_genscripts_dir
$tpcc_sqlplus $tpcc_user_pass
@${tpcc_genscripts_dir}/createtabledistsql >
junk 2>&1

if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi

.....
App_E/createtabledistsql
.....

spool createtabledistsql.log
@tkvcin.sql
spool off
exit sql.sqlcode;

.....
App_E/createtabledistsql
.....

set timing on
set sqlblanklines on
spool createtabledistsql.log
set echo on
drop cluster custcluster including tables ;

create cluster custcluster (
  c_id number
, c_d_id number
, c_w_id number
, 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 char(500)
)
cluster custcluster (
  c_id
, c_d_id
, c_w_id
);
set echo off
spool off
exit sql.sqlcode;

.....
App_E/createtabledistsql
.....

set timing on
set sqlblanklines on
spool createtabledistsql.log
set echo on
drop cluster distcluster including tables ;

create cluster distcluster (
  d_id number
, d_w_id number
)
single table
hashkeys 2083200
hash is (((d_w_id * 10) + d_id))
size 1448
intrans 4
storage ( buffer_pool default )
tablespace dist_0;

create table dist (
  d_id number
, d_w_id number
, d_ytd number
, d_next_o_id number
, d_tax 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_id
, d_w_id
);
set echo off
spool off
exit sql.sqlcode;

.....
App_E/createtabledistsql
.....

set timing on
set sqlblanklines on
spool createtabledistsql.log
set echo on
drop table hist ;

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)
)
pctfree 5 intrans 4
storage ( buffer_pool recycle )
tablespace hist_0
partition by range (h_c_w_id)
(
  partition wh1 values less than (26040) ,
  partition wh2 values less than (52080) ,
  partition wh3 values less than (78120) ,
  partition wh4 values less than (104160) ,
  partition wh5 values less than (130200) ,
  partition wh6 values less than (156240) ,
  partition wh7 values less than (182280) ,
  partition wh8 values less than (MAXVALUE)
);
set echo off
spool off
exit sql.sqlcode;

.....
App_E/createtabledistsql
.....

set timing on
set sqlblanklines on
spool createtabledistsql.log
set echo on
drop cluster itemcluster including tables ;

create cluster itemcluster (
  i_id number(6,0)
)
single table
hashkeys 100000
hash is ( (i_id) )
size 120
pctfree 0 intrans 3
storage ( buffer_pool keep )
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
);
set echo off
spool off
exit sql.sqlcode;

```

```

.....
App_E/createtable_nord.sql
.....

set timing on
set sqlblanklines on
spool createtable_nord.log
set echo on
drop cluster nordcluster_queue including
tables ;

create cluster nordcluster_queue (
no_w_id number
,no_d_id number
,no_o_id number SORT
)

hashkeys 2083200
hash is ((no_w_id - 1) * 10 + no_d_id - 1)
size 190
tablespace nord_0;

create table nord (
no_w_id number
,no_d_id number
,no_o_id number sort
,constraint nord_uk primary key (no_w_id
,no_d_id
,no_o_id)
)
cluster nordcluster_queue (
no_w_id
,no_d_id
,no_o_id
);
set echo off
spool off
exit sql.sqlcode;

```

```

.....
App_E/createtable_ordr.sql
.....

set timing on
set sqlblanklines on
spool createtable_ordr.log
set echo on
create table ordl (
ol_w_id number
,ol_d_id number
,ol_o_id number sort
,ol_number number sort
,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 ordl_uk primary key (ol_w_id,
ol_d_id,ol_o_id,ol_number )) CLUSTER
ordrcluster_queue(ol_w_id, ol_d_id, ol_o_id,
ol_number) ;
set echo off
spool off
exit sql.sqlcode;

```

```

.....
App_E/createtable_ordr.sql
.....

set timing on
set sqlblanklines on

```

```

spool createtable_ordr.log
set echo on
drop cluster ordrcluster_queue including
tables ;

create cluster ordrcluster_queue (
o_w_id number
,o_d_id number
,o_id number SORT
,o_number number SORT
)

hashkeys 2083200
hash is ((o_w_id - 1) * 10 + o_d_id - 1)
size 1490
tablespace ordr_0;

create table ordr (
o_id number sort
,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
,constraint ordr_uk primary key (o_w_id
,o_d_id
,o_id)
)
cluster ordrcluster_queue (
o_w_id
,o_d_id
,o_id
);
set echo off
spool off
exit sql.sqlcode;

```

```

.....
App_E/createtable_stok.sql
.....

set timing on
set sqlblanklines on
spool createtable_stok.log
set echo on
drop cluster stokcluster including tables ;

create cluster stokcluster (
s_i_id number
,s_w_id number
)
single table
hashkeys 20832000000
hash is ((abs(s_i_id - 1) * 26040 +
mod((s_w_id - 1), 26040) + trunc ((s_w_id - 1) /
26040) * 26040 * 100000) )
size 256
pctfree 0 initrans 2 maxtrans 2
storage ( initial 1937502k next 1937500k
maxextents unlimited pctincrease 0 freelist
groups 4 buffer_pool keep ) parallel ( degree
64 )
tablespace stok_0;

create table stok (
s_i_id number
,s_w_id number
,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
);
set echo off
spool off
exit sql.sqlcode;

```

```

.....
App_E/createtable_ware.sql
.....

set timing on
set sqlblanklines on
spool createtable_ware.log
set echo on
drop cluster warecluster including tables ;

create cluster warecluster (
w_id number
)
single table
hashkeys 208320
hash is ((w_id - 1) )
size 1448
initrans 2
storage ( buffer_pool default )
tablespace ware_0;

```

```

create table ware (
w_id number
,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
);
set echo off
spool off
exit sql.sqlcode;

.....
App_E/createts.sh
.....

# Tablespace ware, ts size 440M (450560K)
# each file 440M (450560K)
# extents 446828K (446828K)
# 1 files

$tpcc_createts ware 1 1 440M 446828K unix
0 0 128 auto t
if expr $? != 0 > /dev/null; then

```

```

    echo Creating tablespace for ware failed.
Exiting.
  exit 0
fi

# Tablespace dist, ts size 4360M (4464640K)
# each file 4360M (4464640K)
# extents 1487040K (1487040K)
# 1 files

$tpcc_createts dist 1 1 4360M 1487040K
unix 0 769 128 auto t
  if expr $? != 0 > /dev/null; then
    echo Creating tablespace for dist failed.
Exiting.
  exit 0
fi

# Tablespace hist, ts size 583080M
(597073920K)
# each file 3390M (3471360K)
# extents 102030K (102030K)
# 172 files

$tpcc_createts hist 172 1 3390M 102030K
unix 0 770 128 4K t
  if expr $? != 0 > /dev/null; then
    echo Creating tablespace for hist failed.
Exiting.
  exit 0
fi

# Tablespace item, ts size 20M (20480K)
# each file 20M (20480K)
# extents 16892K (16892K)
# 1 files

$tpcc_createts item 1 1 20M 16892K
unix 0 1902 128 auto t
  if expr $? != 0 > /dev/null; then
    echo Creating tablespace for item failed.
Exiting.
  exit 0
fi

# Tablespace ordr, ts size 8152320M
(8347975680K)
# each file 42460M (43479040K)
# extents 103264K (103264K)
# 192 files

$tpcc_createts ordr 192 1 42460M 103264K
unix 0 1903 128 16K t
  if expr $? != 0 > /dev/null; then
    echo Creating tablespace for ordr failed.
Exiting.
  exit 0
fi

# Tablespace nord, ts size 72800M (74547200K)
# each file 3640M (3727360K)
# extents 372528K (372528K)
# 20 files

$tpcc_createts nord 20 1 3640M 372528K
unix 0 2095 128 auto t
  if expr $? != 0 > /dev/null; then
    echo Creating tablespace for nord failed.
Exiting.
  exit 0
fi

# Tablespace iware, ts size 260M (266240K)
# each file 260M (266240K)
# extents 261424K (261424K)

```

```

# 1 files

$tpcc_createts iware 1 1 260M 261424K
unix 0 2115 128 auto t
  if expr $? != 0 > /dev/null; then
    echo Creating tablespace for iware failed.
Exiting.
  exit 0
fi

# Tablespace icust1, ts size 196320M
(201031680K)
# each file 4090M (4188160K)
# extents 4080K (4080K)
# 48 files

$tpcc_createts icust1 48 1 4090M 4080K
unix 0 2116 128 16K t
  if expr $? != 0 > /dev/null; then
    echo Creating tablespace for icust1 failed.
Exiting.
  exit 0
fi

# Tablespace icust2, ts size 516480M
(528875520K)
# each file 2690M (2754560K)
# extents 2688K (2688K)
# 192 files

$tpcc_createts icust2 192 1 2690M 2688K
unix 0 2164 128 16K t
  if expr $? != 0 > /dev/null; then
    echo Creating tablespace for icust2 failed.
Exiting.
  exit 0
fi

# Tablespace idist, ts size 1030M (1054720K)
# each file 1030M (1054720K)
# extents 1042624K (1042624K)
# 1 files

$tpcc_createts idist 1 1 1030M 1042624K
unix 0 2356 128 auto t
  if expr $? != 0 > /dev/null; then
    echo Creating tablespace for idist failed.
Exiting.
  exit 0
fi

# Tablespace istok, ts size 583200M
(597196800K)
# each file 4050M (4147200K)
# extents 4048K (4048K)
# 144 files

$tpcc_createts istok 144 1 4050M 4048K
unix 0 2357 128 16K t
  if expr $? != 0 > /dev/null; then
    echo Creating tablespace for istok failed.
Exiting.
  exit 0
fi

# Tablespace iitem, ts size 20M (20480K)
# each file 20M (20480K)
# extents 11264K (11264K)
# 1 files

$tpcc_createts iitem 1 1 20M 11264K
unix 0 2501 128 auto t
  if expr $? != 0 > /dev/null; then
    echo Creating tablespace for iitem failed.
Exiting.

```

```

  exit 0
fi

# Tablespace iordr2, ts size 535680M
(548536320K)
# each file 2790M (2856960K)
# extents 2784K (2784K)
# 192 files

$tpcc_createts iordr2 192 1 2790M 2784K
unix 0 2502 128 16K t
  if expr $? != 0 > /dev/null; then
    echo Creating tablespace for iordr2 failed.
Exiting.
  exit 0
fi

# Tablespace temp, ts size 1092480M
(1118699520K)
# each file 5690M (5826560K)
# extents 200564K (200564K)
# 192 files

$tpcc_createts temp 192 1 5690M 200564K
unix 1 2694 128 auto t
  if expr $? != 0 > /dev/null; then
    echo Creating tablespace for temp failed.
Exiting.
  exit 0
fi

.....
App_E/createuser.sh
.....

#!/bin/sh

echo Creating user tpcc...
$tpcc_sqlplus $tpcc_dba_user_pass
@$tpcc_sql_dir/createuser > junk 2>&1
if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi

.....
App_E/createuser.sql
.....

spool createusertpcc.log;

set echo on;

create user tpcc identified by tpcc;

grant dba to tpcc;

set echo off;
spool off;

exit ;

.....
App_E/ddview.sh
.....

#!/bin/sh

$tpcc_sqlplus $tpcc_sqlplus_args << !
$tpcc_internal_connect

```

```

spool ddview.log

REM
REM In an ade/nde view we might need to run
standard.sql and dbmsstdx manually
REM catalog and catproc suppose to take care
of it
REM

@$ORACLE_HOME/plsql/admin/standard
@$ORACLE_HOME/rdbms/admin/dbmsstdx

@$ORACLE_HOME/rdbms/admin/catalog
@$ORACLE_HOME/rdbms/admin/catproc

REM
REM In an ade/nde view we might need to run
publd manually
REM catalog and catproc suppose to take care
of it
REM

connect system/manager
REM @$ORACLE_HOME/sqlplus/admin/publd

REM
REM Oracle
REM

REM if test $NUMBER_ORACLE_NODE -qt 1
REM then

REM @$ORACLE_HOME/rdbms/admin/catparr

REM fi

spool off
!

.....
App_E/dpbccore.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 tpc
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

*/

```

```

#ifndef __dpbcore__
#define __dpbcore__

#include <stdio.h>
#include "dpbpcntl.h"

#ifdef __STDC__ /* ANSI C
*/
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
*/
clock_t dpbcpu(void); /* get CPU
time */
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 /* K&R C
*/
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
*/
clock_t dpbcpu(); /* get cpu
time */
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__ */

.....
App_E/dpbpcntl.h
.....

/* Copyright (c) Oracle Corporation 1993, 1992.
All Rights Reserved. */

/*
NAME DPBPCNTL.H

```

```

DESCRIPTION
OSD structures for process control

NOTES
Desktop Performance Group

MODIFIED (MM/DD/YY)
R Keller 02/03/93 - initial version

*/

#ifndef __dpbpcntl__
#define __dpbpcntl__

#ifdef ORA_OS2 /* IBM
OS/2 2.x */
#define INCL_DOSPROCESS
#include <os2.h>
typedef struct _pcntl
{
RESULTCODES rcodes;
} pcntl;
#endif /* ORA_OS2 */ /* IBM
OS/2 2.x */

#ifdef ORA_NT /* Microsoft
Windows NT */
#include <windows.h> /*
*/
typedef struct _pcntl
{
PROCESS_INFORMATION proc_info;
} pcntl;
#endif /* ORA_NT */ /*
Microsoft Windows NT */

#ifdef ORA_AUX /* Apple
A/UX */
typedef struct _pcntl
{
int dummy;
} pcntl;
#endif /* ORA_AUX */ /* Apple
A/UX */

#ifdef ORA_NW /* Novell
Netware */
typedef struct _pcntl
{
int dummy;
} pcntl;
#endif /* ORA_NW */ /* Novell
Netware */

#endif /* __dpbpcntl__ */

.....
App_E/estsize.sh
.....

#!/bin/sh

```



```

echo ${amount}G

.....
App_E/isneg.sh
.....

#!/bin/sh
# exit true if negative, else false

if test `$pcc_bcxpr "$*" | cut -b1` = -; then
    exit 0
else
    exit 1
fi

.....
App_E/lcm.sh
.....

#!/bin/sh
# echo the lcm of two numbers

if expr $2 \> $1 > /dev/null; then
    set $2 $1
# now $1 is guaranteed to be bigger
fi

lcm=$1
while expr \( (\($lcm % $1) ) + (\($lcm % $2) ) \) \>
0 > /dev/null; do
    lcm=`expr $lcm + $1`
done

echo $lcm

.....
App_E/loadcust.sh
.....

rm -f loadcust*.log
cd $pcc_bench
$tpcc_load -M 205632 -c -l 1 -m 101 >>
loadcust0.log 2>&1 &
$tpcc_load -M 205632 -c -l 102 -m 201 >>
loadcust1.log 2>&1 &
$tpcc_load -M 205632 -c -l 202 -m 301 >>
loadcust2.log 2>&1 &
$tpcc_load -M 205632 -c -l 302 -m 401 >>
loadcust3.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 402 -m 501 >>
loadcust4.log 2>&1 &
$tpcc_load -M 205632 -c -l 502 -m 601 >>
loadcust5.log 2>&1 &
$tpcc_load -M 205632 -c -l 602 -m 701 >>
loadcust6.log 2>&1 &
$tpcc_load -M 205632 -c -l 702 -m 801 >>
loadcust7.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 802 -m 901 >>
loadcust8.log 2>&1 &
$tpcc_load -M 205632 -c -l 902 -m 1001 >>
loadcust9.log 2>&1 &
$tpcc_load -M 205632 -c -l 1002 -m 1101 >>
loadcust10.log 2>&1 &
$tpcc_load -M 205632 -c -l 1102 -m 1201 >>
loadcust11.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 1202 -m 1301 >>
loadcust12.log 2>&1 &

```

```

$tpcc_load -M 205632 -c -l 1302 -m 1401 >>
loadcust13.log 2>&1 &
$tpcc_load -M 205632 -c -l 1402 -m 1501 >>
loadcust14.log 2>&1 &
$tpcc_load -M 205632 -c -l 1502 -m 1601 >>
loadcust15.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 1602 -m 1701 >>
loadcust16.log 2>&1 &
$tpcc_load -M 205632 -c -l 1702 -m 1801 >>
loadcust17.log 2>&1 &
$tpcc_load -M 205632 -c -l 1802 -m 1901 >>
loadcust18.log 2>&1 &
$tpcc_load -M 205632 -c -l 1902 -m 2001 >>
loadcust19.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 2002 -m 2101 >>
loadcust20.log 2>&1 &
$tpcc_load -M 205632 -c -l 2102 -m 2201 >>
loadcust21.log 2>&1 &
$tpcc_load -M 205632 -c -l 2202 -m 2301 >>
loadcust22.log 2>&1 &
$tpcc_load -M 205632 -c -l 2302 -m 2401 >>
loadcust23.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 2402 -m 2501 >>
loadcust24.log 2>&1 &
$tpcc_load -M 205632 -c -l 2502 -m 2601 >>
loadcust25.log 2>&1 &
$tpcc_load -M 205632 -c -l 2602 -m 2701 >>
loadcust26.log 2>&1 &
$tpcc_load -M 205632 -c -l 2702 -m 2801 >>
loadcust27.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 2802 -m 2901 >>
loadcust28.log 2>&1 &
$tpcc_load -M 205632 -c -l 2902 -m 3001 >>
loadcust29.log 2>&1 &
$tpcc_load -M 205632 -c -l 3002 -m 3101 >>
loadcust30.log 2>&1 &
$tpcc_load -M 205632 -c -l 3102 -m 3201 >>
loadcust31.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 3202 -m 3301 >>
loadcust32.log 2>&1 &
$tpcc_load -M 205632 -c -l 3302 -m 3401 >>
loadcust33.log 2>&1 &
$tpcc_load -M 205632 -c -l 3402 -m 3501 >>
loadcust34.log 2>&1 &
$tpcc_load -M 205632 -c -l 3502 -m 3601 >>
loadcust35.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 3602 -m 3701 >>
loadcust36.log 2>&1 &
$tpcc_load -M 205632 -c -l 3702 -m 3801 >>
loadcust37.log 2>&1 &
$tpcc_load -M 205632 -c -l 3802 -m 3901 >>
loadcust38.log 2>&1 &
$tpcc_load -M 205632 -c -l 3902 -m 4001 >>
loadcust39.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 4002 -m 4101 >>
loadcust40.log 2>&1 &
$tpcc_load -M 205632 -c -l 4102 -m 4201 >>
loadcust41.log 2>&1 &
$tpcc_load -M 205632 -c -l 4202 -m 4301 >>
loadcust42.log 2>&1 &
$tpcc_load -M 205632 -c -l 4302 -m 4401 >>
loadcust43.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 4402 -m 4501 >>
loadcust44.log 2>&1 &
$tpcc_load -M 205632 -c -l 4502 -m 4601 >>
loadcust45.log 2>&1 &

```

```

$tpcc_load -M 205632 -c -l 4602 -m 4701 >>
loadcust46.log 2>&1 &
$tpcc_load -M 205632 -c -l 4702 -m 4801 >>
loadcust47.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 4802 -m 4901 >>
loadcust48.log 2>&1 &
$tpcc_load -M 205632 -c -l 4902 -m 5001 >>
loadcust49.log 2>&1 &
$tpcc_load -M 205632 -c -l 5002 -m 5101 >>
loadcust50.log 2>&1 &
$tpcc_load -M 205632 -c -l 5102 -m 5201 >>
loadcust51.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 5202 -m 5301 >>
loadcust52.log 2>&1 &
$tpcc_load -M 205632 -c -l 5302 -m 5401 >>
loadcust53.log 2>&1 &
$tpcc_load -M 205632 -c -l 5402 -m 5501 >>
loadcust54.log 2>&1 &
$tpcc_load -M 205632 -c -l 5502 -m 5601 >>
loadcust55.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 5602 -m 5701 >>
loadcust56.log 2>&1 &
$tpcc_load -M 205632 -c -l 5702 -m 5801 >>
loadcust57.log 2>&1 &
$tpcc_load -M 205632 -c -l 5802 -m 5901 >>
loadcust58.log 2>&1 &
$tpcc_load -M 205632 -c -l 5902 -m 6001 >>
loadcust59.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 6002 -m 6101 >>
loadcust60.log 2>&1 &
$tpcc_load -M 205632 -c -l 6102 -m 6201 >>
loadcust61.log 2>&1 &
$tpcc_load -M 205632 -c -l 6202 -m 6301 >>
loadcust62.log 2>&1 &
$tpcc_load -M 205632 -c -l 6302 -m 6401 >>
loadcust63.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 6402 -m 6501 >>
loadcust64.log 2>&1 &
$tpcc_load -M 205632 -c -l 6502 -m 6601 >>
loadcust65.log 2>&1 &
$tpcc_load -M 205632 -c -l 6602 -m 6701 >>
loadcust66.log 2>&1 &
$tpcc_load -M 205632 -c -l 6702 -m 6801 >>
loadcust67.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 6802 -m 6901 >>
loadcust68.log 2>&1 &
$tpcc_load -M 205632 -c -l 6902 -m 7001 >>
loadcust69.log 2>&1 &
$tpcc_load -M 205632 -c -l 7002 -m 7101 >>
loadcust70.log 2>&1 &
$tpcc_load -M 205632 -c -l 7102 -m 7201 >>
loadcust71.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 7202 -m 7301 >>
loadcust72.log 2>&1 &
$tpcc_load -M 205632 -c -l 7302 -m 7401 >>
loadcust73.log 2>&1 &
$tpcc_load -M 205632 -c -l 7402 -m 7501 >>
loadcust74.log 2>&1 &
$tpcc_load -M 205632 -c -l 7502 -m 7601 >>
loadcust75.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 7602 -m 7701 >>
loadcust76.log 2>&1 &
$tpcc_load -M 205632 -c -l 7702 -m 7801 >>
loadcust77.log 2>&1 &
$tpcc_load -M 205632 -c -l 7802 -m 7901 >>
loadcust78.log 2>&1 &

```



```

$tpcc_load -M 205632 -c -l 196745 -m 196845
>> loadcust1960.log 2>&1 &
$tpcc_load -M 205632 -c -l 196846 -m 196946
>> loadcust1961.log 2>&1 &
$tpcc_load -M 205632 -c -l 196947 -m 197047
>> loadcust1962.log 2>&1 &
$tpcc_load -M 205632 -c -l 197048 -m 197148
>> loadcust1963.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 197149 -m 197249
>> loadcust1964.log 2>&1 &
$tpcc_load -M 205632 -c -l 197250 -m 197350
>> loadcust1965.log 2>&1 &
$tpcc_load -M 205632 -c -l 197351 -m 197451
>> loadcust1966.log 2>&1 &
$tpcc_load -M 205632 -c -l 197452 -m 197552
>> loadcust1967.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 197553 -m 197653
>> loadcust1968.log 2>&1 &
$tpcc_load -M 205632 -c -l 197654 -m 197754
>> loadcust1969.log 2>&1 &
$tpcc_load -M 205632 -c -l 197755 -m 197855
>> loadcust1970.log 2>&1 &
$tpcc_load -M 205632 -c -l 197856 -m 197956
>> loadcust1971.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 197957 -m 198057
>> loadcust1972.log 2>&1 &
$tpcc_load -M 205632 -c -l 198058 -m 198158
>> loadcust1973.log 2>&1 &
$tpcc_load -M 205632 -c -l 198159 -m 198259
>> loadcust1974.log 2>&1 &
$tpcc_load -M 205632 -c -l 198260 -m 198360
>> loadcust1975.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 198361 -m 198461
>> loadcust1976.log 2>&1 &
$tpcc_load -M 205632 -c -l 198462 -m 198562
>> loadcust1977.log 2>&1 &
$tpcc_load -M 205632 -c -l 198563 -m 198663
>> loadcust1978.log 2>&1 &
$tpcc_load -M 205632 -c -l 198664 -m 198764
>> loadcust1979.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 198765 -m 198865
>> loadcust1980.log 2>&1 &
$tpcc_load -M 205632 -c -l 198866 -m 198966
>> loadcust1981.log 2>&1 &
$tpcc_load -M 205632 -c -l 198967 -m 199067
>> loadcust1982.log 2>&1 &
$tpcc_load -M 205632 -c -l 199068 -m 199168
>> loadcust1983.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 199169 -m 199269
>> loadcust1984.log 2>&1 &
$tpcc_load -M 205632 -c -l 199270 -m 199370
>> loadcust1985.log 2>&1 &
$tpcc_load -M 205632 -c -l 199371 -m 199471
>> loadcust1986.log 2>&1 &
$tpcc_load -M 205632 -c -l 199472 -m 199572
>> loadcust1987.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 199573 -m 199673
>> loadcust1988.log 2>&1 &
$tpcc_load -M 205632 -c -l 199674 -m 199774
>> loadcust1989.log 2>&1 &
$tpcc_load -M 205632 -c -l 199775 -m 199875
>> loadcust1990.log 2>&1 &
$tpcc_load -M 205632 -c -l 199876 -m 199976
>> loadcust1991.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 199977 -m 200077
>> loadcust1992.log 2>&1 &

```

```

$tpcc_load -M 205632 -c -l 200078 -m 200178
>> loadcust1993.log 2>&1 &
$tpcc_load -M 205632 -c -l 200179 -m 200279
>> loadcust1994.log 2>&1 &
$tpcc_load -M 205632 -c -l 200280 -m 200380
>> loadcust1995.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 200381 -m 200481
>> loadcust1996.log 2>&1 &
$tpcc_load -M 205632 -c -l 200482 -m 200582
>> loadcust1997.log 2>&1 &
$tpcc_load -M 205632 -c -l 200583 -m 200683
>> loadcust1998.log 2>&1 &
$tpcc_load -M 205632 -c -l 200684 -m 200784
>> loadcust1999.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 200785 -m 200885
>> loadcust2000.log 2>&1 &
$tpcc_load -M 205632 -c -l 200886 -m 200986
>> loadcust2001.log 2>&1 &
$tpcc_load -M 205632 -c -l 200987 -m 201087
>> loadcust2002.log 2>&1 &
$tpcc_load -M 205632 -c -l 201088 -m 201188
>> loadcust2003.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 201189 -m 201289
>> loadcust2004.log 2>&1 &
$tpcc_load -M 205632 -c -l 201290 -m 201390
>> loadcust2005.log 2>&1 &
$tpcc_load -M 205632 -c -l 201391 -m 201491
>> loadcust2006.log 2>&1 &
$tpcc_load -M 205632 -c -l 201492 -m 201592
>> loadcust2007.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 201593 -m 201693
>> loadcust2008.log 2>&1 &
$tpcc_load -M 205632 -c -l 201694 -m 201794
>> loadcust2009.log 2>&1 &
$tpcc_load -M 205632 -c -l 201795 -m 201895
>> loadcust2010.log 2>&1 &
$tpcc_load -M 205632 -c -l 201896 -m 201996
>> loadcust2011.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 201997 -m 202097
>> loadcust2012.log 2>&1 &
$tpcc_load -M 205632 -c -l 202098 -m 202198
>> loadcust2013.log 2>&1 &
$tpcc_load -M 205632 -c -l 202199 -m 202299
>> loadcust2014.log 2>&1 &
$tpcc_load -M 205632 -c -l 202300 -m 202400
>> loadcust2015.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 202401 -m 202501
>> loadcust2016.log 2>&1 &
$tpcc_load -M 205632 -c -l 202502 -m 202602
>> loadcust2017.log 2>&1 &
$tpcc_load -M 205632 -c -l 202603 -m 202703
>> loadcust2018.log 2>&1 &
$tpcc_load -M 205632 -c -l 202704 -m 202804
>> loadcust2019.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 202805 -m 202905
>> loadcust2020.log 2>&1 &
$tpcc_load -M 205632 -c -l 202906 -m 203006
>> loadcust2021.log 2>&1 &
$tpcc_load -M 205632 -c -l 203007 -m 203107
>> loadcust2022.log 2>&1 &
$tpcc_load -M 205632 -c -l 203108 -m 203208
>> loadcust2023.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 203209 -m 203309
>> loadcust2024.log 2>&1 &
$tpcc_load -M 205632 -c -l 203310 -m 203410
>> loadcust2025.log 2>&1 &

```

```

$tpcc_load -M 205632 -c -l 203411 -m 203511
>> loadcust2026.log 2>&1 &
$tpcc_load -M 205632 -c -l 203512 -m 203612
>> loadcust2027.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 203613 -m 203713
>> loadcust2028.log 2>&1 &
$tpcc_load -M 205632 -c -l 203714 -m 203814
>> loadcust2029.log 2>&1 &
$tpcc_load -M 205632 -c -l 203815 -m 203915
>> loadcust2030.log 2>&1 &
$tpcc_load -M 205632 -c -l 203916 -m 204016
>> loadcust2031.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 204017 -m 204117
>> loadcust2032.log 2>&1 &
$tpcc_load -M 205632 -c -l 204118 -m 204218
>> loadcust2033.log 2>&1 &
$tpcc_load -M 205632 -c -l 204219 -m 204319
>> loadcust2034.log 2>&1 &
$tpcc_load -M 205632 -c -l 204320 -m 204420
>> loadcust2035.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 204421 -m 204521
>> loadcust2036.log 2>&1 &
$tpcc_load -M 205632 -c -l 204522 -m 204622
>> loadcust2037.log 2>&1 &
$tpcc_load -M 205632 -c -l 204623 -m 204723
>> loadcust2038.log 2>&1 &
$tpcc_load -M 205632 -c -l 204724 -m 204824
>> loadcust2039.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 204825 -m 204925
>> loadcust2040.log 2>&1 &
$tpcc_load -M 205632 -c -l 204926 -m 205026
>> loadcust2041.log 2>&1 &
$tpcc_load -M 205632 -c -l 205027 -m 205127
>> loadcust2042.log 2>&1 &
$tpcc_load -M 205632 -c -l 205128 -m 205228
>> loadcust2043.log 2>&1 &
wait
$tpcc_load -M 205632 -c -l 205229 -m 205329
>> loadcust2044.log 2>&1 &
$tpcc_load -M 205632 -c -l 205330 -m 205430
>> loadcust2045.log 2>&1 &
$tpcc_load -M 205632 -c -l 205431 -m 205531
>> loadcust2046.log 2>&1 &
$tpcc_load -M 205632 -c -l 205532 -m 205632
>> loadcust2047.log 2>&1 &
wait

```

```

.....
App_E/loaddist.sh
.....

```

```

cd $tpcc_bench
$tpcc_load -M 208320 -d > loaddist.log 2>&1

```

```

.....
App_E/loadhist.sh
.....

```

```

rm -f loadhist*.log
cd $tpcc_bench
allprocs=
$tpcc_load -M 208320 -h -b 1 -e 813 >>
loadhist0.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -h -b 814 -e 1626 >>
loadhist1.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -h -b 1627 -e 2439 >>
loadhist2.log 2>&1 &

```



```

allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -h -b 185529 -e 186342
>> loadhist228.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -h -b 186343 -e 187156
>> loadhist229.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -h -b 187157 -e 187970
>> loadhist230.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -h -b 187971 -e 188784
>> loadhist231.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -h -b 188785 -e 189598
>> loadhist232.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -h -b 189599 -e 190412
>> loadhist233.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -h -b 190413 -e 191226
>> loadhist234.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -h -b 191227 -e 192040
>> loadhist235.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -h -b 192041 -e 192854
>> loadhist236.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -h -b 192855 -e 193668
>> loadhist237.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -h -b 193669 -e 194482
>> loadhist238.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -h -b 194483 -e 195296
>> loadhist239.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -h -b 195297 -e 196110
>> loadhist240.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -h -b 196111 -e 196924
>> loadhist241.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -h -b 196925 -e 197738
>> loadhist242.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -h -b 197739 -e 198552
>> loadhist243.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -h -b 198553 -e 199366
>> loadhist244.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -h -b 199367 -e 200180
>> loadhist245.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -h -b 200181 -e 200994
>> loadhist246.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -h -b 200995 -e 201808
>> loadhist247.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -h -b 201809 -e 202622
>> loadhist248.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -h -b 202623 -e 203436
>> loadhist249.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -h -b 203437 -e 204250
>> loadhist250.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -h -b 204251 -e 205064
>> loadhist251.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -h -b 205065 -e 205878
>> loadhist252.log 2>&1 &

```

```

allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -h -b 205879 -e 206692
>> loadhist253.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -h -b 206693 -e 207506
>> loadhist254.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -h -b 207507 -e 208320
>> loadhist255.log 2>&1 &
allprocs="$allprocs ${!}"
error=0
for curproc in $allprocs; do
  wait $curproc
  error=`expr $? + $error`
done
exit `expr $error != 0`

.....
App_E/loaditem.sh
.....

cd $tpcc_bench
$tpcc_load -M 208320 -i > loaditem.log 2>&1

.....
App_E/loadnord.sh
.....

rm -f loadnord*.log
cd $tpcc_bench
allprocs=
$tpcc_load -M 208320 -n -b 1 -e 208320 >>
loadnord0.log 2>&1 &
allprocs="$allprocs ${!}"
error=0
for curproc in $allprocs; do
  wait $curproc
  error=`expr $? + $error`
done
exit `expr $error != 0`

.....
App_E/loadordrdl.sh
.....

rm -f loadordrdl*.log
cd $tpcc_bench
allprocs=
$tpcc_load -M 208320 -o
$(tpcc_disks_location)dummy0.dat -b 1 -e 813
>> loadordrdl0.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
$(tpcc_disks_location)dummy1.dat -b 814 -e
1626 >> loadordrdl1.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
$(tpcc_disks_location)dummy2.dat -b 1627 -e
2439 >> loadordrdl2.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
$(tpcc_disks_location)dummy3.dat -b 2440 -e
3252 >> loadordrdl3.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
$(tpcc_disks_location)dummy4.dat -b 3253 -e
4065 >> loadordrdl4.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
$(tpcc_disks_location)dummy5.dat -b 4066 -e
4878 >> loadordrdl5.log 2>&1 &
allprocs="$allprocs ${!}"

```

```

$tpcc_load -M 208320 -o
$(tpcc_disks_location)dummy6.dat -b 4879 -e
5691 >> loadordrdl6.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
$(tpcc_disks_location)dummy7.dat -b 5692 -e
6504 >> loadordrdl7.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
$(tpcc_disks_location)dummy8.dat -b 6505 -e
7317 >> loadordrdl8.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
$(tpcc_disks_location)dummy9.dat -b 7318 -e
8130 >> loadordrdl9.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
$(tpcc_disks_location)dummy10.dat -b 8131 -e
8943 >> loadordrdl10.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
$(tpcc_disks_location)dummy11.dat -b 8944 -e
9756 >> loadordrdl11.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
$(tpcc_disks_location)dummy12.dat -b 9757 -e
10569 >> loadordrdl12.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
$(tpcc_disks_location)dummy13.dat -b 10570 -e
11382 >> loadordrdl13.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
$(tpcc_disks_location)dummy14.dat -b 11383 -e
12195 >> loadordrdl14.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
$(tpcc_disks_location)dummy15.dat -b 12196 -e
13008 >> loadordrdl15.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
$(tpcc_disks_location)dummy16.dat -b 13009 -e
13821 >> loadordrdl16.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
$(tpcc_disks_location)dummy17.dat -b 13822 -e
14634 >> loadordrdl17.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
$(tpcc_disks_location)dummy18.dat -b 14635 -e
15447 >> loadordrdl18.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
$(tpcc_disks_location)dummy19.dat -b 15448 -e
16260 >> loadordrdl19.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
$(tpcc_disks_location)dummy20.dat -b 16261 -e
17073 >> loadordrdl20.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
$(tpcc_disks_location)dummy21.dat -b 17074 -e
17886 >> loadordrdl21.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
$(tpcc_disks_location)dummy22.dat -b 17887 -e
18699 >> loadordrdl22.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
$(tpcc_disks_location)dummy23.dat -b 18700 -e
19512 >> loadordrdl23.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
$(tpcc_disks_location)dummy24.dat -b 19513 -e
20325 >> loadordrdl24.log 2>&1 &

```



```

allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
${tpcc_disks_location}dummy247.dat -b 200995
-e 201808 >> loadordrordl247.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
${tpcc_disks_location}dummy248.dat -b 201809
-e 202622 >> loadordrordl248.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
${tpcc_disks_location}dummy249.dat -b 202623
-e 203436 >> loadordrordl249.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
${tpcc_disks_location}dummy250.dat -b 203437
-e 204250 >> loadordrordl250.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
${tpcc_disks_location}dummy251.dat -b 204251
-e 205064 >> loadordrordl251.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
${tpcc_disks_location}dummy252.dat -b 205065
-e 205878 >> loadordrordl252.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
${tpcc_disks_location}dummy253.dat -b 205879
-e 206692 >> loadordrordl253.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
${tpcc_disks_location}dummy254.dat -b 206693
-e 207506 >> loadordrordl254.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -o
${tpcc_disks_location}dummy255.dat -b 207507
-e 208320 >> loadordrordl255.log 2>&1 &
allprocs="$allprocs ${!}"
error=0
for curproc in $allprocs; do
  wait $curproc
  error=`expr $? + $error`
done
exit `expr $error != 0`

.....
App_E/loadstok.sh
.....

rm -f loadstok*.log
cd $tpcc_bench
allprocs=
$tpcc_load -M 208320 -S -j 1 -k 390 >>
loadstok0.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 391 -k 780 >>
loadstok1.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 781 -k 1170 >>
loadstok2.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 1171 -k 1560 >>
loadstok3.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 1561 -k 1950 >>
loadstok4.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 1951 -k 2340 >>
loadstok5.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 2341 -k 2730 >>
loadstok6.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 2731 -k 3120 >>
loadstok7.log 2>&1 &

```

```

allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 3121 -k 3510 >>
loadstok8.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 3511 -k 3900 >>
loadstok9.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 3901 -k 4290 >>
loadstok10.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 4291 -k 4680 >>
loadstok11.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 4681 -k 5070 >>
loadstok12.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 5071 -k 5460 >>
loadstok13.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 5461 -k 5850 >>
loadstok14.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 5851 -k 6240 >>
loadstok15.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 6241 -k 6630 >>
loadstok16.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 6631 -k 7020 >>
loadstok17.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 7021 -k 7410 >>
loadstok18.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 7411 -k 7800 >>
loadstok19.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 7801 -k 8190 >>
loadstok20.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 8191 -k 8580 >>
loadstok21.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 8581 -k 8970 >>
loadstok22.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 8971 -k 9360 >>
loadstok23.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 9361 -k 9750 >>
loadstok24.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 9751 -k 10140 >>
loadstok25.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 10141 -k 10530 >>
loadstok26.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 10531 -k 10920 >>
loadstok27.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 10921 -k 11310 >>
loadstok28.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 11311 -k 11700 >>
loadstok29.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 11701 -k 12090 >>
loadstok30.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 12091 -k 12480 >>
loadstok31.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 12481 -k 12870 >>
loadstok32.log 2>&1 &

```

```

allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 12871 -k 13260 >>
loadstok33.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 13261 -k 13650 >>
loadstok34.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 13651 -k 14040 >>
loadstok35.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 14041 -k 14430 >>
loadstok36.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 14431 -k 14820 >>
loadstok37.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 14821 -k 15210 >>
loadstok38.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 15211 -k 15600 >>
loadstok39.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 15601 -k 15990 >>
loadstok40.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 15991 -k 16380 >>
loadstok41.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 16381 -k 16770 >>
loadstok42.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 16771 -k 17160 >>
loadstok43.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 17161 -k 17550 >>
loadstok44.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 17551 -k 17940 >>
loadstok45.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 17941 -k 18330 >>
loadstok46.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 18331 -k 18720 >>
loadstok47.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 18721 -k 19110 >>
loadstok48.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 19111 -k 19500 >>
loadstok49.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 19501 -k 19890 >>
loadstok50.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 19891 -k 20280 >>
loadstok51.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 20281 -k 20670 >>
loadstok52.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 20671 -k 21060 >>
loadstok53.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 21061 -k 21450 >>
loadstok54.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 21451 -k 21840 >>
loadstok55.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 21841 -k 22230 >>
loadstok56.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 22231 -k 22620 >>
loadstok57.log 2>&1 &

```



```

allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 81233 -k 81623 >>
loadstok208.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 81624 -k 82014 >>
loadstok209.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 82015 -k 82405 >>
loadstok210.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 82406 -k 82796 >>
loadstok211.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 82797 -k 83187 >>
loadstok212.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 83188 -k 83578 >>
loadstok213.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 83579 -k 83969 >>
loadstok214.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 83970 -k 84360 >>
loadstok215.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 84361 -k 84751 >>
loadstok216.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 84752 -k 85142 >>
loadstok217.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 85143 -k 85533 >>
loadstok218.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 85534 -k 85924 >>
loadstok219.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 85925 -k 86315 >>
loadstok220.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 86316 -k 86706 >>
loadstok221.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 86707 -k 87097 >>
loadstok222.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 87098 -k 87488 >>
loadstok223.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 87489 -k 87879 >>
loadstok224.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 87880 -k 88270 >>
loadstok225.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 88271 -k 88661 >>
loadstok226.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 88662 -k 89052 >>
loadstok227.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 89053 -k 89443 >>
loadstok228.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 89444 -k 89834 >>
loadstok229.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 89835 -k 90225 >>
loadstok230.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 90226 -k 90616 >>
loadstok231.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 90617 -k 91007 >>
loadstok232.log 2>&1 &

```

```

allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 91008 -k 91398 >>
loadstok233.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 91399 -k 91789 >>
loadstok234.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 91790 -k 92180 >>
loadstok235.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 92181 -k 92571 >>
loadstok236.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 92572 -k 92962 >>
loadstok237.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 92963 -k 93353 >>
loadstok238.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 93354 -k 93744 >>
loadstok239.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 93745 -k 94135 >>
loadstok240.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 94136 -k 94526 >>
loadstok241.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 94527 -k 94917 >>
loadstok242.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 94918 -k 95308 >>
loadstok243.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 95309 -k 95699 >>
loadstok244.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 95700 -k 96090 >>
loadstok245.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 96091 -k 96481 >>
loadstok246.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 96482 -k 96872 >>
loadstok247.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 96873 -k 97263 >>
loadstok248.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 97264 -k 97654 >>
loadstok249.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 97655 -k 98045 >>
loadstok250.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 98046 -k 98436 >>
loadstok251.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 98437 -k 98827 >>
loadstok252.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 98828 -k 99218 >>
loadstok253.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 99219 -k 99609 >>
loadstok254.log 2>&1 &
allprocs="$allprocs ${!}"
$tpcc_load -M 208320 -S -j 99610 -k 100000 >>
loadstok255.log 2>&1 &
allprocs="$allprocs ${!}"
error=0
for curproc in $allprocs; do
    wait $curproc
    error=`expr $? + $error`
done

```

```

exit `expr $error != 0`

.....
App_E/loadware.sh
.....

cd $tpcc_bench
$tpcc_load -M 208320 -w > loadware.log 2>&1

.....
App_E/p_build.ora
.....

compatible = 10.1.0.0.0
db_name = tpcc
control_files =
(ora_dev/control_001,/ora_dev/control_002)
parallel_max_servers = 100
recovery_parallelism = 40
db_files = 2621
db_cache_size = 50400M
db_8k_cache_size = 1890M
db_16k_cache_size = 5040M
db_recycle_cache_size = 7700M
dml_locks = 500
statistics_level = basic
log_buffer = 1048576
processes = 3200
sessions = 3200
transactions = 3200
shared_pool_size = 5000M
cursor_space_for_time = TRUE
db_block_size = 2048
undo_management = auto
undo_retention = 2
plsql_optimize_level=2

UNDO_TABLESPACE = undo_1
db_4k_cache_size = 2000M

.....
App_E/p_create.ora
.....

compatible = 10.1.0.0.0
db_name = tpcc
control_files = (/ora_dev/control_001,
/ora_dev/control_002)
db_block_size = 2048
db_cache_size = 85333M
db_8k_cache_size = 32000M
log_buffer = 1048576
db_16k_cache_size = 85333M
undo_management = manual
statistics_level = basic
shared_pool_size = 16000M
plsql_optimize_level=2
db_4k_cache_size = 20M

.....
App_E/shutdowndb.sh
.....

#!/bin/sh

echo "Shutting down database..."

$tpcc_sqlplus $tpcc_sqlplus_args << !
$tpcc_internal_connect

spool shutdowndb.log;

```

```

set echo on;

alter system switch logfile;
alter system switch logfile;

shutdown immediate;

set echo off;
spool off;

exit
!

.....
App_E/startupdb.sh
.....

#!/bin/sh

echo "Starting up database using $1..."

init_file=${1}.ora

if test $tpcc_np -gt 1 ; then
  init_file=build_init_${tpcc_rac_id}.ora
fi

$tpcc_sqlplus $tpcc_sqlplus_args << !
$tpcc_internal_connect

spool startdb.log

set echo on

startup pfile=$init_file open

spool off
set echo off
exit sql.sqlcode
!

.....
App_E/tkvcin.sql
.....

-- The initnew package for storing variables used
in the
-- New Order anonymous block

CREATE OR REPLACE PACKAGE inittpcc
AS
  TYPE intarray IS TABLE OF INTEGER INDEX
  BY BINARY_INTEGER;
  TYPE distarray IS TABLE OF VARCHAR(24)
  INDEX BY BINARY_INTEGER;
  nulldate DATE;
  TYPE rowidarray IS TABLE OF ROWID INDEX
  BY PLS_INTEGER;
  s_dist distarray;
  idx1arr intarray;
  s_remote intarray;
  dist intarray;
  row_id rowidarray;
  cust_rowid rowid;
  dist_name VARCHAR2(11);
  ware_name VARCHAR2(11);
  c_num PLS_INTEGER;

  PROCEDURE init_no(idxarr intarray);
  PROCEDURE init_del;
  PROCEDURE init_pay;
END inittpcc;

```

```

/
show errors;

CREATE OR REPLACE PACKAGE BODY
inittpcc AS
  PROCEDURE init_no (idxarr intarray)
  IS
  BEGIN
    -- initialize null date
    nulldate := TO_DATE('01-01-1811', 'MM-DD-
YYYY');
    idx1arr := idxarr;
  END init_no;

  PROCEDURE init_del
  IS
  BEGIN
    FOR i IN 1 .. 10 LOOP
      dist(i) := i;
    END LOOP;
  END init_del;

  PROCEDURE init_pay IS
  BEGIN
    NULL;
  END init_pay;

END inittpcc;
/
show errors
exit

.....
App_E/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.
+=====
+=====*/

#ifdef TPCC_H
#define TPCC_H

#ifdef FALSE
# define FALSE 0
#endif

#ifdef TRUE
# define TRUE 1
#endif

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

```

```

#ifdef boolean
#define boolean int
#endif

#include "tpccflags.h"

#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 ldatef;

/* 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 ();
extern void userlog(char* ftmp, ...);

/* 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.hh24:mi:ss"
#define SHORTDATE "dd-mm-yyyy"

#define DELRT 80.0

extern int tkvcninit ();
extern int tkvcpinit ();
extern int tkvcoint ();
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 tkvodone ();
extern void tkvcdone ();

```

```

extern void tkvcsdone ();

extern int tkvcs (); /* for alter session to get
memory size and trace */
extern boolean multitransx;
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 OCISlmt *curmtest;
/* 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;
#ifdef USE_IEEE_NUMBER
extern float threshold;
#else
extern int threshold;
#endif /* USE_IEEE_NUMBER */
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];
#ifdef USE_IEEE_NUMBER
extern float ol_quantity[15];
extern float ol_amount[15];
#else
extern int ol_quantity[15];
extern int ol_amount[15];
#endif /* USE_IEEE_NUMBER */
ub4 ol_del_len[15];
extern text ol_delivery_d[15][11];
/* xnie - begin */
extern OCIRowid *o_rowid;

```

```

/* xnie - end */

/* for payment transaction */

extern int c_w_id;
extern int c_d_id;
#ifdef USE_IEEE_NUMBER
extern float h_amount;
#else
extern int h_amount;
#endif /* USE_IEEE_NUMBER */
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];
#ifdef USE_IEEE_NUMBER
extern float nol_quantity[15];
extern float nol_amount[15];
extern float s_quantity[15];
extern float i_price[15];
#else
extern int nol_quantity[15];
extern int nol_amount[15];
extern int s_quantity[15];
extern int i_price[15];
#endif /* USE_IEEE_NUMBER */
extern int nol_quant10[15];
extern int nol_quant19[15];
extern int nol_ytdqty[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 char brand_gen[15];
extern ub2 brand_gen_len[15];
extern ub2 brand_gen_rcode[15];
extern ub4 brand_gen_csize;
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

#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(x) (x * )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,
progvl, ftype)\
ocierror(__FILE__,__LINE__(errp), \
OCIHandleAlloc((stmp),(dvoid**)&(bndp),OCI_H
TYPE_BIND,0,(dvoid**0)); \
ocierror(__FILE__,__LINE__(errp), \
OCIBindByName((stmp), &(bndp), (errp), \
\
(text *) (sqlvar), strlen((sqlvar)), \
(progvl), (progvl),
(ftype),0,0,0,0,OCI_DEFAULT));

/* bind arrays for sql */
#define
OCIBNDRA(stmp,bndp,errp,sqlvar,progvl,progvl,f
type,indp,alen,rcode) \
DISCARD
ocierror(__FILE__,__LINE__(errp), \
OCIHandleAlloc((stmp),(dvoid**)&(bndp),OCI_H
TYPE_BIND,0,(dvoid**0)); \
DISCARD
ocierror(__FILE__,__LINE__(errp), \
OCIBindByName((stmp),&(bndp),(errp),(text
*)(sqlvar),strlen((sqlvar)), \
(progvl),(progvl),(ftype),(indp),(alen),(rcode),0,0,
OCI_DEFAULT));

/* use with callback data */

```



```
#define
OCIBNDRAD(stmp,bndp,errp,sqlvar,progv,ftype,
indp,ctxp,\
    cbf_nodata,cbf_data) \
    DISCARD
ocierror(__FILE__,__LINE__,(errp), \

OCIBindAlloc((stmp),(dvoid*)&(bndp),OCI_H
TYPE_BIND,0,(dvoid**0)); \
    DISCARD
ocierror(__FILE__,__LINE__,(errp), \

OCIBindByName((stmp),&(bndp),(errp),(text
*)(sqlvar), \
    strlen((sqlvar)),0,(progv),(ftype),
\
indp,0,0,0,OCI_DATA_AT_EXEC)); \
    DISCARD
ocierror(__FILE__,__LINE__,(errp), \

OCIBindDynamic((bndp),(errp),(ctxp),(cbf_nodat
a),(ctxp),(cbf_data));

/* bind in/out for plsql without indicator and rcode
*/
#define
OCIBNDPL(stmp,bndp,errp,sqlvar,progv,progl,f
type,alen) \
    DISCARD ocierror(__FILE__,__LINE__,(errp),
\

OCIBindAlloc((stmp),(dvoid*)&(bndp),OCI_H
TYPE_BIND,0,(dvoid**0)); \
    DISCARD ocierror(__FILE__,__LINE__,(errp),
\

OCIBindByName((stmp),&(bndp),(errp),(CONST
text*)(sqlvar), \
    (sb4)strlen((CONST char*)(sqlvar)),
(dvoid*)(progv),(progl),(ftype),\
    NULLP(dvoid),(alen), NULLP(ub2),
0,NULLP(ub4),OCI_DEFAULT));

/* bind in values for plsql with indicator and
rcode */
#define
OCIBNDR(stmp,bndp,errp,sqlvar,progv,progl,ft
ype,indp,alen,arcode) \
    DISCARD
ocierror(__FILE__,__LINE__,(errp), \

OCIBindAlloc((stmp),(dvoid*)&(bndp),OCI_H
TYPE_BIND,0,(dvoid**0)); \
    DISCARD
ocierror(__FILE__,__LINE__,(errp), \

OCIBindByName((stmp),&(bndp),(errp),(text
*)(sqlvar),strlen((sqlvar)),\

(progv),(progl),(ftype),(indp),(alen),(arcode),0,0,
\
    OCI_DEFAULT));

/* bind in/out for plsql arrays without indicator and
rcode */
#define
OCIBNDPLA(stmp,bndp,errp,sqlvar,progv,progl,
ftype,alen,ms,cu) \
    DISCARD ocierror(__FILE__,__LINE__,
(errp), \
```

```
OCIBindAlloc((stmp),(dvoid*)&(bndp),OCI_H
TYPE_BIND,0,(dvoid**0));\
    DISCARD
ocierror(__FILE__,__LINE__,(errp),\

OCIBindByName((stmp),&(bndp),(errp),(CONST
text*)(sqlvar), \
    (sb4)strlen((CONST char *)
(sqlvar)),(void*)(progv), \

(progv),(ftype),NULL,(alen),NULL,(ms),(cu),OCI
_DEFAULT));

/* bind in/out values for plsql with indicator and
rcode */
#define
OCIBNDRAA(stmp,bndp,errp,sqlvar,progv,progl,
ftype,indp,alen,arcode,\
    ms,cu) \
    ocierror(__FILE__,__LINE__,(errp), \

OCIBindAlloc((stmp),(dvoid*)&(bndp),OCI_H
TYPE_BIND,0,(dvoid**0));\
    ocierror(__FILE__,__LINE__,(errp),\

OCIBindByName((stmp),&(bndp),(errp),(text
*)(sqlvar),strlen((sqlvar)),\

(progv),(progl),(ftype),(indp),(alen),(arcode),(ms
),(cu),OCI_DEFAULT));

#define
OCIDEFINE(stmp,dfnp,errp,pos,progv,progl,ftyp
e)\

OCIDefineByPos((stmp),&(dfnp),(errp),(pos),(pro
gv),(progl),(ftype),\
    0,0,0,OCI_DEFAULT);

#define
OCIDEF(stmp,dfnp,errp,pos,progv,progl,ftype) \

OCIBindAlloc((stmp),(dvoid*)&(dfnp),OCI_H
TYPE_DEFINE,0,\
    (dvoid**0));\

OCIDefineByPos((stmp),&(dfnp),(errp),(pos),(pro
gv),(progl),\

(ftype),NULL,NULL,NULL,OCI_DEFAULT); \

#define
OCIDFNRA(stmp,dfnp,errp,pos,progv,progl,ftyp
e,indp,alen,arcode) \

OCIBindAlloc((stmp),(dvoid*)&(dfnp),OCI_H
TYPE_DEFINE,0,\
    (dvoid**0));\

OCIDefineByPos((stmp),&(dfnp),(errp),(pos),(pro
gv),\
    (progl),(ftype),(indp),(alen),\
    (arcode),OCI_DEFAULT);

#define
OCIDFNDR(stmp,dfnp,errp,pos,progv,progl,ftyp
e,indp,ctxp,cbf_data) \
    ocierror(__FILE__,__LINE__,(errp), \

OCIBindAlloc((stmp),(dvoid*)&(dfnp),OCI_H
TYPE_DEFINE,0,\
    (dvoid**0));\
```

```
ocierror(__FILE__,__LINE__,(errp), \

OCIDefineByPos((stmp),&(dfnp),(errp),(pos),(pro
gv),(progl),(ftype),\
    (indp),NULL,NULL,
OCI_DYNAMIC_FETCH));\
    ocierror(__FILE__,__LINE__,(errp), \

OCIDefineDynamic((dfnp),(errp),(ctxp),(cbf_data
));

/* 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_of_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 ordinstr {
    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 ordinstr ordin;
    struct ordoutstruct ordout;
};

/* Delivery */

struct delinstr {
    int w_id;
    int o_carrier_id;
    double qtime;
    int in_timing_int;
    int plsflag;
};

struct deloutstruct {
    int terror;
    int retry;
};

};

struct delstruct {
    struct delinstr 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

.....
App_E/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 (cluster around
c_w_id)
| -C load cust table (cluster
around c_id)
| -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)
| -l <cid#> beginning cid number
(with -C)
| -m <cid#> ending cid number
(with -C)
| -g generate rows to standard
output
+=====
=====*/

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <time.h>
#include <sys/types.h>
#include "tpcc.h"

#ifdef ORA_NT
#undef boolean
#include <process.h>
#include "dpbccore.h"
# define gettime dpbtimef
# define getcpu dpbcpcu
# define lrand48() ((long)rand() <<15 | rand())
#ifdef __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. dist id */
#define CUSTFAC 3000 /* max. cust id */
#define STOCFAC 100000 /* max. stok id */
#define ITEMFAC 100000 /* max. item id */
#define HISTFAC 30000 /* history /
warehouse */
#define ORDEFAC 3000 /* order / district
*/
#define NEWOFAC 900 /* new order /
district */

#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 NOT_SERIALIZABLE 8177 /* ORA-
08177: transaction not serializable */
#define SNAPSHOT_TOO_OLD 1555 /* ORA-
01555: snapshot too old */
#define RECOVERR -10
#define IRRECERR -20

#define SOLTXTW "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 SOLTXTD "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, 30000000, :d_tax, \
3001, :d_name, :d_street_1, :d_street_2, :d_city,
:d_state, :d_zip)"

#define SOLTXTCQUERY "select /*+ HASH
(cust) */ count(*) from cust where c_w_id
=:s_c_w_id and c_d_id = :s_c_d_id and c_id
=:s_c_id"

#define SOLTXTC "INSERT INTO cust (C_ID,
C_D_ID, C_W_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_YTD_PAYMENT, C_PAYMENT_CNT,
C_DELIVERY_CNT, C_DATA) VALUES
(:c_id, :c_d_id, :c_w_id, \
:c_first,
'OE', :c_last, :c_street_1, :c_street_2, :c_city, :c_
state, \
:c_zip, :c_phone, SYSDATE, :c_credit,
5000000, :c_discount, -1000, 1000, 1, \
0, :c_data)"

#define SOLTXTH "INSERT INTO hist (h_c_id,
h_c_d_id, h_c_w_id, h_d_id, h_w_id, h_date,
h_amount, h_data) VALUES
(:h_c_id, :h_c_d_id, :h_c_w_id, \
:h_d_id, :h_w_id, SYSDATE, 1000, :h_data)"

#define SOLTXTSQUERY "select /*+ HASH
(stok) */ count(*) from stok where s_w_id
=:s_s_w_id and s_i_id = :s_s_i_id"

#define SOLTXTS "INSERT INTO stok (s_i_id,
s_w_id, s_quantity, s_dist_01, s_dist_02,
s_dist_03, s_dist_04, s_dist_05, s_dist_06,
s_dist_07, s_dist_08, s_dist_09, s_dist_10,
s_ytd, s_order_cnt, s_remote_cnt, s_data) \
VALUES (:s_i_id, :s_w_id, :s_quantity, \
: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,
0, 0, 0, :s_data)"

#define SOLTXTI "INSERT INTO item
(I_ID, I_IM_ID, I_NAME, I_PRICE, I_DATA)
VALUES (:i_id, :i_im_id, :i_name, :i_price, \
:i_data)"

#define SOLTXTO1 "INSERT INTO ord1 (O_ID,
O_D_ID, O_W_ID, O_C_ID, O_ENTRY_D, O_CAR
RIER_ID, O_OL_CNT, O_ALL_LOCAL) \
VALUES (:o_id, :o_d_id, :o_w_id, :o_c_id, \
SYSDATE, :o_carrier_id, :o_ol_cnt, 1)"
```

```
#define SOLTXTO2 "INSERT INTO ord1 (O_ID,
O_D_ID, O_W_ID, O_C_ID, O_ENTRY_D, O_CAR
RIER_ID, O_OL_CNT, O_ALL_LOCAL) \
VALUES (:o_id, :o_d_id, :o_w_id, :o_c_id, \
SYSDATE, 11, :o_ol_cnt, 1)"

#define SOLTXTOL1 "INSERT INTO ord1
(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,
SYSDATE, :ol_i_id, :ol_supply_w_id, 5, 0, \
:ol_dist_info)"

#define SOLTXTOL2 "INSERT INTO ord1
(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, to_date('01-Jan-
1811'), :ol_i_id, :ol_supply_w_id, 5, :ol_amount, \
:ol_dist_info)"

#define SOLTXTNO "INSERT INTO nord
(no_o_id, no_d_id, no_w_id) VALUES
(:no_o_id, :no_d_id, :no_w_id)"

#define SOLTXTENHA "alter session set
'_enable_hash_overflow'=true"
#define SOLTXTDIHA "alter session set
'_enable_hash_overflow'=false"

static char *lastname[] = {
"BAR",
"OUGHT",
"ABLE",
"PRI",
"PRES",
"ESE",
"ANTI",
"CALLY",
"ATION",
"EING"
};

char num9[10];
char num16[17];
char str2[3];
char str24[15][25];
int randperm3000[3000];

void initperm();
void randstr();
void randdatastr();
void randnum();
void randlastname(char*, int);
int NURand();
void sysdate();

OCIEEnv *tpcenv;
OCIServer *tpcsrv;
OCIError *errhp;
OCISvcCtx *tpcsvc;
OCISession *tpcsrv;

OCISmt *curw;
OCISmt *curd;
OCISmt *curc;
OCISmt *curcs;
OCISmt *curh;
OCISmt *curs;
OCISmt *curss;
```

```
OCISmt *curi;
OCISmt *curo1;
OCISmt *curo2;
OCISmt *curo11;
OCISmt *curo12;
OCISmt *curno;

OCIBind *w_id_bp = (OCIBind *) 0;
OCIBind *w_name_bp = (OCIBind *) 0;
OCIBind *w_street1_bp = (OCIBind *) 0;
OCIBind *w_street2_bp = (OCIBind *) 0;
OCIBind *w_city_bp = (OCIBind *) 0;
OCIBind *w_state_bp = (OCIBind *) 0;
OCIBind *w_zip_bp = (OCIBind *) 0;
OCIBind *w_tax_bp = (OCIBind *) 0;

OCIBind *d_id_bp = (OCIBind *) 0;
OCIBind *d_w_id_bp = (OCIBind *) 0;
OCIBind *d_name_bp = (OCIBind *) 0;
OCIBind *d_street1_bp = (OCIBind *) 0;
OCIBind *d_street2_bp = (OCIBind *) 0;
OCIBind *d_city_bp = (OCIBind *) 0;
OCIBind *d_state_bp = (OCIBind *) 0;
OCIBind *d_zip_bp = (OCIBind *) 0;
OCIBind *d_tax_bp = (OCIBind *) 0;

OCIDefine *s_c_ret_bp = (OCIDefine *) 0;
OCIBind *s_c_id_bp = (OCIBind *) 0;
OCIBind *s_c_d_id_bp = (OCIBind *) 0;
OCIBind *s_c_w_id_bp = (OCIBind *) 0;

OCIBind *c_id_bp = (OCIBind *) 0;
OCIBind *c_d_id_bp = (OCIBind *) 0;
OCIBind *c_w_id_bp = (OCIBind *) 0;
OCIBind *c_first_bp = (OCIBind *) 0;
OCIBind *c_street1_bp = (OCIBind *) 0;
OCIBind *c_street2_bp = (OCIBind *) 0;
OCIBind *c_city_bp = (OCIBind *) 0;
OCIBind *c_state_bp = (OCIBind *) 0;
OCIBind *c_zip_bp = (OCIBind *) 0;
OCIBind *c_phone_bp = (OCIBind *) 0;
OCIBind *c_discount_bp = (OCIBind *) 0;
OCIBind *c_credit_bp = (OCIBind *) 0;
OCIBind *c_data_bp = (OCIBind *) 0;

OCIBind *i_id_bp = (OCIBind *) 0;
OCIBind *i_im_id_bp = (OCIBind *) 0;
OCIBind *i_name_bp = (OCIBind *) 0;
OCIBind *i_price_bp = (OCIBind *) 0;
OCIBind *i_data_bp = (OCIBind *) 0;

OCIDefine *s_s_ret_bp = (OCIDefine *) 0;
OCIBind *s_s_i_id_bp = (OCIBind *) 0;
OCIBind *s_s_w_id_bp = (OCIBind *) 0;

OCIBind *s_i_id_bp = (OCIBind *) 0;
OCIBind *s_w_id_bp = (OCIBind *) 0;
OCIBind *s_quantity_bp = (OCIBind *) 0;
OCIBind *s_dist_01_bp = (OCIBind *) 0;
OCIBind *s_dist_02_bp = (OCIBind *) 0;
OCIBind *s_dist_03_bp = (OCIBind *) 0;
OCIBind *s_dist_04_bp = (OCIBind *) 0;
OCIBind *s_dist_05_bp = (OCIBind *) 0;
OCIBind *s_dist_06_bp = (OCIBind *) 0;
OCIBind *s_dist_07_bp = (OCIBind *) 0;
OCIBind *s_dist_08_bp = (OCIBind *) 0;
OCIBind *s_dist_09_bp = (OCIBind *) 0;
OCIBind *s_dist_10_bp = (OCIBind *) 0;
OCIBind *s_data_bp = (OCIBind *) 0;

OCIBind *h_c_id_bp = (OCIBind *) 0;
OCIBind *h_c_d_id_bp = (OCIBind *) 0;
OCIBind *h_c_w_id_bp = (OCIBind *) 0;
OCIBind *h_d_id_bp = (OCIBind *) 0;
```

```

OCIBind *h_w_id_bp = (OCIBind *) 0;
OCIBind *h_data_bp = (OCIBind *) 0;

OCIBind *ol_o_id_bp = (OCIBind *) 0;
OCIBind *ol_d_id_bp = (OCIBind *) 0;
OCIBind *ol_w_id_bp = (OCIBind *) 0;
OCIBind *ol_i_id_bp = (OCIBind *) 0;
OCIBind *ol_number_bp = (OCIBind *) 0;
OCIBind *ol_supply_w_id_bp = (OCIBind *) 0;
OCIBind *ol_dist_info_bp = (OCIBind *) 0;
OCIBind *ol_amount_bp = (OCIBind *) 0;

OCIBind *o_id_bp = (OCIBind *) 0;
OCIBind *o_d_id_bp = (OCIBind *) 0;
OCIBind *o_w_id_bp = (OCIBind *) 0;
OCIBind *o_c_id_bp = (OCIBind *) 0;
OCIBind *o_carrier_id_bp = (OCIBind *) 0;
OCIBind *o_ol_cnt_bp = (OCIBind *) 0;
OCIBind *o_ocnt_bp = (OCIBind *) 0;
OCIBind *o_olcnt_bp = (OCIBind *) 0;

OCIBind *no_o_id_bp = (OCIBind *) 0;
OCIBind *no_d_id_bp = (OCIBind *) 0;
OCIBind *no_w_id_bp = (OCIBind *) 0;

void myusage()
{
    fprintf(stderr, "\n");
    fprintf(stderr, "Usage: ttpccload -M
<multiplier> [options]\n");
    fprintf(stderr, "options:\n");
    fprintf(stderr, "\t-A :tload all tables\n");
    fprintf(stderr, "\t-w :tload ware table\n");
    fprintf(stderr, "\t-d :tload dist table\n");
    fprintf(stderr, "\t-c :tload cust table (cluster
around c_w_id)\n");
    fprintf(stderr, "\t-C :tload cust table (cluster
around c_id)\n");
    fprintf(stderr, "\t-i :tload item table\n");
    fprintf(stderr, "\t-s :tload stok table (cluster
around s_w_id)\n");
    fprintf(stderr, "\t-S :tload stok table (cluster
around s_i_id)\n");
    fprintf(stderr, "\t-h :tload hist table\n");
    fprintf(stderr, "\t-n :tload new-order table\n");
    fprintf(stderr, "\t-o <oline file> :tload order and
order-line table\n");
    fprintf(stderr, "\t-b <ware#> :tbeginning ware
number\n");
    fprintf(stderr, "\t-e <ware#> :tending ware
number\n");
    fprintf(stderr, "\t-j <item#> :tbeginning item
number (with -S)\n");
    fprintf(stderr, "\t-k <item#> :tending item
number (with -S)\n");
    fprintf(stderr, "\t-l <cid#> :tbeginning cid
number (with -C)\n");
    fprintf(stderr, "\t-m <cid#> :tending cid
number (with -C)\n");
    fprintf(stderr, "\t-g :tgenerate rows to standard
output\n");
    fprintf(stderr, "\t $tpcc_bench must be set to
the location of the kit\n");
    fprintf(stderr, "\n");
    exit(1);
}

int sqlfile(fnam, linebuf)
char *fnam;
text *linebuf;
{
    FILE *fd;
    int nulpt = 0;
    char realfile[512];

```

```

    sprintf(realfile, "%s", fnam);
    fd = fopen(realfile, "r");
    if (!fd)
    {
        return (0);
    }
    while (fgets((char *)linebuf+nulpt,
SQL_BUF_SIZE, fd))
    {
        nulpt = strlen((char *)linebuf);
    }
    return(nulpt);
}

void quit()
{
    OCIERROR(errhp, OCISessionEnd
(tpcsvc, errhp, tpcusr, OCI_DEFAULT));
    OCIERROR(errhp, OCIServerDetach ( tpcsrv,
errhp, OCI_DEFAULT));
    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);
}

void main (argc, argv)
int argc;
char *argv[];
{
    char *uid="tpcc";
    char *pwd="tpcc";
    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 s_c_id;
    int s_c_d_id;
    int s_c_w_id;

```

```

    int s_c_count;

    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_s_count;
    int s_s_i_id;
    int s_s_w_id;

    int s_i_id[100];
    int s_w_id[100];
    int s_quantity[100];
    char s_dist_01[100][25];
    char s_dist_02[100][25];
    char s_dist_03[100][25];
    char s_dist_04[100][25];
    char s_dist_05[100][25];
    char s_dist_06[100][25];
    char s_dist_07[100][25];
    char s_dist_08[100][25];
    char s_dist_09[100][25];
    char s_dist_10[100][25];
    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[1500];
    int ol_d_id[1500];
    int ol_w_id[1500];
    int ol_number[1500];
    int ol_i_id[1500];
    int ol_supply_w_id[1500];
    int ol_amount[1500];
    char ol_dist_info[1500][24];
    int o_cnt;
    int ol_cnt;

    ub2 ol_o_id_len[1500];
    ub2 ol_d_id_len[1500];
    ub2 ol_w_id_len[1500];
    ub2 ol_number_len[1500];
    ub2 ol_i_id_len[1500];
    ub2 ol_supply_w_id_len[1500];
    ub2 ol_dist_info_len[1500];
    ub2 ol_amount_len[1500];

    ub4 ol_o_id_clen;

```

```

ub4 ol_d_id_clen;
ub4 ol_w_id_clen;
ub4 ol_number_clen;
ub4 ol_i_id_clen;
ub4 ol_supply_w_id_clen;
ub4 ol_dist_info_clen;
ub4 ol_amount_clen;

ub2 o_id_len[100];
ub2 o_d_id_len[100];
ub2 o_w_id_len[100];
ub2 o_c_id_len[100];
ub2 o_carrier_id_len[100];
ub2 o_ol_cnt_len[100];

ub4 o_id_clen;
ub4 o_d_id_clen;
ub4 o_w_id_clen;
ub4 o_c_id_clen;
ub4 o_carrier_id_clen;
ub4 o_ol_cnt_clen;

text stmbuff[16*1024];

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:AwdcCisShno:b:e:j:k:l:m:g";
int do_A=0;
int do_w=0;
int do_d=0;
int do_i=0;
int do_c=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;
int bcid=1;
int ecid=0;

FILE *olfp=NULL;
char olfname[100];
char* basename;
int status;
#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 'S': do_s = 1;
            break;
        case 'H': do_h = 1;
            break;
        case 'O': do_o = 1;
            break;
        case 'N': do_n = 1;
            break;
        case 'G': gen = 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 'L': bcid = atoi (*argv++);
            break;
        case 'M': ecid = atoi (*argv++);
            break;
        case 'G': gen = 1;
            strcpy (fname, *argv++);
            break;
        case 'L': logfile=fopen(*argv+,"w");
            break;
        default: printf (stderr, "THIS SHOULD
            NEVER HAPPEN!!!\n");
            printf (stderr, "(reached default case
            in getopt ())\n");
            myusage ();
        }
    }
}
#endif /* ORA_NT */

/*-----*
| Rudimentary error checking |
|-----*

if (scale < 1) {
    printf (stderr, "Invalid scale factor: %d\n",
    scale);
    myusage ();
}

if (!(do_A || do_w || do_d || do_c || do_C || do_i
|| do_s || do_S || do_h || do_o ||
do_n)) {
    printf (stderr, "What should I load???\n");
    myusage ();
}

if (gen && (do_A || (do_w + do_d + do_c +
do_C + do_i + do_s + do_S + do_h + do_o +

```



```

DISCARD strcat(fname, "/");
DISCARD strcat(fname,
"benchrun/blocks/load_ordord.sql");
stat = sqlfile(fname, stmbuf);
if (!stat)
{
    fprintf(stderr, "unable to open %s
\n", fname);
    quit();
    exit(1);
}
OCIERROR(errhp, OCIStmtPrepare(curo1,
errhp, stmbuf,
    strlen((char *)stmbuf), (ub4)
OCI_NTV_SYNTAX, (ub4) OCI_DEFAULT));
}

if (do_A || do_n) {

OCIERROR(errhp, OCIHandleAlloc(tpcenv, (dvoi
d **)&curno, OCI_HTYPE_STMT, 0,
(dvoid **)0);
OCIERROR(errhp, OCIStmtPrepare(curno,
errhp, (text *)SQLTXTNO,
    strlen((char *)SQLTXTNO), (ub4)
OCI_NTV_SYNTAX, (ub4) OCI_DEFAULT));
}

/* bind variables */

/* warehouse */

if (do_A || do_w) {
OCIERROR(errhp, OCIBindByName(curw,
&w_id_bp, errhp, (text *)":w_id",
strlen(":w_id"),
    (ub1 *)&w_id, sizeof(w_id),
SQLT_INT, (dvoid *) 0, (ub2 *)0, (ub2 *)0,
    (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curw,
&w_name_bp, errhp, (text *)":w_name",
strlen(":w_name"),
    (ub1 *)w_name, 11, SQLT_STR,
(dvoid *) 0, (ub2 *)0, (ub2 *)0,
    (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curw,
&w_street1_bp, errhp, (text *)":w_street_1",
strlen(":w_street_1"), (ub1 *)w_street_1,
21, SQLT_STR,
    (dvoid *) 0, (ub2 *)0, (ub2 *)0,
    (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curw,
&w_street2_bp, errhp, (text *)":w_street_2",
strlen(":w_street_2"), (ub1 *)w_street_2,
21, SQLT_STR,
    (dvoid *) 0, (ub2 *)0, (ub2 *)0,
    (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curw,
&w_city_bp, errhp, (text *)":w_city",
strlen(":w_city"), (ub1 *)w_city, 21,
SQLT_STR,
    (dvoid *) 0, (ub2 *)0, (ub2 *)0,
    (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curw,
&w_state_bp, errhp, (text *)":w_state",

```

```

    strlen(":w_state"), (ub1 *)w_state, 2,
SQLT_CHAR,
    (dvoid *) 0, (ub2 *)0, (ub2 *)0,
    (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curw,
&w_zip_bp, errhp, (text *)":w_zip",
strlen(":w_zip"), (ub1 *)w_zip, 9,
SQLT_CHAR,
    (dvoid *) 0, (ub2 *)0, (ub2 *)0,
    (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curw,
&w_tax_bp, errhp, (text *)":w_tax",
strlen(":w_tax"), (ub1 *) &w_tax,
sizeof(w_tax), SQLT_FLT,
    (dvoid *) 0, (ub2 *)0, (ub2 *)0,
    (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));
}

/* district */

if (do_A || do_d) {
OCIERROR(errhp, OCIBindByName(curd,
&d_id_bp, errhp, (text *)":d_id",
strlen(":d_id"), (ub1 *)d_id, sizeof(int),
SQLT_INT,
    (dvoid *) 0, (ub2 *)0, (ub2 *)0,
    (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curd,
&d_w_id_bp, errhp, (text *)":d_w_id",
strlen(":d_w_id"), (ub1 *)d_w_id,
sizeof(int), SQLT_INT,
    (dvoid *) 0, (ub2 *)0, (ub2 *)0,
    (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curd,
&d_name_bp, errhp, (text *)":d_name",
strlen(":d_name"), (ub1 *)d_name, 11,
SQLT_STR,
    (dvoid *) 0, (ub2 *)0, (ub2 *)0,
    (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curd,
&d_street1_bp, errhp, (text *)":d_street_1",
strlen(":d_street_1"), (ub1 *)d_street_1,
21, SQLT_STR,
    (dvoid *) 0, (ub2 *)0, (ub2 *)0,
    (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curd,
&d_street2_bp, errhp, (text *)":d_street_2",
strlen(":d_street_2"), (ub1 *)d_street_2,
21, SQLT_STR,
    (dvoid *) 0, (ub2 *)0, (ub2 *)0,
    (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curd,
&d_city_bp, errhp, (text *)":d_city",
strlen(":d_city"), (ub1 *)d_city, 21,
SQLT_STR,
    (dvoid *) 0, (ub2 *)0, (ub2 *)0,
    (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

```

```

OCIERROR(errhp, OCIBindByName(curd,
&d_state_bp, errhp, (text *)":d_state",
strlen(":d_state"), (ub1 *)d_state, 2,
SQLT_CHAR,
    (dvoid *) 0, (ub2 *)0, (ub2 *)0,
    (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curd,
&d_zip_bp, errhp, (text *)":d_zip",
strlen(":d_zip"), (ub1 *)d_zip, 9,
SQLT_CHAR,
    (dvoid *) 0, (ub2 *)0, (ub2 *)0,
    (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curd,
&d_tax_bp, errhp, (text *)":d_tax",
strlen(":d_tax"), (ub1 *)d_tax,
sizeof(float), SQLT_FLT,
    (dvoid *) 0, (ub2 *)0, (ub2 *)0,
    (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));
}

/* customer */

if (do_A || do_c || do_c) {
OCIERROR(errhp, OCIBindByName(curcs,
&s_c_id_bp, errhp, (text *)":s_c_id",
strlen(":s_c_id"), (ub1 *)&s_c_id,
sizeof(int), SQLT_INT,
    (dvoid *) 0, (ub2 *)0, (ub2 *)0,
    (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curcs,
&s_c_w_id_bp, errhp, (text *)":s_c_w_id",
strlen(":s_c_w_id"), (ub1
*)&s_c_w_id, sizeof(int), SQLT_INT,
    (dvoid *) 0, (ub2 *)0, (ub2 *)0,
    (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curcs,
&s_c_d_id_bp, errhp, (text *)":s_c_d_id",
strlen(":s_c_d_id"), (ub1
*)&s_c_d_id, sizeof(int), SQLT_INT,
    (dvoid *) 0, (ub2 *)0, (ub2 *)0,
    (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

OCIDefineByPos(curcs, &s_c_ret_bp, errhp, 1, &s_
c_count, sizeof(int), SQLT_INT, \
    0, 0, 0, OCI_DEFAULT);

OCIERROR(errhp, OCIBindByName(curc,
&c_id_bp, errhp, (text *)":c_id",
strlen(":c_id"), (ub1 *)c_id,
sizeof(int), SQLT_INT,
    (dvoid *) 0, (ub2 *)0, (ub2 *)0,
    (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curc,
&c_d_id_bp, errhp, (text *)":c_d_id",
strlen(":c_d_id"), (ub1 *)c_d_id,
sizeof(int), SQLT_INT,
    (dvoid *) 0, (ub2 *)0, (ub2 *)0,
    (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curc,
&c_w_id_bp, errhp, (text *)":c_w_id",

```

```

        strlen(":c_w_id"), (ub1 *)c_w_id,
sizeof(int), SOLT_INT,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curc,
&c_first_bp, errhp, (text *)":c_first",
        strlen(":c_first"), (ub1 *)c_first, 17,
SOLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curc,
&c_last_bp, errhp, (text *)":c_last",
        strlen(":c_last"), (ub1 *)c_last, 17,
SOLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curc,
&c_street1_bp, errhp, (text *)":c_street_1",
        strlen(":c_street_1"), (ub1
*)c_street_1, 21, SOLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curc,
&c_street2_bp, errhp, (text *)":c_street_2",
        strlen(":c_street_2"), (ub1
*)c_street_2, 21, SOLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curc,
&c_city_bp, errhp, (text *)":c_city",
        strlen(":c_city"), (ub1 *)c_city, 21,
SOLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curc,
&c_state_bp, errhp, (text *)":c_state",
        strlen(":c_state"), (ub1 *)c_state, 2,
SOLT_CHR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curc,
&c_zip_bp, errhp, (text *)":c_zip",
        strlen(":c_zip"), (ub1 *)c_zip, 9,
SOLT_CHR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curc,
&c_phone_bp, errhp, (text *)":c_phone",
        strlen(":c_phone"), (ub1 *)c_phone,
16, SOLT_CHR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curc,
&c_credit_bp, errhp, (text *)":c_credit",
        strlen(":c_credit"), (ub1 *)c_credit,
2, SOLT_CHR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,

```

```

        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curc,
&c_discount_bp, errhp, (text *)":c_discount",
        strlen(":c_discount"), (ub1
*)c_discount, sizeof(float), SOLT_FLT,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curc,
&c_data_bp, errhp, (text *)":c_data",
        strlen(":c_data"), (ub1 *)c_data,
501, SOLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));
    }
}

/* item */

    if (do_A || do_i) {
        OCIERROR(errhp, OCIBindByName(curi,
&i_id_bp, errhp, (text *)":i_id",
        strlen(":i_id"), (ub1 *)i_id, sizeof(int),
SOLT_INT,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curi,
&i_im_id_bp, errhp, (text *)":i_im_id",
        strlen(":i_im_id"), (ub1 *)i_im_id,
sizeof(int), SOLT_INT,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curi,
&i_name_bp, errhp, (text *)":i_name",
        strlen(":i_name"), (ub1 *)i_name,
25, SOLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curi,
&i_price_bp, errhp, (text *)":i_price",
        strlen(":i_price"), (ub1 *)i_price,
sizeof(int), SOLT_INT,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curi,
&i_data_bp, errhp, (text *)":i_data",
        strlen(":i_data"), (ub1 *)i_data, 51,
SOLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));
    }
}

/* stock */

    if (do_A || do_s || do_S) {
        OCIERROR(errhp, OCIBindByName(curss,
&s_s_i_id_bp, errhp, (text *)":s_s_i_id",
        strlen(":s_s_i_id"), (ub1
*)&s_s_i_id, sizeof(int), SOLT_INT,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

```

```

        OCIERROR(errhp, OCIBindByName(curss,
&s_s_w_id_bp, errhp, (text *)":s_s_w_id",
        strlen(":s_s_w_id"), (ub1
*)&s_s_w_id, sizeof(int), SOLT_INT,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIDefineByPos(curss, &s_s_ret_bp, errhp, 1, &s_
s_count, sizeof(int), SOLT_INT, \
        0, 0, 0, OCI_DEFAULT);

        OCIERROR(errhp, OCIBindByName(curs,
&s_i_id_bp, errhp, (text *)":s_i_id",
        strlen(":s_i_id"), (ub1 *)s_i_id,
sizeof(int), SOLT_INT,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curs,
&s_w_id_bp, errhp, (text *)":s_w_id",
        strlen(":s_w_id"), (ub1 *)s_w_id,
sizeof(int), SOLT_INT,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curs,
&s_quantity_bp, errhp, (text *)":s_quantity",
        strlen(":s_quantity"), (ub1
*)s_quantity, sizeof(int), SOLT_INT,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curs,
&s_dist_01_bp, errhp, (text *)":s_dist_01",
        strlen(":s_dist_01"), (ub1
*)s_dist_01, 25, SOLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curs,
&s_dist_02_bp, errhp, (text *)":s_dist_02",
        strlen(":s_dist_02"), (ub1
*)s_dist_02, 25, SOLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curs,
&s_dist_03_bp, errhp, (text *)":s_dist_03",
        strlen(":s_dist_03"), (ub1
*)s_dist_03, 25, SOLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curs,
&s_dist_04_bp, errhp, (text *)":s_dist_04",
        strlen(":s_dist_04"), (ub1
*)s_dist_04, 25, SOLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curs,
&s_dist_05_bp, errhp, (text *)":s_dist_05",

```



```

        strlen(":s_dist_05"), (ub1
*)s_dist_05, 25, SQLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curs,
&s_dist_06_bp, errhp, (text *)":s_dist_06",
        strlen(":s_dist_06"), (ub1
*)s_dist_06, 25, SQLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curs,
&s_dist_07_bp, errhp, (text *)":s_dist_07",
        strlen(":s_dist_07"), (ub1
*)s_dist_07, 25, SQLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curs,
&s_dist_08_bp, errhp, (text *)":s_dist_08",
        strlen(":s_dist_08"), (ub1
*)s_dist_08, 25, SQLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curs,
&s_dist_09_bp, errhp, (text *)":s_dist_09",
        strlen(":s_dist_09"), (ub1
*)s_dist_09, 25, SQLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curs,
&s_dist_10_bp, errhp, (text *)":s_dist_10",
        strlen(":s_dist_10"), (ub1
*)s_dist_10, 25, SQLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curs,
&s_data_bp, errhp, (text *)":s_data",
        strlen(":s_data"), (ub1 *)s_data, 51,
        SQLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));
    }

    /* history */

    if (do_A || do_h) {
        OCIERROR(errhp, OCIBindByName(curh,
&h_c_id_bp, errhp, (text *)":h_c_id",
        strlen(":h_c_id"), (ub1 *)h_c_id,
        sizeof(int), SQLT_INT,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curh,
&h_c_d_id_bp, errhp, (text *)":h_c_d_id",
        strlen(":h_c_d_id"), (ub1 *)h_d_id,
        sizeof(int), SQLT_INT,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curo1,
&ol_w_id_bp, errhp, (text *)":ol_w_id",
        strlen(":ol_w_id"), (ub1 *)ol_w_id,
        sizeof(int), SQLT_INT,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curo1,
&ol_number_bp, errhp, (text *)":ol_number",
        strlen(":ol_number"), (ub1
*)ol_number, sizeof(int), SQLT_INT,
        (dvoid *) 0, (ub2 *)ol_number_len,
        (ub2 *)0,
        (ub4) 15*ORDEARR, (ub4 *)
&ol_number_clen, (ub4) OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curo1,
&ol_i_id_bp, errhp, (text *)":ol_i_id",
        strlen(":ol_i_id"), (ub1 *)ol_i_id,
        sizeof(int), SQLT_INT,
        (dvoid *) 0, (ub2 *)ol_i_id_len, (ub2
*)0,
        (ub4) 15*ORDEARR, (ub4 *)
&ol_i_id_clen, (ub4) OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curo1,
&ol_supply_w_id_bp, errhp, (text
*)":ol_supply_w_id",
        strlen(":ol_supply_w_id"), (ub1
*)ol_supply_w_id, sizeof(int), SQLT_INT,
        (dvoid *) 0, (ub2
*)ol_supply_w_id_len, (ub2 *)0,
        (ub4) 15*ORDEARR, (ub4 *)
&ol_supply_w_id_clen, (ub4) OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curo1,
&ol_dist_info_bp, errhp, (text *)":ol_dist_info",
        strlen(":ol_dist_info"), (ub1
*)ol_dist_info, 24, SQLT_CHR,
        (dvoid *) 0, (ub2 *)ol_dist_info_len,
        (ub2 *)0,
        (ub4) 15*ORDEARR, (ub4 *)
&ol_dist_info_clen, (ub4) OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curo1,
&ol_amount_bp, errhp, (text *)":ol_amount",
        strlen(":ol_amount"), (ub1
*)ol_amount, sizeof(int), SQLT_INT,
        (dvoid *) 0, (ub2 *)ol_amount_len,
        (ub2 *)0,
        (ub4) 15*ORDEARR, (ub4 *)
&ol_amount_clen, (ub4) OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curo1,
&o_id_bp, errhp, (text *)":o_id",
        strlen(":o_id"), (ub1 *)o_id,
        sizeof(int), SQLT_INT,
        (dvoid *) 0, (ub2 *)o_id_len, (ub2
*)0,
        (ub4) ORDEARR, (ub4 *)
&o_id_clen, (ub4) OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curo1,
&o_d_id_bp, errhp, (text *)":o_d_id",
        strlen(":o_d_id"), (ub1 *)o_d_id,
        sizeof(int), SQLT_INT,
        (dvoid *) 0, (ub2 *)o_d_id_len, (ub2
*)0,
        (ub4) ORDEARR, (ub4 *)
&o_d_id_clen, (ub4) OCI_DEFAULT));

        OCIERROR(errhp, OCIBindByName(curo1,
&o_w_id_bp, errhp, (text *)":o_w_id",
        strlen(":o_w_id"), (ub1 *)o_w_id,
        sizeof(int), SQLT_INT,
        (dvoid *) 0, (ub2 *)o_w_id_len,
        (ub2 *)0,
        (ub4) ORDEARR, (ub4 *)
&o_w_id_clen, (ub4) OCI_DEFAULT));
    }

    (dvoid *) 0, (ub2 *)0, (ub2 *)0,
    (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

    OCIERROR(errhp, OCIBindByName(curh,
&h_c_w_id_bp, errhp, (text *)":h_c_w_id",
        strlen(":h_c_w_id"), (ub1 *)h_w_id,
        sizeof(int), SQLT_INT,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

    OCIERROR(errhp, OCIBindByName(curh,
&h_d_id_bp, errhp, (text *)":h_d_id",
        strlen(":h_d_id"), (ub1 *)h_d_id,
        sizeof(int), SQLT_INT,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

    OCIERROR(errhp, OCIBindByName(curh,
&h_w_id_bp, errhp, (text *)":h_w_id",
        strlen(":h_w_id"), (ub1 *)h_w_id,
        sizeof(int), SQLT_INT,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));

    OCIERROR(errhp, OCIBindByName(curh,
&h_data_bp, errhp, (text *)":h_data",
        strlen(":h_data"), (ub1 *)h_data, 25,
        SQLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4)
OCI_DEFAULT));
}

/* order and order_line (delivered) */

if (do_A || do_o) {
    for (i = 0; i < ORDEARR; i++) {
        o_id_len[i] = sizeof(int);
        o_d_id_len[i] = sizeof(int);
        o_w_id_len[i] = sizeof(int);
        o_c_id_len[i] = sizeof(int);
        o_carrier_id_len[i] = sizeof(int);
        o_ol_cnt_len[i] = sizeof(int);
    }

    OCIERROR(errhp, OCIBindByName(curo1,
&ol_o_id_bp, errhp, (text *)":ol_o_id",
        strlen(":ol_o_id"), (ub1 *)ol_o_id,
        sizeof(int), SQLT_INT,
        (dvoid *) 0, (ub2 *)ol_o_id_len,
        (ub2 *)0,
        (ub4) 15*ORDEARR, (ub4
*)&ol_o_id_clen, (ub4) OCI_DEFAULT));

    OCIERROR(errhp, OCIBindByName(curo1,
&ol_d_id_bp, errhp, (text *)":ol_d_id",
        strlen(":ol_d_id"), (ub1 *)ol_d_id,
        sizeof(int), SQLT_INT,
        (dvoid *) 0, (ub2 *)ol_d_id_len,
        (ub2 *)0,
        (ub4) 15*ORDEARR, (ub4 *)
&ol_d_id_clen, (ub4) OCI_DEFAULT));

    OCIERROR(errhp, OCIBindByName(curo1,
&ol_w_id_bp, errhp, (text *)":ol_w_id",
        strlen(":ol_w_id"), (ub1 *)ol_w_id,
        sizeof(int), SQLT_INT,
        (dvoid *) 0, (ub2 *)ol_w_id_len,
        (ub2 *)0,
        (ub4) 15*ORDEARR, (ub4 *)
&ol_w_id_clen, (ub4) OCI_DEFAULT));
}

```



```

}
/*-----+
| Load the CUSTOMER table. |
+-----*/

if (do_A || do_c) {

    nrows = (eware - bware + 1) * CUSTFAC *
DISTFAC;

    fprintf (stderr, "Loading/generating customer:
w%d - w%d (%d rows)\n ",
        bware, eware, nrows);

    if (getenv("tpcc_hash_overflow")) {
        fprintf(stderr, "Hash overflow is enabled\n");
        OCIHandleAlloc(tpcenv, (dvoid **)&curi,
OCI_HTYPE_STMT, 0, (dvoid**)0);
        sprintf ((char *) stmbuf, SQLTXTENHA);
        OCIStmtPrepare(cur, errhp, stmbuf,
strlen((char *)stmbuf),
OCI_NTV_SYNTAX,
OCI_DEFAULT);
        OCIERROR(errhp, OCIStmtExecute(tpcsvc,
cur, errhp, 1, 0, 0, OCI_DEFAULT));
        OCIHandleFree(cur, OCI_HTYPE_STMT);
        fprintf (stderr, "Customer loaded for
horizontal partitioning\n");
    }
    else
    {
        fprintf (stderr, "Customer not loaded for
horizontal partitioning\n");
    }
    begin_time = gettime ();
    begin_cpu = getcpu ();

    s_c_id = 1;
    s_c_d_id = 1;
    s_c_w_id = bware;

    while (s_c_w_id <= eware) {
        status = OCIStmtExecute(tpcsvc, curcs,
errhp, (ub4) 1, (ub4) 0,
(CONST OCISnapshot*) 0,
(OCISnapshot*) 0,
(ub4) OCI_DEFAULT |
OCI_COMMIT_ON_SUCCESS);
        if (status != OCI_SUCCESS) {
            OCIERROR(errhp, status);
            quit ();
            exit (1);
        }

        if (s_c_count == 0) {
            s_c_w_id--;
            break;
        }
        else s_c_w_id++;
    }

    if (s_c_w_id < bware) s_c_w_id = bware;
    else {
        if (s_c_w_id > eware) s_c_w_id = eware;
        while (s_c_d_id <= DISTFAC) {
            status = OCIStmtExecute(tpcsvc, curcs,
errhp, (ub4) 1, (ub4) 0,
(CONST OCISnapshot*) 0,
(OCISnapshot*) 0,
(ub4) OCI_DEFAULT |
OCI_COMMIT_ON_SUCCESS);
            if (status != OCI_SUCCESS) {
                fprintf (stderr, "Select failed\n");

```

```

OCIERROR(errhp, status);
            quit ();
            exit (1);
        }
    }
    if (s_c_count == 0) {
        s_c_d_id--;
        break;
    }
    else s_c_d_id++;
}
if (s_c_d_id > DISTFAC) s_c_d_id =
DISTFAC;

while (s_c_id <= CUSTFAC) {
    status = OCIStmtExecute(tpcsvc, curcs,
errhp, (ub4) 1, (ub4) 0,
(CONST OCISnapshot*) 0,
(OCISnapshot*) 0,
(ub4) OCI_DEFAULT |
OCI_COMMIT_ON_SUCCESS);
    if (status != OCI_SUCCESS) {
        OCIERROR(errhp, status);
        quit ();
        exit (1);
    }
    if (s_c_count == 0) break;
    else s_c_id++;
}
if (s_c_id > CUSTFAC) {
    if (s_c_d_id == DISTFAC) {
        s_c_d_id=1;
        s_c_w_id++;
    } else {
        s_c_d_id++;
    }
    s_c_id=1;
}

fprintf (stderr, "start at wid: %d, did: %d,
cid: %d\n ", s_c_w_id, s_c_d_id, s_c_id);
cid = s_c_id - 1;
cdid = s_c_d_id;
cwid = s_c_w_id;
nrows = (eware - s_c_w_id + 1) * DISTFAC *
CUSTFAC - (s_c_d_id - 1) * CUSTFAC - s_c_id
+ 1;
fprintf (stderr, "remaining rows: %d\n ",
nrows);
loopcount = 0;

for (row = 0; row < nrows; ) {
    for (i = 0; i < CUSTARR && row < nrows;
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 */
            }
            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 %s %s %s\n",
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 {
        status = OCIStmtExecute(tpcsvc, curc,
errhp, (ub4) i, (ub4) 0,
(CONST OCISnapshot*) 0,
(OCISnapshot*) 0,
(ub4) OCI_DEFAULT |
OCI_COMMIT_ON_SUCCESS);

        if (status != OCI_SUCCESS) {
            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]);
            OCIERROR(errhp, status);
            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 %10.2f sec. (%10.2f
cpu)\n\n",
    nrows < 0 ? 0 : nrows, end_time -
begin_time, end_cpu - begin_cpu);
if (getenv("tpcc_hash_overflow")) {
    fprintf(stderr, "Hash overflow is disabled\n");
    OCIHandleAlloc(tpcenv, (dvoid **)&curi,
OCI_HTYPE_STMT, 0, (dvoid**)0);
    sprintf ((char *) stmbuf, SQLTXTDIHA);
    OCIStmtPrepare(cur, errhp, stmbuf,
strlen((char *)stmbuf),

```

```

OCI_NTV_SYNTAX,
OCI_DEFAULT);
OCIERROR(errhp,OCIStmtExecute(tpcsvc,
curi, errhp,1,0,0,OCI_DEFAULT));
OCIHandleFree(curi, OCI_HTYPE_STMT);
}

/*-----+
| Load the CUSTOMER table (cluster around
c_id) |
+-----*/

if (do_C) {

    srand (bcid);
#ifdef ORA_NT
    srand48 (bcid);
#endif

    nrows = (ecid - bcid + 1) * (eware - bware
+1) * DISTFAC;

    fprintf (stderr, "Loading/generating customer:
c%d - c%d, w%d - w%d (%d rows)\n ",
            bcid, ecid, bware, eware, nrows);

    if (getenv("tpcc_hash_overflow")) {
        fprintf(stderr, "Hash overflow is enabled\n");
        OCIHandleAlloc(tpcenv, (dvoid **)&curi,
OCI_HTYPE_STMT, 0, (dvoid**)0);
        sprintf ((char *) stmbuf, SQLTXTENHA);
        OCIStmtPrepare(curi, errhp, stmbuf,
strlen((char *)stmbuf),
OCI_NTV_SYNTAX,
OCI_DEFAULT);
        OCIERROR(errhp,OCIStmtExecute(tpcsvc,
curi, errhp,1,0,0,OCI_DEFAULT));
        OCIHandleFree(curi, OCI_HTYPE_STMT);
        fprintf (stderr, "Customer loaded for
horizontal partitioning\n");
    }
    else
    {
        fprintf (stderr, "Customer not loaded for
horizontal partitioning\n");
    }
    begin_time = gettime ();
    begin_cpu = getcpu ();

    s_c_id = bcid;
    s_c_d_id = 1;
    s_c_w_id = bware;

    while (s_c_id <= ecid) {
        status = OCIStmtExecute(tpcsvc, curcs,
errhp, (ub4) 1, (ub4) 0,
(CONST OCISnapshot*) 0,
(OCISnapshot*) 0,
(ub4) OCI_DEFAULT |
OCI_COMMIT_ON_SUCCESS);
        if (status != OCI_SUCCESS) {
            OCIERROR(errhp, status);
            quit ();
            exit (1);
        }

        if (s_c_count == 0) {
            s_c_id--;
            break;
        }
        else s_c_id++;
    }
}

```

```

if (s_c_id < bcid) s_c_id = bcid;
else {
    if (s_c_id > ecid) s_c_id = ecid;
    while (s_c_w_id <= eware) {
        status = OCIStmtExecute(tpcsvc, curcs,
errhp, (ub4) 1, (ub4) 0,
(CONST OCISnapshot*) 0,
(OCISnapshot*) 0,
(ub4) OCI_DEFAULT |
OCI_COMMIT_ON_SUCCESS);
        if (status != OCI_SUCCESS) {
            fprintf (stderr, "Select failed\n");
            OCIERROR(errhp, status);
            quit ();
            exit (1);
        }
        if (s_c_count == 0) {
            s_c_w_id--;
            break;
        }
        else s_c_w_id++;
    }
    if (s_c_w_id > eware) s_c_w_id = eware;
    else if (s_c_w_id < bware) s_c_w_id =
bware;

    while (s_c_d_id <= DISTFAC) {
        status = OCIStmtExecute(tpcsvc, curcs,
errhp, (ub4) 1, (ub4) 0,
(CONST OCISnapshot*) 0,
(OCISnapshot*) 0,
(ub4) OCI_DEFAULT |
OCI_COMMIT_ON_SUCCESS);
        if (status != OCI_SUCCESS) {
            OCIERROR(errhp, status);
            quit ();
            exit (1);
        }
        if (s_c_count == 0) break;
        else s_c_d_id++;
    }
}

if (s_c_d_id > DISTFAC) {
    s_c_d_id=1;
    if (s_c_w_id==eware) {
        s_c_w_id=bware;
        s_c_id++;
    }
    else s_c_w_id++;
}

fprintf (stderr, "start at cid: %d, wid: %d,
did: %d\n ", s_c_id, s_c_w_id, s_c_d_id);
cid = s_c_id;
cdid = s_c_d_id-1;
cwid = s_c_w_id;
nrows = (ecid - s_c_id + 1) * (eware - bware
+ 1) * DISTFAC - (s_c_w_id - 1) * DISTFAC -
s_c_d_id + 1;
fprintf (stderr, "remaining rows: %d\n ",
nrows);
loopcount = 0;

for (row = 0; row < nrows; ) {
    for (i = 0; i < CUSTARR && row < nrows;
i++, row++) {
        cdid++;
        if (cdid > DISTFAC) { /* cycle dist id
*/
            cdid = 1; /* cheap mod */
            cwid++; /* shift dist cycle */
            if (cwid > eware) {

```

```

        cwid = bware; /* shift ware
cycle */
            cid++;
        }
    }
    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 {
    status = OCIStmtExecute(tpcsvc, curc,
errhp, (ub4) i, (ub4) 0,
(CONST OCISnapshot*) 0,
(OCISnapshot*) 0,
(ub4) OCI_DEFAULT |
OCI_COMMIT_ON_SUCCESS);

    if (status != OCI_SUCCESS) {
        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]);
        OCIERROR(errhp, status);
        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 %10.2f sec. (%10.2f
cpu)\n\n",
        nrows < 0 ? 0 : nrows, end_time -
begin_time, end_cpu - begin_cpu);
    if (getenv("tpcc_hash_overflow")) {
        fprintf(stderr, "Hash overflow is disabled\n");
        OCIHandleAlloc(tpcenv, (dvoid **)&curi,
OCI_HTYPE_STMT, 0, (dvoid**)0);
        sprintf ((char *) stmbuf, SQLTXDIHA);
        OCIStmtPrepare(cur, errhp, stmbuf,
strlen((char *)stmbuf),
            OCI_NTV_SYNTAX,
OCI_DEFAULT);
        OCIERROR(errhp, OCIStmtExecute(tpcsvc,
cur, errhp, 1, 0, 0, OCI_DEFAULT));
        OCIHandleFree(cur, OCI_HTYPE_STMT);
    }

/*-----+
| 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 {
            status = OCIStmtExecute(tpcsvc, curi,
errhp, (ub4) ITEMARR, (ub4) 0,
                (CONST OCISnapshot*) 0,
(OCISnapshot*) 0,
                    (ub4) OCI_DEFAULT |
OCI_COMMIT_ON_SUCCESS);
            if (status != OCI_SUCCESS) {
                fprintf (stderr, "Aborted at i_id %d\n",
i_id[0]);
                OCIERROR(errhp, status);
                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 %10.2f sec. (%10.2f
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 stock:
w%d - w%d (%d rows)\n ",
        bware, eware, nrows);

    begin_time = gettime ();
    begin_cpu = getcpu ();

    s_s_i_id = 1;
    s_s_w_id = bware;

    while (s_s_w_id <= eware) {
        status = OCIStmtExecute(tpcsvc, curss,
errhp, (ub4) 1, (ub4) 0,
            (CONST OCISnapshot*) 0,
(OCISnapshot*) 0,
                (ub4) OCI_DEFAULT |
OCI_COMMIT_ON_SUCCESS);
        if (status != OCI_SUCCESS) {
            OCIERROR(errhp, status);
            quit ();
            exit (1);
        }
        if (s_s_count == 0) {
            s_s_w_id--;
            break;
        }
        else s_s_w_id++;

        if (s_s_w_id < bware) s_s_w_id = bware;
        else {
            if (s_s_w_id > eware) s_s_w_id = eware;
            while (s_s_i_id <= STOCFAC) {
                status = OCIStmtExecute(tpcsvc, curss,
errhp, (ub4) 1, (ub4) 0,
                    (CONST OCISnapshot*) 0,
(OCISnapshot*) 0,
                        (ub4) OCI_DEFAULT |
OCI_COMMIT_ON_SUCCESS);
                if (status != OCI_SUCCESS) {
                    OCIERROR(errhp, status);
                    quit ();
                    exit (1);
                }
                if (s_s_count == 0) {
                    break;
                }
                else s_s_i_id++;
            }
        }
    }
}

```

```

    }
}
if (s_s_i_id > STOCFAC) {
    s_s_i_id = 1;
    s_s_w_id++;
}

    fprintf(stderr, "start at s_i_id: %d,
s_w_id: %d\n ", s_s_i_id, s_s_w_id);

    sid = s_s_i_id - 1;
    swid = s_s_w_id;
    nrows = (eware - s_s_w_id + 1) * STOCFAC
- (s_s_i_id - 1);
    fprintf (stderr, "remaining rows: %d\n ",
nrows);
    loopcount = 0;

    for (row = 0; row < nrows; ) {
        /* added row < nrows condition on next
line - alex.ni */
        for (i = 0; (i < STOCARR) && (row < nrows);
i++, row++) {
            if (++sid > STOCFAC) { /* cheap
mod */
                sid = 1;
                swid++;
            }
            s_quantity[i] = (lrand48 () % 91) + 10;
            randstr (s_dist_01[i], 24, 24);
            randstr (s_dist_02[i], 24, 24);
            randstr (s_dist_03[i], 24, 24);
            randstr (s_dist_04[i], 24, 24);
            randstr (s_dist_05[i], 24, 24);
            randstr (s_dist_06[i], 24, 24);
            randstr (s_dist_07[i], 24, 24);
            randstr (s_dist_08[i], 24, 24);
            randstr (s_dist_09[i], 24, 24);
            randstr (s_dist_10[i], 24, 24);
            randdatastr (s_data[i], 26, 50);

            if (gen) {
                printf
("%d %d %d %s %s %s %s %s %s %s %s %s %s
s 0 0 0 %s\n",
                    sid, swid, s_quantity[i], s_dist_01[i],
s_dist_02[i],
                        s_dist_03[i], s_dist_04[i],
s_dist_05[i], s_dist_06[i],
                            s_dist_07[i], s_dist_08[i],
s_dist_09[i], s_dist_10[i],
                                s_data[i]);
            }
            else {
                s_i_id[i] = sid;
                s_w_id[i] = swid;
            }
        }

        if (gen) {
            fflush (stdout);
        }
        else {
            /* Changed to STOCKARR to i - alex.ni */
            status = OCIStmtExecute(tpcsvc, cur,
errhp, (ub4) i, (ub4) 0,
                (CONST OCISnapshot*) 0,
(OCISnapshot*) 0,
                    (ub4) OCI_DEFAULT |
OCI_COMMIT_ON_SUCCESS);
            if (status != OCI_SUCCESS) {
                fprintf (stderr, "Aborted at w_id %d,
s_i_id %d\n", s_w_id[0], s_i_id[0]);
                OCIERROR(errhp, status);
                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 %10.2f sec. (%10.2f
cpu)\n\n",
nrows < 0 ? 0 : 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 stock: i%d
-i%d, w%d - w%d (%d rows)\n ",
bitem, eitem, bware, eware, nrows);

begin_time = gettime ();
begin_cpu = getcpu ();

s_s_i_id = bitem;
s_s_w_id = bware;

while (s_s_i_id <= eitem) {
status = OCISmtExecute(tpcsvc, curss,
errhp, (ub4) 1, (ub4) 0,
(CONST OCISnapshot*) 0,
(OCISnapshot*) 0,
(ub4) OCI_DEFAULT |
OCI_COMMIT_ON_SUCCESS);
if (status != OCI_SUCCESS) {
OCIERROR(errhp, status);
quit ();
exit (1);
}
if (s_s_count == 0) {
s_s_i_id--;
break;
}
else s_s_i_id++;
}

if (s_s_i_id < bitem) s_s_i_id = bitem;
else {
if (s_s_i_id > eitem) s_s_i_id = eitem;
while (s_s_w_id <= eware) {
status = OCISmtExecute(tpcsvc, curss,
errhp, (ub4) 1, (ub4) 0,
(CONST OCISnapshot*) 0,
(OCISnapshot*) 0,
(ub4) OCI_DEFAULT |
OCI_COMMIT_ON_SUCCESS);
if (status != OCI_SUCCESS) {
OCIERROR(errhp, status);
quit ();
exit (1);
}
}
}
}

```

```

if (s_s_count == 0) {
break;
}
else s_s_w_id++;
}
}

if (s_s_w_id > eware) {
s_s_w_id = bware;
s_s_i_id++;
}

fprintf(stderr, "start at s_i_id: %d,
s_w_id: %d\n ", s_s_i_id, s_s_w_id);

sid = s_s_i_id;
swid = s_s_w_id - 1;
nrows = (eitem - s_s_i_id + 1) * (eware -
bware + 1) - (s_s_w_id - bware);
fprintf (stderr, "remaining rows: %d\n ",
nrows);
loopcount = 0;

for (row = 0; row < nrows; ) {
for (i = 0; i < STOCARR && row < nrows;
i++, row++) {
if (++swid > eware) { /* cheap mod
*/
swid = bware;
swid++;
}
s_quantity[i] = (lrand48 () % 91) + 10;
randstr (s_dist_01[i], 24, 24);
randstr (s_dist_02[i], 24, 24);
randstr (s_dist_03[i], 24, 24);
randstr (s_dist_04[i], 24, 24);
randstr (s_dist_05[i], 24, 24);
randstr (s_dist_06[i], 24, 24);
randstr (s_dist_07[i], 24, 24);
randstr (s_dist_08[i], 24, 24);
randstr (s_dist_09[i], 24, 24);
randstr (s_dist_10[i], 24, 24);
randdatastr (s_data[i], 26, 50);

if (gen) {
printf
("%d %d %d %d %s %s %s %s %s %s %s %s %s %s %s
s 0 0 0 %s\n",
sid, swid, s_quantity[i], s_dist_01[i],
s_dist_02[i],
s_dist_03[i], s_dist_04[i],
s_dist_05[i], s_dist_06[i],
s_dist_07[i], s_dist_08[i],
s_dist_09[i], s_dist_10[i],
s_data[i]);
}
else {
s_i_id[i] = sid;
s_w_id[i] = swid;
}
}

if (gen) {
fflush (stdout);
}
else {
status = OCISmtExecute(tpcsvc, curs,
errhp, (ub4) i, (ub4) 0,
(CONST OCISnapshot*) 0,
(OCISnapshot*) 0,
(ub4) OCI_DEFAULT |
OCI_COMMIT_ON_SUCCESS);
if (status != OCI_SUCCESS) {
fprintf (stderr, "Aborted at w_id %d,
s_i_id %d\n", s_w_id[0], s_i_id[0]);
OCIERROR(errhp, status);
}
}
}
}
}

```

```

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 %10.2f sec. (%10.2f
cpu)\n\n",
nrows < 0 ? 0 : 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 history:
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 district cycle
*/

if (cdid > DISTFAC) {
cdid = 1;
cwid++; /* shift warehouse
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 %d %s
1000 %s\n", cid, cdid, cwid, cdid,
cwid, sdate, h_data[i]);
}
}
}

if (gen) {
fflush (stdout);
}
else {
status = OCISmtExecute(tpcsvc, curh,
errhp, (ub4) HISTARR, (ub4) 0,
(CONST OCISnapshot*) 0,
(OCISnapshot*) 0,
}
}
}
}
}

```

```

        (ub4) OCI_DEFAULT |
OCI_COMMIT_ON_SUCCESS);
    if (status != OCI_SUCCESS) {
        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]);
        OCIERROR(errhp, status);
        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 %10.2f sec. (%10.2f
cpu)\n\n",
            nrows, end_time - begin_time, end_cpu
- begin_cpu);
}

/*-----+
| Load the ORDERS and ORDER-LINE table.
|
+-----*/

if (do_A || do_o) {

    int batch_olcnt;

    nrows = (eware - bware + 1) * ORDEFAC *
DISTFAC;

    fprintf (stderr, "Loading/generating orders
and order-line: w%d - w%d (%d ord, ~%d ord)\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; ) {

        batch_olcnt = 0;

        for (i = 0; i < ORDEARR; i++, row++) {
            cid++;
            if (cid > ORDEFAC) { /* cycle cust id
*/
                cid = 1; /* cheap mod */
                cdid++; /* shift district cycle
*/
                if (cdid > DISTFAC) {
                    cdid = 1;
                    cwid++; /* shift warehouse
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];
                if (cid >= 2101) {
                    o_carrier_id[i] = 11;
                }
            }

            for (j = 0; j < o_ol_cnt[i]; j++,
batch_olcnt++) {
                ol_i_id[batch_olcnt] = sid = lrand48 () %
100000 + 1;
                if (cid < 2101)
                    ol_amount[batch_olcnt] = 0;
                else
                    ol_amount[batch_olcnt] = (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[batch_olcnt], cwid,
ol_amount[batch_olcnt],
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[batch_olcnt], cwid,
ol_amount[batch_olcnt],
str24[j]);
                    }
                }
            }
            else {
                ol_o_id[batch_olcnt] = cid;
                ol_d_id[batch_olcnt] = cdid;
                ol_w_id[batch_olcnt] = cwid;
                ol_number[batch_olcnt] = j + 1;
                ol_supply_w_id[batch_olcnt] = cwid;
                strncpy (ol_dist_info[batch_olcnt],
str24[j], 24);
            }
        }
        if (gen) {
            fflush (olfp);
        }
    }

    o_cnt = ORDEARR;
    ol_cnt = batch_olcnt;

    for (j = 0; j < batch_olcnt; j++) {

```

```

        ol_o_id_len[j] = sizeof(int);
        ol_d_id_len[j] = sizeof(int);
        ol_w_id_len[j] = sizeof(int);
        ol_number_len[j] = sizeof(int);
        ol_i_id_len[j] = sizeof(int);
        ol_supply_w_id_len[j] = sizeof(int);
        ol_dist_info_len[j] = 24;
        ol_amount_len[j] = sizeof(int);
    }
    for (j = batch_olcnt; j < 15*ORDEARR; j++)
    {
        ol_o_id_len[j] = 0;
        ol_d_id_len[j] = 0;
        ol_w_id_len[j] = 0;
        ol_number_len[j] = 0;
        ol_i_id_len[j] = 0;
        ol_supply_w_id_len[j] = 0;
        ol_dist_info_len[j] = 0;
        ol_amount_len[j] = 0;
    }

    o_id_clen = ORDEARR;
    o_d_id_clen = ORDEARR;
    o_w_id_clen = ORDEARR;
    o_c_id_clen = ORDEARR;
    o_carrier_id_clen = ORDEARR;
    o_ol_cnt_clen = ORDEARR;

    ol_o_id_clen = batch_olcnt;
    ol_d_id_clen = batch_olcnt;
    ol_w_id_clen = batch_olcnt;
    ol_number_clen = batch_olcnt;
    ol_i_id_clen = batch_olcnt;
    ol_supply_w_id_clen = batch_olcnt;
    ol_dist_info_clen = batch_olcnt;
    ol_amount_clen = batch_olcnt;

    OCIERROR(errhp,
OCIStmtExecute(tpcsvc, cur01, errhp, (ub4) 1,
(ub4) 0,
                (CONST OCISnapshot*) 0,
(OCISnapshot*) 0,
                (ub4) OCI_DEFAULT |
OCI_COMMIT_ON_SUCCESS));

    if ((++loopcount) % 50) {
        fprintf (stderr, ".");
    }
    else {
        fprintf (stderr, "%d orders committed\n
", row);
    }
}

    end_time = gettime ();
    end_cpu = getcpu ();
    fprintf (stderr, "Done. %d orders
loaded/generated in %10.2f sec. (%10.2f
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, aware, 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 {
        status = OCISmtExecute(tpcsvc, curno,
errhp, (ub4) NEWOARR, (ub4) 0,
(CONST OCISnapshot*) 0,
(OCISnapshot*) 0,
(ub4) OCI_DEFAULT |
OCI_COMMIT_ON_SUCCESS);
        if (status != OCI_SUCCESS) {
            fprintf (stderr, "Aborted at w_id %d, d_id %d,
o_id %d\n", cwid, cdid, cid + 2100);
            OCIERROR(errhp, status);
            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 %10.2f sec. (%10.2f
cpu)\n\n",
nrows, end_time - begin_time, end_cpu
- begin_cpu);
}

/*-----+
| clean up and exit. |
+-----+
*/

if (olfp)
    fclose (olfp);
if (!lgen)
    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);
}

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:
            fprintf(stderr, "Module %s Line %d\n", fname,
lineno);
            fprintf(stderr, "Error -
OCI_SUCCESS_WITH_INFO\n");
            lstat = OCIErrorGet (errhp, recno++, (text *)
NULL, &errcode, errbuf,
(ub4) sizeof(errbuf),
OCI_HTYPE_ERROR);
            fprintf(stderr, "Error - %s\n", errbuf);
            break;
        case OCI_NEED_DATA:
            fprintf(stderr, "Module %s Line %d\n", fname,
lineno);
            fprintf(stderr, "Error - OCI_NEED_DATA\n");
    }
}

```



```

return (IRRECERR);
case OCI_NO_DATA:
    fprintf(stderr,"Module %s Line %d\n", fname,
lineno);
    fprintf(stderr,"Error - OCI_NO_DATA\n");
    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);
    if (errcode == SNAPSHOT_TOO_OLD) return
(errcode);
    while (lstat != OCI_NO_DATA)
    {
        fprintf(stderr,"Module %s Line %d\n", fname,
lineno);
        fprintf(stderr,"Error - %s\n", errbuf);
        lstat = OCIErrorGet (errhp, recno++, (text *)
NULL, &errcode, errbuf,
        (ub4) sizeof(errbuf),
OCI_HTYPE_ERROR);
    }
    return (errcode);
case OCI_INVALID_HANDLE:
    fprintf(stderr,"Module %s Line %d\n", fname,
lineno);
    fprintf(stderr,"Error -
OCI_INVALID_HANDLE\n");
    exit(-1);
case OCI_STILL_EXECUTING:
    fprintf(stderr,"Module %s Line %d\n", fname,
lineno);
    fprintf(stderr,"Error -
OCI_STILL_EXECUTE\n");
    return (IRRECERR);
case OCI_CONTINUE:
    fprintf(stderr,"Module %s Line %d\n", fname,
lineno);
    fprintf(stderr,"Error - OCI_CONTINUE\n");
    return (IRRECERR);
default:
    fprintf(stderr,"Module %s Line %d\n", fname,
lineno);
    fprintf(stderr,"Status - %s\n", status);
    return (IRRECERR);
}
return (RECOVERR);
}

```

```

.....
App_E/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 /*+ leading(s) use_nl(i) */
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;

```

```

.....
App_E/lib/Makefile.lib
.....

#=====
# Copyright (c) 1996 Oracle Corp,
Redwood Shores, CA |
# OPEN SYSTEMS PERFORMANCE
GROUP |
# All Rights
Reserved |
#=====
# FILENAME
# Makefile
# DESCRIPTION
# Makefile for lib for batch driver, load
program and tx testing.
#=====
#
# Programs:
#
# dpblibunix.o
#

```

```

all: compile dpblibunix.o

```

```

#include
$(ORACLE_HOME)/bench/buildtools/prefix.mk
I_SYM=-l
#include
$(ORACLE_HOME)/rdbsms/lib/env_rdbms.mk
REMOVE=rm
#CC=/opt/SunProd/SUNWwspro6.1/bin/..WS6U1/
bin/cc
CC=/usr/bin/gcc

TARGETS=compile cleanup

TPCBIN=.
INCLUDE=$(I_SYM).
$(I_SYM)$(ORACLE_HOME)/rdbsms/demo \
$(I_SYM)$(ORACLE_HOME)/rdbsms/public \
$(I_SYM)$(ORACLE_HOME)/rdbsms/include \
$(I_SYM)$(ORACLE_HOME)/plssql/public \
$(I_SYM)$(ORACLE_HOME)/network/public
ITUX=$(I_SYM)$(ROOTDIR)/include

```

```

MEMBS=
OBSJ=gettime.o dpbproc.o dpbwait.o dpbpchk.o
dpbltimef.o

```

```

CFLAGS=

```

```

files:

```

```

compile: $(OBSJ)
@-$(DOTARGS)

```

```

cleanup:
$(REMOVE) $(OBSJ) dpblibunix.o

```

```

dpbltimef.o: dpbltimef.c
$(CC) $(CFLAGS) -DORA_PC $(INCLUDE) -
c dpbltimef.c

```

```

dpbproc.o: dpbproc.c
$(CC) $(CFLAGS) -DORA_AUX $(INCLUDE)
-c dpbproc.c

```

```

dpbwait.o: dpbwait.c
$(CC) $(CFLAGS) -DORA_AUX $(INCLUDE)
-c dpbwait.c

```

```

dpbpchk.o: dpbpchk.c
$(CC) $(CFLAGS) -DORA_AUX $(INCLUDE)
-c dpbpchk.c

```

```

gettime.o: gettime.c
$(CC) $(CFLAGS) $(INCLUDE) -c gettime.c

```

```

trigger.o: trigger.c

```

```

dpblibunix.o: $(OBSJ)
$(LD) -r -o $@ $(OBSJ)

```

```

c_trans_tux.o: $(CTRANTUX_OBSJ)
$(LD) -r -o $@ $(CTRANTUX_OBSJ)

```

```

.....
App_E/lib/dpbcpre.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 dpbpcpu() 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 __dpbcpre__
#define __dpbcpre__

```

```

#include <stdio.h>
#include "dpbpcntl.h"

#ifdef __STDC__
C
*/
ANSI
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 */
clock_t dpbcpu(void); /* get CPU
time */
void dpbwait(clock_t); /* wait
routine in millisecc */
void dpbxtpgm(void); /* pgm
exit routine */
int dpboradt(char *); /* sys date
time in ora form*/
clock_t dpbetime(void); /* elapsed
time */
#else
/* K&R
C
*/
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 */
clock_t dpbcpu(); /* get cpu
time */
void dpbwait(); /* wait routine
in millisecc */
void dpbxtpgm(); /* pgm exit
routine */
int dpboradt(); /* sys date
time in ora form*/
clock_t dpbetime(); /* elapsed
time */
#endif /* __STDC__ */

#endif /* __dpbcore__ */

.....
App_E/lib/dpbcpu.c
.....

/* Copyright (c) Oracle Corporation 1993. All
Rights Reserved. */

/*
NAME DPBTIME.C

DESCRIPTION

```

```

Get time in seconds.

NOTES
Desktop Performance Group

MODIFIED (MM/DD/YY)
bmoriart 05/10/95 - V4.7 Convert from
double to clock_t
MBHULLAR 02/06/95 - V4.5
*/

#ifdef ORA_NT
#include <windows.h>
#include <time.h>

clock_t dpbcpu(void)
{
clock_t begin_cpu;

begin_cpu = clock();
return(begin_cpu);
}
#endif /* ORA_NT */

.....
App_E/lib/dpbtime.c
.....

/* Copyright (c) Oracle Corporation 1995. All
Rights Reserved. */
/*
NAME DPBETIME.C

DESCRIPTION
Get elapsed time in 10ths of milliseconds as a
clock_t.

NOTES
Desktop Performance Group

MODIFIED (MM/DD/YY)
B Moriarty 06/02/95 - V4.8 Initial Version
*/

#ifdef ORA_OS2
#endif /* ORA_OS2 */

#ifdef ORA_NT
#include <windows.h>
#include <sys/types.h>
#include <time.h>
#include <stdio.h>

BOOL First = TRUE;
LARGE_INTEGER ICount; /* Initial Time */
LARGE_INTEGER Tptms; /* Ticks per tenth
of millisecond */
#endif /* ORA_NT */

#ifdef __STDC__
clock_t dpbetime(void)
#else
clock_t dpbetime()
#endif /* __STDC__ */
{
#ifdef ORA_NT

```

```

LARGE_INTEGER PFreq; /* Ticks per
Second */
LARGE_INTEGER PCount; /* Ticks Since
1970 */
clock_t etime; /* Elapsed time in tenths of
milliseconds */

if (First) {
if (!QueryPerformanceFrequency(&PFreq))
return((clock_t)-1);
if (!QueryPerformanceCounter(&ICount))
return((clock_t)-1);
Tptms.QuadPart = PFreq.QuadPart / 10000;
First = FALSE;
return((clock_t)0);
}
if (!QueryPerformanceCounter(&PCount))
return((clock_t)-1);
etime = (clock_t) ((PCount.QuadPart -
ICount.QuadPart) / Tptms.QuadPart);
return(etime);

#endif /* ORA_NT */
}

.....
App_E/lib/dpbfsync.c
.....

/* Copyright (c) Oracle Corporation 1993,
1992. All Rights Reserved. */

/*
NAME DPBFSYNC.C

DESCRIPTION

Flush o/s buffers to disk for a file.

Calling fclose() or fflush() is not
enough. These calls will only flush
the buffer in the FILE structure by making a
write() call to the o/s, and
the o/s will probably place these data in its own
disk buffers.
dpbfsync() will cause the o/s disk buffers for a
file to be written to disk.

This function should normally be called *after*
an fflush() is done, or you
will miss the data that is buffered in the FILE
structure.

NOTES
Desktop Performance Group

MODIFIED (MM/DD/YY)
C Kelly 02/24/94 - V4.4 initial version
*/

#include <stdio.h>

#ifdef ORA_OS2
int dpbfsync(FILE *fp)
{
return 0;
}
#endif /* ORA_OS2 */

```

```

#ifdef ORA_NT
#include <windows.h>

int dpbfsync(FILE *fp)
{
    if (FlushFileBuffers((HANDLE)(fp->_file)) ==
FALSE)
    {
        return 1;
    };
    return 0;
}
#endif /* ORA_NT */

#ifdef ORA_AUX
int dpbfsync(fp)
FILE *fp;
{
    if (fsync(fp->_file) == -1)
    {
        return 1;
    };
    return 0;
}
#endif /* ORA_AUX */

#ifdef ORA_NW
int dpbfsync(FILE *fp)
{
    return 0;
}
#endif /* ORA_NW */

#ifdef ORA_DOS
int dpbfsync(FILE *fp)
{
    return 0;
}
#endif /* ORA_DOS */

#ifdef ORA_MAC
#endif /* ORA_MAC */

.....
App_E/lib/dpbinpgm.c
.....

/* Copyright (c) Oracle Corporation 1994. All
Rights Reserved. */

/*
NAME    DPBINPGM.C

DESCRIPTION
    Routine that performs any o/s specific program
initialization.

NOTES
    Desktop Performance Group

MODIFIED   (MM/DD/YY)
    bmoriart 05/26/95 - V4.8 Created
*/

#ifdef ORA_NT
#include <windows.h>
#endif /* ORA_NT */

#ifdef __STDC__
void dpborad(char *oradt)
#else
void dpborad(oradt)

```

```

C Kelly 04/21/94 - V4.4 created to support
Netware NLMs

*/

#ifdef ORA_NW
#include <process.h>
#include <library.h>

extern int samtid;
extern int samtgid;

#else /* ORA_NW */
#endif /* ORA_NW */

#ifdef __STDC__
void dpbinpgm(void)
#else
void dpbinpgm()
#endif /* __STDC__ */
{
    # ifdef ORA_NW

        samtid = GetThreadID(); /* get this
program's thread id */
        samtgid = GetThreadGroupID(); /* get this
program's thread group id */

    # else /* ORA_NW */

        return; /* do nothing for everything else
*/

    # endif /* ORA_NW */
}

.....
App_E/lib/dpborad.c
.....

/* Copyright (c) Oracle Corporation 1993. All
Rights Reserved. */

/*
NAME    DPBORADT.C

DESCRIPTION
    Get System Date and Time and
Return in Oracle External SQLT_DAT (Date)
Format
    Returns 1-JAN-2000 00:00:00
when not implemented or when conversion
fails

NOTES
    Desktop Performance Group

MODIFIED   (MM/DD/YY)
    bmoriart 05/26/95 - V4.8 Created
*/

#ifdef ORA_NT
#include <windows.h>
#endif /* ORA_NT */

#ifdef __STDC__
void dpborad(char *oradt)
#else
void dpborad(oradt)

```

```

unsigned char *oradt;
#endif /* __STDC__ */
{
    char cnvrtOK=TRUE;

#ifdef ORA_NT
    SYSTEMTIME lpst;

    GetLocalTime(&lpst);
    *oradt = (unsigned char)(lpst.wYear / 100) +
100;
    if (*oradt < 119 || *oradt > 120)
cnvrtOK=FALSE;
    *(++oradt) = (unsigned char)(lpst.wYear %
100) + 100;
    if (*oradt < 100 || *oradt > 199)
cnvrtOK=FALSE;
    *(++oradt) = (unsigned char)(lpst.wMonth);
    if (*oradt < 1 || *oradt > 12) cnvrtOK=FALSE;
    *(++oradt) = (unsigned char)(lpst.wDay);
    if (*oradt < 1 || *oradt > 31) cnvrtOK=FALSE;
    *(++oradt) = (unsigned char)(lpst.wHour) + 1;
    if (*oradt < 1 || *oradt > 24) cnvrtOK=FALSE;
    *(++oradt) = (unsigned char)(lpst.wMinute) + 1;
    if (*oradt < 1 || *oradt > 60) cnvrtOK=FALSE;
    *(++oradt) = (unsigned char)(lpst.wSecond) +
1;
    if (*oradt < 1 || *oradt > 60) cnvrtOK=FALSE;
#else /* ORA_NT */
    cnvrtOK = FALSE;
#endif /* ORA_NT */

    if(!cnvrtOK) { /* Use 1-JAN-2000 00:00:00 */
        *oradt++ = 120;
        *oradt++ = 100;
        *oradt++ = 1;
        *oradt++ = 1;
        *oradt++ = 1;
        *oradt++ = 1;
        *oradt++ = 1;
        *oradt++ = 1;
        *oradt++ = 1;
        *oradt++ = 1;
    }
    return; /* do nothing for everything
else */
}

.....
App_E/lib/dpbcchk.c
.....

/* Copyright (c) Oracle Corporation 1993,
1992. All Rights Reserved. */

/*
NAME    DPBCCHK.C

DESCRIPTION
    Check New Process

NOTES
    Desktop Performance Group

MODIFIED   (MM/DD/YY)
    W Brumiller 02/08/93 - Correct error handling
for NT
    R Keller 01/08/92 - Initial version

*/

#include "dpbcnttl.h"

```

```

#ifdef ORA_OS2                /* IBM
OS/2 2.0                */
#define
INCL_DOSPROCESS          /*
*/
#include
<os2.h>                /*          */

unsigned long dpbpchk(pcntl *info)
{
    ULONG pid;
    APIRET rc;

    rc = DosWaitChild(DCWA_PROCESS,
        DCWW_WAIT,
        &info->rcodes,
        &pid,
        0);

    return(info->rcodes.codeResult);
};
#endif /* ORA_OS2 */

#ifdef ORA_NT
#include <windows.h>

int dpbpchk(pcntl *info)
{
    DWORD rc;

    if (WaitForSingleObject(info->proc_info.hProcess, INFINITE) ==
    0xFFFFFFFF)
    {
        return -1;
    };

    if (GetExitCodeProcess(info->proc_info.hProcess, &rc) == FALSE)
    {
        return -1;
    };

    (void)CloseHandle(info->proc_info.hProcess);
    (void)CloseHandle(info->proc_info.hThread);

    return((int)rc);
}
#endif /* ORA_NT */

#ifdef ORA_AUX
#include <errno.h>

int dpbpchk(info)
pcntl *info;
{
    extern int errno;
    int byte_mask;
    int status;
    int high_byte;
    int child;
    int i;

    byte_mask = 255; /* low order 8 bits are 1,
bits 8..31 are 0 */

do
{
    child = wait(&status);
    if (errno != ECHILD)
    {
        high_byte = ((status & (byte_mask << 8)) >>
8);
    };
    } while (errno != ECHILD);

    return high_byte;
}
#endif /* ORA_AUX */

.....
App_E/lib/dpbpcntl.h
.....

/* Copyright (c) Oracle Corporation 1993,
1992. All Rights Reserved. */

/*
NAME   DPBPCNTL.H

DESCRIPTION
    OSD structures for process control

NOTES
    Desktop Performance Group

MODIFIED (MM/DD/YY)
    R Keller 02/03/93 - initial version

*/

#ifdef __dpbpcntl__
#define __dpbpcntl__

#ifdef ORA_OS2                /* IBM
OS/2 2.x                */
#define INCL_DOSPROCESS
#include <os2.h>
typedef struct _pcntl
{
    RESULTCODES rcodes;
} pcntl;
#endif /* ORA_OS2 */          /* IBM
OS/2 2.x                */

#ifdef ORA_NT                /* Microsoft
Windows NT                */
#include
<windows.h>                /*          *
/
typedef struct _pcntl
{
    PROCESS_INFORMATION proc_info;
} pcntl;
#endif /* ORA_NT */          /*
Microsoft Windows NT    */

#ifdef ORA_AUX                /* Apple
A/UX                    */
typedef struct _pcntl
{
    int dummy;
} pcntl;

#endif /* ORA_AUX */          /* Apple
A/UX                    */

#ifdef ORA_NW                /* Novell
Netware                    */
typedef struct _pcntl
{
    int dummy;
} pcntl;
#endif /* ORA_NW */          /* Novell
Netware                    */

.....
__dpbpcntl__ */

.....
App_E/lib/dpbproc.c
.....

/* Copyright (c) Oracle Corporation 1993,
1992. All Rights Reserved. */

/*
NAME   DPBPROC.C

DESCRIPTION
    Create New Process

NOTES
    Desktop Performance Group

MODIFIED (MM/DD/YY)
    W Brumiller 02/08/93 - Add flags for
minimized window under NT
    R Keller 01/08/92 - Initial version

*/

#include "dpbpcntl.h"

#ifdef ORA_OS2                /* IBM
OS/2 2.0                */
#define INCL_DOSPROCESS
#include
<os2.h>                /*          */
#include
<stdlib.h>                /*          */
#include
<string.h>                /*          */

unsigned long dpbproc(char *i_argv[], pcntl *info)
{
    char *args;
    char *args2;
    char load_error[100];
    char pgm[44];
    APIRET rc;
    int i;

    args2 = args = (char *)malloc(128);

    strcpy(args, i_argv[0]);
    strcpy(pgm, i_argv[0]);
    strcat(pgm, ".exe");

    args2 += strlen(args) + 1;

```

```

if (i_argv[1] != NULL)
{
strcpy(args2, i_argv[1]);
for (i = 2; i_argv[i] != NULL; i++)
{
strcat(args2, " ");
strcat(args2, i_argv[i]);
};
}
else
{
*args2 = '\0';
};

rc = DosExecPgm(load_error, /*
spawn process */
sizeof(load_error),
EXEC_ASYNCRESULT,
args,
0,
&info->rcodes,
pgm);

free(args);

return rc;
}
#endif /* ORA_OS2 */

```

```

#ifdef ORA_NT /* Microsoft
Windows NT */
#include <windows.h>
#include
<stdlib.h> /* */
#include
<string.h> /* */

```

```

int dpbproc(char *i_argv[], pcntl *info)
{
BOOL rc;
int i;
char *args;
STARTUPINFO start_info;

args = (char *)malloc(128);

memset(&start_info, 0x0,
sizeof(STARTUPINFO));
start_info.cb = sizeof(STARTUPINFO);
start_info.lpTitle = i_argv[0];
start_info.dwFlags =
STARTF_USESHOWWINDOW;
start_info.wShowWindow =
SW_SHOWMINNOACTIVE;

```

```

strcpy(args, i_argv[0]); /* get
first str */

for (i = 1; i_argv[i] != NULL; i++)
{
strcat(args, " ");
strcat(args, i_argv[i]);
};

```

```

if ((rc = CreateProcess(NULL, /*
image name
args, // command line
NULL, // process
security attr
NULL, // thread

```

```

security attr
TRUE, // inherit
handles
CREATE_NEW_CONSOLE, //
creation flags
NULL, // environment
blocks
NULL, // current
directory
&start_info,
&info->proc_info) == FALSE)
{
return rc;
};

```

```

return 0;
};
#endif /* ORA_NT */

```

```

#ifdef ORA_AUX
#include <stdio.h>
#include <stdlib.h>
#include <sys/types.h>

```

```

int dpbproc(arg_list, info)
char *arg_list[];
pcntl *info;
{
char *path = (char *)malloc(strlen(arg_list[0]) +
3);
pid_t child;

```

```

sprintf(path, "%s", arg_list[0]);

if ((child = fork()) == (pid_t)-1)
{
free(path);
return -1;
}
else if (child == (pid_t)0)
{
return execv(path, arg_list);
}
else
{
free(path);
return 0;
};
}
#endif /* ORA_AUX */

```

```

.....
App_E/lib/dpbprty.c
.....

```

```

/* Copyright (c) Oracle Corporation 1993. All
Rights Reserved. */

```

```

/*
NAME DPBPRTY.C

```

```

DESCRIPTION
Set O/S Priority.

```

```

NOTES
Desktop Performance Group

```

```

MODIFIED (MM/DD/YY)

```

```

MBHULLAR 03/25/94 - Change prty_str(1)
to case statement
B Moriarty 11/11/93 - Add Get Priority
R Keller 10/18/93 - Redesign
R Keller 10/16/93 - Initial version
*/

```

```

#ifdef ORA_OS2
#include <string.h>
#include <sys/types.h>
#endif /* ORA_OS2 */

```

```

#ifdef ORA_NW
#endif /* ORA_NW */

```

```

#ifdef ORA_NT
# include <windows.h>
# include <string.h>
# define REALCLASS 'R'
# define HIGHCLASS 'H'
# define NORMALCLASS 'N'
# define IDLECLASS 'I'
#endif /* ORA_NT */

```

```

#ifdef ORA_AUX
#endif /* ORA_AUX */

```

```

#ifdef __STDC__
int dpbprty(char *prty_str)
#else
int dpbprty(prty_str)
char *prty_str;
#endif
{
#ifdef ORA_OS2
return 0;
#endif /* ORA_OS2 */

```

```

#ifdef ORA_AUX
return 0;
#endif /* ORA_AUX */

```

```

#ifdef ORA_NW
return 0;
#endif /* ORA_NW */

```

```

#ifdef ORA_NT

```

```

HANDLE this_process, this_thread;

```

```

DWORD class;

```

```

int prios;

```

```

if ( ( strlen(prty_str) > 2) || prty_str[0] == '0')
{
return(0); /* return if invalid length
or 0 */
};

```

```

this_process = GetCurrentProcess();

```

```

switch (prty_str[0])
{
case IDLECLASS:
case 'I':
class = IDLE_PRIORITY_CLASS;
break;

```

```

case NORMALCLASS:
case 'n':
    class = NORMAL_PRIORITY_CLASS;
    break;

case HIGHCLASS:
case 'h':
    class = HIGH_PRIORITY_CLASS;
    break;

case REALCLASS:
case 'r':
    class = REALTIME_PRIORITY_CLASS;
    break;
};

if (!SetPriorityClass(this_process, class))
{
    return(1);
};

this_thread = GetCurrentThread();
switch(prty_str[1])
{
    case '1':
        prios = THREAD_PRIORITY_IDLE;
        break;

    case '2':
        prios = THREAD_PRIORITY_LOWEST;
        break;

    case '3':
        prios =
        THREAD_PRIORITY_BELOW_NORMAL;
        break;

    case '4':
        prios = THREAD_PRIORITY_NORMAL;
        break;

    case '5':
        prios =
        THREAD_PRIORITY_ABOVE_NORMAL;
        break;

    case '6':
        prios = THREAD_PRIORITY_HIGHEST;
        break;

    case '7':
        prios = THREAD_PRIORITY_TIME_CRITICAL;
        break;

    default:
        break;
} /* End of switch statement */

if (!SetThreadPriority(this_thread, prios))
{
    return(2);
}

return 0;

# endif /* ORA_NT */

}

#ifdef __STDC__
int dpbgetprty(char *os_pri, char *prty_str, int

```

```

os_pri_len)
#else
int dpbgetprty(os_pri, prty_str, os_pri_len)
char *os_pri;
char *prty_str;
int os_pri_len;
#endif /* __STDC__ */
{
#ifdef ORA_OS2
    strncpy(os_pri,prty_str,(size_t)os_pri_len);
    return 0;
#endif /* ORA_OS2 */

#ifdef ORA_AUX
    strncpy(os_pri,prty_str,os_pri_len);
    return 0;
#endif /* ORA_AUX */

#ifdef ORA_NW
    strncpy(os_pri, prty_str, os_pri_len);
    return 0;
#endif /* ORA_NW */

#ifdef ORA_NT
    HANDLE this_process, this_thread;
    DWORD pclass;
    int tpri;

    this_process = GetCurrentProcess();
    pclass = GetPriorityClass(this_process);

    switch (pclass)
    {
        case IDLE_PRIORITY_CLASS:
            strcpy(os_pri,"I");
            break;

        case NORMAL_PRIORITY_CLASS:
            strcpy(os_pri,"N");
            break;

        case HIGH_PRIORITY_CLASS:
            strcpy(os_pri,"H");
            break;

        case REALTIME_PRIORITY_CLASS:
            strcpy(os_pri,"R");
            break;

        default:
            strcpy(os_pri,"?");
            break;
    };

    this_thread=GetCurrentThread();
    tpri=GetThreadPriority(this_thread);
    switch (tpri)
    {
        case THREAD_PRIORITY_IDLE:
            strcat(os_pri,"1");
            break;

        case THREAD_PRIORITY_LOWEST:
            strcat(os_pri,"2");
            break;

        case THREAD_PRIORITY_BELOW_NORMAL:
            strcat(os_pri,"3");
            break;

        case THREAD_PRIORITY_NORMAL:
            strcat(os_pri,"4");
            break;

```

```

case THREAD_PRIORITY_ABOVE_NORMAL:
    strcat(os_pri,"5");
    break;

case THREAD_PRIORITY_HIGHEST:
    strcat(os_pri,"6");
    break;

case THREAD_PRIORITY_TIME_CRITICAL:
    strcat(os_pri,"7");
    break;

default:
    strcat(os_pri,"?");
    break;
};

return 0;

# endif /* ORA_NT */
}

.....
App_E/lib/dpbtimf.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__
clock_t dpbtimf(void)
# else
clock_t dpbtimf()
# endif /* __STDC__ */
{
    struct timeb buf;

    ftime(&buf);

```

```

return((clock_t) (buf.time));
}
#endif /* ORA_PC */

#ifdef ORA_AUX
#include <sys/time.h>
double dpbtimedf()
{
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 dpbtimedf()
{
return (double)time(NULL); /* there is no
function with greater precision */
}
#endif /* ORA_NW */

#ifdef ORA_MAC
#include <types.h>
#include <OSUtils.h>

double dpbtimedf()
{
unsigned long secs;
GetDateTime(&secs);
return((double) secs);
}
#endif /* ORA_MAC */

.....
App_E/lib/dpbwait.c
.....

/* Copyright (c) Oracle Corporation 1993. All
Rights Reserved. */

/*
NAME DPBWAIT.C

DESCRIPTION
Wait for n milliseconds.

NOTES
Desktop Performance Group

MODIFIED (MM/DD/YY)
R Keller 03/02/92 - V3.0
*/

#ifdef ORA_OS2
#define INCL_DOS
#include <os2.h>

```

```

#include <time.h>

void dpbwait(clock_t i)
{
DosSleep(i);
}
#endif /* ORA_OS2 */

#ifdef ORA_NW
#include <process.h>
void dpbwait(long i)
{
delay((unsigned)i);
};
#endif /* ORA_NW */

#ifdef ORA_AUX
void dpbwait(wait_time)
long wait_time;
{
unsigned secs = (unsigned)(wait_time / 1000);

while (secs)
{
secs = sleep(secs);
};
}
#endif /* ORA_AUX */

#ifdef ORA_NT
#include <windows.h>

void dpbwait(long i)
{
Sleep(i);
}
#endif /* ORA_NT */

#ifdef ORA_DOS
#include <time.h>

void dpbwait(long i)
{
long current_time;
long target_time;

current_time = time(NULL);
target_time = current_time + i/1000;

while (current_time < target_time)
{
current_time = time(NULL);
};
}
#endif /* ORA_DOS */

.....
App_E/lib/dpbxtpgm.c
.....

/* Copyright (c) Oracle Corporation 1994. All
Rights Reserved. */

/*
NAME DPBXTPGM.C

```

```

DESCRIPTION
Routine that performs any o/s specific program
exit operations.

NOTES
Desktop Performance Group

MODIFIED (MM/DD/YY)
C Kelly 04/21/94 - V4.4 created to support
Netware NLMs

*/

#ifdef ORA_NW
#include <process.h>
#include <library.h>

extern int samtid;
extern int samtgid;

#else /* ORA_NW */
#endif /* ORA_NW */

#ifdef __STDC__
void dpbxtpgm(void)
#else
void dpbxtpgm()
#endif /* __STDC__ */
{
#ifdef ORA_NW

/*
** Cleanup code for NetWare.
** This routine will cleanup any Oracle
connection should the module
** be unexpectedly unloaded.
*/

int oldtgid;

oldtgid = SetThreadGroupID(samtgid); /*
switch to this NLM's thread group */
OraClientExit(samtid); /* cleanup
Oracle connection */
SetThreadGroupID(oldtgid); /* reset the
thread group */

#else /* ORA_NW */

return; /* do nothing for everything else
*/

#endif /* ORA_NW */
}

.....
App_E/lib/gettime.c
.....

#ifdef RCSID
static char *RCSid =
"$Header: gettime.c 7030100.1 96/05/21
15:31:36 plai Generic<base> $ Copyr (c) 1993
Oracle";
#endif /* RCSID */

/*=====
=====+

```

```

| Copyright (c) 1996 Oracle Corp,
Redwood Shores, CA
| OPEN SYSTEMS
PERFORMANCE GROUP
| All Rights
Reserved
|
+-----+
+-----+
|
| FILENAME
| gettimeofday
|
| ROUTINES
| gettimeofday
| getcpu
| DESCRIPTION
| get wall clock time.
| get cpu time.
| NOTES
| Both routines return time in seconds as a
double.
+-----+
+-----+
/*
** Options:
** TIME_W_TIMES: implement gettimeofday()
with times().
** TIME_W_GETTIME: implement gettimeofday()
with gettimeofday().
** CPU_W_TIMES: implement
getcpu() with times().
** CPU_W_GETRU: implement
getcpu() with getrusage().
** GETRU_STATS: collect getrusage
statistics
** GET_P_STATS: collect
get_process_stats statistics
*/

#if defined(sequent) || defined(SEQ_P SX)
# define GET_P_STATS
#endif /* sequent */

#if defined(aix) || defined(AIXRIOS)
# define TIME_W_GETTIME
# define CPU_W_TIMES
# define GETRU_STATS
#endif /* AIXRIOS */

#if defined(a_osf) || defined(A_OSF)
# define TIME_W_GETTIME
# define CPU_W_GETRU
# define GETRU_STATS
#endif /* AIXRIOS */

#if !defined(TIME_W_GETTIME)
&& !defined(TIME_W_TIMES)
# define TIME_W_TIMES
#endif

#if !defined(CPU_W_GETRU)
&& !defined(CPU_W_TIMES)
# define CPU_W_TIMES
#endif

#ifdef GET_P_STATS
# ifdef GETRU_STATS
# undef GETRU_STATS
# endif
#endif

#if defined(TIME_W_GETTIME) ||
defined(CPU_W_GETRU) ||
defined(GETRU_STATS)
# include <sys/time.h>

```

```

#endif /* TIME_W_GETTIME || CPU_W_GETRU
|| GETRU_STATS */

#if defined(CPU_W_GETRU) ||
defined(GETRU_STATS)
# include <sys/resource.h>
#endif /* CPU_W_GETRU || GETRU_STATS */

#if defined(TIME_W_TIMES) || defined
(CPU_W_TIMES)
# include <sys/types.h>
# include <sys/times.h>
# include <sys/param.h> /* most systems define
HZ here */
#if !defined(_SC_CLK_TCK)
# include <unistd.h>
#endif
#endif /* TIME_W_TIMES or CPU_W_TIMES */

#ifdef GET_P_STATS
# include <sys/types.h>
# include <sys/procstats.h>
#endif /* GET_P_STATS */

# include <stdio.h>

#ifdef GETRU_STATS
struct rusage selfru;
struct rusage kidsru;
#endif /* GETRU_STATS */

#ifdef GET_P_STATS
struct process_stats selfru;
struct process_stats kidsru;
#endif /* GET_P_STATS */

void getwait(clock_t secs)
{
printf("sleep = %lu\n", (secs/1000) / HZ);
printf("hz = %lu\n", HZ);
sleep((secs/1000) / HZ);
}

clock_t getetime()
{
struct tms buf;

return ((times (&buf) / HZ)*10000);
}

double gettime ()
{
#ifdef TIME_W_GETTIME
struct timeval tv;

(void) gettimeofday (&tv, (struct timezone *) 0);
return ((double) tv.tv_sec + (1.0e-6 * (double)
tv.tv_usec));
#endif /* TIME_W_GETTIME */

#ifdef TIME_W_TIMES
struct tms buf;

return ((double) times (&buf) / HZ);
#endif /* TIME_W_TIMES */
}

```

```

double getcpu ()
{
#ifdef CPU_W_TIMES
struct tms buf;

(void) times (&buf);
return (((double) buf.tms_utime + (double)
buf.tms_stime) / HZ);
#endif /* CPU_W_TIMES */

#ifdef CPU_W_GETRU
struct rusage ru;
double usecs;

(void) getrusage (0, &ru);
usecs = 1.0e-6 * (double) (ru.ru_utime.tv_usec
+ ru.ru_stime.tv_usec);
return ((double) (ru.ru_utime.tv_sec +
ru.ru_stime.tv_sec) + usecs);
#endif /* CPU_W_GETRU */
}

getru (fp, kids, config, runname, proc_no)

FILE *fp;
int kids;
char *config;
char *runname;
int proc_no;

{
#ifdef GETRU_STATS
struct rusage ru;

fprintf (fp, "%-10.10s %-10.10s %10d %10d ",
config, runname, proc_no, kids);
getrusage (kids ? RUSAGE_CHILDREN :
RUSAGE_SELF, &ru);
print_ru (fp, &ru);
fprintf (fp, "\n");
#endif /* GETRU_STATS */

#ifdef GET_P_STATS
timeval_t tv;
struct process_stats ru;

fprintf (fp, "%-10.10s %-10.10s %10d %10d ",
config, runname, proc_no, kids);
if (kids)
get_process_stats (&tv, PS_SELF, (struct
process_stats *) 0, &ru);
else
get_process_stats (&tv, PS_SELF, &ru,
(struct process_stats *) 0);
print_ru (fp, &ru);
fprintf (fp, "\n");
#endif /* GET_P_STATS */
}

getru1 (kids)

int kids;

{
#ifdef GETRU_STATS

```



```

if (kids) {
    memset (&kidsru, 0, sizeof (kidsru));
    getrusage (RUSAGE_CHILDREN, &kidsru);
}
else {
    memset (&selfru, 0, sizeof (selfru));
    getrusage (RUSAGE_SELF, &selfru);
}
#endif /* GETRU_STATS */

#ifdef GET_P_STATS
timeval_t tv;

if (kids) {
    memset (&kidsru, 0, sizeof (kidsru));
    get_process_stats (&tv, PS_SELF, (struct
process_stats *) 0, &kidsru);
}
else {
    memset (&selfru, 0, sizeof (selfru));
    get_process_stats (&tv, PS_SELF, &selfru,
(struct process_stats *) 0);
}
#endif /* GET_P_STATS */

}

getru2 (fp, kids, config, runname, proc_no)

FILE *fp;
int kids;
char *config;
char *runname;
int proc_no;

{

#ifdef GETRU_STATS
    struct rusage ru;

    fprintf (fp, "%-10.10s %-10.10s %10d %10d ",
config, runname, proc_no, kids);
    getrusage (kids ? RUSAGE_CHILDREN :
RUSAGE_SELF, &ru);
    if (kids)
        diffru (&ru, &kidsru);
    else
        diffru (&ru, &selfru);
    print_ru (fp, &ru);
    fprintf (fp, "\n");
#endif /* GETRU_STATS */

#ifdef GET_P_STATS
    timeval_t tv;
    struct process_stats ru;

    fprintf (fp, "%-10.10s %-10.10s %10d %10d ",
config, runname, proc_no, kids);
    if (kids)
        get_process_stats (&tv, PS_SELF, (struct
process_stats *) 0, &ru);
    else
        get_process_stats (&tv, PS_SELF, &ru,
(struct process_stats *) 0);
    if (kids)
        diffru (&ru, &kidsru);
    else
        diffru (&ru, &selfru);
    print_ru (fp, &ru);
    fprintf (fp, "\n");
#endif /* GET_P_STATS */

}

```

```

#ifndef GETRU_STATS

print_ru (fp, ru)

FILE *fp;
struct rusage *ru;

{

    fprintf (fp, "%10ld ", ru->ru_utime.tv_sec * 1000
+
            (ru->ru_utime.tv_usec/1000));
    fprintf (fp, "%10ld ", ru->ru_stime.tv_sec * 1000
+
            (ru->ru_stime.tv_usec/1000));
    fprintf (fp, "%10ld ", ru->ru_maxrss);
    fprintf (fp, "%10ld ", ru->ru_majflt);
    fprintf (fp, "%10ld ", ru->ru_minflt);
    fprintf (fp, "%10ld ", 0);
    fprintf (fp, "%10ld ", 0);
    fprintf (fp, "%10ld ", 0);
    fprintf (fp, "%10ld ", ru->ru_nswap);
    fprintf (fp, "%10ld ", 0);
    fprintf (fp, "%10ld ", ru->ru_nvcsw);
    fprintf (fp, "%10ld ", ru->ru_nivcsw);
    fprintf (fp, "%10ld ", ru->ru_nsignals);
    fprintf (fp, "%10ld ", 0);
    fprintf (fp, "%10ld ", 0);
    fprintf (fp, "%10ld ", ru->ru_inblock);
    fprintf (fp, "%10ld ", ru->ru_oublock);
    fprintf (fp, "%10ld ", 0);
    fprintf (fp, "%10ld ", 0);

}

diffru (ru2, ru)

struct rusage *ru2;
struct rusage *ru;

{

    ru2->ru_utime.tv_sec -= ru->ru_utime.tv_sec;
    ru2->ru_utime.tv_usec -= ru-
>ps_utime.tv_usec;
    ru2->ps_stime.tv_sec -= ru->ps_stime.tv_sec;
    ru2->ps_stime.tv_usec -= ru-
>ps_stime.tv_usec;
    ru2->ps_maxrss -= ru->ps_maxrss;
    ru2->ps_pagein -= ru->ps_pagein;
    ru2->ps_reclaim -= ru->ps_reclaim;
    ru2->ps_zerofill -= ru->ps_zerofill;
    ru2->ps_pffincr -= ru->ps_pffincr;
    ru2->ps_pffdecr -= ru->ps_pffdecr;
    ru2->ps_swap -= ru->ps_swap;
    ru2->ps_syscall -= ru->ps_syscall;
    ru2->ps_volcsw -= ru->ps_volcsw;
    ru2->ps_involcsw -= ru->ps_involcsw;
    ru2->ps_signal -= ru->ps_signal;
    ru2->ps_lread -= ru->ps_lread;
    ru2->ps_lwrite -= ru->ps_lwrite;
    ru2->ps_bread -= ru->ps_bread;
    ru2->ps_bwrite -= ru->ps_bwrite;
    ru2->ps_phread -= ru->ps_phread;
    ru2->ps_phwrite -= ru->ps_phwrite;

}

#endif /* GETRU_STATS */

#ifdef GET_P_STATS

```

```

print_ru (fp, ps)

FILE *fp;
struct process_stats *ps;

{

    fprintf (fp, "%lu ", ps->ps_utime.tv_sec * 1000
+
            (ps->ps_utime.tv_usec/1000));
    fprintf (fp, "%lu ", ps->ps_stime.tv_sec * 1000
+
            (ps->ps_stime.tv_usec/1000));
    fprintf (fp, "%lu ", ps->ps_maxrss);
    fprintf (fp, "%lu ", ps->ps_pagein);
    fprintf (fp, "%lu ", ps->ps_reclaim);
    fprintf (fp, "%lu ", ps->ps_zerofill);
    fprintf (fp, "%lu ", ps->ps_pffincr);
    fprintf (fp, "%lu ", ps->ps_pffdecr);
    fprintf (fp, "%lu ", ps->ps_swap);
    fprintf (fp, "%lu ", ps->ps_syscall);
    fprintf (fp, "%lu ", ps->ps_volcsw);
    fprintf (fp, "%lu ", ps->ps_involcsw);
    fprintf (fp, "%lu ", ps->ps_signal);
    fprintf (fp, "%lu ", ps->ps_lread);
    fprintf (fp, "%lu ", ps->ps_lwrite);
    fprintf (fp, "%lu ", ps->ps_bread);
    fprintf (fp, "%lu ", ps->ps_bwrite);
    fprintf (fp, "%lu ", ps->ps_phread);
    fprintf (fp, "%lu ", ps->ps_phwrite);

}

diffru (ru2, ru)

struct process_stats *ru2;
struct process_stats *ru;

{

    ru2->ps_utime.tv_sec -= ru->ps_utime.tv_sec;
    ru2->ps_utime.tv_usec -= ru-
>ps_utime.tv_usec;
    ru2->ps_stime.tv_sec -= ru->ps_stime.tv_sec;
    ru2->ps_stime.tv_usec -= ru-
>ps_stime.tv_usec;
    ru2->ps_maxrss -= ru->ps_maxrss;
    ru2->ps_pagein -= ru->ps_pagein;
    ru2->ps_reclaim -= ru->ps_reclaim;
    ru2->ps_zerofill -= ru->ps_zerofill;
    ru2->ps_pffincr -= ru->ps_pffincr;
    ru2->ps_pffdecr -= ru->ps_pffdecr;
    ru2->ps_swap -= ru->ps_swap;
    ru2->ps_syscall -= ru->ps_syscall;
    ru2->ps_volcsw -= ru->ps_volcsw;
    ru2->ps_involcsw -= ru->ps_involcsw;
    ru2->ps_signal -= ru->ps_signal;
    ru2->ps_lread -= ru->ps_lread;
    ru2->ps_lwrite -= ru->ps_lwrite;
    ru2->ps_bread -= ru->ps_bread;
    ru2->ps_bwrite -= ru->ps_bwrite;
    ru2->ps_phread -= ru->ps_phread;
    ru2->ps_phwrite -= ru->ps_phwrite;

}

#endif /* GET_P_STATS */

.....
App_E/lib/tstetime.c
.....

```

```
#include <windows.h>
#include <systypes.h>
#include <time.h>

clock_t dpbetime();

main()
{
    clock_t begin, middle, end;

    begin = dpbetime();
    Sleep(2000);
    middle = dpbetime();
    Sleep(2000);
    end = dpbetime();
    printf(" begin = %lu\n middle = %lu\n end
= %lu\n",begin,middle,end);
}
```

Appendix F: 60 Day Space Calculation

SEGMENT	TYPE	TSPACE	BLOCKS	BLOCK_SIZE	KB	FIVE_PCT (KB)	DAILY_GROW (KB)	TOTAL (KB)	TOTAL (Block)
CUSTCLUSTER	CLUSTER	CUST	3,333,771,005	2,048	6,667,542,010	333,377,100	0	7,000,919,110	3,500,459,555
DB_STAT	SYS	SYSTEM	1,048,576	2,048	2,097,152	0	0	2,097,152	1,048,576
DCLUSTER	CLUSTER	DIST	2,230,560	2,048	4,461,120	223,056	0	4,684,176	2,342,088
HIST	TABLE	HIST	92,696,072	4,096	370,784,288	0	72,993,392	443,777,680	110,944,420
ICUST1	INDEX	ICUST1	8,595,540	16,384	137,528,640	6,876,432	0	144,405,072	9,025,317
ICUST2	INDEX	ICUST2	18,811,296	16,384	300,980,736	15,049,040	0	316,029,776	19,751,861
IDIST	INDEX	IDIST	521,312	2,048	1,042,624	52,132	0	1,094,756	547,378
IITEM	INDEX	ITEMS	5,632	2,048	11,264	564	0	11,828	5,914
IORDR2	INDEX	IORD2	13,457,508	16,384	215,320,128	10,766,000	0	226,086,128	14,130,383
ISTOK	INDEX	ISTK	25,903,750	16,384	414,460,000	20,723,008	0	435,183,008	27,198,938
ITEMCLUSTER	CLUSTER	ITEMS	8,446	2,048	16,892	844	0	17,736	8,868
IWARE	INDEX	WARE	130,712	2,048	261,424	13,072	0	274,496	137,248
NORDCLUSTER_QUEUE	CLUSTER	NORD	23,283,000	2,048	46,566,000	2,328,300	0	48,894,300	24,447,150
ORDRCLUSTER_QUEUE	CLUSTER	ORDR	321,699,630	16,384	5,147,194,080	0	1,013,287,712	6,160,481,792	385,030,112
STOCKCLUSTER	CLUSTER	STOK	2,976,968,755	2,048	5,953,937,510	297,696,876	0	6,251,634,386	3,125,817,193
SYS_IQ0000009734\$\$	INDEX	SYSTEM	372,528	2,048	745,056	37,252	0	782,308	391,154
SYS_IQ0000009836\$\$	INDEX	SYSTEM	1,419,880	16,384	22,718,080	1,135,904	0	23,853,984	1,490,874
SYSAUX	SYS	SYSTEM	61,440	2,048	122,880	0	0	122,880	61,440
SYSTEM	SYS	SYSTEM	204,800	2,048	409,600	0	0	409,600	204,800
WCLUSTER	CLUSTER	WARE	223,414	2,048	446,828	22,342	0	469,170	234,585
Total			6,821,413,856		19,286,646,312	688,301,922	1,086,281,104	21,061,229,338	7,223,277,854
Dynamic space(KB)		5,517,978,368							
Static space(KB)		14,456,969,866							
Free space(KB)		1,086,281,104							
Daily growth(KB)		1,086,281,104							
Daily spread		0	Oracle may be configured such that daily spread is 0						
60-day (GB)		75,944.74							
Log KB/tpmC		5.10	KB of log used per New-Order						
8-hour log (GB)		5,561.08							
		Capacity	NumDisk				Total GB		
Database Disks	ETERNUS8000	67.99	3,456				234,973.44		
							234,973.44		
		Capacity	NumDisk(RAID0+1)						
8-Hr Log Disks	ETERNUS8000	67.99	384				26,108.16		
							26,108.16		

Appendix G: Numerical Quantities Summary per Client

Client	Priced	Substituted										
	RX200 S4 c069	C200 c007	c008	c009	c010	c011	c012	c013	c014	c017	c018	c019
tpmC user	27762.94 22000	13315.03 11000	12831.29 11000	13589.71 11000	13399.82 11000	13466.30 11000	13272.24 11000	13844.52 11000	13561.19 11000	13266.20 11000	12794.27 11000	13385.79 11000
Menu												
average response	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103
90%ile response	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104
New Order												
average response	0.259	1.197	2.075	0.743	1.058	0.956	1.295	0.325	0.779	1.293	2.160	1.081
90%ile response	0.301	1.489	2.451	1.037	1.334	1.258	1.620	0.485	1.047	1.564	2.557	1.407
average think time	12.022	12.020	12.017	12.009	12.021	12.003	12.011	12.015	12.018	12.015	12.008	12.006
average keying time	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012
mixture %	44.93	44.92	44.92	44.94	44.92	44.94	44.97	44.93	44.90	44.94	44.96	44.92
Payment												
average response	0.242	1.184	2.061	0.731	1.044	0.941	1.280	0.313	0.761	1.278	2.145	1.069
90%ile response	0.291	1.480	2.440	1.027	1.324	1.249	1.609	0.477	1.037	1.555	2.545	1.397
average think time	12.024	12.021	12.019	12.025	12.003	12.010	12.011	12.007	12.014	12.020	12.012	12.023
average keying time	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012
mixture %	43.03	43.03	43.03	43.00	43.04	43.04	43.00	43.02	43.04	42.99	43.00	43.02
Order Status												
average response	0.247	1.188	2.068	0.737	1.049	0.948	1.289	0.318	0.765	1.285	2.154	1.073
90%ile response	0.298	1.483	2.445	1.032	1.331	1.257	1.619	0.482	1.042	1.562	2.555	1.402
average think time	9.998	9.978	10.031	10.006	10.063	10.007	10.024	9.989	9.971	10.020	10.007	10.019
average keying time	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012
mixture %	4.01	4.02	4.01	4.04	4.01	4.01	4.02	4.04	4.02	4.02	4.00	4.02
Delivery												
average response	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103
90%ile response	0.103	0.104	0.103	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.103	0.104
average think time	5.011	5.018	4.996	5.038	5.015	5.017	5.030	5.033	5.031	5.027	5.040	4.999
average keying time	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012
mixture %	4.02	4.02	4.02	4.02	4.01	4.00	4.00	4.01	4.02	4.02	4.01	4.03
Stock Level												
average response	0.233	1.176	2.051	0.721	1.033	0.934	1.268	0.303	0.751	1.268	2.137	1.056
90%ile response	0.281	1.472	2.429	1.019	1.314	1.241	1.601	0.466	1.027	1.546	2.534	1.385
average think time	5.014	5.019	4.997	5.014	4.994	5.023	5.015	5.002	5.031	5.021	5.026	5.034
average keying time	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012
mixture %	4.01	4.01	4.02	4.00	4.02	4.01	4.01	4.00	4.02	4.03	4.03	4.01
# of New Order	3331553	1597804	1539755	1630766	1607979	1615956	1592669	1661343	1627343	1591944	1535313	1606295

Client	Substituted											F250 cI034
	C200											
	cI020	cI021	cI022	cI025	cI026	cI027	cI028	cI029	cI030	cI051	cI052	
tpmC user	13036.99 11000	13072.25 11000	12882.97 11000	13651.59 11000	13342.83 11000	12797.10 11000	13581.05 11000	13399.63 11000	12957.90 11000	13480.91 11000	13252.97 11000	26454.12 22000
Menu												
average response	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.104
90%ile response	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104
New Order												
average response	1.691	1.649	1.978	0.622	1.151	2.150	0.747	1.078	1.847	0.920	1.332	1.370
90%ile response	2.035	1.986	2.296	0.917	1.443	2.490	1.030	1.363	2.185	1.181	1.642	1.589
average think time	12.019	12.010	12.018	12.002	12.006	12.004	12.016	12.013	12.002	12.029	11.998	12.013
average keying time	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012
mixture %	44.92	44.96	44.92	44.87	44.90	44.94	44.93	44.96	44.93	44.96	44.94	44.93
Payment												
average response	1.678	1.636	1.961	0.610	1.139	2.135	0.733	1.062	1.834	0.905	1.319	1.352
90%ile response	2.025	1.974	2.286	0.906	1.435	2.478	1.021	1.354	2.174	1.173	1.633	1.579
average think time	12.021	12.019	12.029	12.020	12.005	12.019	12.024	12.001	12.019	12.027	12.012	12.005
average keying time	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012
mixture %	43.05	42.98	43.00	43.06	43.04	43.03	43.04	42.99	43.03	42.99	42.97	43.03
Order Status												
average response	1.684	1.644	1.972	0.617	1.143	2.145	0.738	1.070	1.844	0.910	1.326	1.356
90%ile response	2.033	1.983	2.297	0.915	1.440	2.488	1.025	1.363	2.182	1.180	1.638	1.585
average think time	10.020	9.987	9.992	10.015	10.073	9.967	10.026	10.013	10.026	9.988	10.026	9.974
average keying time	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012
mixture %	4.02	4.02	4.03	4.03	4.02	4.01	4.01	4.02	4.01	4.02	4.04	4.01
Delivery												
average response	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.104
90%ile response	0.103	0.103	0.103	0.104	0.104	0.103	0.104	0.104	0.103	0.104	0.104	0.104
average think time	4.989	5.019	5.026	4.987	5.029	5.010	5.019	5.036	5.033	5.005	5.006	5.019
average keying time	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012
mixture %	3.99	4.02	4.02	4.02	4.03	4.02	4.02	4.01	4.02	4.00	4.02	4.01
Stock Level												
average response	1.666	1.627	1.950	0.598	1.128	2.126	0.722	1.053	1.827	0.899	1.312	1.340
90%ile response	2.015	1.965	2.277	0.894	1.425	2.469	1.013	1.345	2.166	1.164	1.623	1.567
average think time	4.995	5.024	5.012	5.019	5.007	5.024	5.042	5.014	5.019	5.021	5.000	5.013
average keying time	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012
mixture %	4.02	4.02	4.03	4.02	4.01	4.00	4.00	4.02	4.01	4.03	4.03	4.02
# of New Order	1564439	1568671	1545957	1638191	1601140	1535652	1629727	1607956	1554948	1617710	1590357	3174495

Client	Substituted												
	F250	d035	d036	d037	d038	d039	d040	d041	d042	d043	d044	d045	d046
tpmC	26198.45	27083.05	26525.31	26448.75	27172.29	27329.18	26754.75	26848.53	27214.10	26713.44	26590.75	26812.15	
user	22000	22000	22000	22000	22000	22000	22000	22000	22000	22000	22000	22000	22000
Menu													
average respon	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104
90%ile respons	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104
New Order													
average respon	1.607	0.814	1.305	1.363	0.745	0.629	1.092	1.177	0.715	1.132	1.251	1.246	
90%ile respons	1.904	1.044	1.598	1.635	0.967	0.859	1.344	1.411	0.960	1.383	1.510	1.501	
average think ti	12.005	12.004	12.012	12.010	12.017	12.016	12.015	12.021	12.014	12.021	12.012	12.011	
average keying	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012
mixture %	44.96	44.89	44.94	44.91	44.94	44.97	44.90	44.90	44.95	44.94	44.94	44.99	
Payment													
average respon	1.592	0.800	1.294	1.350	0.730	0.614	1.080	1.163	0.701	1.118	1.235	1.231	
90%ile respons	1.894	1.034	1.588	1.624	0.957	0.849	1.335	1.401	0.950	1.374	1.499	1.491	
average think ti	12.008	12.022	12.011	12.014	12.017	12.016	12.016	12.022	12.025	12.019	12.010	12.007	
average keying	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012
mixture %	43.00	43.04	43.01	43.03	43.00	42.97	43.03	43.04	43.01	43.03	43.01	42.97	
Order Status													
average respon	1.599	0.805	1.301	1.359	0.735	0.618	1.084	1.169	0.707	1.125	1.242	1.237	
90%ile respons	1.903	1.042	1.595	1.633	0.963	0.855	1.340	1.406	0.956	1.380	1.508	1.497	
average think ti	10.013	9.994	10.009	9.974	10.008	9.994	10.005	9.978	10.008	9.990	9.995	10.026	
average keying	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012
mixture %	4.02	4.02	4.02	4.03	4.02	4.03	4.03	4.03	4.01	4.01	4.02	4.02	
Delivery													
average respon	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104
90%ile respons	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104
average think ti	5.021	5.016	5.027	5.020	5.024	5.002	5.023	5.007	5.020	5.015	5.022	5.016	
average keying	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012
mixture %	4.01	4.03	4.02	4.01	4.02	4.03	4.02	4.01	4.02	4.01	4.02	4.00	
Stock Level													
average respon	1.580	0.790	1.284	1.342	0.721	0.603	1.071	1.154	0.689	1.109	1.225	1.221	
90%ile respons	1.882	1.024	1.576	1.614	0.946	0.840	1.325	1.390	0.941	1.364	1.490	1.481	
average think ti	5.010	5.019	5.007	5.005	5.012	5.031	5.021	5.016	5.013	5.019	5.017	5.017	
average keying	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012
mixture %	4.01	4.02	4.01	4.02	4.02	4.00	4.02	4.02	4.01	4.01	4.01	4.02	
# of New Order	3143575	3249967	3183038	3173850	3260675	3279500	3210570	3197824	3265693	3205613	3190891	3193458	

Client	Substituted											
	F250 d047	RX200 S2 d081	d062	d063	d064	d065	d066	d067	d068	d071	d072	d074
tpmC user	26618.95 22000	26675.08 22000	27514.83 22000	27561.55 22000	26248.98 22000	27561.39 22000	27732.54 22000	27363.05 22000	27723.97 22000	27498.35 22000	27193.18 22000	27206.79 22000
Menu												
average respon	0.104	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103
90%ile respons	0.104	0.104	0.104	0.104	0.104	0.103	0.103	0.103	0.103	0.103	0.103	0.103
New Order												
average respon	1.223	1.162	0.456	0.434	1.546	0.434	0.290	0.593	0.303	0.476	0.737	0.729
90%ile respons	1.470	1.426	0.653	0.620	1.849	0.623	0.361	0.791	0.415	0.677	0.944	0.968
average think ti	12.019	12.014	12.016	12.016	12.018	12.020	12.014	12.004	12.018	12.021	12.016	12.002
average keying	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012
mixture %	44.97	44.93	44.92	44.95	44.95	44.93	44.93	44.96	44.95	44.94	44.96	44.95
Payment												
average respon	1.207	1.150	0.444	0.423	1.534	0.420	0.273	0.579	0.288	0.462	0.723	0.716
90%ile respons	1.459	1.417	0.644	0.610	1.839	0.614	0.350	0.781	0.404	0.668	0.934	0.958
average think ti	12.024	12.025	12.014	12.020	12.013	12.007	12.007	12.027	12.006	12.020	12.017	12.014
average keying	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012
mixture %	43.00	43.01	43.04	43.00	43.01	43.03	43.03	43.01	43.01	43.01	42.99	43.00
Order Status												
average respon	1.218	1.158	0.452	0.429	1.542	0.428	0.279	0.588	0.298	0.470	0.730	0.722
90%ile respons	1.466	1.422	0.651	0.618	1.847	0.621	0.357	0.788	0.413	0.676	0.942	0.965
average think ti	10.072	10.058	10.010	10.019	10.011	10.044	10.045	10.032	9.995	10.000	10.024	10.009
average keying	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012
mixture %	4.01	4.02	4.01	4.01	4.02	4.02	4.03	4.01	4.01	4.02	4.02	4.01
Delivery												
average respon	0.104	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103
90%ile respons	0.104	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103
average think ti	5.012	5.027	5.021	5.009	5.021	5.007	5.014	5.041	5.028	5.020	5.023	5.018
average keying	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012
mixture %	4.00	4.01	4.02	4.02	4.02	4.01	4.00	4.01	4.01	4.01	4.03	4.02
Stock Level												
average respon	1.199	1.141	0.433	0.413	1.524	0.412	0.261	0.569	0.279	0.452	0.714	0.706
90%ile respons	1.450	1.407	0.632	0.600	1.827	0.604	0.340	0.771	0.394	0.657	0.925	0.948
average think ti	4.999	5.002	5.006	5.019	5.029	5.022	5.013	5.010	5.021	5.028	5.022	5.036
average keying	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012
mixture %	4.02	4.03	4.01	4.01	4.00	4.01	4.01	4.01	4.02	4.02	4.00	4.02
# of New Order	3194274	3201010	3301780	3307386	3149878	3306167	3327905	3283567	3326877	3299803	3263182	3264815

Client	Substituted											
	cl075	cl076	cl077	cl078	cl079	cl080	cl081	cl082	cl083	cl084	cl085	cl086
tpmC	27704.05	27512.37	27294.87	27183.34	26913.33	27193.4	26233.55	27678.55	27382	27113.92	27113.51	27598.89
user	22000	22000	22000	22000	22000	22000	22000	22000	22000	22000	22000	22000
Menu												
average respon	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.104	0.104
90%ile respons	0.103	0.103	0.103	0.103	0.103	0.103	0.104	0.104	0.104	0.104	0.104	0.104
New Order												
average respon	0.313	0.621	0.650	0.737	0.964	0.725	1.563	0.331	0.572	0.792	0.808	0.399
90%ile respons	0.443	0.847	0.888	0.965	1.21	0.943	1.841	0.465	0.804	1.029	1.022	0.567
average think ti	12.020	12.026	12.013	12.026	12.024	12.016	12.019	12.018	12.01	12.02	12.009	12.029
average keying	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012
mixture %	44.95	44.94	44.93	44.94	44.94	44.94	44.94	44.94	44.93	44.94	44.93	44.93
Payment												
average respon	0.300	0.806	0.634	0.721	0.948	0.712	1.55	0.315	0.558	0.778	0.79	0.382
90%ile respons	0.433	0.837	0.858	0.956	1.2	0.933	1.832	0.455	0.795	1.018	1.012	0.557
average think ti	12.012	12.033	12.009	12.013	12.013	12.02	12.01	12.017	12.018	12.021	12.01	12.008
average keying	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012
mixture %	43.03	43.01	43.03	43.03	43	43.02	43.01	43.02	43.04	43	43.02	43
Order Status												
average respon	0.306	0.615	0.644	0.726	0.955	0.717	1.558	0.321	0.564	0.785	0.797	0.386
90%ile respons	0.440	0.845	0.867	0.962	1.208	0.939	1.839	0.461	0.8	1.026	1.021	0.562
average think ti	9.999	10.005	10.034	9.992	10.01	10.033	9.998	10.044	10.028	9.995	9.998	9.99
average keying	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012
mixture %	4.01	4.02	4.01	4.01	4.01	4.01	4.02	4.01	4.01	4.02	4.02	4.02
Delivery												
average respon	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103
90%ile respons	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.104	0.104
average think ti	5.024	5.013	4.998	5.012	5.025	5.027	4.998	5.014	5.014	5.014	5.006	4.999
average keying	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012
mixture %	4.00	4.01	4.02	4.02	4.03	4.01	4.03	4.02	4.01	4.02	4.02	4.03
Stock Level												
average respon	0.289	0.595	0.624	0.711	0.939	0.702	1.54	0.308	0.548	0.767	0.778	0.371
90%ile respons	0.422	0.826	0.847	0.945	1.189	0.923	1.821	0.445	0.784	1.008	1.002	0.547
average think ti	5.026	5.023	5.013	5.011	5.031	5.02	5.02	5.018	5	5.012	5.007	5.028
average keying	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012
mixture %	4.01	4.02	4.01	4	4.02	4.02	4	4.01	4.01	4.02	4.01	4.02
# of New Order	3324486	3277485	3275361	3262001	3229600	3263209	3148026	3321426	3285840	3253671	3253622	3311867

Client	Substituted												
	RX200 S2	d087	d088	d089	d090	d091	d092	d095	d105	d106	d108	d109	d110
tpmC user	26778.17	27044.95	26263.63	27481.92	27540.58	27533.9	26246.79	27198.44	27701.43	27769.92	27366.8	25908.2	
	22000	22000	22000	22000	22000	22000	22000	22000	22000	22000	22000	22000	22000
Menu													
average respon	0.104	0.104	0.103	0.103	0.104	0.104	0.103	0.103	0.103	0.103	0.103	0.103	0.103
90%ile respons	0.104	0.104	0.103	0.103	0.104	0.104	0.103	0.103	0.103	0.103	0.103	0.103	0.103
New Order													
average respon	1.076	0.856	1.528	0.497	0.448	0.441	1.539	0.716	0.307	0.263	0.586	1.851	
90%ile respons	1.319	1.057	1.782	0.688	0.647	0.636	1.84	0.929	0.408	0.34	0.765	2.126	
average think ti	12.018	12.016	12.016	12.013	12.02	12.017	12.022	12.025	12.007	12.018	12.029	12.011	
average keying	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	
mixture %	44.96	44.95	44.93	44.95	44.97	44.91	44.93	44.9	44.89	44.95	44.95	44.93	
Payment													
average respon	1.081	0.838	1.513	0.482	0.431	0.428	1.528	0.7	0.289	0.249	0.568	1.835	
90%ile respons	1.309	1.046	1.772	0.678	0.636	0.626	1.831	0.919	0.399	0.33	0.755	2.116	
average think ti	12.033	12.02	12.019	12.014	12.024	12.011	12.021	12.006	12.013	12.009	12.02	12.022	
average keying	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	
mixture %	43.02	42.98	43.03	43.01	42.99	43.03	43.01	43.04	43.04	43.01	43	43.03	
Order Status													
average respon	1.071	0.843	1.519	0.49	0.44	0.435	1.535	0.709	0.293	0.255	0.575	1.84	
90%ile respons	1.318	1.052	1.779	0.686	0.644	0.633	1.837	0.927	0.405	0.337	0.762	2.123	
average think ti	10.048	10.049	9.996	10.005	10.04	10.019	10.023	10.017	10.043	9.98	10.025	10.042	
average keying	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	
mixture %	4	4.02	4.01	4.01	4.02	4.01	4.02	4.02	4.01	4.01	4.02	4.01	
Delivery													
average respon	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	
90%ile respons	0.104	0.104	0.103	0.103	0.104	0.104	0.103	0.103	0.103	0.103	0.103	0.103	
average think ti	5.014	5.019	5.018	5.026	5.031	5.026	5.023	5.012	5.009	5.017	5.009	5.003	
average keying	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	
mixture %	4.01	4.03	4.02	4.02	4	4.03	4.02	4.02	4.04	4.02	4.02	4.01	
Stock Level													
average respon	1.051	0.831	1.505	0.473	0.421	0.419	1.518	0.691	0.28	0.24	0.559	1.825	
90%ile respons	1.299	1.037	1.764	0.668	0.625	0.615	1.82	0.909	0.389	0.32	0.746	2.107	
average think ti	5.011	5.023	5.031	5.02	5.008	5.018	5.015	5.016	5.01	5.01	5.02	5.01	
average keying	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	
mixture %	4.01	4.02	4.01	4.01	4.02	4.02	4.02	4.02	4.02	4.01	4.01	4.02	
# of New Order	3213381	3245394	3151636	3297831	3304870	3304069	3149615	3263813	3324172	3332391	3284016	3108984	

Client	Substituted											
	RX200 S2											
	cl111	cl112	cl113	cl114	cl115	cl116	cl117	cl118	cl119	cl121	cl122	cl124
tpmc user	27633.85 22000	27632.61 22000	27509.22 22000	27224.44 22000	27773.53 22000	27668.73 22000	27154.1 22000	28282.06 22000	27647.61 22000	27666.1 22000	28952.36 22000	27249.81 22000
Menu												
average response	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.103	0.104	0.104	0.104	0.104
90%ile response	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104
New Order												
average response	0.38	0.381	0.466	0.709	0.266	0.343	0.777	1.545	0.362	0.349	0.926	0.683
90%ile response	0.558	0.547	0.659	0.933	0.341	0.49	0.981	1.817	0.516	0.478	1.181	0.924
average think tim	12.006	12.01	12.021	12.02	12.012	12.017	12.014	12.01	12.014	12.019	12.015	12.022
average keying ti	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012
mixture %	44.97	44.97	44.94	44.94	44.97	44.94	44.96	44.92	44.95	44.96	44.92	44.95
Payment												
average response	0.367	0.365	0.453	0.695	0.252	0.328	0.762	1.531	0.348	0.331	0.913	0.67
90%ile response	0.548	0.537	0.648	0.922	0.331	0.481	0.972	1.807	0.506	0.467	1.171	0.913
average think tim	12.018	12.017	12.01	11.997	12.02	12.012	12.008	12.005	12.008	12.023	12.014	12.009
average keying ti	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012
mixture %	43	42.98	43.02	43.03	42.97	43.03	43	43.05	43.01	43	43.03	43.01
Order Status												
average response	0.373	0.374	0.461	0.703	0.258	0.334	0.768	1.536	0.355	0.339	0.919	0.678
90%ile response	0.553	0.544	0.657	0.929	0.337	0.487	0.978	1.813	0.511	0.474	1.177	0.923
average think tim	10.016	9.998	10.003	10.002	10.02	9.977	10.019	10.028	10.005	10.015	9.991	10.045
average keying ti	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012
mixture %	4.01	4.01	4.02	4.01	4.04	4.01	4.01	4	4.01	4	4.02	4.02
Delivery												
average response	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103
90%ile response	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.103	0.104	0.104	0.104	0.104
average think tim	5.007	5.025	5.024	5.011	5.009	5.021	4.994	5.008	5.019	5.023	5.032	4.992
average keying ti	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012
mixture %	4.01	4.02	4.02	4.01	4.01	4.01	4.02	4.01	4.01	4.01	4.01	4
Stock Level												
average response	0.357	0.354	0.444	0.687	0.242	0.319	0.749	1.519	0.339	0.319	0.902	0.66
90%ile response	0.537	0.525	0.639	0.913	0.321	0.47	0.96	1.795	0.496	0.456	1.16	0.904
average think tim	5.018	5.029	5.007	5.022	5.027	5.026	5.015	5.009	5.023	5.026	5.033	5.022
average keying ti	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012
mixture %	4.01	4.02	4	4.01	4.01	4.01	4.01	4.02	4.02	4.01	4.02	4.02
# of New Order	3316063	3315914	3301107	3266933	3332824	3320248	3258492	3150248	3317714	3319933	3234284	3269978

Client	Substituted											
	RX200 S2											
	d125	d126	d127	d128	d129	d130	d131	d132	d133	d134	d135	d136
tpmC	27734.83	27229.62	27157.95	25391.13	27617.84	27613.21	27098.35	27251.27	27640.39	27721.1	27264.25	25512.4
user	22000	22000	22000	22000	22000	22000	22000	22000	22000	22000	22000	22000
Menu												
average respon	0.104	0.104	0.104	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103
90%ile respons	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104	0.104
New Order												
average respon	0.285	0.716	0.755	2.328	0.385	0.401	0.816	0.895	0.363	0.295	0.666	2.215
90%ile respons	0.387	0.951	0.978	2.893	0.548	0.587	1.053	0.924	0.51	0.429	0.877	2.566
average think ti	12.026	12.01	12.019	12.019	12.009	12.005	12.015	12.012	12.017	12.007	12.004	12.011
average keying	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012	18.012
mixture %	44.94	44.97	44.95	44.91	44.94	44.96	44.95	44.97	44.93	44.92	44.91	44.93
Payment												
average respon	0.271	0.701	0.741	2.314	0.372	0.388	0.8	0.68	0.346	0.282	0.654	2.203
90%ile respons	0.377	0.941	0.968	2.883	0.539	0.578	1.042	0.913	0.5	0.418	0.868	2.556
average think ti	12.016	12.016	12.02	12.022	12.018	12.012	12.013	12.024	12.015	12.016	12.018	12.015
average keying	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012	3.012
mixture %	43.02	42.98	43.03	43.01	43.01	42.98	42.99	42.98	43.03	43.05	43.03	43.03
Order Status												
average respon	0.276	0.706	0.748	2.32	0.377	0.393	0.807	0.688	0.353	0.287	0.66	2.21
90%ile respons	0.384	0.946	0.975	2.887	0.544	0.584	1.051	0.921	0.506	0.424	0.874	2.563
average think ti	10.012	10.004	10.023	9.999	10.05	9.98	10.015	9.986	10.03	10.01	10.038	10.018
average keying	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012
mixture %	4.02	4.02	4	4.03	4.01	4.02	4.03	4.01	4.01	4.01	4.02	4.01
Delivery												
average respon	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103
90%ile respons	0.104	0.104	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103	0.103
average think ti	5.02	5.014	5.018	5.016	5.016	5.02	5.013	5.025	5.019	5.028	5.018	5.02
average keying	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012
mixture %	4.01	4.02	4	4.03	4.02	4.02	4.01	4.03	4.02	4.02	4.02	4.01
Stock Level												
average respon	0.26	0.691	0.73	2.301	0.363	0.378	0.792	0.67	0.337	0.271	0.643	2.191
90%ile respons	0.385	0.931	0.958	2.867	0.529	0.567	1.032	0.903	0.491	0.406	0.856	2.544
average think ti	5.012	5.01	5.001	5.017	5.03	5.026	5.002	5.011	5.02	5.037	5.025	5.02
average keying	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012	2.012
mixture %	4.01	4.01	4.02	4.02	4.02	4.02	4.02	4.03	4.01	4	4.02	4.02
# of New Order	3328180	3267555	3258954	3046936	3314141	3313586	3251803	3270153	3316847	3326533	3271711	3061488

Client	Substituted		
	RX200 S2		
	cl137	cl138	cl139
tpmC	27552.48	27502.07	27247.07
user	22000	22000	22000
Menu			
average response	0.103	0.103	0.103
90%ile response	0.104	0.104	0.104
New Order			
average response	0.435	0.474	0.695
90%ile response	0.626	0.68	0.915
average think time	12.018	12.015	12.012
average keying	18.012	18.012	18.012
mixture %	44.95	44.94	44.96
Payment			
average response	0.422	0.459	0.682
90%ile response	0.616	0.669	0.904
average think time	12.027	12.014	12.01
average keying	3.012	3.012	3.012
mixture %	42.99	43.02	42.99
Order Status			
average response	0.427	0.466	0.69
90%ile response	0.622	0.676	0.912
average think time	10.006	10.044	9.979
average keying	2.012	2.012	2.012
mixture %	4.02	4.02	4.02
Delivery			
average response	0.103	0.103	0.103
90%ile response	0.103	0.103	0.103
average think time	5.03	5.021	5.019
average keying	2.012	2.012	2.012
mixture %	4.02	4.01	4.01
Stock Level			
average response	0.412	0.45	0.673
90%ile response	0.606	0.659	0.895
average think time	5.012	5.019	5.024
average keying	2.012	2.012	2.012
mixture %	4.02	4.01	4.02
# of New Order	3306298	3300249	3269649

Appendix H: Price Quotes

From: MaryBeth Pierantoni [mailto:mary.beth.pierantoni@oracle.com]
 Sent: Saturday, November 22, 2008 3:35 AM
 To: Shin'ichi Kurogi
 Subject: Oracle Pricing

Product	Price	Quantity	Extended Price
Oracle Database 10g Enterprise Edition, Per Processor, Unlimited Users for 3 years	\$23,750	32*	\$760,000
Partitioning, Per Processor, Unlimited Users for 3 years	\$5,750	32*	\$184,000
Oracle Database Server Support for 3 years	\$2,300	3	\$6,900
Tuxedo CFS-R Tier 1	\$1,800	99	\$178,200
Oracle Premium Support for 3 years	\$39,204	3	\$117,612
Oracle Mandatory E-Business Discount			<\$311,678>
Oracle TOTAL			\$935,034

(*32 = 0.50 * 64. Explanation: For the purpose of counting the number of processors which require licensing, an Intel multicore chip with "n" cores shall be determined by multiplying "n" cores by a factor of 0.50).

Oracle pricing contact: MaryBeth Pierantoni, mary.beth.pierantoni@oracle.com, 916-315-5081

WS-C2950T-24 24 Port Networking Switch - WS-C2950T-24 - Mozilla Firefox

http://www.getitnew.com/index.asp?PageAction=VIEWPROD&ProdID=137

getitnew TECHNOLOGY

Manufacturers: --Select--

Join Mailing List: Enter Email Here GO

Call Us: 800-567-9121

Search Products: GO

Home | Login

products

- Cisco Blade
- Cisco GBIC
- Cisco Module
- Cisco Power Supply
- Cisco Router
- Cisco Security
- Cisco Switch
- Cisco Telephony
- Cisco WIC
- Nortel Baystack
- Nortel Cable
- Nortel Chassis
- Nortel Module
- Nortel Passport
- Nortel Switch
- Specials of the Week

my cart

Your Cart Is Empty

SPECIALS of the WEEK

FREE GROUND SHIPPING ON MOST ORDERS WITH IN THE U.S.A

McAfee SECURE TESTED DAILY 06-OCT

VISA MasterCard American Express

Home | Login | Order Tracking

Browse All Products

Powered By MonsterCommerce Shopping Cart Software

Cisco WS-C2950T-24

WS-C2950T-24

Cisco Catalyst 24 Port 10/100 2PT GETH Switch W/ EI S/W

Qty In Stock: 3

Your Price: \$839.00

Qty: 1

buy

Techonweb.com - CISCO CON-SNTP-C2950T24 Cisco SMARTnet P12158 - Mozilla Firefox

http://www.techonweb.com/products/productdetail.aspx?id=B20287&bcsi_scan_D7AAAD91BBD8FCDA=ZcV98K

About | Policy | Sign In | My Account | Cart

TechOnWeb.com where geeks go shopping

Shop by Manufacture

Home Computers Printers Electronics Peripherals Games Software Storage Networking Components Misc.

Cable/Tool | Mounting Kit | POS Data | Desk Option | Cellular Option | Book | Education | Service | Tablet | Other

Misc. -> Service -> HW Maintenance Agreement -> Networking

Search

Misc. 90

Advanced Search

Service

Configuration Service

HW Maintenance Agreement

- ▶ Computer
- ▶ Networking
- ▶ Printer
- ▶ Projector
- ▶ Scanner
- ▶ UPS
- ▶ Other
- ▶ Storage

On-Line Service

- SW Maintenance Agreement
- Technical Support

CISCO - Cisco SMARTnet

Part #: **CON-SNTP-C2950T24**

SKU: **B20287**

Condition: **New**

Status: **Ship within 2~3 weeks**

List Price: ~~\$n/a~~

Reg. Price: **\$119.99**

Price: **\$119.99** BUY NOW!

Product Alert Email Friend

Product Description

SMARTnet support provides customers with software updates and upgrades, registered access to Cisco Connection Online (CCO), advance replacement of hardware, and technical support. Cisco Systems developed SMARTnet services to meet the maintenance requirements of customers with internal maintenance support staff. The SMARTnet support product is based on Cisco's philosophy of moving beyond traditional business barriers to achieve the following: make all of Cisco's information, services, and support available to its customers on demand, deliver fast response and high customer satisfaction, improve user productivity, significantly lower the cost of doing business. SMARTnet Premium service provides all of the features of SMARTnet service, plus the fastest response time for replacement parts, 4-hour or 2-hour response time for advance replacement parts 24 hours a day, 7 days a week, including Cisco-observed holidays.

Product Specifications

Product Description	Cisco SMARTnet Premium extended service agreement - 1 year
Type	Extended service agreement
Service Included	Replacement
Full Contract Period	1 year
Response Time	4 hour(s) 4 hours
Service Availability	24 hours a day / 7 days a week

Newsletter

Enter your EMAIL to receive our monthly specials.

Subscribe

Other Products

HP-COMPAQ - CARE
PACK 1YR PW 4H
13X5-PROCURVE
8116FL H/W SUP
\$ 10,383.56

HP-COMPAQ - CARE
PACK 3YR 13X5
4HR-PROCURVE
2600-8 PWR
\$ 184.92

HP-COMPAQ - HP
Software License
UD543E
\$ 195.87

Rebate Items

RZ363A#ABA - HP
Software License
RZ363A#ABA
\$ 264.25

L1943A#B1H - HP
ScanJet 7650n
Networked ...
\$ 814.92

L2690A#201 -
SCANJET N8460
FBSCAN 110V-U...
\$ 1,310.65

TOSHIBA
Leading Innovation 30

introducing
the new
**PORTÉGÉ®
R500**

NOTEBOOK PC WITH
INTEL® CENTRINO®
DUO PROCESSOR
TECHNOLOGY

**WORLD'S
LIGHTEST**
UNUSUAL 2.2"

Appendix I: Auditor's attestation letter



Benchmark Sponsor: Shin'ichi Kurogi
 Manager, TRIOLE Technology Development Division
 Software Unit
 Fujitsu Limited
 Shin-Yokohama TECH Bldg.
 3-9-18 Shin-Yokohama, Kohoku-ku, Yokohama
 Kanagawa Pref. 222-0033, Japan

November 25, 2008

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

Platform: PRIMEQUEST 580A c/s
 Operating system: Red Hat Enterprise Linux 4 AS
 Database Manager: Oracle 10g Release 2 Enterprise Edition with Partitioning
 Transaction Manager: Tuxedo 8.1 CFS-R Tier 1

The results were:

CPU's Speed	Memory	Disks	New Order 90% Response Time	tpmC
Server: PRIMEQUEST 580A c/s				
32 x Itanium Dual-Core 9150M (1.66GHz)	2 TB (24MB L3 cache)	3840 x 73.4 GB 15K rpm	1.597 Seconds	2,382,032.06
99 Client: PRIMERGY RX200 S4 (each with)				
1 x Intel Xeon X5260 DC (3.33 GHz)	3 GB (6MB L2 cache)	1 x 73 GB SAS	n/a	n/a

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

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

Additional Audit Notes:

Server Storage Substitution

The measured system included (9) ETERNUS6000 Model900 storage enclosures; (8) were populated with 400 36GB disks each; (1) was populated with 256 36GB disks. These were substituted in the priced configuration by (9) ETERNUS8000 Model900 storage enclosures; (8) were populated with 400 73GB disks each; (1) was populated with 256 73GB disks. Based on I/O data collected during testing, it is my opinion that this substitution has no significant effect on performance.

Client Substitution

The tested configuration included (1) priced client model PRIMERGY RX200 S4 (3 GB), (62) non-priced clients model PRIMERGY RX200 S2 (3GB), (14) non-priced clients model PRIMERGY F250 (3 GB) and (22) non-priced clients model PRIMERGY C200 (2GB). The priced configuration includes (99) PRIMERGY RX200 S4 systems. Based on data analysis done for each type of client, it is my opinion that this substitution has no significant effect on performance.

Respectfully Yours,



François Raab, President