



TPC BenchmarkTMC

Full Disclosure Report

*Fujitsu
GP7000F model 2000 c/s
w/ 24 Front-Ends*

running

*Oracle8i
Enterprise Edition 8.1.5*

July 22, 2000

The benchmark results contained in this document were submitted for compliance with version 3.5 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 July 22, 2000.

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 2000 Fujitsu

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.

First Edition July 22, 2000

SPARC64 is a trademark of SPARC International, Inc, used under license by HAL Computer Systems.

Solaris is a trademark of Sun Microsystems, Inc.

ORACLE, SQL*DBA, SQL*Loader, SQL*net, SQL*Plus, Oracle8i, Pro*c and PL/SQL are trademarks of Oracle Corporation.

Pentium is a trademark of Intel, Inc.

Microsoft, Windows, Windows NT 4.0 Enterprise Edition, MS-DOS and the Microsoft logo are registered trademarks of Microsoft Corporation.

TUXEDO 6.4 CFS, is Copyright © 1996-2000 BEA Systems, Inc. All rights reserved.

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

Preface

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

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.

Abstract

Overview

This report documents the methodology and results of the TPC Benchmark C test conducted by Fujitsu Ltd. on the Fujitsu GP7000F model 2000 c/s w/ 24 Front-Ends. The operating system used for the benchmark was Solaris7. The DBMS used was Oracle8i Enterprise Edition 8.1.5.

TPC Benchmark C Metrics

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

138,735.06 tpmC
\$74.32 per tpmC
January 14, 2001


Standard and Executive Summary Statements

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

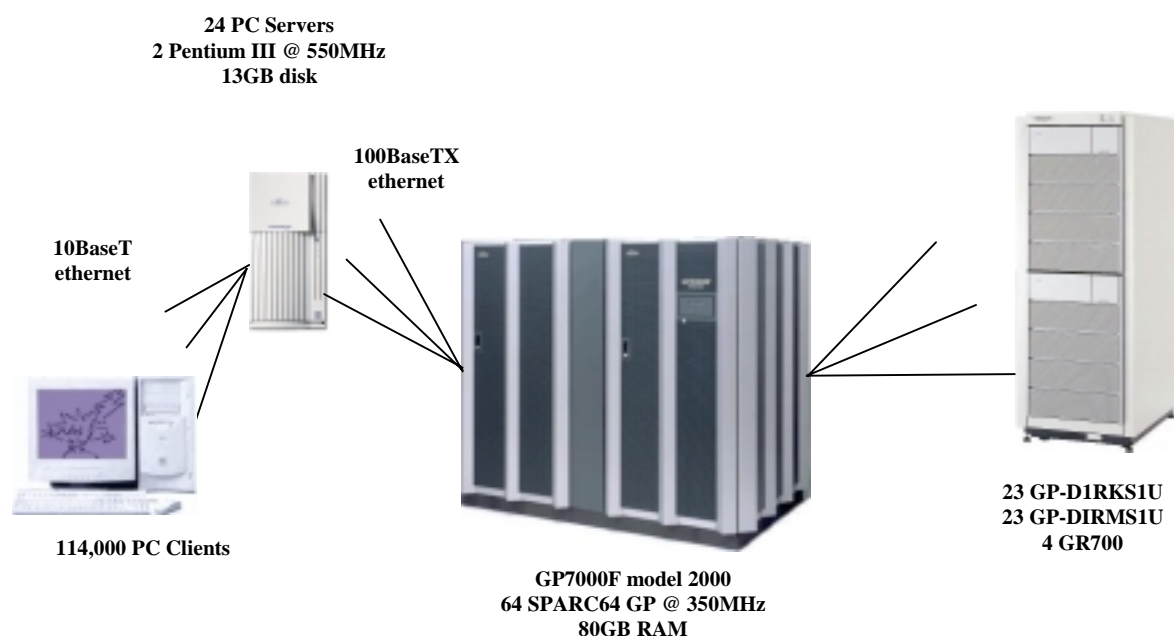
Auditor

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

Priced Configuration


		GP7000F model 2000 c/s w/ 24 Front-Ends		TPC-C Rev 3.5
				Report Date: July 22, 2000
Total System Cost		TPC-C Throughput	Price/Performance	Availability Date
\$10,310,471		138,735.06 tpmC	\$74.32/tpmC	January 14, 2001
Processors	Database Manager	Operating system	Other Software	Number of users
64 SPARC64 GP @ 350MHz	Oracle8i Enterprise Edition 8.1.5	Solaris7	BEA Tuxedo 6.4 CFS, Microsoft Internet Information Server	114,000

GP7000F model 2000 Priced Configuration



RDBMS SERVER			CLIENTS(EACH OF 24)	
	QTY	DESCRIPTION	QTY	DESCRIPTION
PROCESSOR	64	SPARC64 GP @ 350MHZ	2	PENTIUM III @ 550MHZ
CACHE MEMORY		8MB (EACH PROCESSOR)		512KB (EACH PROCESSOR)
MEMORY		80GB		640MB
DISK CONTROLLER	54	ULTRA WIDE DIFFERENTIAL SCSI	1	ULTRA ATA
DISKS	235	18GB (10,000RPM)	1	13GB
	1,222	9GB (10,000RPM)		
TOTAL		15,290 GB		
TERMINAL	1	SYSTEM CONSOLE	1	15" MONITOR
NETWORK INTERFACE	5	FASTETHERNET	3	FASTETHERNET
			24	10/100 ETHERNET SWITCH (24PORTS)
HUBS	5	10/100 ETHERNET SWITCH	14,400	10 BASET HUB (8 PORTS)

Detailed Pricing Information

	<h2 style="margin: 0;">Detailed Pricing Information</h2> <h3 style="margin: 0;">GP7000F model 2000 c/s</h3> <h3 style="margin: 0;">w/ 24 Front-Ends</h3>	<p>TPC-C Rev 3.5</p> <p>Report Date:</p> <p>July 22, 2000</p>
---	--	---

Order Number	Description	Quantity	Third Party	Unit Price	Extended Price	Maintenance rate/unit		5 Years Maintenance
Server Hardware								
GP7N0B1U	GP7000F Model 2000	1		593,507.00	593,507.00	65,286.00	/yr *1	261,144.00
	w/ 8 x SPARC64 GP 350MHz/8MB, 4GB memory, 2 x 9GB disks,							
	2 Dual Channel UltraSCSI/D Cards							
GP7N7ER1U	System Expansion Cabinet for Model 2000	1		127,769.00	127,769.00	14,055.00	/yr *1	56,220.00
GP7N7SB1U	System Board	14		38,735.00	542,290.00	4,261.00	/yr *1	238,616.00
GP7N1B11U	CPU Module (SPARC64 GP 350MHz/8MB)	56		11,502.00	644,112.00	1,265.25	/yr *1	283,416.00
GP7N2M51U	Additional Memory (512MB)	88		2,600.00	228,800.00	286.00	/yr *1	100,672.00
GP7N2M61U	Additional Memory (1024MB)	32		6,000.00	192,000.00	660.00	/yr *1	84,480.00
GP7N7FT1U	Additional Fan Tray	7		1,450.00	10,150.00	160.00	/yr *1	4,480.00
GP7N7PU1U	Additional Power Supply	3		3,475.00	10,425.00	382.25	/yr *1	4,587.00
X1034A-X	Quad FastEthernet Card	1		1,795.00	1,795.00	197.50	/yr *1	790.00
X6541A-X	Dual Channel UltraSCSI/D Card	25		1,440.00	36,000.00	158.50	/yr *1	15,850.00
DCBL-SCR10	VHDCI - 68P SCSI cable	92		131.00	12,052.00	20.00	/5yr	1,840.00
GP-D1RKS1U	Disk Array (Rack, w/ 32MB cache)	23		24,096.00	554,208.00	3,614.40	/5yr	83,131.20
GP-D1RMS1U	Additional Disk Array (Rack-mount, w/ 32MB cache)	23		20,954.00	481,942.00	3,143.10	/5yr	72,291.30
GP-DHD92U	5 x 9GB 10,000rpm disks	204		3,927.00	801,108.00	589.00	/5yr	120,156.00
GP-DHD182U	5 x 18GB 10,000rpm disks	47		7,845.00	368,715.00	1,177.00	/5yr	55,319.00
GP-D1SC1U	Additional RAID Controller (w/ 32MB cache)	46		6,012.00	276,552.00	902.00	/5yr	41,492.00
GP-DPW1U	Additional Power Supply Unit	92		530.00	48,760.00	80.00	/5yr	7,360.00
F6403A3L	Disk array Rackmount model A3 base unit	4		29,418.00	117,672.00	4,413.00	/5yr	17,652.00
	(w/ Disk Drive Modules & cables)							
F6403A12L	SCSI RAID controller pair M128 (128MB cache)	8		22,120.00	176,960.00	3,318.00	/5yr	26,544.00
F6403A23L	10,000rpm 9GB disk drive	200		2,235.00	447,000.00	335.25	/5yr	67,050.00
F6403A41L	19inch Rack (Data Center Cabinet)	4		3,829.00	15,316.00	574.35	/5yr	2,297.40
X6541A-X	Dual Channel UltraSCSI/D Card	4		1,440.00	5,760.00	158.50	/yr *1	2,536.00
DCBL-SCR10	VHDCI - 68P SCSI cable	8		131.00	1,048.00	20.00	/5yr	160.00
GP7N7CL3U	System Console	1		15,275.00	15,275.00	1,680.25	/yr *1	6,721.00
	12/24GB 4mm DDS3 tape drive	1		1,365.00	1,365.00	205.00	/5yr	205.00
Server Hardware Subtotals					5,710,581.00			1,555,009.90
Server Software								
	Solaris 7	1		100.00	100.00	0.00	/yr	0.00
GR72A901	GRmgr for Solaris 2.6/7	1		1,325.00	1,325.00	0.00	/yr	0.00
	Oracle8i Enterprise Edition 8.1.5	1		1,027,900.00	1,027,900.00	1,027,900.00	/5yr	1,027,900.00
	VERITAS Volume Manager License / Direct Assist Support	1	2	53,743.00	53,743.00	9,673.00	/yr	48,365.00
	VERITAS Volume Manager Media / Documentation Kit	1	2	187.00	187.00	0.00	/5yr	0.00
Server Software Subtotals					1,083,255.00			1,076,265.00
Client Hardware								
GP528VIN5	GRANPOWER 5000 Model 280	24		1,772.00	42,528.00	265.80	/5yr	6,379.20
	w/ 1 x Pentium III 550MHz, 1 10/100Base-TX							
GP5-FG10V	Pentium III (550MHz)	24		882.00	21,168.00	132.30	/5yr	3,175.20
GP5-RM25F	256MB memory	48		802.00	38,496.00	120.30	/5yr	5,774.40
GP5-RM12F	128MB memory	24		260.00	6,240.00	39.00	/5yr	936.00
GP5-HD3E1	13GB Disk (EIDE)	24		265.00	6,360.00	39.75	/5yr	954.00
GP5-185	FastEthernet Adapter (10/100Base-TX)	48		120.00	5,760.00	18.00	/5yr	864.00
FMV-DP846	15" Monitor	24		265.00	6,360.00	39.75	/5yr	954.00
Client Hardware Subtotals					126,912.00			19,036.80
Client Software								
	Microsoft Windows NT Server 4.0, include 5 CALs	24	3	809.00	19,416.00	0.00	/yr	0.00
	Microsoft Visual C++ Professional 6.0	1	3	549.00	549.00	0.00	/yr	0.00
	Microsoft Software Maintenance	1	3	0.00	0.00	2,095.00	/yr	10,475.00
	BEA Tuxedo 6.4 CFS	24	4	3,000.00	72,000.00	480.00	/yr	57,600.00
Client Software Subtotals					91,965.00			68,075.00
User Connectivity								
WS-C2924-XL-EN	CISCO Catalyst 2924M-XL-A 10/100 Ethernet Switch (24ports) *	33	5	1,316.70	43,451.10	560.00	/yr	92,400.00
TP1008C	8-Port RJ45 & BNC Ethernet Hub *	15840	6	28.00	443,520.00	0.00	/yr	0.00
User Connectivity Subtotals					486,971.10			92,400.00
Totals					7,499,684.10			2,810,786.70
5 Year cost								10,310,471
tpmC								138,735.06
\$/ tpmC								74.32
*1 : 1st year warranty included.								
*2 : 10% or minimum of 2 spares are included.								
Third Party :								
1 = Oracle Corporation								
2 = VERITAS Software Corporation								
3 = Microsoft Corporation								
4 = BEA Systems								
5 = Fujitsu Systems Business of America, Inc.								

Notes:

Results independently audited by Bradley J. Askins of InfoSizing, Inc.

Prices used in TPC benchmarks reflect the actual prices a customer would pay for a one-time purchase of the stated components. Individually negotiated discounts are not permitted. Special prices based on assumptions about past or future purchases are not permitted. All discounts 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						
GP7000F model 2000 c/s w/ 24 Front-Ends						
Oracle8i Enterprise Edition 8.1.5						
MQTH, Computed Maximum Qualified Throughput					138,735.06 tpmC	
Response Times (in seconds)		Average		90%		Max.
New-Order		1.58		3.58		12.81
Payment		0.32		0.39		4.29
Order-Status		0.28		0.33		4.15
Delivery (interactive portion)		0.18		0.16		3.97
Delivery (deferred portion)		0.34		0.53		4.18
Stock-Level		0.26		0.42		4.36
Menu		0.15		0.15		4.01
Transaction Mix, in percent of total transaction						
New-Order						44.79
Payment						43.10
Order-Status						4.06
Delivery						4.02
Stock-Level						4.03
Emulation Delay (in seconds)				Resp. Time		Menu
New-Order				.1		.1
Payment				.1		.1
Order-Status				.1		.1
Delivery (interactive)				.1		.1
Stock-Level				.1		.1
Keying/Think Times (in seconds)		Min.		Average		Max.
New-Order		18.06 0.02		18.08 12.03		18.44 120.24
Payment		3.05 0.02		3.06 12.04		3.41 120.27
Order-Status		2.05 0.02		2.06 10.07		2.41 100.00
Delivery (interactive)		2.05 0.02		2.06 5.05		2.38 50.41
Stock-Level		2.05 0.02		2.06 5.05		2.37 50.34
Test Duration						
Ramp-up time (seconds)						2,615
Measurement interval						1,800
Transactions during measurement interval						4,162,052
Checkpointing						
Number of checkpoints						1
Checkpoint interval						1,800
Reproducibility Run						
Reported measurement						138,735.06
Reproducibility measurement						137,801.56
Difference						0.67%

Table Of Contents

PREFACE	I
TPC BENCHMARK C OVERVIEW	I
ABSTRACT	III
OVERVIEW	III
TPC BENCHMARK C METRICS	III
STANDARD AND EXECUTIVE SUMMARY STATEMENTS	III
AUDITOR	III
PRICED CONFIGURATION.....	IV
DETAILED PRICING INFORMATION.....	V
NUMERICAL QUANTITIES SUMMARY	VI
TABLE OF CONTENTS	VII
GENERAL ITEMS	11
APPLICATION CODE AND DEFINITION STATEMENTS.....	11
TEST SPONSOR.....	11
PARAMETER SETTINGS	11
CONFIGURATION ITEMS	12
CLAUSE 1 RELATED ITEMS.....	14
1.1 TABLE DEFINITIONS.....	14
1.2 PHYSICAL ORGANIZATION OF DATABASE.....	14
1.3 INSERT AND DELETE OPERATIONS	14
1.4 PARTITIONING	14
1.5 REPLICATION, DUPLICATION OR ADDITIONS	14
CLAUSE 2 RELATED ITEMS.....	15
2.1 RANDOM NUMBER GENERATION	15

2.2 INPUT/OUTPUT SCREEN LAYOUT15

2.3 PRICED TERMINAL FEATURE VERIFICATION15

2.4 PRESENTATION MANAGER OR INTELLIGENT TERMINAL15

2.5 TRANSACTION STATISTICS16

2.6 QUEUEING MECHANISM16

CLAUSE 3 RELATED ITEMS17

3.1 TRANSACTION SYSTEM PROPERTIES (ACID)17

3.2 ATOMICITY17

3.3 CONSISTENCY18

3.4 ISOLATION.....18

3.5 DURABILITY19

CLAUSE 4 RELATED ITEMS21

4.1 INITIAL CARDINALITY OF TABLES21

4.2 DATABASE LAYOUT22

4.3 TYPE OF DATABASE22

4.4 DATABASE MAPPING23

4.5 180 DAY SPACE23

CLAUSE 5 RELATED ITEMS25

5.1 THROUGHPUT25

5.2 RESPONSE TIMES.....25

5.3 KEYING AND THINK TIMES.....26

5.4 RESPONSE TIME FREQUENCY DISTRIBUTION CURVES AND OTHER GRAPHS.....26

5.5 STEADY STATE DETERMINATION.....31

5.6 WORK PERFORMED DURING STEADY STATE31

5.7 REPRODUCIBILITY.....32

5.8 MEASUREMENT PERIOD DURATION.....32

5.9 REGULATION OF TRANSACTION MIX32

5.10 TRANSACTION STATISTICS.....32

5.11 CHECKPOINT COUNT AND LOCATION33

CLAUSE 6 RELATED ITEMS35

6.1 RTE DESCRIPTIONS35

6.2 EMULATED COMPONENTS35

6.3 FUNCTIONAL DIAGRAMS.....35

6.4 NETWORKS36

6.5 OPERATOR INTERVENTION36

CLAUSE 7 RELATED ITEMS37

7.1 SYSTEM PRICING	37
7.2 AVAILABILITY	37
7.3 THROUGHPUT AND PRICE PERFORMANCE	38
7.4 COUNTRY SPECIFIC PRICING.....	38
7.5 USAGE PRICING	38
CLAUSE 9 RELATED ITEMS	39
9.1 AUDITOR'S REPORT.....	39
9.2 AVAILABILITY OF THE FULL DISCLOSURE REPORT	39
APPENDIX A: CLIENT SOURCE CODE.....	41
APPENDIX B: SERVER SOURCE CODE.....	79
APPENDIX C: RTE SCRIPTS	179
APPENDIX D: SYSTEM TUNABLES	187
APPENDIX E: DATABASE CREATION CODE	237
APPENDIX F: 180 DAY SPACE CALCULATION.....	256
APPENDIX G: PRICE QUOTES.....	257
APPENDIX H: AUDITOR'S ATTESTATION LETTER.....	265

General Items

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 code implemented in this benchmark.

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.

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.*

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

Appendix D contains the parameters for the database, the operating system, and the configuration for the transaction monitor.

Configuration Items

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

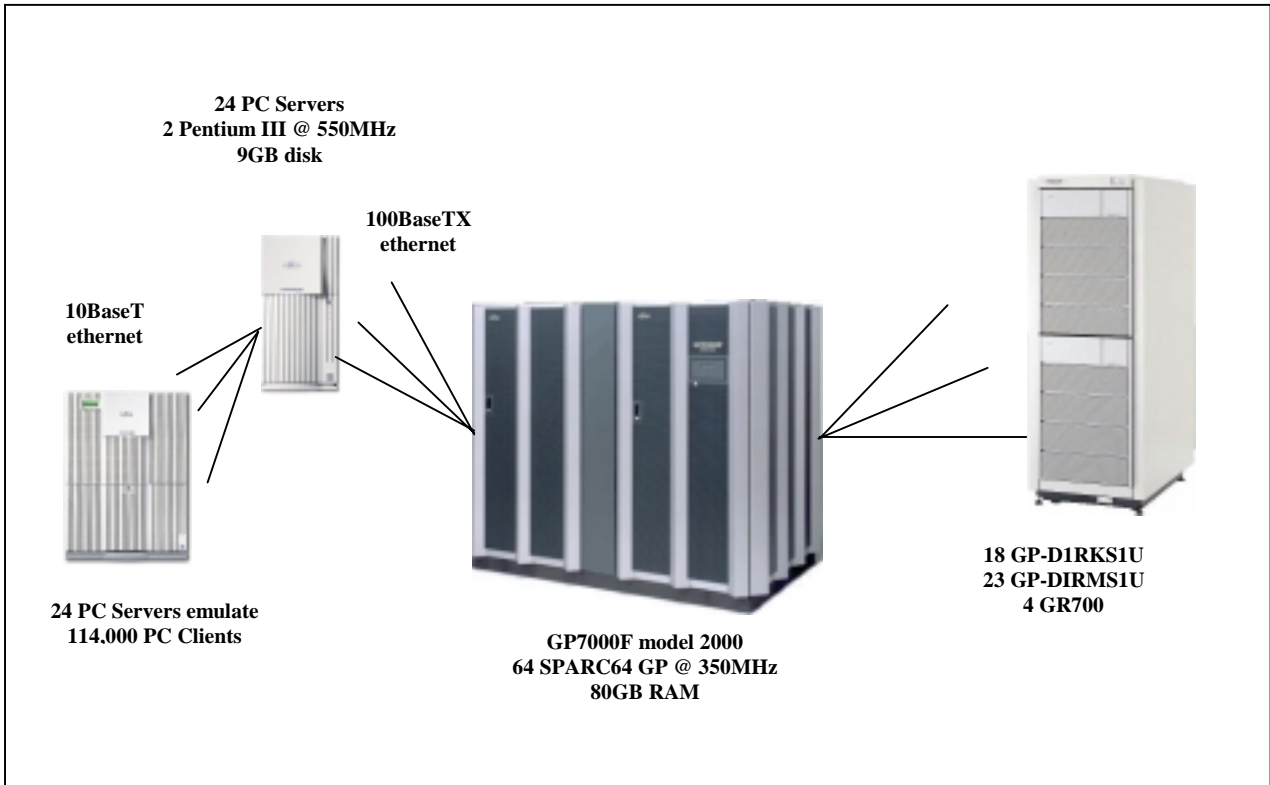
The System Under Test (SUT), a GP7000F model 2000 c/s w/ 24 Front-Ends, is depicted in the following diagrams.

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

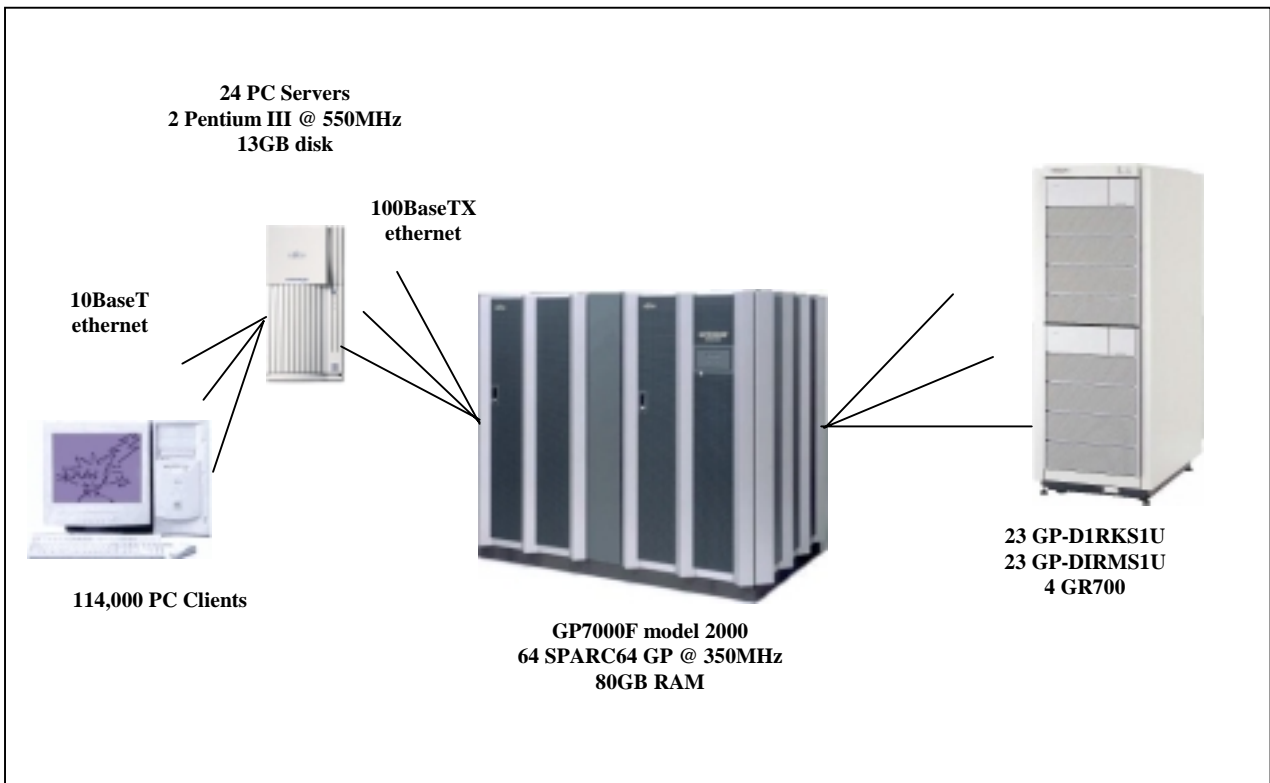
There were differences between the priced and measured configurations. The differences are:

- A RTE was used in the tested configuration.
- The clients that use 9GB disks in the measurement were replaced by those that use 13GB disks in the priced configuration.

GP7000F model 2000 Tested Configuration



GP7000F model 2000 Priced Configuration



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 Oracle8i Server on the server disks according to the details provided in section 5.2. 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.

Partitioning was not used for any of the measurement reported in this full disclosure.

1.5 Replication, Duplication or Additions

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.

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 process id. Each RTE machine was given a number incremented by 30,000. The process id was appended to this number to ensure uniqueness across all RTE machines. These seeds were printed to a file and verified by the auditor to be unique.

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 Statistics

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	84.99%
	Remote warehouse	15.01%
	Accessed by last name	59.98%
Order Status	Accessed by last name	59.97%
Delivery	Skipped transactions	None
Transaction Mix	New Order	44.79%
	Payment	43.10%
	Order status	4.06%
	Delivery	4.02%
	Stock level	4.03%

2.6 Queueing Mechanism

The queueing 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. The only difference was that the Tuxedo call to the server process was asynchronous, i.e., control would return to the client process immediately and the deferred delivery part would complete asynchronously on the server.

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 as specified in Clause 2.5.1.2) 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 as specified in Clause 2.5.1.2) and substitute a ROLLBACK of the transaction for the COMMIT of the transaction. Verify that the records in 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 30 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

Isolation can be defined in terms of phenomena that can occur during the execution of concurrent transactions. These phenomena are P0 (“Dirty Write”), P1 (“Dirty Read”), P2 (“non-repeatable Read”), and P3 (“Phantom”). The table in Clause 3.4.1 of the TPC-C specifications defines the isolation requirements which must be met by the TPC-C transactions. 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 Durable Media Failure

3.5.1.1 Loss of Log And Data

To demonstrate recovery from a permanent failure of durable media containing the Oracle8i recovery log data and TPC-C tables, the following steps were executed on a database of 10 warehouses:

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 100 users.
4. The test was allowed to run for a minimum of 5 minutes.
5. One of the log disks was powered off by removing it from the cabinet. Since the log was mirrored, the transactions continued to run without interruption.
6. The test was allowed to run for another 5 minutes and a disk array failure was caused by removing a disk from the disk array cabinet.
7. The RTE was shut down
8. Oracle8i was shutdown abort.
9. The removed disk was returned into the disk cabinet and the data disk was reformatted to simulate a complete loss of data.
10. Data from the backup disks was restored.
11. Oracle8i was restarted and the media recovery utility started.
12. Step 2 was repeated and the difference between the first and second counts noted.
13. The success file was used to determine the number of NEW_ORDERS successfully returned to the RTE.
14. The counts in step 12 and 13 were compared, verifying that all committed transactions were successfully recovered.
15. 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.

3.5.2 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 11,400 warehouses under a full load of 114,000 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 114,000 users.
3. The test was allowed to run for a minimum of 5 minutes.
4. The primary power to the server was shutdown.
5. Power was restored and the system performed an automatic recovery.
6. Oracle8i 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 8 and 9 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.

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, must be disclosed. If the database was over-scaled and inactive rows of the WAREHOUSE table were deleted, the cardinality of the WAREHOUSE table as initially configured and the number of rows deleted must be disclosed.

The TPC-C database was initially configured with 13,500 warehouses; w-id 11,401 through 13,500 were deleted before performance runs were conducted.

Table 4.1 Number of Rows for Server

Table	Occurrences
Warehouse	11,400
District	135,000
Customer	405,000,000
History	405,000,000
Order	405,000,000
New Order	121,500,000
Order Line	4,050,029,750
Stock	135,000,000
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.

Table 4.2 depicts the layout of the database for the tested system. The data was striped evenly across a total of 1070 disks in 29 DF400's with 30 drives each and 4 GR700's with 50 drives each.

Log and mirrors were located on 41 RAID0 LUNs with 5 8GB drives each and 7 RAID0 LUNs with 5 16GB drives each.

Table 4.2 Database Layout

Device Name	No. of	Physical Disks
Log Devices	4	Each striped across 6 5-way RAID0 LUNs
Log Mirrors	4	Each striped across 6 5-way RAID0 LUNs
Ware	1	Each striped across 214 5-way RAID0 LUNs
District	1	Each striped across 214 5-way RAID0 LUNs
Stock	210	Each striped across 214 5-way RAID0 LUNs
Customer	90	Each striped across 214 5-way RAID0 LUNs
Order Line	90	Each striped across 214 5-way RAID0 LUNs
Order	9	Each striped across 214 5-way RAID0 LUNs
New Order	3	Each striped across 214 5-way RAID0 LUNs
History	7	Each striped across 214 5-way RAID0 LUNs
Item	1	Each striped across 214 5-way RAID0 LUNs
IWare	1	Each striped across 214 5-way RAID0 LUNs
IDist	1	Each striped across 214 5-way RAID0 LUNs
IItem	1	Each striped across 214 5-way RAID0 LUNs
IStock	15	Each striped across 214 5-way RAID0 LUNs
ICust1	15	Each striped across 214 5-way RAID0 LUNs
ICust2	7	Each striped across 214 5-way RAID0 LUNs
IOrder1	15	Each striped across 214 5-way RAID0 LUNs
IOrder2	15	Each striped across 214 5-way RAID0 LUNs
Roll	2	Each striped across 214 5-way RAID0 LUNs

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/1, COBOL read/write used to implement the TPC-C transaction. If more than one interface/access language is used to implement TPC-C, each interface/access language must be described and a list of which interface/access language is used with which transaction type must be disclosed.*

Oracle8i is a relational DBMS.

The interface used was Oracle8i 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.

The database was neither partitioned nor replicated.

4.5 180 Day Space

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

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 previous 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 180 day space requirement is shown in Appendix F.

Clause 5 Related Items

5.1 Throughput

Measured tpmC must be reported.

Measured tpmC	138,735.06
Price per tpmC	\$74.32

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

Type	Average	Maximum	90th %
New-Order	1.58	12.81	3.58
Payment	0.32	4.29	0.39
Order-Status	0.28	4.15	0.33
Interactive Delivery	0.18	3.97	0.16
Deferred Delivery	0.34	4.18	0.53
Stock-Level	0.26	4.36	0.42
Menu	0.15	4.01	0.15

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	18.06	18.08	18.44
Payment	3.05	3.06	3.41
Order-Status	2.05	2.06	2.41
Interactive Delivery	2.05	2.06	2.38
Stock-Level	2.05	2.06	2.37

Table 5.3 Think Times

Type	Minimum	Average	Maximum
New-Order	0.02	12.03	120.24
Payment	0.02	12.04	120.27
Order-Status	0.02	10.07	100.00
Interactive Delivery	0.02	5.05	50.41
Stock-Level	0.02	5.05	50.34

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

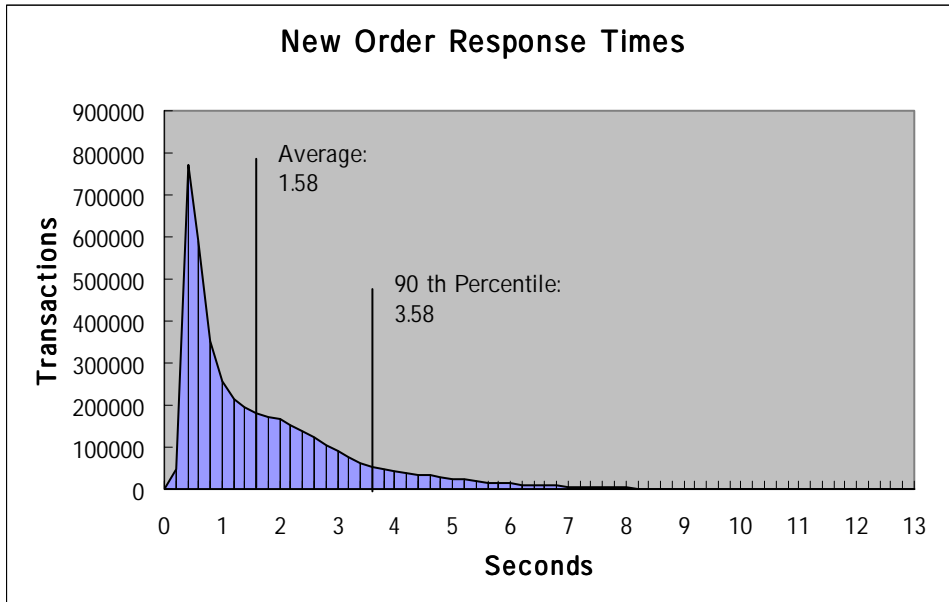


Figure 5.2: Payment Response Time Distribution

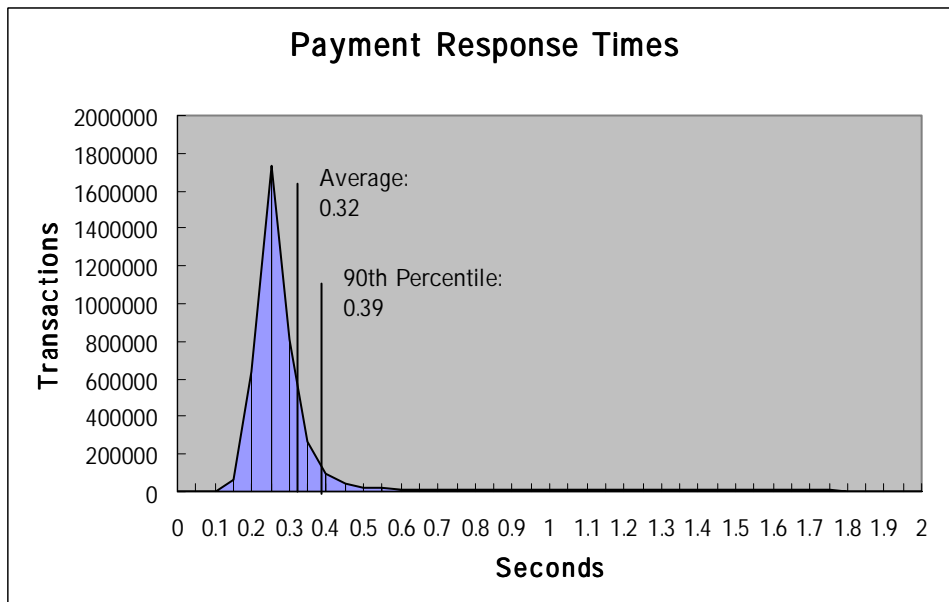


Figure 5.3: Order Status Response Time Distribution

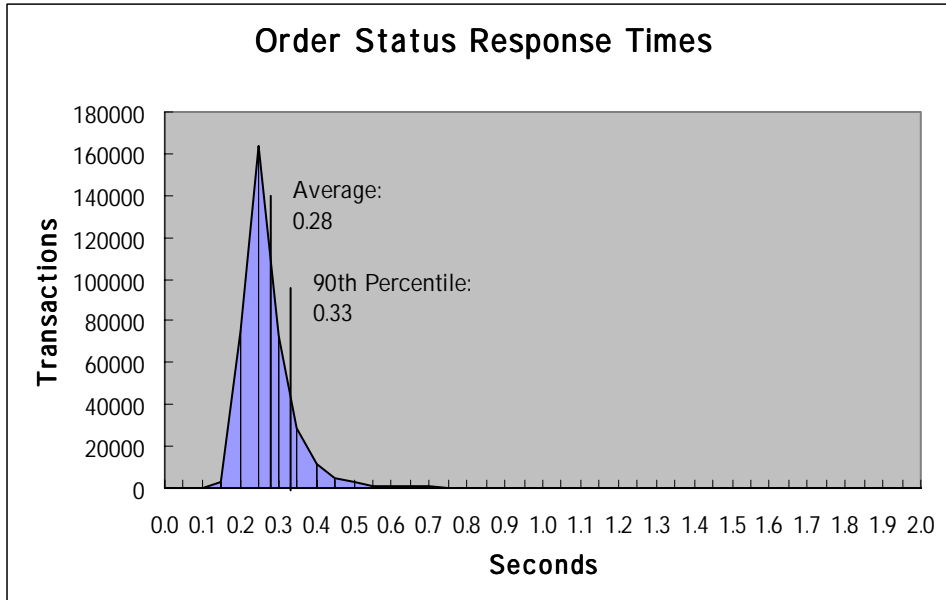


Figure 5.4: Delivery Response Time Distribution

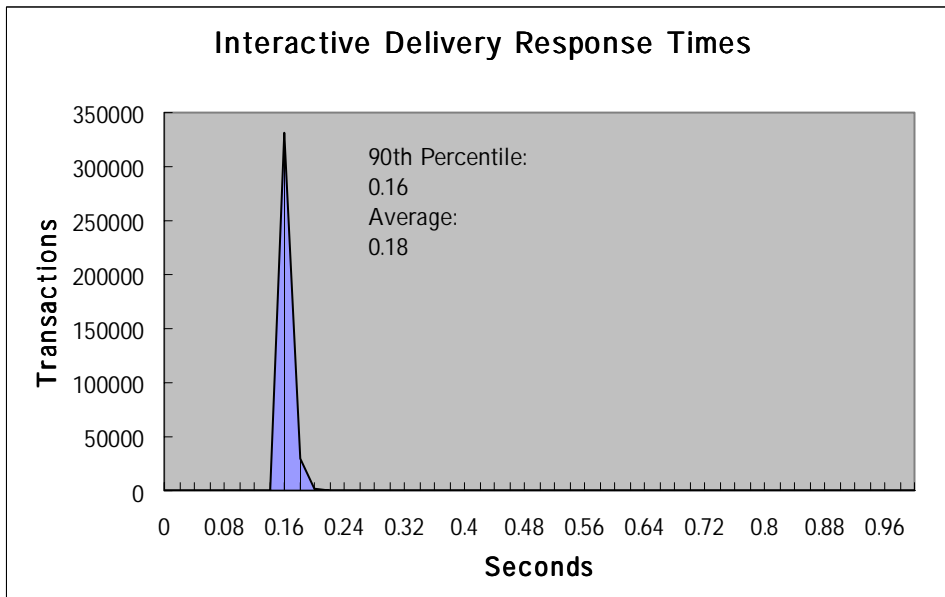


Figure 5.5: Stock Level Response Time Distribution

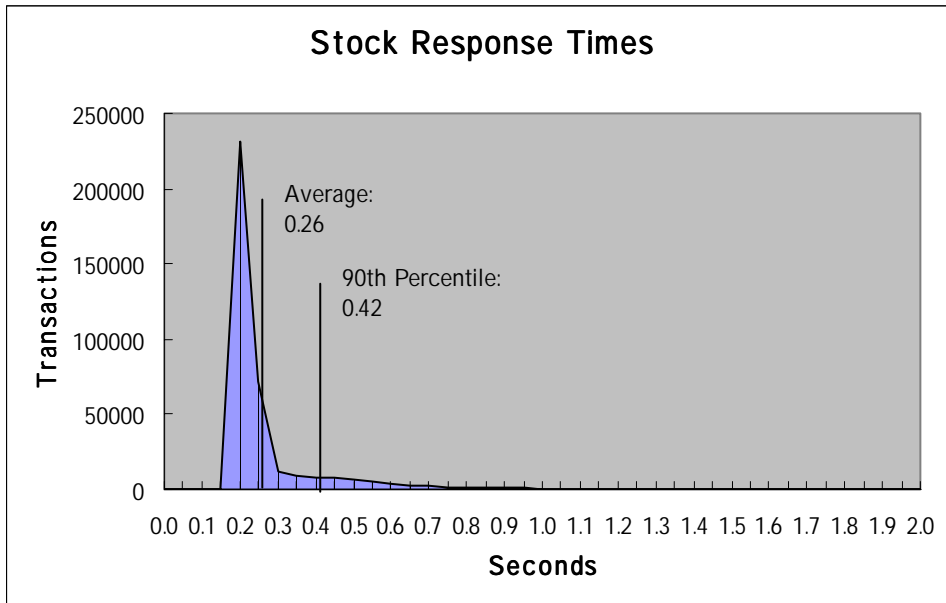


Figure 5.6: New Order Think Time Frequency Distribution

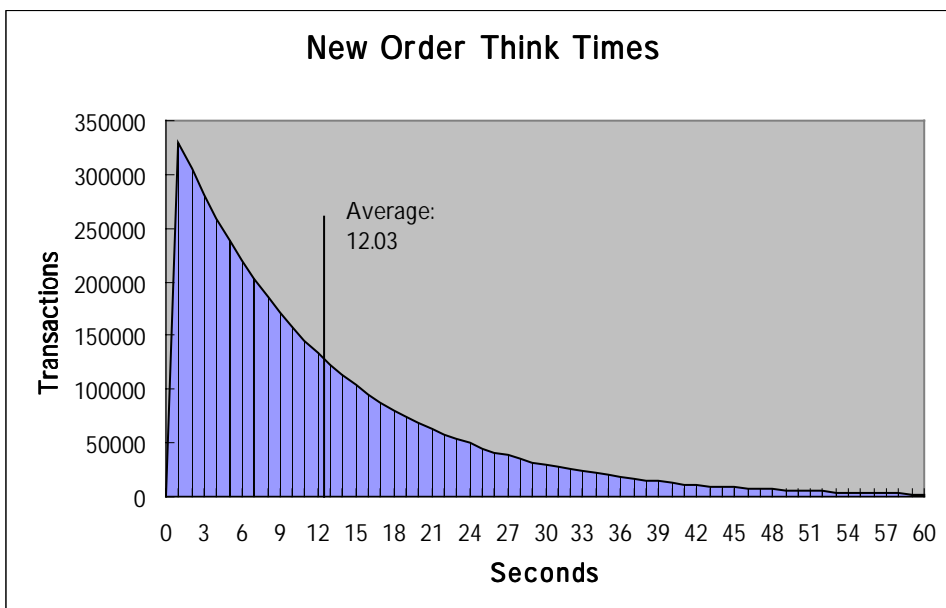


Figure 5.7: Response time versus Throughput

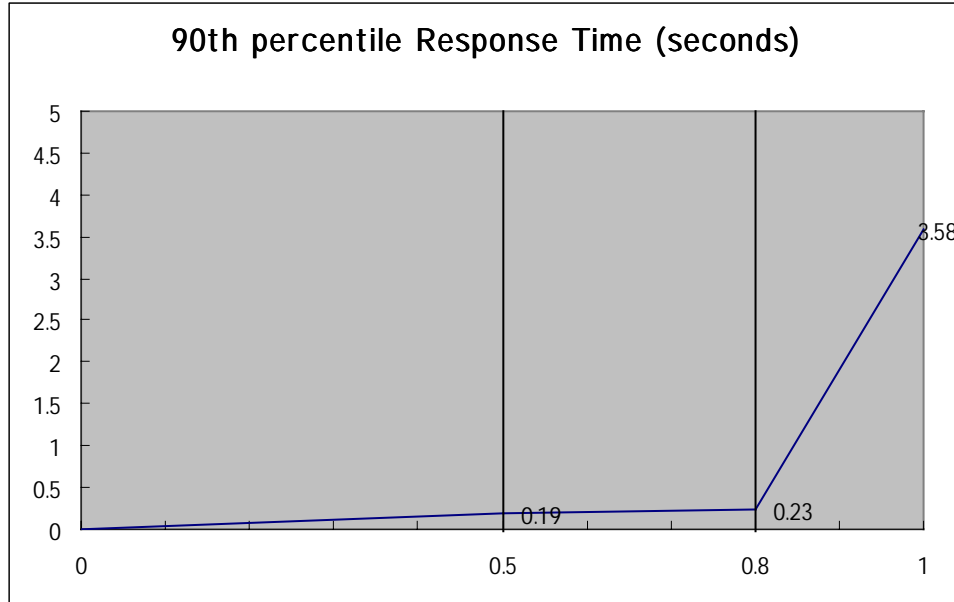
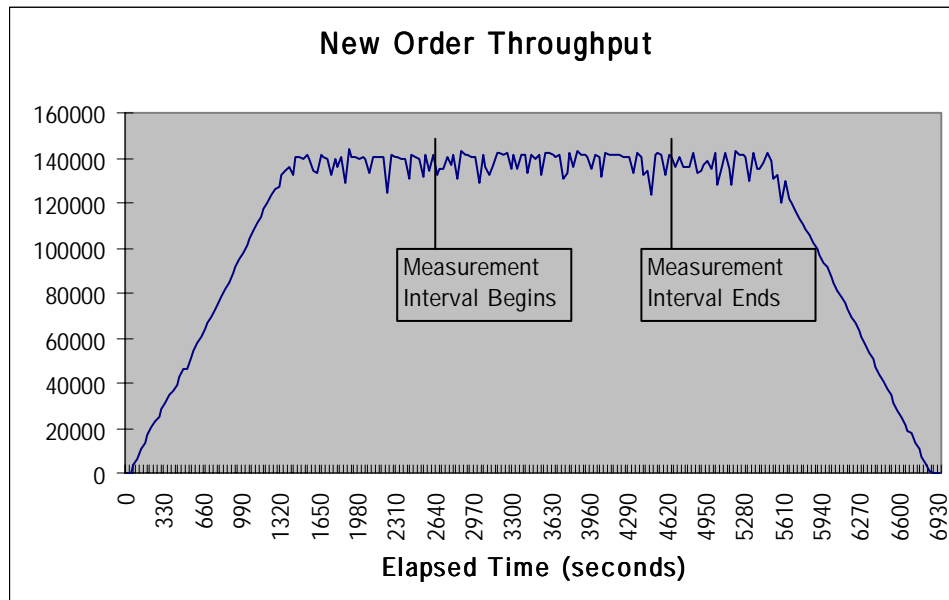


Figure 5.8: New Order Sustained Throughput



5.5 Steady State Determination

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

Steady state was 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 RAID1 array. To perform checkpoints at specific intervals, we set Oracle's checkpoint interval to the maximum allowable value and wrote a script to schedule multiple log switches at specific intervals, which forced checkpoints to occur. Oracle automatically logs all checkpoints to an alert file on the server. The scripts included a wait time between each checkpoint equal to the measurement interval, which was 30 minutes. The checkpoint script was started manually after the RTE had all users logged in and sending transactions. At each checkpoint, Oracle wrote to disk all buffer pages that had been updated but not yet physically written to disk. The positioning of the checkpoint was verified to be clear of the guard zones and depicted on the graph in Figure 5.8.

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.

The measurement procedure was repeated and the throughput verified to be within less than 2% of the reported measurement.

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 30 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 used the UNIX function `irand48()` to control the transaction mix, and 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	44.79%
	Payment	43.10%
	Order status	4.06%
	Delivery	4.02%
	Stock level	4.03%
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	84.99%
	Remote warehouse	15.01%
	Accessed by last name	59.98%
Order Status	Accessed by last name	59.97%
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 another checkpoint was started 687 seconds inside the measured window. Both checkpoints were clear of the guard zone. Checkpoints were started exactly 30 minutes apart.

Clause 6 Related Items

6.1 RTE Descriptions

If the RTE is commercially available, then its inputs must be specified. Otherwise, a description must be supplied of what inputs (e.g., scripts) to the RTE had been used.

The RTE used was developed at Fujitsu Limited and is proprietary. It consists of an RTE management process as shown in Appendix C, which forks off the individual RTE processes and controls the run. After the run completes, a separate report generator program collects all the log files and generates the final statistics of a run.

Inputs to the RTE include the names of the RTE machine to run, client machines to attach to, the database scale, the ramp-up, measurement and ramp-down times. These come from the configuration script file for the RTE management process.

6.2 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.

6.3 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.

6.4 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 replaced with the Driver System must be disclosed.

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

A 100Mbps ethernet LAN connection was used between each client and the server. Twenty-four 10Mbps ethernet LAN connections were used between the emulated users and the client machines.

6.5 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 System Pricing

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

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

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.

All hardware and software components will be available no later than January 14, 2001.

7.3 Throughput and Price Performance

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

Maximum Qualified Throughput (tpmC):	138,735.06
Price per tpmC	\$74.32
Available	January 14, 2001

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.*

None.

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 Benchmark C was audited by Bradley J. Askins of InfoSizing.

InfoSizing
1373 North Franklin St.
Colorado Springs, CO 80903-2527
(voice) 650/533-7357
(fax) 719/473-7554
<http://www.sizing.com>

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

Appendix A: Client Source Code

```
===== tpapl.mak =====
```

```
# Microsoft Developer Studio Generated NMAKE File,
Based on tpapl.dsp
!IF "$(CFG)" == ""
CFG=tpapl - Win32 test
!MESSAGE 構成が指定されていません。デフォルト
の tpapl - Win32 test を設定します。
!ENDIF
```

```
!IF "$(CFG)" != "tpapl - Win32 Release" && "$(CFG)" !=
"tpapl - Win32 keep" && "$(CFG)" != "tpapl - Win32
test" && "$(CFG)" != "tpapl - Win32 Release_dll"
!MESSAGE 指定されたビルドモード "$(CFG)" は
正しくありません。
```

```
!MESSAGE NMAKE の実行時に構成を指定でき
ます
```

```
!MESSAGE コマンドライン上でマクロの設定を定義し
ます。例:
```

```
!MESSAGE
!MESSAGE NMAKE /f "tpapl.mak" CFG="tpapl - Win32
test"
!MESSAGE
```

```
!MESSAGE 選択可能なビルドモード:
```

```
!MESSAGE
!MESSAGE "tpapl - Win32 Release" ("Win32 (x86)
Dynamic-Link Library" 用)
```

```
!MESSAGE "tpapl - Win32 keep" ("Win32 (x86)
Dynamic-Link Library" 用)
```

```
!MESSAGE "tpapl - Win32 test" ("Win32 (x86)
Dynamic-Link Library" 用)
```

```
!MESSAGE "tpapl - Win32 Release_dll" ("Win32 (x86)
Dynamic-Link Library" 用)
```

```
!MESSAGE
!ERROR 無効な構成が指定されています。
!ENDIF
```

```
!IF "$(OS)" == "Windows_NT"
```

```
NULL=
!ELSE
NULL=nul
!ENDIF
```

```
!IF "$(CFG)" == "tpapl - Win32 Release"
```

```
OUTDIR=.\Release
INTDIR=.\Release
# Begin Custom Macros
OutDir=.\Release
# End Custom Macros
```

```
ALL : "$(OUTDIR)\tpapl.dll"
```

```
CLEAN :
```

```
-@erase "$(INTDIR)\StdAfx.obj"
```

```
-@erase "$(INTDIR)\tpapl.obj"
-@erase "$(INTDIR)\tpapl.pch"
-@erase "$(INTDIR)\tpapl.res"
-@erase "$(INTDIR)\vc60.idb"
-@erase "$(OUTDIR)\tpapl.dll"
-@erase "$(OUTDIR)\tpapl.exp"
-@erase "$(OUTDIR)\tpapl.lib"
```

```
"$(OUTDIR)" :
if not exist "$(OUTDIR)\$(NULL)" mkdir "$(OUTDIR)"
```

```
CPP=cl.exe
CPP_PROJ=/nologo /MD /W3 /GX /I "c:\tuxedo\include"
/D "WIN32" /D "NDEBUG" /D "_WINDOWS" /D
"_WINDLL" /D "_AFXDLL" /D "_USRDLL" /D
"_TMSTHEADS" /Fp"$(INTDIR)\tpapl.pch"
/Yu"stdafx.h" /Fo"$(INTDIR)\\" /Fd"$(INTDIR)\\" /FD /c
```

```
.c$(INTDIR).obj:
$(CPP) @<<
$(CPP_PROJ) $<
<<
```

```
.cpp$(INTDIR).obj:
$(CPP) @<<
$(CPP_PROJ) $<
<<
```

```
.cxx$(INTDIR).obj:
$(CPP) @<<
$(CPP_PROJ) $<
<<
```

```
.c$(INTDIR).sbr:
$(CPP) @<<
$(CPP_PROJ) $<
<<
```

```
.cpp$(INTDIR).sbr:
$(CPP) @<<
$(CPP_PROJ) $<
<<
```

```
.cxx$(INTDIR).sbr:
$(CPP) @<<
$(CPP_PROJ) $<
<<
```

```
MTL=midl.exe
MTL_PROJ=/nologo /D "NDEBUG" /mktyplib203 /o
/win32 "NUL"
RSC=rc.exe
RSC_PROJ=/I 0x411 /fo"$(INTDIR)\tpapl.res" /d
"NDEBUG" /d "_AFXDLL"
BSC32=bscmake.exe
BSC32_FLAGS=/nologo /o"$(OUTDIR)\tpapl.bsc"
BSC32_SBRS= \
```

```
LINK32=link.exe
LINK32_FLAGS=libtux.lib libbuff.lib libtux2.lib libfml.lib
libfml32.lib libgp.lib /nologo /subsystem:windows /dll
/incremental:no /pdb:"$(OUTDIR)\tpapl.pdb"
/machine:I386 /def:".\tpapl.def"
/out:"$(OUTDIR)\tpapl.dll" /implib:"$(OUTDIR)\tpapl.lib"
/libpath:"c:\tuxedo\lib"
DEF_FILE= \
".\tpapl.def"
LINK32_OBJS= \
"$(INTDIR)\StdAfx.obj" \
```

```
"$(INTDIR)\tpapl.obj" \
"$(INTDIR)\tpapl.res"
"$(OUTDIR)\tpapl.dll" : "$(OUTDIR)" $(DEF_FILE)
$(LINK32_OBJS)
$(LINK32) @<<
$(LINK32_FLAGS) $(LINK32_OBJS)
<<
```

```
!ELSEIF "$(CFG)" == "tpapl - Win32 keep"
```

```
OUTDIR=.\tpaplkeep
INTDIR=.\tpaplkeep
# Begin Custom Macros
OutDir=.\tpaplkeep
# End Custom Macros
```

```
ALL : "$(OUTDIR)\tpapl.dll" "$(OUTDIR)\tpapl.pch"
```

```
CLEAN :
```

```
-@erase "$(INTDIR)\StdAfx.obj"
-@erase "$(INTDIR)\tpapl.obj"
-@erase "$(INTDIR)\tpapl.pch"
-@erase "$(INTDIR)\tpapl.res"
-@erase "$(INTDIR)\vc60.idb"
-@erase "$(INTDIR)\vc60.pdb"
-@erase "$(OUTDIR)\tpapl.dll"
-@erase "$(OUTDIR)\tpapl.exp"
-@erase "$(OUTDIR)\tpapl.ilk"
-@erase "$(OUTDIR)\tpapl.lib"
-@erase "$(OUTDIR)\tpapl.pdb"
```

```
"$(OUTDIR)" :
if not exist "$(OUTDIR)\$(NULL)" mkdir "$(OUTDIR)"
```

```
CPP=cl.exe
CPP_PROJ=/nologo /Mdd /W3 /GX /ZI /Od /I
"c:\tuxedo\include" /D "WIN32" /D "_DEBUG" /D
"_WINDOWS" /D "_WINDLL" /D "_AFXDLL" /D
"_USRDLL" /D "_TMSTHEADS"
/Fp"$(INTDIR)\tpapl.pch" /Yu"stdafx.h" /Fo"$(INTDIR)\\"
/Fd"$(INTDIR)\\" /FD /c
```

```
.c$(INTDIR).obj:
$(CPP) @<<
$(CPP_PROJ) $<
<<
```

```
.cpp$(INTDIR).obj:
$(CPP) @<<
$(CPP_PROJ) $<
<<
```

```
.cxx$(INTDIR).obj:
$(CPP) @<<
$(CPP_PROJ) $<
<<
```

```
.c$(INTDIR).sbr:
$(CPP) @<<
$(CPP_PROJ) $<
<<
```

```
.cpp$(INTDIR).sbr:
$(CPP) @<<
$(CPP_PROJ) $<
<<
```

```
.cxx$(INTDIR).sbr:
$(CPP) @<<
$(CPP_PROJ) $<
<<

MTL=midl.exe
MTL_PROJ=/nologo /D "_DEBUG" /mktyplib203 /o
/win32 "NUL"
RSC=rc.exe
RSC_PROJ=/I 0x411 /fo"$(INTDIR)\tpapl.res" /d
"_DEBUG" /d "_AFXDLL"
BSC32=bscmake.exe
BSC32_FLAGS=/nologo /o"$(OUTDIR)\tpapl.bsc"
BSC32_SBRS= \

LINK32=link.exe
LINK32_FLAGS=libtux.lib libbuff.lib libtux2.lib libfml.lib
libfml32.lib libgp.lib /nologo /subsystem:windows /dll
/incremental:yes /pdb:"$(OUTDIR)\tpapl.pdb" /debug
/machine:i386 /def:".\tpapl.def"
/out:"$(OUTDIR)\tpapl.dll" /implib:"$(OUTDIR)\tpapl.lib"
/pdbtype:sept /libpath:"c:\tuxedo\lib"
DEF_FILE= \
    ".\tpapl.def"
LINK32_OBJS= \
    "$(INTDIR)\StdAfx.obj" \
    "$(INTDIR)\tpapl.obj" \
    "$(INTDIR)\tpapl.res"

"$(OUTDIR)\tpapl.dll" : "$(OUTDIR)" $(DEF_FILE)
$(LINK32_OBJS)
$(LINK32) @<<
$(LINK32_FLAGS) $(LINK32_OBJS)
<<

!ELSEIF "$(CFG)" == "tpapl - Win32 test"

OUTDIR=. \tpapl__Win32_test
INTDIR=. \tpapl__Win32_test
# Begin Custom Macros
OutDir=. \tpapl__Win32_test
# End Custom Macros

ALL : "$(OUTDIR)\tpapl.dll" "$(OUTDIR)\tpapl.pch"

CLEAN :
-@erase "$(INTDIR)\StdAfx.obj"
-@erase "$(INTDIR)\tpapl.obj"
-@erase "$(INTDIR)\tpapl.pch"
-@erase "$(INTDIR)\tpapl.res"
-@erase "$(INTDIR)\vc60.idb"
-@erase "$(INTDIR)\vc60.pdb"
-@erase "$(OUTDIR)\tpapl.dll"
-@erase "$(OUTDIR)\tpapl.exp"
-@erase "$(OUTDIR)\tpapl.ilk"
-@erase "$(OUTDIR)\tpapl.lib"
-@erase "$(OUTDIR)\tpapl.pdb"

"$(OUTDIR)" :
if not exist "$(OUTDIR)\$(NULL)" mkdir "$(OUTDIR)"

CPP=cl.exe
CPP_PROJ=/nologo /MDd /W3 /GX /ZI /Od /I
"c:\tuxedo\include" /D "WIN32" /D "_DEBUG" /D
"_WINDOWS" /D "_WINDLL" /D "_AFXDLL" /D
"_USRDLL" /D "_TMSTHEADS" /D "SCRTEST" /D
"USE_FML" /Fp"$(INTDIR)\tpapl.pch" /Yu"stdafx.h"
/Fo"$(INTDIR)\\" /Fd"$(INTDIR)\\" /FD /c
```

```
.c$(INTDIR).obj:
$(CPP) @<<
$(CPP_PROJ) $<
<<

.cpp$(INTDIR).obj:
$(CPP) @<<
$(CPP_PROJ) $<
<<

.cxx$(INTDIR).obj:
$(CPP) @<<
$(CPP_PROJ) $<
<<

.c$(INTDIR).sbr:
$(CPP) @<<
$(CPP_PROJ) $<
<<

.cpp$(INTDIR).sbr:
$(CPP) @<<
$(CPP_PROJ) $<
<<

.cxx$(INTDIR).sbr:
$(CPP) @<<
$(CPP_PROJ) $<
<<

MTL=midl.exe
MTL_PROJ=/nologo /D "_DEBUG" /mktyplib203 /o
/win32 "NUL"
RSC=rc.exe
RSC_PROJ=/I 0x411 /fo"$(INTDIR)\tpapl.res" /d
"_DEBUG" /d "_AFXDLL"
BSC32=bscmake.exe
BSC32_FLAGS=/nologo /o"$(OUTDIR)\tpapl.bsc"
BSC32_SBRS= \

LINK32=link.exe
LINK32_FLAGS=libtux.lib libbuff.lib libtux2.lib libfml.lib
libfml32.lib libgp.lib /nologo /subsystem:windows /dll
/incremental:yes /pdb:"$(OUTDIR)\tpapl.pdb" /debug
/machine:i386 /def:".\tpapl.def"
/out:"$(OUTDIR)\tpapl.dll" /implib:"$(OUTDIR)\tpapl.lib"
/pdbtype:sept /libpath:"c:\tuxedo\lib"
DEF_FILE= \
    ".\tpapl.def"
LINK32_OBJS= \
    "$(INTDIR)\StdAfx.obj" \
    "$(INTDIR)\tpapl.obj" \
    "$(INTDIR)\tpapl.res"

"$(OUTDIR)\tpapl.dll" : "$(OUTDIR)" $(DEF_FILE)
$(LINK32_OBJS)
$(LINK32) @<<
$(LINK32_FLAGS) $(LINK32_OBJS)
<<

!ELSEIF "$(CFG)" == "tpapl - Win32 Release_dll"

OUTDIR=. \Release_dll
INTDIR=. \Release_dll
# Begin Custom Macros
OutDir=. \Release_dll
# End Custom Macros
```

```
ALL : "$(OUTDIR)\tpapl.dll"

CLEAN :
-@erase "$(INTDIR)\StdAfx.obj"
-@erase "$(INTDIR)\tpapl.obj"
-@erase "$(INTDIR)\tpapl.pch"
-@erase "$(INTDIR)\tpapl.res"
-@erase "$(INTDIR)\vc60.idb"
-@erase "$(OUTDIR)\tpapl.dll"
-@erase "$(OUTDIR)\tpapl.exp"
-@erase "$(OUTDIR)\tpapl.lib"

"$(OUTDIR)" :
if not exist "$(OUTDIR)\$(NULL)" mkdir "$(OUTDIR)"

CPP=cl.exe
CPP_PROJ=/nologo /MD /W3 /GX /O2 /I
"c:\tuxedo\include" /D "WIN32" /D "NDEBUG" /D
"_WINDOWS" /D "_WINDLL" /D "_AFXDLL" /D
"_USRDLL" /D "_TMSTHEADS" /D "SCRTEST" /D
"USE_FML" /Fp"$(INTDIR)\tpapl.pch" /Yu"stdafx.h"
/Fo"$(INTDIR)\\" /Fd"$(INTDIR)\\" /FD /c

.c$(INTDIR).obj:
$(CPP) @<<
$(CPP_PROJ) $<
<<

.cpp$(INTDIR).obj:
$(CPP) @<<
$(CPP_PROJ) $<
<<

.cxx$(INTDIR).obj:
$(CPP) @<<
$(CPP_PROJ) $<
<<

.c$(INTDIR).sbr:
$(CPP) @<<
$(CPP_PROJ) $<
<<

.cpp$(INTDIR).sbr:
$(CPP) @<<
$(CPP_PROJ) $<
<<

.cxx$(INTDIR).sbr:
$(CPP) @<<
$(CPP_PROJ) $<
<<

MTL=midl.exe
MTL_PROJ=/nologo /D "NDEBUG" /mktyplib203 /o
/win32 "NUL"
RSC=rc.exe
RSC_PROJ=/I 0x411 /fo"$(INTDIR)\tpapl.res" /d
"NDEBUG" /d "_AFXDLL"
BSC32=bscmake.exe
BSC32_FLAGS=/nologo /o"$(OUTDIR)\tpapl.bsc"
BSC32_SBRS= \

LINK32=link.exe
LINK32_FLAGS=libtux.lib libbuff.lib libtux2.lib libfml.lib
libfml32.lib libgp.lib /nologo /subsystem:windows /dll
/incremental:no /pdb:"$(OUTDIR)\tpapl.pdb"
```

```

/machine:1386 /def:. \tpapl.def"
/out:"$(OUTDIR)\tpapl.dll" /implib:"$(OUTDIR)\tpapl.lib"
/libpath:"c:\tuxedo\lib"
DEF_FILE= \
    ".\tpapl.def"
LINK32_OBJS= \
    "$(INTDIR)\StdAfx.obj" \
    "$(INTDIR)\tpapl.obj" \
    "$(INTDIR)\tpapl.res"

"$$(OUTDIR)\tpapl.dll" : "$$(OUTDIR)" $(DEF_FILE)
$(LINK32_OBJS)
    $(LINK32) @<<
    $(LINK32_FLAGS) $(LINK32_OBJS)
<<

!ENDIF

!IF "$$(NO_EXTERNAL_DEPS)" != "1"
!IF EXISTS("tpapl.dep")
!INCLUDE "tpapl.dep"
!ELSE
!MESSAGE Warning: cannot find "tpapl.dep"
!ENDIF
!ENDIF

!IF "$(CFG)" == "tpapl - Win32 Release" || "$(CFG)" ==
"tpapl - Win32 keep" || "$(CFG)" == "tpapl - Win32 test"
|| "$(CFG)" == "tpapl - Win32 Release_dll"
SOURCE=. \StdAfx.cpp

!IF "$(CFG)" == "tpapl - Win32 Release"

CPP_SWITCHES=/nologo /MD /W3 /GX /I
"c:\tuxedo\include" /D "WIN32" /D "NDEBUG" /D
"_WINDOWS" /D "_WINDLL" /D "_AFXDLL" /D
"_USRDLL" /D "_TMSTHEADS"
/Fp"$(INTDIR)\tpapl.pch" /Yc"stdafx.h" /Fo"$(INTDIR)\
/Fd"$(INTDIR)\ /FD /c

"$$(INTDIR)\StdAfx.obj" "$$(INTDIR)\tpapl.pch" :
$(SOURCE) "$(INTDIR)"
    $(CPP) @<<
    $(CPP_SWITCHES) $(SOURCE)
<<

!ENDIF

SOURCE=. \tpapl.cpp

!IF "$(CFG)" == "tpapl - Win32 Release"

CPP_SWITCHES=/nologo /MD /W3 /GX /I
"c:\tuxedo\include" /D "WIN32" /D "NDEBUG" /D
"_WINDOWS" /D "_WINDLL" /D "_AFXDLL" /D
"_USRDLL" /D "_TMSTHEADS"
/Fp"$(INTDIR)\tpapl.pch" /Yc"stdafx.h" /Fo"$(INTDIR)\
/Fd"$(INTDIR)\ /FD /c

"$$(INTDIR)\tpapl.obj" : $(SOURCE) "$(INTDIR)"
"$$(INTDIR)\tpapl.pch"
    $(CPP) @<<
    $(CPP_SWITCHES) $(SOURCE)
<<

!ELSEIF "$(CFG)" == "tpapl - Win32 keep"

CPP_SWITCHES=/nologo /MDd /W3 /GX /ZI /Od /I
"c:\tuxedo\include" /D "WIN32" /D "_DEBUG" /D
"_WINDOWS" /D "_WINDLL" /D "_AFXDLL" /D
"_USRDLL" /D "_TMSTHEADS"
/Fp"$(INTDIR)\tpapl.pch" /Yc"stdafx.h" /Fo"$(INTDIR)\
/Fd"$(INTDIR)\ /FD /c

"$$(INTDIR)\StdAfx.obj" "$$(INTDIR)\tpapl.pch" :
$(SOURCE) "$(INTDIR)"
    $(CPP) @<<
    $(CPP_SWITCHES) $(SOURCE)
<<

!ELSEIF "$(CFG)" == "tpapl - Win32 test"

CPP_SWITCHES=/nologo /MDd /W3 /GX /ZI /Od /I
"c:\tuxedo\include" /D "WIN32" /D "_DEBUG" /D
"_WINDOWS" /D "_WINDLL" /D "_AFXDLL" /D
"_USRDLL" /D "_TMSTHEADS" /D "SCRTEST" /D
"USE_FML" /Fo"$(INTDIR)\ /Fd"$(INTDIR)\ /FD /c

"$$(INTDIR)\tpapl.obj" : $(SOURCE) "$(INTDIR)"
    $(CPP) @<<
    $(CPP_SWITCHES) $(SOURCE)
<<

!ELSEIF "$(CFG)" == "tpapl - Win32 Release_dll"

CPP_SWITCHES=/nologo /MD /W3 /GX /O2 /I
"c:\tuxedo\include" /D "WIN32" /D "NDEBUG" /D
"_WINDOWS" /D "_WINDLL" /D "_AFXDLL" /D
"_USRDLL" /D "_TMSTHEADS" /D "SCRTEST" /D
"USE_FML" /Fp"$(INTDIR)\tpapl.pch" /Yu"stdafx.h"
/Fo"$(INTDIR)\ /Fd"$(INTDIR)\ /FD /c

"$$(INTDIR)\tpapl.obj" : $(SOURCE) "$(INTDIR)"
"$$(INTDIR)\tpapl.pch"
    $(CPP) @<<
    $(CPP_SWITCHES) $(SOURCE)
<<

!ENDIF

SOURCE=. \tpapl.rc

"$$(INTDIR)\tpapl.res" : $(SOURCE) "$(INTDIR)"
    $(RSC) $(RSC_PROJ) $(SOURCE)
<<

!ENDIF

===== Resource.h =====

//{{NO_DEPENDENCIES}}
// Microsoft Visual C++ generated include file.
// Used by tpapl.rc
//

#define IDS_SERVER 102

#define _APS_NEXT_RESOURCE_VALUE 103
#define _APS_NEXT_CONTROL_VALUE 103
#define _APS_NEXT_SYMED_VALUE 103
#define _APS_NEXT_COMMAND_VALUE 32768

===== StdAfx.h =====

#if !defined(AFX_STDAFX_H__13B36D5C_C1F8_11D
1_8326_00A0C906407A__INCLUDED_)
#define
AFX_STDAFX_H__13B36D5C_C1F8_11D1_8326_00
A0C906407A__INCLUDED_

// stdafx.h : 標準のシステム インクルード ファ
イル、
// または参照回数が多く、かつあまり変
更されない
// プロジェクト専用のインクルード フ
ァイルを記述します。
//

```



```

void dummy_delivery( delstruct *bp )
{

#ifdef Symfo
// bp->delout.terror = 1;
bp->delout.terror = NOERR;
#else
bp->delout.terror = NOERR;
#endif

return;
}

void dummy_stockvl( stostruct *bp )
{
int i;

#ifdef Symfo
// bp->stoout.terror = 1;
bp->stoout.terror = NOERR;
#else
bp->stoout.terror = NOERR;
#endif

do{
i = rand()%1000;
} while ( i > bp->stoin.threshold );

bp->stoout.low_stock = i;

return;
}

void dummy_payment( paystruct *bp )
{

#ifdef Symfo
// bp->payout.terror = 1;
bp->payout.terror = NOERR;
#else
bp->payout.terror = NOERR;
#endif

// get_datetimestr( bp->h_date );
// check
strcpy( bp->payout.h_date, "12345678901234" );
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" );

bp->payin.c_id = 777;
strcpy( bp->payout.c_first, "John" );
strcpy( bp->payout.c_middle, "H" );
strcpy( bp->payin.c_last, "Watson" );
strcpy( bp->payout.c_street_1, "Baker street" );
strcpy( bp->payout.c_street_2, "221B" );
strcpy( bp->payout.c_credit, "GC" );

#ifdef Symfo
bp->payout.c_discount = 20;
#else
bp->payout.c_discount = (float)0.20;
// check
#endif

strcpy( bp->payout.c_city, "London" );
strcpy( bp->payout.c_state, "GB" );
strcpy( bp->payout.c_zip, "888" );
strcpy( bp->payout.c_phone,
"1234567890123456" );
bp->payout.c_balance = 67876;
bp->payout.c_credit_lim = 77777;
// get_datestr( bp->c_since );
// check
strcpy( bp->payout.c_since, "123456789" );

strcpy( bp->payout.c_data,
"Migyamigyamigyamigyamigya"
"migyamigyamigyamigyamigya" );
return;
}

void dummy_orderstat( ordstruct *bp )
{
int i, j;

#ifdef Symfo
// bp->ordout.terror = 1;
bp->ordout.terror = NOERR;
#else
bp->ordout.terror = NOERR;
#endif

bp->ordin.c_id = rand()%10000;
strcpy( bp->ordout.c_first, "Robert" );
strcpy( bp->ordout.c_middle, "L" );
strcpy( bp->ordin.c_last, "Fish" );
bp->ordout.c_balance =
(( rand()*rand()%19999999 - 9999999 ) /
(double)100.0;
/*
fprintf( stderr, "ordout.c_balance = %12.4fn", bp-
>ordout.c_balance );
bp->c_balance = -1;
*/

bp->ordout.o_id = rand()%10000;
get_datetimestr( bp->o_entry_d );
// check
strcpy( bp->ordout.o_entry_d, "12345678790" );
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;
}

#ifdef Symfo
bp->ordout.ol_amount[i] = (float)rand() *
(float)100; // check
#else
bp->ordout.ol_amount[i] = (float)rand();
// check
#endif

#ifdef Symfo
// bp->newout.terror = 1;
bp->newout.terror = NOERR;
#else
bp->newout.terror = NOERR;
#endif

// *( bp->status ) = '\0';
// This value is nothing

strcpy( bp->newout.c_last, "Holmes" );
strcpy( bp->newout.c_credit, "GC" );
bp->newout.o_id = o_id++;

/*
bp->newout.o_id = ( rand()%100000 )+1;
*/

// get_datetimestr( bp->o_entry_d ); //
check
strcpy( bp->newout.o_entry_d, "1234567879" );

#ifdef Symfo
bp->newout.c_discount = (float)(rand()%101 );
// check
bp->newout.w_tax = (float)( rand()%2001 );
// check
bp->newout.d_tax = (float)( rand()%2001 );
// check
#else
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
#endif

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 ) {

```

```

//          strcpy( bp->status, "Item number is
not valid" );
}

        bp->newout.i_name[i][0] = '\0';
        bp->newout.s_quantity[i] =
( rand()%10)+1;
        bp->newout.brand_generic[i] =
( rand()%26)+'A';

#ifdef Symfo
        bp->newout.i_price[i] =
(float)(( rand()%10000)+1); // check
#else
        bp->newout.i_price[i] =
(float)((( rand()%10000)+1)/100.0); // check
#endif
        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;
}

#endif

//
// 内部データ確認用のサブ関数の定義部分
//

#ifdef DBPRT
void oder_dsp(rte_input_data *in_data,
              ordstruct *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_id = %s ", in_data-
>C_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);
        if ( bp->c_last[0] == '\0' ) {
            fprintf (test_fp, "byname = %d \n", bp-
>ordin.bylastname);
        } else {
            fprintf (test_fp, "c_last = %s :byname
= %d\n", bp->ordin.c_last,
            bp->ordin.bylastname);
        }
    } 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,
            paystruct *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);
        if ( bp->c_last[0] == '\0' ) {
            fprintf (test_fp, "byname = %d \n", bp-
>payin.bylastname);
        } else {
            fprintf (test_fp, "c_last = %s :byname
= %d\n", bp->payin.c_last,
            bp->payin.bylastname);
        }
    }
}

}

        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,
            stostruct *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,
            newstruct *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->ol_i_id[i]);
            fprintf(test_fp, "i_name=%s \n", &bp-
>i_name[i][0]);
            fprintf(test_fp, "o_quant=%d ", bp->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->ol_amount[i]);
            // check
        }
        // fprintf(test_fp, "status=%s \n", bp->status);
        fprintf(test_fp, "total_a=%f \n", bp->total_amount);
        // check
    }
}

#ifdefif
int tsp(int id, char flag, char type){

    // struct tm times;
    SYSTEMTIME systemTime; // for IIS Version

    GetLocalTime(&systemTime);

    // GetLocalTimeの結果を, mktime で数値デー
タに変換する
    // 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(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; }

#endifif

===== fldtbl.h =====
/* fname fldid */
/* ----- */

```



```

HKEY m_tpccregkey;

void Termfree (int Cookie);
void Termfree (char *Cookie);
int Terminit (int w_id, int d_id, int cookie);

CRITICAL_SECTION crit;

#ifdef WIZDEF
void Default(CHttpServerContext* pCtxt, LPTSTR
pszName);
DECLARE_PARSE_MAP()
#endif
//{{AFX_MSG(CTpaplExtension)
//}}AFX_MSG
};

CTpaplExtension *srv;

//{{AFX_INSERT_LOCATION}}
// Microsoft Developer Studio は前行の直前に追加
//の宣言を挿入します。

#endif
// !defined(AFX_TPAPL_H_04D0797B_A452_11D1_8
D77_0000E20BF509__INCLUDED)

===== tpcarea.h =====

/* Client context area (oracle web server) */

typedef struct {
// void *trans_b; /* pointer of interface area with TP
application */
int prt_cnt; /* print counter : for debug */
int clent_num; /* maximam client matchine
number */
int max_user; /* maximam user number */
int ott_num; /* maximam TP applicaton
program of 1 client */
int clent[MAXCLIENT][2]; /* client matchine
infomation */
int ott [MAXCLIENT][MAXOTT]; /* TP application
program information */
}tpc_struct;

===== tpc.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

#ifndef FALSE
#define FALSE 0
#endif

#ifndef TRUE
#define TRUE 1
#endif

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

#ifndef boolean
#define boolean int
#endif

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

typedef struct cda_def csrdef;
typedef struct cda_def ldadef;

/* TPC-C transaction functions */

extern int TPCinit ();
extern int TPCnew ();
extern int TPCpay ();
extern int TPCord ();
extern int TPCdel ();
extern int TPCsto ();
extern void TPCexit ();
extern int TPCdumpinit ();
extern void TPCdumpnew ();
extern void TPCdumpppay ();
extern void TPCdumpord ();
extern void TPCdumpdel ();
extern void TPCdumpsto ();
extern void TPCdumpexit ();
extern void userlog();

/* Error codes */

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

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

#define DELRT 80.0

extern int tkvcninit ();
extern int tkvcpinit ();
extern int 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 tkvcodone ();
extern void tkvcdone ();
extern void tkvcsdone ();

extern int tkvcss (); /* for alter session to get memory
size and trace */
extern boolean multitrax;
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 *curntest;
/* The bind and define handles for each transaction are
included in their respective header files. */

/* for stock-level transaction */

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

/* for delivery transaction */

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

```

```
/* for order-status transaction */
```

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

```
/* for payment transaction */
```

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

```
/* for new order transaction */
```

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

```
extern int i_price[15];
extern char brand_generic[15][1];
extern int status;
extern int tracelevel;
```

```
/* Miscellaneous */
```

```
extern OCIDate cr_date;
extern OCIDate c_since;
extern OCIDate o_entry_d_base;
extern OCIDate ol_d_base[15];
```

```
#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 (void *)NULL
#endif /* NULLP */
```

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

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

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

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

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

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

```
(progvl),(progvl),(ftype),(indp),(alen),(arcode),0,0,OCI_DEFAULT));
```

```
#define OCIBNDRAD(stmp,bndp,errp,sqlvar,progvl,ftype,indp,ctxp,cbf_nodata,cbf_data)\
ocierror(__FILE__,__LINE__,(errp),\
OCIHandleAlloc((stmp),(dvoid**) &(bndp),OCI_HTYPE_BIND,0,(dvoid**)0));\
ocierror(__FILE__,__LINE__,(errp),\
OCIBindByName((stmp), &(bndp), (errp), (text*)(sqlvar),\
strlen((sqlvar)),0,(progvl),(ftype),\
indp,0,0,0,0,OCI_DATA_AT_EXEC));\
ocierror(__FILE__,__LINE__,(errp),\
OCIBindDynamic((bndp), (errp), (ctxp), (cbf_nodata), (ctxp), (cbf_data));
```

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

```
(progvl),(progvl),(ftype),(indp),(alen),(arcode),0,0,OCI_DEFAULT));
```

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

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

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

```
#define OCIDFNRA(stmp,dfnp,errp,pos,progvl,progvl,ftype,indp,alen,arcode)\
OCIHandleAlloc((stmp),(dvoid**) &(dfnp),OCI_HTYPE_DEFINE,0,\
```

```

                (dvoid**)0);\
    OCIDefineByPos((stmp), &(dfnp), (errp), (pos), (progv
), \
    (progv), (ftype), (indp), (alen), \
                (arcod), OCI_DEFAULT);

#define
OCIDFNNDYN(stmp, dfnp, errp, pos, progv, progv, ftype, ind
p, ctp, cbf_data) \
    ocierror(__FILE__, __LINE__, (errp), \
    OCIHandleAlloc((stmp), (dvoid**) &(dfnp), OCI_HTY
PE_DEFINE, 0, \
                (dvoid**)0);\
    ocierror(__FILE__, __LINE__, (errp), \
    OCIDefineByPos((stmp), &(dfnp), (errp), (pos), (progv
), (progv), (ftype), \
                (indp), NULL, NULL,
OCI_DYNAMIC_FETCH));\
    ocierror(__FILE__, __LINE__, (errp), \
    OCIDefineDynamic((dfnp), (errp), (ctp), (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 ol_cnt;
    char c_last[17];
    char c_credit[3];
    float c_discount;
    float w_tax;
    float d_tax;
    char o_entry_d[20];
    float total_amount;
    char i_name[15][25];
    int s_quantity[15];
    char brand_generic[15];
    float i_price[15];
    float ol_amount[15];
    char status[26];
    int retry;
};

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

/* Payment */

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

struct payoutstruct {
    int terror;
    char w_street_1[21];
    char w_street_2[21];
    char w_city[21];
    char w_state[3];
    char w_zip[10];
    char d_street_1[21];
    char d_street_2[21];
    char d_city[21];
    char d_state[3];
    char d_zip[10];
    int c_id;
    char c_first[17];
    char c_middle[3];
    char c_last[17];
    char c_street_1[21];
    char c_street_2[21];
    char c_city[21];
    char c_state[3];
    char c_zip[10];
    char c_phone[17];
    char c_since[11];
    char c_credit[3];
    double c_credit_lim;
    float c_discount;
    double c_balance;
    char c_data[201];
    char h_date[20];
    int retry;
};

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

/* Order status */

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

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

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

/* Delivery */

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

struct deloutstruct {
    int terror;
    int retry;
};

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

/* Stock level */

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

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

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

#endif

*/

===== 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 ALLTXNODEL 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;
    long startsec;
    long startusec;
    /* double qtime;
    int in_timing_int;*/
};

struct deloutstruct {
    int terror;
    int retry;
};

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

/* Stock level */

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

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

struct stostruct {
    int retval;
    struct stoinstruct stoin;
    struct stoostruct stooout;
};

/* 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

```

```

===== tpccis.h =====

//
// Client Application Header file
//
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <sys/types.h>
#include <time.h>
#include <math.h>
#include <stdarg.h>
#include <signal.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <malloc.h>
#include <process.h>

/* TPC-C transaction functions
extern int TPCinit ();
extern int TPCnew ();
extern int TPCpay ();
extern int TPCord ();
extern int TPCdel ();
extern int TPCsto ();
extern int TPCexit ();
extern int TPCdumpinit ();
extern int TPCdumpnew ();
extern int TPCdumppay ();
extern int TPCdumpord ();
extern int TPCdumpdel ();
extern int TPCdumpsto ();
extern int TPCdumpexit ();
*/

/* 99.8.19 w_id 5 keta *
#define MAXCONNECT 25000
#define MAXTERM 6400
#define MAXWH 2500 /* Max Warehouse
scale */
*/

#define MAXCONNECT 250000
#define MAXTERM 64000
#define MAXWH 25000 /* Max Warehouse
scale */

#ifndef Symfo
/* The maximum value of client machine which can be
processed */
#define MAXCLIENT 10

/* The maximum value of TP application program of 1
client machine
which can be processed */
#define MAXOTT 80
#endif

/* number of Transaction */
#define TRANNEW 1
#define TRANPAY 2
#define TRANORD 3
#define TRANDEL 4
#define TRANSTO 5

/* Error codes : send from TP application program */
#define RECOVERR -10

#define IRRECERR -20
#define NOERR 111

#define NG 0
#define OK 1

#define BUF_S 4096 /* size of the send buffer area
*/
#define WORK_S 2400 /* size of the work buffer area
*/

#define TERM_V 0x1 /* Terminal Verification
#define PRT_ELOG 0x8 /* Output Debug
Message

// SymfoWare Use Only
#define TX_NEWORDER 1
#define TX_PAYMENT 2
#define TX_ORDERSTAT 3
#define TX_DELIVERY 4
#define TX_STOCKLVL 5

/* Debug Print proc define : debug use only */

#ifdef DBPRT

#if (DBPRT > 5)
#define DBGP(proc) proc
#define DBGR(proc) proc

#else
#define DBGP(proc)
#define DBGR(proc) proc
#endif

#else
#define DBGP(proc)
#define DBGR(proc)
#endif

#ifdef TPCC_INFO_H
#define TPCC_INFO_H

#define trans_size 1104 /* interfase area size */

#ifndef Symfo
/* Oracle use only
The external variable is declared. (this file use tpcc.c
only) */

long olen;
void *trans_buf;
//int trans_size = 1024;
int svrnum;

int logincnt = 0;

int base_cok = 0;
char NewOrdername[20];
char Paymentname[20];
char OrderStatusname[20];
char Deliveryname[20];
char StockLevelname[20];

char s_buf[BUF_S];
char s_work[WORK_S];

int now_cookie = 0;
int now_w_id = 0;
int now_d_id = 0;
char now_ottname[8] =
{0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00};
#endif

#define VLDATA "Ver 2.1 keep-fml"

#ifndef Symfo
#define INTNULL -32768
#else
#define INTNULL 0
#endif

#ifndef SCRTEST

#ifdef DBPRT
char SOPATH[] = "/dbgSD/tpapl.dll"; /*
DEBG Mode: SCRTEST & DBPRT */
#define MDDATA "SCR And DP"
#else
char SOPATH[] = "/tpc/tpapl.dll"; /* DEBG
Mode: SCRTEST */
#define MDDATA "SCR"
#endif

#else

#ifdef NOSCR
char SOPATH[] = "/tpc/tpapl.dll"; /* DEGB
Mode: */
#define MDDATA "DBG"
#else
char SOPATH[] = "/tpc/tpapl.dll"; /* Release
Mode: */
#define MDDATA "REL"
#endif

#endif

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

#endif

===== tpicinweb.h =====

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

```



```

* -----*/
#define in_paypage "\
<HTML><HEAD><TITLE>TPC-C:
Payment</TITLE></HEAD>\r\n
<BODY><FORM ACTION="%s"
METHOD="GET">\r\n
<INPUT TYPE="hidden" NAME="f"
VALUE="P">\r\n
<INPUT TYPE="hidden" NAME="c"
VALUE="%d">\r\n
<center>Payment<br></center>\r\n
<font size=4><PRE>Date:\r\n
\r\n
Warehouse:%5d            District:<INPUT
NAME="DI" SIZE=2 maxlength=2>\r\n
\r\n
\r\n
\r\n
\r\n
Customer: <INPUT NAME="CI" SIZE=4
maxlength=4> Cust-Warehouse: <INPUT
NAME="CW" SIZE=5 maxlength=5> 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="HI" 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:%5d
District: %2d\r\n
\r\n
Stock Level Threshold: <INPUT NAME="tl" 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>\
Warehouse:%5d    District:%2d\r\n
\r\n
Stock Level Threshold:<INPUT NAME="tl" SIZE=2
maxlength=2>\r\n
\r\n
low stock:\r\n
</PRE>\r\n
<INPUT
TYPE="submit"></FORM></BODY></HTML>\r\n"

===== tpcweb.h =====

/* -----
   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 luxerr "\
<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 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="%d">\
</BODY></HTML>"


```

```

<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>"

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

===== trans.h =====

/* =====
   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_I_ID[15],
  *OL_QUANTITY[15];
} rte_input_data;

===== Stdafx.cpp =====

// stdafx.cpp : 標準インクルードファイルを含む
ソース ファイル
// tpaql.pch : 生成されるプリコンパイル済ヘ
ッダー
// stdafx.obj : 生成されるプリコンパイル済
タイプ情報

#include "stdafx.h"


```

```

===== tpapl.cpp =====

//
// TPC-C Client Application Program Source
//
// tpapl Extension
//
#include "stdafx.h"
#include "tpapl.h"

#include "tpccis.h"
#include "tpccinf.h"
#include "trans.h"
#include "tpcc_info.h"
// #include "bench2.h"
#include "dbgprt.h"

// HTML-Page Data
#include "tpcweb.h"
#include "tpcinweb.h"
#include "menupage.h"
#include "newpage.h"
#include "paypage.h"
#include "odrpge.h"
#include "delpage.h"
#include "stpage.h"

#include "ATML.H" // TP-BASE include File

#ifdef USE_FML
# include "fml.h"
# include "fldtbl.h" // Create by mkfldhdr commnd.
#endif

char *point;
FILE *envget;
FILE *errfile;

#include "dmy.h" // For debug

static TPINIT *tpinf;
static DWORD TLSsTpnitedKey;
static int ThrTpnit();

//
// 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 )
{
    struct tm tim;
    time_t tt = ( time_t )time;

    tim = *( localtime( &tt ) );

    sprintf( save_p, "%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 )
{
    struct tm tim;
    time_t tt = ( time_t )time;

    tim = *( localtime( &tt ) );

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

/*
Thread 毎の tpinit()を実施する
該当する Thread が既に tpinit()済みであるか
どうかは、 TLS インデックス域にフラグを設
定し
されているかどうかで判断する
tpinit()が未実施の場合、フラグは設定され
ていない。
tpinit()実施後に、フラグを設定する。

Tpinit() is executed with each thread.
The flag in the TLS region is checked, and whether
corresponding thread has executed
function tpinit is judged.
When function tpinit is unexecution, the flag is not
set.
When function tpinit is executed, the flag is set.
*/
static int ThrTpnit()

static int num_tpnits=0;
static int x=1;
static int once=0;
static CRITICAL_SECTION TpCriticalSection;
int lasterr, iRc, TpRc;
int retry = 0;
BOOL Success = FALSE;

#ifdef SCRTEST
return 0;
#endif

// Whether the key data is set is checked.
if(!TlsGetValue(TLSsTpnitedKey)) {

// If the key data is not set
FILE *fp;
fp = fopen("c:\\luxlog\\tpapl.log", "ab");

if (!once)

InitializeCriticalSection(&TpCriticalSection);
once=1;

fprintf(fp, "- Start -----
\n");
}

#ifdef DBPRT
fprintf(fp, "** In ThrTpnit Thread %d * \n",
GetCurrentThreadId());
#endif

fclose(fp);

while ( retry < 10 )

```

```

EnterCriticalSection(&TpCriticalSection);

// Execute tpalloc()
if(tpinf == NULL)

if ((tpinf = ( TPINIT *)tpalloc("TPINIT",
NULL, sizeof(TPINIT))) == NULL)

LeaveCriticalSection(&TpCriticalSection);
TpRc = tperrno;

{
if((C_FLAG &
PRT_ELOG)!= 0){
FILE *fp;
fp =
fopen("c:\\luxlog\\tpapl.log", "ab");
fprintf(fp, " >
ThrTpnit:%d : tpalloc of tpinit failed: %d : %s\r\n",
GetCurrentThreadId(), TpRc, tpsterror(TpRc));
fclose(fp);
}
}
retry++;
continue;
}

tpinf-
>flags|=TPMULTICONTEXTS;
}

// Execute tpinit()
iRc = tpinit(tpinf);
TpRc = tperrno;

// check return code
if (iRc < 0)
// if tpinit abnormal end

LeaveCriticalSection(&TpCriticalSection);
retry++;
lasterr = GetLastError();
}
else
// if tpinit() normal end
Success = TRUE;

LeaveCriticalSection(&TpCriticalSection);
break;
}
Sleep(5); // Relinquish thread
timeslice
} // retry the tpinit if it failed the first time

// if tpinit() abnormal end
if ( Success == FALSE )
{
if((C_FLAG & PRT_ELOG)!=
0){
char ebuf[128];
sprintf(ebuf,
"False : ThrTpnit %d :
Cannot tpinit after %d tries iRc = %d LastErr = %d \n",

```

```

GetCurrentThreadId(),
1, iRc, lasterr);

        FILE *fp;
        fp =
fopen("c:\\luxlog\\tpapl.log", "ab");
        fprintf(fp,"%s\n", ebuf);
        fprintf(fp,"%d\n", TpRc);
        fclose(fp);
    }
    return -1;
}

// if tpinit() normal end
if ( Success == TRUE )

    if ( retry > 0 )
        if((C_FLAG & PRT_ELOG)!=
0){
        char ebuf[128];
        sprintf(ebuf,
"Success : ThrTpInit %d:
Cannot tpinit after %d tries iRc = %d LastErr = %d\n",
GetCurrentThreadId(),
10, iRc, lasterr);
        sprintf(ebuf,
"Success : ThrTpInit
Thread %d Success retry count %d with LastErr = %d
\n",
GetCurrentThreadId(), retry,
lasterr);

        FILE *fp;
        fp =
fopen("c:\\luxlog\\tpapl.log", "ab");
        fprintf(fp,"%s\n", ebuf);
        fclose(fp);
    }

    if
(( iRc=TlsSetValue(TLSiTpInitKey,&x) == 0) { //?
    {
        if((C_FLAG & PRT_ELOG)!=
0){
        FILE *fp;
        fp =
fopen("c:\\luxlog\\tpapl.log", "ab");

        fprintf(fp, " >
ThrTpInit %d : TlsSetValue Failed iRc: %d \n",
GetCurrentThreadId(),
iRc);

        fclose(fp);
    }
    }
}
}
else
// If the key data is set
#ifndef DBORT
FILE *fp;
fp = fopen("c:\\luxlog\\tpapl.log", "ab");
fprintf(fp, "ThrTpInit Thread %d already
tpinited * (%x) \n",
GetCurrentThreadId(), tpinf->flags);
fclose(fp);
#endif
#endif
}

#endif
return 0;
}

/*
check HTML form
*/
int checkHTMLform( char *str, char *buffer)
{
    int length;
    int cnt1;
    int cnt2 = 0;
    int newlength = 0;

    length = strlen( str );

    for (cnt1 = 0; cnt1 < length; cnt1++){

        if ( *(str + cnt1) == '&') {
            *(buffer + cnt2) = '&'; cnt2++;
            *(buffer + cnt2) = 'a'; cnt2++;
            *(buffer + cnt2) = 'm'; cnt2++;
            *(buffer + cnt2) = 'p'; cnt2++;
            *(buffer + cnt2) = ';'; cnt2++;
        }
        else if ( *(str + cnt1) == '<') {
            *(buffer + cnt2) = '&'; cnt2++;
            *(buffer + cnt2) = 'l'; cnt2++;
            *(buffer + cnt2) = 't'; cnt2++;
            *(buffer + cnt2) = ';'; cnt2++;
        }
        else if ( *(str + cnt1) == '>') {
            *(buffer + cnt2) = '&'; cnt2++;
            *(buffer + cnt2) = 'g'; cnt2++;
            *(buffer + cnt2) = 't'; cnt2++;
            *(buffer + cnt2) = ';'; cnt2++;
        }
        else if ( *(str + cnt1) == '"') {
            *(buffer + cnt2) = '&'; cnt2++;
            *(buffer + cnt2) = 'q'; cnt2++;
            *(buffer + cnt2) = 'u'; cnt2++;
            *(buffer + cnt2) = 'a'; cnt2++;
            *(buffer + cnt2) = 't'; cnt2++;
            *(buffer + cnt2) = ';'; cnt2++;
        }
        else {
            *(buffer + cnt2) = *(str + cnt1);
            cnt2++;
        }
    }

    *(buffer + cnt2) = 0;
    return ( strlen ( buffer ) );
}

/*
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 分に満たない場合は空白を詰める */
    /* If not coming up to the specified length then set the
blank. */
    if ( len - cnt > 0 )
        memset ( &str[cnt], ' ', len - cnt);
}

/*
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] = (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] = (num % 10) + '0';
        num /= 10;
    }
}

/*
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);
}

/*
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
*/
#ifdef symfo
void time2str (char *str, char *time)
{
    int year, month, day, hour, min, sec;

#ifdef DBPRT
    fprintf (test_fp, "time2: %s\n", time);
#endif
    sscanf( time, "%d-%d-%d %d:%d:%d",
            &day, &month, &year, &hour, &min, &sec );

    int3str (str, 2, day);
    str[2] = ':';

    int3str (&str[3], 2, month);
    str[5] = ':';

    int3str (&str[6], 4, year);
    str[10] = ':';

    int3str (&str[11], 2, hour);
    str[13] = ':';

    int3str (&str[14], 2, min);
    str[16] = ':';

    int3str (&str[17], 2, sec);
}

#else
void time2str (char *str, char *time)
{
    short mon;
    int year, day, hour, min, sec;
    char month[4];

#ifdef DBPRT
    fprintf (test_fp, "time2: %s\n", time);
#endif
    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] = ':';

    /* 西暦 2 桁対応 */
    if (year >= 70)
        year += 1900;
    else
        year += 2000;

    int3str (&str[6], 4, year);

    str[10] = ':';

    int3str (&str[11], 2, hour);
    str[13] = ':';

    int3str (&str[14], 2, min);
    str[16] = ':';

    int3str (&str[17], 2, sec);
}
#endif

/*
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;
    char *string;

    string = ecvt(num, len-1, &dec, &sign);

    /* dec = 整数部の桁数, sign = 正は 0, 負は
    1, string=小数点を除いた数字が設定される */

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

        /* 整数部の数字でゼロを空白に設定する:
        "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: 整数部の数字を設定する */
        for (; i < (len - 3);
            str[i] = *(string++);
        }
    }
    else
        /* If the integer part is zero ... Exsample:
        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';

        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);
    }
}
```

/*
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]);
}
```

```
#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;

    if(str == 0 || !(x = strlen(str))) return -3;
    if(x > field_len){
        if (strchr (str, '%') != 0) /* 98.8.3 :特
        殊記号文字対応 */
            return -2;
        else
            return -1;
    }
    else{
        for( ; x ; 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;

    if(str == 0 || !(x = strlen(str))) return -3;
    if(x > field_len){
        if (strchr (str, '%') != 0) /* 98.8.3 :特
        殊記号文字対応 */
            return -2;
        else
            return -1;
    }
    else {
        for( ; x ; x--){
            if (!numcheck(str[x-1]))
                return -2;
        }
    }
    x = atoi(str);
    return (short)x;
}
```

/*
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;

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

    if(x > field_len) {
        if (strchr (str, '%') != 0) /* 98.8.3 :特
        殊記号文字対応 */
            return -2;
        else
            return -1;
    }
    else {
        for( ; x ; x--){
            if (!alpcheck(str[x-1]))
                return -2;
        }
    }
    return 1;
}
```

/*
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;
    char NUM[7];

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

    if(str == 0 || !(x = strlen(str))) return -3;
    len = x;

    if(x > field_len){
        if (strchr (str, '%') != 0) /* 98.8.3 :特
        殊記号文字対応 */
            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.: 小数点以下の数字が無いま
    たは 1 桁の場合の処理 */
    for ( ; fcnt > 0 ; fcnt--){
        NUM[cnt++] = '0';
    }
}
else if ( pointf == 0 ) {
    /* There is no decimal point.: 小数点がな
    い場合の処理 */
    NUM[cnt++] = '0'; NUM[cnt++] = '0';
}

NUM[cnt] = 0;

return (atoi(NUM));
}

/*
struct_init :
init_ptrs sets everything in the annoyingly long
raw_form_data structure
to zero.
*/

void struct_init (rte_input_data *in_data) {

    int cnt = 0;

    in_data->button = 0;
    in_data->cookie = 0;
    in_data->form = 0;
    in_data->O_CARRIER_ID = 0;
    in_data->threshold = 0;
    in_data->D_ID = 0;
    in_data->C_ID = 0;
    in_data->C_W_ID = 0;
    in_data->C_D_ID = 0;
    in_data->C_LAST = 0;
    in_data->H_AMOUNT = 0;

    for (cnt = 0; cnt < 15; cnt++){
        in_data->OL_SUPPLY_W_ID[cnt] = 0;

    for (cnt = 0; cnt < 15; cnt++){
        in_data->OL_I_ID[cnt] = 0;

    for (cnt = 0; cnt < 15; cnt++){

```

```

        in_data->OL_QUANTITY[cnt] = 0;
    }
}

/*
para_split :

文字列 (QueryString)の中から指定されたデリ
ミタのアドレスをサーチする

復帰値: デリミタの次のアドレス.その時、
デリミタを NULL に変換する
デリミタが無ければ、NULL を返す

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;

    /* The address of the delimitation character is
    calculated */
    /* 区切り文字のアドレスを求める */
    if ((point = strchr (para, delimita)) == NULL)
        return (char *)0;

    /* The delimitation character is replaced with NULL*/
    *point = '\0'; /* 区切り文字を NULL に
    変換する */

    /* The first position of the analyzed variable is
    returned.*/
    return (point + 1); /* データの先頭アドレス
    を返す */
}

/*
anly_para :
QueryString から変数名と設定値を分割する
それぞれの変数名に該当する設定値のアド
レスを作業用の構造体に設定する

復帰値: 処理を要求した端末番号を返す

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 anly_para (char *para, rte_input_data *in_data) {
    char *val, *rest;
    int num = 0;

    if(!para) 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 '0':
                                if (para[3] >= 0x31 &&
                                para[3] <= 0x39){
                                    num = (int)(para[3]
                                    - 0x30);

                                    if (strlen(val) != 0 )
                                        in_data-
                                        >OL_SUPPLY_W_ID[num - 1] = val;
                                }
                                break;

```

<pre> 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[num - 1] = val; } break; } break; case 'I': switch(para[2]) { case '0': if (para[3] >= 0x31 && para[3] <= 0x39){ num = (int)(para[3] - 0x30); if (strlen(val) != 0) in_data- >OL_I_ID[num - 1] = val; } break; } break; case '1': if (para[3] >= 0x30 && para[3] <= 0x35){ num = (int)(para[3] - 0x30) + 10; if (strlen(val) != 0) in_data- >OL_I_ID[num - 1] = val; } break; } break; case 'Q': switch(para[2]) { case '0': if (para[3] >= 0x31 && para[3] <= 0x39){ num = (int)(para[3] - 0x30); if (strlen(val) != 0) in_data- >OL_QUANTITY[num - 1] = val; } break; } break; case '1': if (para[3] >= 0x30 && para[3] <= 0x35){ num = (int)(para[3] - 0x30) + 10; if (strlen(val) != 0) in_data- >OL_QUANTITY[num - 1] = val; } break; } break; } } } </pre>	<pre> para = rest; } if (in_data->cookie != 0) return(atoi(in_data->cookie)); else return(0); } /* Error message list : these are notified from CLINET to RTE */ /* 98.8.3 : メッセージ内容の変更 */ char errstrings[23][166] = { "The function you selected doesn't exist.\r\n" "Don't enter URLs manually!\r\n%s", /* 0 */ "You seem to have responded to a form that doesn't exist.\r\n" "Don't enter URLs manually!\r\n%s", /* 1 */ "The District ID you entered isn't valid.\r\n%s\r\n" "It must be an integer in the range 1 to 10.\r\n", /* 2 */ "The threshold value you entered isn't valid.\r\n%s\r\n" "It must be an integer in the range 10 to 20.\r\n", /* 3 */ "The terminal number you entered isn't valid.\r\n%s\r\n" "It must be an integer in the range 1 to %d.\r\n", /* 4 */ "The Carrier ID you entered isn't valid.\r\n%s\r\n" "It must be an integer in the range 1 to 10.\r\n", /* 5 */ "The Customer ID you entered isn't valid.\r\n%s\r\n" "It must be an integer of 4 or fewer digits.\r\n", // "It must be an integer in the 1 to 3000.\r\n", /* 6 */ "The Customer Last Name you entered isn't valid.\r\n%s\r\n" "It must be a string shorter than 16 characters.\r\n", /* 7 */ "The Payment Amount you entered isn't valid.\r\n%s\r\n" "It must be a dollar amount, without the dollar sign," " between \$1.00 and \$5000.00.\r\n", /* 8 */ "The Customer Warehouse ID you entered isn't valid.\r\n%s\r\n" "It must be an integer in the range 1 to %d.\r\n", /* 9 */ "The Customer District ID you entered isn't valid.\r\n%s\r\n" "It must be an integer in the 1 to 10.\r\n", /* 10 */ "You must enter either a Customer ID or a Customer Last Name.\r\n" "You left both fields blank.\r\n%s", /* 11 */ </pre>	<pre> "The Warehouse ID you entered isn't valid.\r\n%s\r\n" "It must be an integer in the range 1 to %d.\r\n", /* 12 */ "On entry line %d, the data you entered for the %s field isn't valid.\r\n%s\r\n", /* 13 */ "Supply Warehouse ID", /* 14 */ "Item ID", /* 15 */ "Quantity", /* 16 */ "Your entry was outside the range.", /* 17 */ "You didn't enter 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 から指定された処理データに異常があ った場合のエラーフォームを作成する 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 : メッセージ内容変更に伴う修正 */ 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; if(err_no >= 13) { /* OrderLine Data(Neworder) is Abnormal */ switch(err_inf) { case -5: /* S_W_ID data is abnormal */ sprintf(errmsg,errstrings[13],err_no- 12,errstrings[14],errstrings[20]); sub_inf2 = maxwh; break; case -8: /* S_W_ID data is uninput */ sprintf(errmsg,errstrings[13],err_no- 12,errstrings[14],errstrings[18]); sub_inf2 = maxwh; break; </pre>
---	---	---


```

        case -15:          /* S_W_ID data is
outside range */
        sprintf(errmsg,errstrings[13],err_no-
12,errstrings[14],errstrings[17]);
        sub_inf2 = maxwh;
        break;

        case -1:          /* I_ID data is
uninput */
        sprintf(errmsg,errstrings[13],err_no-
12,errstrings[15],errstrings[18]);
        sub_inf2 = 100000;
        break;

        case -6:          /* I_ID data is
abnormal */
        sprintf(errmsg,errstrings[13],err_no-
12,errstrings[15],errstrings[20]);
        sub_inf2 = 100000;
        break;

        case -16:         /* I_ID data is
outside range */
        sprintf(errmsg,errstrings[13],err_no-
12,errstrings[15],errstrings[17]);
        sub_inf2 = 100000;
        break;

        case -7:          /* Quantity data is
abnormal */
        sprintf(errmsg,errstrings[13],err_no-
12,errstrings[16],errstrings[20]);
        sub_inf2 = 10;
        break;

        case -2:          /* Quantity data is
uninput */
        sprintf(errmsg,errstrings[13],err_no-
12,errstrings[16],errstrings[18]);
        sub_inf2 = 10;
        break;

        case -17:         /* Quantity data is
outside range */
        sprintf(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 */
                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(fprintf(test_fp, "This Transaction is parameter
ERROR\n"));
        return 0;
    }

    /*
    set_tuxerr :
    this function make error message of the TP-
application program.
*/
int set_tuxerr (char *page, char *err_inf, int cookie) {
#ifdef SCRTEST

// tpterm ();

#endif

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

    return 0;
}

/*
set_oraerr :
this function make error message of the Oracle
application program.
*/
int set_oraerr (char *page, char *err_inf, int cookie ) {
#ifdef Symfo
    sprintf(page, symfoerr, err_inf, SOPATH, cookie );
#else
    sprintf(page, oraerr, err_inf, SOPATH, cookie );
#endif

    return 0;
}

/*
set_symfoerr :
this function make error message of the Oracle
application program.
*/
void set_symfoerr (char *page, int errorpos , int
sqlstate , int cookie )
{
    char *sqlfunc[4] ={"Failure on insert of a new
record",
                    "Failure on select of an existing record",
                    "Failure on update of an existing record",
                    "Failure to delete an existing record"};

    char buf[80];
    int pos;

    if ( errorpos == 0 ){
        sprintf( buf,"SQLERROR occured ...
(SQLSTATE : %05d)",sqlstate );
    }
    else{
        pos = errorpos / 100 ;
        sprintf( buf,"%s ...
(SQLSTATE : %05d)",sqlfunc[pos-1],sqlstate ) ;
    }

    sprintf(page, symfoerr, buf, SOPATH, cookie );
}

#endif USE_FML
// アプリケーション番号を設定する処理
void term_id ( int cookie ){

    int transaction_id;
    int num;
    int loop;
}

```

<pre>// トランザクション毎のアプリケーション // 番号を設定する for (loop = 0; loop < 5; loop++){ num = Term_Base + TRN_ID[loop]; transaction_id = 1; while (num <= cookie) { num += TRN_ID[loop]; transaction_id++; } srv->m_tcctx[cookie - Term_Base].trn_id[loop] = (char)transaction_id; } // DBGP(fprintf(test_fp," svn=%d ott-cnt=%d, num=%d)\n", svnum, loop, num)); return; } #endif /* ----- --- The function number of the TP application program which requests processing is acquired. (Get TPCCxx name in tpccsvr.ott) ----- */ int getsvnam (int cookie){ int svnum = 1; int num; int loop = 0; num = 0; if (maxconnect < cookie cookie == 0){ DBGP(fprintf (test_fp, "Term NO(%d) is not support!\n", cookie)); return(-1); } /* COMMENT OUT : 98.01.13 ===== = for (cnt = 1; cnt < tpc_area->clnt_num; cnt++){ if (cookie < tpc_area->clnt[cnt][0]){ break; } } cnt--; num = tpc_area->clnt[cnt][0]; DBGP(fprintf (test_fp, "cookie=%d (client=%d, max=%d, num=%d ->" \ cookie, cnt, tpc_area->clnt[cnt][0], num)); for (loop = 0; loop < tpc_area->clnt[cnt][1]; loop++){ num += tpc_area->ott[cnt][loop]; if (num > cookie){ break; } else{ svnum++; </pre>	<pre>} } } ===== */ DBGP(fprintf(test_fp," svn=%d ott-cnt=%d, num=%d)\n", svnum, loop, num)); return (svnum); } /* 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){ int w_id, d_id, user_id; char flag=0; // make w_id, d_id w_id = (cookie - 1)/10 + 1; d_id = (cookie - 1)%10 + 1; #ifndef USE_FML // 従来のアプリケーション番号を設定す // る処理 // term_id (cookie); #endif // execute tpalloc ... : set w_id, d_id, and trans_buffer pointer if ((user_id = srv->Terminet (w_id, d_id, cookie)) == -1){ // tpalloc terminated abnormally sprintf(s_buf, luxierr, "tpalloc", SOPATH); return -1; } sprintf(s_buf, h_menu, SOPATH, cookie); return 0; } /* ----- --- The w_id, d_id are acquired. and call the function getsvnam ----- */ int idget (char *s_buf, int cookie){ int w_id, d_id, user_id, sarvice_name; /* make w_id, d_id */ w_id = (cookie - 1)/10 + 1; d_id = (cookie - 1)%10 + 1; if ((sarvice_name = getsvnam (cookie)) < 0){ /*The terminal number exceeded the maximum value */ sprintf(s_buf, noconnt, maxconnect, cookie); return -1; } } sprintf(s_buf, h_menu, SOPATH, cookie); 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 : 入力データ範囲チェック強化 (-15, -16, -17 追加: outside range) ----- ----- */ int chk_NOdata (newstruct *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){ /* Find last order line : 1行分の入力データ無 し */ /* comment out : 98.08.25 bp->newin.ol_i_id[cnt] = 0; bp->ol_quantity[cnt] = 0; bp->ol_supply_w_id[cnt] = 0; return 0; */ return 16; /* change return code */ } if(in_data->OL_SUPPLY_W_ID[cnt] != 0){ if((bp->newin.ol_supply_w_id[svcnt] = /* 99.8.19 w_id 5 keta */ str2int (in_data- >OL_SUPPLY_W_ID[cnt], 5) < 1) /* 99.6.8 no_range */ return -5; /* w_id abnormal */ /* if((bp->ol_supply_w_id[svcnt] = str2int (in_data->OL_SUPPLY_W_ID[cnt], 4) < 1 bp->ol_supply_w_id[svcnt] > maxwh) { if (bp->ol_supply_w_id[svcnt] < 0) return -5; /* w_id abnormal * else return -15; /* outside range * } } */ /* 99.6.8 */ </pre>
--	---

```

    }
else {
    flag |= SUPPLY_NG;
}

if(in_data->OL_ID[cnt] != 0){

    if((bp->newin.ol_id[svcnt] =
        str2int (in_data->OL_ID[cnt], 6)) <
0) /* 99.6.8 no_range */
// str2int (in_data->OL_ID[cnt], 6) < -
1) /* 99.12.17 roll back*/
return -6; /* i_id abnormal
value */

/* sv-apl ローパックデータを作成する用
99.12.20 */
else if (bp->newin.ol_id[svcnt] == 0)
bp->newin.ol_id[svcnt] = -1;

/* if((bp->ol_id[svcnt] =
str2int (in_data->OL_ID[cnt], 6) < 0
||
bp->ol_id[svcnt] > 100000) {

if (bp->ol_id[svcnt] < 0)
return -6; /* i_id abnormal
value *
else
return -16; /* outside range *
}
else if (bp->ol_id[svcnt] == 0){
/* Convert 0 into -1. if this function set
0 then the TP application
send the return code of abnormal
end : Oracle use only *
#ifdef Oracle
bp->ol_id[cnt] = -1;
#endif
}
*/ /* 99.6.8 */
}
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 */

DBGR(sprintf(test_fp, "neworder ol data check
flag=%d\n", flag));

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,
newstruct *bp,rte_input_data
*in_data)
{
#ifdef Symfo
/* 99.8.19 w_id 5 keta */
int2str((s_work + OF + newp[11]), 5, (int)bp-
>newin.ol_supply_w_id[cnt]);

int2str((s_work + OF + newp[12]), 6, (int)bp-
>newin.ol_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)((double)bp->newout.i_price[cnt] /
(double)100); // check
dec2str((s_work + OF + newp[18]), 7,
(double)((double)bp-
>newout.ol_amount[cnt] / (double)100); // check

return 0;
#else
if(!bp->newin.ol_id[cnt]) {
alp2str ((s_work + OF + newp[11]), 78, " ");
return -1;
}
else {
/* 99.8.19 w_id 5 keta */
int2str((s_work + OF + newp[11]), 5,
(int)bp->newin.ol_supply_w_id[cnt]);
/* 99.12.17 roll back */
if (bp->newin.ol_id[cnt] == -1)

bp->newin.ol_id[cnt] = 0;
// alp2str((s_work + OF + newp[12]), 6,
in_data->OL_ID[cnt]);
int2str((s_work + OF + newp[12]), 6, bp-
>newin.ol_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;
}
}
#endif

neworder : this function processes the NewOrder
transaction.
-----
*/

int neworder (char *s_buf, rte_input_data *in_data, int
cookie)
{
newstruct *bp;
// neworder_trans *bp;
long olen;
int user_id, i;
int ol_cnt, cnt, rtn;

char S_WORK[WORK_S];
char TPAPL[12];
char time_data[64];
double entry;

#ifdef USE_FML
newstruct tbuf;
int w_id;

bp = &tbuf;
user_id = cookie - Term_Base;
memset (bp, 0, sizeof(newstruct));
/* 98.7.29 */
#else
user_id = cookie - Term_Base;
bp = ( newstruct * )srv->m_tctx[user_id].trans_b;
#endif

bp->retval = TX_NEWORDER;
bp->newout.terror = 0;

#ifdef USE_FML
sprintf (TPAPL, "TPCC"); // TP-Base
Application Name (View)
#else
sprintf (TPAPL, "NEWORDER");

```

```

// sprintf (TPAPL, "TPCC%d", (int)srv-
>m_tcctxt[user_id].trn_id[0]);
#endif

if ((rtn = ThrTplnit()) < 0){
    sprintf(S_WORK, "Thread init abort NEW (%d) \n",
rtn);
    set_or Kerr(s_buf, S_WORK, cookie);
    return (-1);
}

/* ----- check the
Input data */
bp->newin.w_id = (short)srv->m_tcctxt[user_id].w_id;

/* 99.6.8 no_range */
// if((bp->d_id = strtoint (in_data->D_ID, 2)) < 1 || bp-
>d_id > 10)
    if((bp->newin.d_id = strtoint (in_data->D_ID, 2)) < 1)
        return set_errpage(s_buf, cookie, 2, (int)bp-
>newin.d_id, 0, 0);

/* 98.8.3 : 範囲チェック強化 */
// if((bp->c_id = strtoint (in_data->C_ID, 4)) < 1 || bp-
>c_id > 3000)
    if((bp->newin.c_id = strtoint (in_data->C_ID, 4)) < 0)
        return set_errpage(s_buf, cookie, 6, bp-
>newin.c_id, 0, 0);

/* 98.8.25 */
ol_cnt = 0;
for (cnt = 0; cnt < 15; cnt++){

    if ((rtn = chk_NOdata (bp, cnt, in_data, ol_cnt)) <
0){
        return set_errpage(s_buf, cookie, 13 + cnt, rtn, 0,
0);
    }
}

else if (rtn == 0 && ol_cnt == 0){
    return set_errpage(s_buf, cookie, 13 + cnt, -8, 0,
0);
}

else if (rtn == 0){
    break;
}

*/
else if (rtn == 1){
    ol_cnt++;
}

// else // Order Line data is normal: rtn == 1
// ol_cnt++;
// }

}

/* nothing order line data */
if (cnt >= 15 && ol_cnt == 0)
    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;

DBGRR(new_dsp(in_data, bp, srv-
>m_tcctxt[user_id].w_id, 0, ol_cnt));

/* ----- Execute NewOrder
transaction */
resend_neworder:

#ifndef SCRTTEST

    DBGRR(tsp(0));

#define USE_FML
    w_id = bp->newin.w_id;

    rtn = Fchg( (Fbfr *)srv->m_tcctxt[user_id].trans_b,
FML_TERM, 0, (char *)&w_id, 0);
    rtn = Fchg( (Fbfr *)srv->m_tcctxt[user_id].trans_b,
FML_TRAN, 0, (char *)&bp->retval, 0);
    rtn = Fchg( (Fbfr *)srv->m_tcctxt[user_id].trans_b,
FML_DATA, 0, (char *)bp,
        ( FLDLEN )sizeof( newstruct ));

    if ( tpcall( TPAPL, ( char * )srv-
>m_tcctxt[user_id].trans_b, 0,
        ( char ** )&srv->m_tcctxt[user_id].trans_b,
        &olen, 0|TPNOTIME ) == -1 ){

        if ( tperno == TPESVCFAIL ) {
            sprintf(S_WORK, "Oracle failed to process
NewOrder Transaction.\n"
                "tperno = %d svc = '%s' d_id = %d c_id = %d
lines = %d\n",
                tperno, TPAPL, bp->newin.d_id, bp-
>newin.c_id, ol_cnt );

            set_or Kerr(s_buf, S_WORK, cookie);
            return (-1);
        }
    }

    set_tuxerr(s_buf, S_WORK, cookie);
    return (-1);
}
DBGRR(tsp(1));

bp = ( newstruct * )srv->m_tcctxt[user_id].trans_b;
#endif

#else
    dummy_neworder( bp );
#endif

DBGRR(new_dsp(in_data, bp, srv-
>m_tcctxt[user_id].w_id, 1, 1));

    sprintf(S_WORK, h_new2);

/* 99.8.19 w_id 5 keta */
int2str ((S_WORK + newp[0]), 5, (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);

// if ( bp->newout.terror == 1 || bp->newout.terror ==
2 ){ // Normal End
    if ( bp->newout.terror == NOERR || bp-
>newout.terror == 2 ||
        bp->newout.terror == 1 ){ // Normal End

        cnt = bp->newout.o_ol_cnt;

#ifdef Symfo
        entry = str2dbl(bp->newout.o_entry_d, 20);
        convert_time (time_data, entry);
        time2str((S_WORK + newp[2]), time_data);
        dec2str ((S_WORK + newp[6]), 5,
            (double)((double)bp-
>newout.c_discount) / (double)100.0 );

```

```

int2str ((S_WORK + newp[8]),2,(int)bp-
>newout.o_ol_cnt);

dec2str ((S_WORK + newp[9]),5,
(double)((double)(bp->newout.w_tax)
/ (double)100.0));
dec2str ((S_WORK + newp[10]),5,
(double)((double)(bp->newout.d_tax)
/ (double)100.0));
#else
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));
#endif

for (i = 0; i < cnt; i++) {
setNOdata (S_WORK, 0x50*i, i, bp, in_data);
}

#ifndef Symfo
if (bp->newout.terror == 2)
alp2str ((S_WORK + newp[19]), 24,
"Item number is not valid");

dec2str ((S_WORK + newp[20]), 8,
(double)((double)(bp-
>newout.total_amount) / (double)100.0));
#else
/* "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
#endif

}

#ifndef SCRTEST
else{

#ifndef Symfo
if ( bp->newout.terror == IRRECERR){
sprintf (S_WORK, "Irrecoverable error in
NewOrder\n");
set_luxerr (s_buf, S_WORK, cookie);
return (-1);
}
else{
goto resend_neworder; /* Retry
NewOrder transaction */
}
}
#else
set_symfoerr (s_buf, bp->errorpos, bp-
>sqlstate, cookie);
return (-1);
}
#endif
}
#endif

/* ----- The execution result data notified RTE is make
by the HTML form */

sprintf(s_buf, h_new1);
strcat (s_buf, S_WORK);

sprintf(S_WORK, h_new3, SOPATH, cookie);
strcat (s_buf, S_WORK);

return (0);
}

/*-----
-
payment : this function processes the Payment
transaction.
-----
*/
int payment (char *s_buf, rte_input_data *in_data, int
cookie)
{
paystruct *bp;
int i, user_id, rtn;
long olen;

float h_amount; /* For work */

char c_id_flag = NG;
char S_WORK[WORK_S];
char TPAPL[12];
char time_data[64];

char buffer[128]; /* check HTML form */
char buffer2[128];
char buffer3[512];
int newlength;
double date, since;

#ifndef USE_FML
paystruct tbuf;
int w_id;
bp = &tbuf;
user_id = cookie - Term_Base;
memset (bp, 0, sizeof(paystruct)); /*
98.7.29 */
#else
user_id = cookie - Term_Base;
bp = ( paystruct * )srv->m_tcctx[user_id].trans_b;
#endif

bp->retval = TX_PAYMENT;
bp->payout.terror = 0;

#ifndef USE_FML
sprintf (TPAPL, "TPCC"); // TP-Base
Application Name (View)
#else
sprintf (TPAPL, "PAYMENT");
// sprintf (TPAPL, "TPCC%d", (int)srv-
>m_tcctx[user_id].trn_id[1]);
#endif

if ((rtn = ThrTplnit()) < 0){
sprintf( S_WORK, "Thread init abort PAY (%d) \n",
rtn);
set_or Kerr(s_buf, S_WORK, cookie);
return (-1);
}

/* ----- check the
Input data */
bp->payin.w_id = (short)srv->m_tcctx[user_id].w_id;

/* 99.6.8 no_range */
/* check d_id data */
// if((bp->d_id = str2short (in_data->D_ID, 2)) < 1 ||
bp->d_id > 10)
if((bp->payin.d_id = str2short (in_data->D_ID, 2) <
1)
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 < 1 || bp->payin.c_id > 3000){
/* 98.8.3 :範囲チェック強化 */
if (bp->payin.c_id < 0) {
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){
// return set_errpage(s_buf, cookie, 7, rtn, 22, 0);
/* 99.6.8 no_range コメントアウト */
c_id_flag = OK; /* 99.6.8 no_range
追加 */
}
else{
if ( rtn == 0 || *(in_data->C_LAST) == '\0' ) {
bp->payin.bylastname = 0;
/* Oracle use only */
} 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)
return set_errpage(s_buf, cookie, 11, -4, 0,
0);

/* 99.6.8 no_range */
/* check c_w_id data */
/* 99.8.19 w_id 5 keta */
// if((bp->payin.c_w_id = str2short (in_data->C_W_ID,
5)) < 1)
if((bp->payin.c_w_id = str2dbl (in_data->C_W_ID, 5)
/ 100) < 1)

```

```

// if((bp->c_w_id = str2short (in_data->C_W_ID, 4)) <
1 || bp->c_w_id > maxwh)
    return set_errpage(s_buf, cookie, 9, (int)bp-
>payin.c_w_id, 0, maxwh);

/* 99.6.8 no_range */
/* check c_d_id data */
if((bp->payin.c_d_id = str2short (in_data->C_D_ID,
2)) < 1)
// if((bp->payin.c_d_id = str2short (in_data->C_D_ID,
2)) < 1 || bp->c_d_id > 10)
    return set_errpage(s_buf, cookie, 10, (int)bp-
>payin.c_d_id, 0, 0);

/* check h_amount data :
str2dbl do hundredfold of the H_AMOUNT. The
pupose of if is to prosess
H_AMOUNT by the integer : 98.8.3 update */

if((bp->payin.h_amount = (long)str2dbl (in_data-
>H_AMOUNT, 7)) < 100 ||
bp->payin.h_amount > 500000)
    return set_errpage(s_buf, cookie, 8, (int)bp-
>payin.h_amount, 0, 0);

DBGP(pay_dsp(in_data, bp, srv-
>m_tcctxt[user_id].w_id, 0));

/* ----- Execute Payment
transaction */
resend_payment:

#ifdef SCRTEST

    DBGPR(tsp(0));

#endif USE_FML
    w_id = bp->payin.w_id;

    rtn = Fchg( (Fbfr *)srv->m_tcctxt[user_id].trans_b,
FML_TERM, 0, (char *)&w_id, 0);
    rtn = Fchg( (Fbfr *)srv->m_tcctxt[user_id].trans_b,
FML_TRAN, 0, (char *)&bp->retval, 0);
    rtn = Fchg( (Fbfr *)srv->m_tcctxt[user_id].trans_b,
FML_DATA, 0, (char *)bp,
(FLDLN)sizeof( paystruct ));

    if ( tpcall( TPAPL, ( char *)srv-
>m_tcctxt[user_id].trans_b, 0,
(char **)&srv->m_tcctxt[user_id].trans_b,
&olen,0|TPNOTIME ) == -1){

        if ( tperno == TPESVCFAIL ) {
            sprintf( S_WORK, "Oracle failed to process
Payment Transaction.\n"
"tperno = %d svc = '%s' d_id = %d c_id = %d
c_last = %s\n"
" c_w_id = %d, c_d_id = %d,
h_amount = %d\n",
tperno, TPAPL,
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 );

            set_orherr( s_buf, S_WORK, cookie );
            return (-1);
        }

        sprintf( S_WORK, "tpcall failed in Payment:
tperno = %d\n"
" svc = '%s' d_id = %d c_id = %d
c_last = %s\n"
" c_w_id = %d c_d_id = %d
h_amount = %f\n",
tperno, TPAPL, 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 );

        set_tuxerr( s_buf, S_WORK, cookie);
        return (-1);
    }
    DBGPR(tsp(1));
    bp = ( paystruct * )srv->m_tcctxt[user_id].trans_b;
#endif

return (-1);
}

sprintf( S_WORK, "tpcall failed in Payment:
tperno = %d\n"
" svc = '%s' d_id = %d c_id = %d
c_last = %s\n"
" c_w_id = %d c_d_id = %d
h_amount = %f\n",
tperno, TPAPL, 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 );

set_tuxerr( s_buf, S_WORK, cookie);
return (-1);
}

DBGPR(tsp(1));
tbuf = *( ( Fstruct * )Ffind( (Fbfr *)srv-
>m_tcctxt[user_id].trans_b, FML_DATA, 0, NULL ));
bp = &tbuf;

#else

    if ( tpcall( TPAPL,(char *)srv-
>m_tcctxt[user_id].trans_b,
sizeof(paystruct),( char **)&srv-
>m_tcctxt[user_id].trans_b,
&olen,0|TPNOTIME ) == -1 )

        if ( tperno == TPESVCFAIL ) {
            sprintf( S_WORK, "Oracle failed to process
Payment Transaction.\n"
"tperno = %d svc = '%s' d_id = %d c_id = %d
c_last = %s\n"
" c_w_id = %d, c_d_id = %d,
h_amount = %d\n",
tperno, TPAPL,
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 );

            set_orherr( s_buf, S_WORK, cookie );
            return (-1);
        }

        sprintf( S_WORK, "tpcall failed in Payment:
tperno = %d\n"
" svc = '%s' d_id = %d c_id = %d
c_last = %s\n"
" c_w_id = %d c_d_id = %d
h_amount = %f\n",
tperno, TPAPL, 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 );

        set_tuxerr( s_buf, S_WORK, cookie);
        return (-1);
    }
    DBGPR(tsp(1));
    bp = ( paystruct * )srv->m_tcctxt[user_id].trans_b;
#endif

/* ----- Check the execution
result */
// if (bp->payout.terror != 1){
if (bp->payout.terror != NOERR || bp->payout.terror
== 1){

#ifdef Symfo
    if ( bp->payout.terror == IRRECERR ){
        sprintf( S_WORK, "Irrecoverable error in
Payment\n" );
        set_tuxerr( s_buf, S_WORK, cookie);
        return (-1);
    }
    goto resend_payment: /* TP_base busy.
Try again */
#else
    set_symfoerr( s_buf, bp->errorpos, bp-
>sqlstate, cookie );
    return (-1);
#endif
}

#else
    dummy_payment( bp );
#endif

    DBGPR(pay_dsp(in_data, bp, srv-
>m_tcctxt[user_id].w_id, 1));

    sprintf( S_WORK, h_pay2);

#ifdef Symfo
    date = str2dbl(bp->payout.h_date,20);
    convert_time (time_data, date);
    time2str ((S_WORK + payp[0]), time_data);
#else
    time2str ((S_WORK + payp[0]), bp-
>payout.h_date );
#endif

/* 99.8.19 w_id 5 keta */
int2str ((S_WORK + payp[1]), 5, (int)bp->payin.w_id);
int2str ((S_WORK + payp[2]), 2, (int)bp->payin.d_id);
/*
alp2str ((S_WORK + payp[3]), 20, bp->w_street_1);
alp2str ((S_WORK + payp[4]), 20, bp->d_street_1);
alp2str ((S_WORK + payp[5]), 20, bp->w_street_2);
alp2str ((S_WORK + payp[6]), 20, bp->d_street_2);
*/

// check HTML form

alp2str (&buffer2[0], 20, bp->payout.w_street_1);
buffer2[20] = 0;
newlength = checkHTMLform ( &buffer2[0],
&buffer[0]);
strcpy (&buffer3[0], &buffer[0]);
strcat (buffer3, " ");

alp2str (buffer2, 20, bp->payout.d_street_1);
buffer2[20] = 0;
newlength = checkHTMLform ( &buffer2[0],
&buffer[0]);
strcat (buffer3, &buffer[0]);
strcat (buffer3, "\n\n");

alp2str (buffer2, 20, bp->payout.w_street_2);
buffer2[20] = 0;
newlength = checkHTMLform ( &buffer2[0],
&buffer[0]);

```



```

    strcpy(S_WORK, buffer3);
/* ----- The execution result data notified RTE is make
by the HTML form */

    sprintf(s_buf, h_pay1); /* set Header Data */
    strcpy(s_buf, S_WORK); /* set Result Data */

    sprintf(S_WORK, h_pay3, SOPATH, cookie); /* set
Tailer Data */
    strcpy(s_buf, S_WORK);

    return (0);
}

/*-----
orderstatus : this function processes the Orderstatus
transaction
-----*/
int orderstatus (char *s_buf, rte_input_data *in_data, int
cookie)
{
    ordstruct *bp;
    long olen;
    int i, user_id, rtn;

    char S_WORK[WORK_S];
    char TPAPL[12];
    char c_id_flag = NG;
    char time_data[64], time_data0[64][64];
    double deliver[15], entry;

#ifdef USE_FML
    ordstruct tbuf;
    int w_id;
    bp = &tbuf;
    user_id = cookie - Term_Base;
    memset (bp, 0, sizeof(ordstruct)); /*
98.8.29 */
#else
    user_id = cookie - Term_Base;
    bp = (ordstruct *)srv->m_tcctxt[user_id].trans_b;
#endif

    bp->retval = TX_ORDERSTAT;
    bp->ordout.terror = 0;

#ifdef USE_FML
    sprintf (TPAPL, "TPCC"); /* TP-Base
Application Name (View)
#else
    sprintf (TPAPL, "ORDERSTATUS");
// sprintf (TPAPL, "TPCC%d", (int)srv-
>m_tcctxt[user_id].trn_id[2]);
#endif

    if ((rtn = ThrTplnit()) < 0){
        sprintf(S_WORK, "Thread init abort ODR (%d) \n",
rtn);
        set_oraerr(s_buf, S_WORK, cookie);
        return (-1);
    }

/* ----- check the
Input data */
    bp->ordin.w_id = (short)srv->m_tcctxt[user_id].w_id;

/* 99.6.8 no_range */

```

```

/* check d_id data */
    if ((bp->ordin.d_id = str2short (in_data->D_ID, 2)) <
1)
// if ((bp->d_id = str2short (in_data->D_ID, 2)) < 1 ||
bp->d_id > 10)
        return set_errpage(s_buf, cookie, 2, (int)bp-
>ordin.d_id, 0, 0);

/* check c_id data : 98.8.3 範囲チェック強化 */
    if ((bp->ordin.c_id = str2int(in_data->C_ID, 4)) != -3){

        DBGP(fprintf (test_fp, "check c_id=%x\n",
bp->ordin.c_id, 0, 0));

// if (bp->c_id < 1 || bp->c_id > 3000){ /* 範
囲チェック強化 */
    if (bp->ordin.c_id < 0){
        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){
// return set_errpage(s_buf, cookie, 7, rtn, 22, 0);
/* 99.6.8 no_range コメントアウト */
    c_id_flag = OK; /* 99.6.8
no_range 追加 */
}
else{
    if (rtn == 0 || *(in_data->C_LAST) == '\0'){
        bp->ordin.bylastname = 0;
/* Oracle use only */
    } 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)
        return set_errpage(s_buf, cookie, 11, -4, 0, 0);

    DBGP(oder_dsp ( in_data, bp, srv-
>m_tcctxt[user_id].w_id, 0));

/* ----- Execute Orderstatus
transaction */
    resend_orderstatus;

#ifdef SCRTST
    DBGR(tsp(0));
#endif

#ifdef USE_FML
    w_id = bp->ordin.w_id;

    rtn = Fchg( (Fbfr *)srv->m_tcctxt[user_id].trans_b,
FML_TERM, 0, (char *)&w_id, 0);

```

```

    rtn = Fchg( (Fbfr *)srv->m_tcctxt[user_id].trans_b,
FML_TRAN, 0, (char *)&bp->retval, 0);
    rtn = Fchg( (Fbfr *)srv->m_tcctxt[user_id].trans_b,
FML_DATA, 0, (char *)bp,
(FLDLEN)sizeof( ordstruct ));

    if (tpcall( TPAPL, ( char *)srv-
>m_tcctxt[user_id].trans_b, 0,
(char **)&srv->m_tcctxt[user_id].trans_b,
&olen, 0|TPNOTIME) == -1){

        if ( tpermo == TPESVCFAIL ) {
            sprintf( S_WORK, "Oracle failed to process
NewOrder Transaction.\n"
                "tpermo = %d svc = '%s' d_id = %d c_id = %d
c_last = '%s'\n",
                tpermo, TPAPL, bp->ordin.d_id, bp-
>ordin.c_id,
                bp->ordin.c_last);

            set_oraerr( s_buf, S_WORK, cookie );
            return (-1);
        }

        sprintf( S_WORK, "tpcall failed in
OrderStatus: tpermo = %d\n"
                " svc = '%s' d_id = %d c_id = %d
c_last = '%s'\n",
                tpermo, TPAPL, bp->ordin.d_id, bp-
>ordin.c_id, bp->ordin.c_last);

        set_tuxerr( s_buf, S_WORK, cookie);
        return (-1);
    }

    DBGR(tsp(1));
    tbuf = *(ordstruct *)Ffind( (Fbfr *)srv-
>m_tcctxt[user_id].trans_b, FML_DATA, 0, NULL));
    bp = &tbuf;
#else

    if ( tpcall( TPAPL, ( char *)srv-
>m_tcctxt[user_id].trans_b,
sizeof(ordstruct),
(char **)&srv->m_tcctxt[user_id].trans_b,
&olen, 0|TPNOTIME) == -1){

        if ( tpermo == TPESVCFAIL ) {
            sprintf( S_WORK, "Oracle failed to process
NewOrder Transaction.\n"
                "tpermo = %d svc = '%s' d_id = %d c_id = %d
c_last = '%s'\n",
                tpermo, TPAPL, bp->ordin.d_id, bp-
>ordin.c_id,
                bp->ordin.c_last);

            set_oraerr( s_buf, S_WORK, cookie );
            return (-1);
        }

        sprintf( S_WORK, "tpcall failed in
OrderStatus: tpermo = %d\n"
                " svc = '%s' d_id = %d c_id = %d
c_last = '%s'\n",
                tpermo, TPAPL, bp->ordin.d_id, bp-
>ordin.c_id, bp->ordin.c_last);

```



```

        set_tuxerr (s_buf, S_WORK, cookie);
        return (-1);
    }
    DBGGR(tsp(1));

    bp = ( ordstruct * )srv->m_tcctxt[user_id].trans_b;

#endif

/* ----- Check the execution
result */
// if ( bp->ordout.terror != 1 ){
    if ( bp->ordout.terror != NOERR || bp->ordout.terror
== 1 ){

#ifdef Symfo
    if ( bp->ordout.terror == IRRECERR ) {

        sprintf ( S_WORK, "Irrecoverable
error in orderstatus.\n" );
        set_tuxerr ( s_buf, S_WORK, cookie );
        return (-1);
    }

    goto resend_orderstatus; /* tuxedo busy ->
retry!! */
#else
    set_symfoerr ( s_buf, bp->errorpos, bp-
>sqlstate, cookie );
    return (-1);
#endif
}

#else
    dummy_orderstat( bp );
#endif
    DBGP(oder_dsp ( in_data, bp, srv-
>m_tcctxt[user_id].w_id, 1));

    sprintf(S_WORK, h_order2);
/* 99.8.19 w_id 5 keta */
    int2str ((S_WORK + orderp[0]), 5, (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-
>ordin.c_last);
    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);

#ifdef Symfo
    entry = str2dbl(bp->ordout.o_entry_d,20);
    convert_time (time_data, entry);
    time2str ((S_WORK + orderp[8]), time_data);
#else
    time2str ((S_WORK + orderp[8]), bp-
>ordout.o_entry_d);
#endif

#endif

    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++ ){

/* 99.8.19 w_id 5 keta */
        int2str ((S_WORK+i*0x3a+orderp[10]), 5, (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]);

#ifdef Symfo
        sigdec2str ((S_WORK+i*0x3a+orderp[13]),
8,
            (double)((double)bp-
>ordout.ol_amount[i] / (double)100.0));

        // if( bp->ol_delivery_d[i] != -1 &&
/* Symfo NG 応急処置 */
        // bp->ol_delivery_d[i] != 77777777 ){

            if( (int)bp->ordout.ol_delivery_d[i] !=
77777777 ){

                deliver[i] = str2dbl(bp-
>ordout.ol_delivery_d[i],20);
                convert_date (&time_data0[i][0],
deliver[i]);
                date2str
((S_WORK+i*0x3a+orderp[14]), time_data0[i]);
            }
        #else
            sigdec2str ((S_WORK+i*0x3a+orderp[13]),
8,(double)bp->ordout.ol_amount[i]);

            if( strncmp( bp->ordout.ol_delivery_d[i],
"NOT DELIVR", 10 ) != 0 ){

                date2str
((S_WORK+i*0x3a+orderp[14]), bp-
>ordout.ol_delivery_d[i]);
            }
        #endif
    }

/* ----- The execution result data notified RTE is make
by the HTML form */

    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);

    return 0;
}

/*
delivery : this function processes the delivery
transaction.
*/
int delivery (char *s_buf, rte_input_data *in_data, int
cookie)
{
    delstruct *bp;
    int user_id, rtn;
    char S_WORK[WORK_S];
    char TPAPL[12];

    struct tm times;
    SYSTEMTIME systemTime;

#ifdef USE_FML
    delstruct tbuf;
    int w_id;

    user_id = cookie - Term_Base;
    bp = &tbuf;

    memset (bp, 0, sizeof(delstruct));
    /* 98.7.29 */
#else
    user_id = cookie - Term_Base;
    bp = ( delstruct * )srv->m_tcctxt[user_id].trans_b;
#endif

    bp->retval = TX_DELIVERY;
    bp->delout.terror = 0;

#ifdef USE_FML
    sprintf (TPAPL, "TPCC"); // TP-Base
Application Name (View)
#else
    sprintf (TPAPL, "DELIVERY");
    // sprintf (TPAPL, "TPCC%d", (int)srv-
>m_tcctxt[user_id].trn_id[3]);
#endif

    if ((rtn = ThrTpnit()) < 0 ){
        sprintf ( S_WORK, "Thread init abort DEL(%d) \n",
rtn);
        set_or Kerr (s_buf, S_WORK, cookie );
        return (-1);
    }

/* ----- Check the Input
data */
    bp->delin.w_id = (short)srv->m_tcctxt[user_id].w_id;

    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) /* 98.6.29:範囲チェック強化 */
        return set_errpage(s_buf, cookie, 5, (int)bp-
>delin.o_carrier_id, 0, 0);

    // bp->delin.in_timing_int = 1; /*
oracle use only */
}

```

```

/* ----- Execute Delivery
transaction */

resend_delivery:

    GetLocalTime(&systemTime);

#ifdef TOOLKIT_ORIGINAL_STRUCTURE /*
1996.08.07 */
/* bp->delin.qtime = ( double )timeque.tv_sec
+ ( double )timeque.tv_usec / 1000000.0;
*/

#else /* !TOOLKIT_ORIGINAL_STRUCTURE */

// GetLocalTime の結果を,mktime で数値デー
タに変換する
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;

/* bp->delout.startsec = (long)mktime (&times);
bp->delout.startusec =
(long)systemTime.wMilliseconds;
*/

bp->delin.startsec = (long)mktime (&times);
bp->delin.startusec =
(long)systemTime.wMilliseconds;

#endif /* !TOOLKIT_ORIGINAL_STRUCTURE */

#ifdef SCRTEST

DBGRR(fprintf ( test_fp, "tpacall delivery!\n" ));

DBGRR(tsp(0));

#endif USE_FML
w_id = bp->delin.w_id;

rtn = Fchg( (Fbfr *)srv->m_tcctxt[user_id].trans_b,
FML_TERM, 0, (char *)&w_id, 0);
rtn = Fchg( (Fbfr *)srv->m_tcctxt[user_id].trans_b,
FML_TRAN, 0, (char *)&bp->retval, 0);
rtn = Fchg( (Fbfr *)srv->m_tcctxt[user_id].trans_b,
FML_DATA, 0, (char *)bp,
( FLDLEN )sizeof( delstruct ) );

rtn = tpacall( TPAPL, ( char *)srv-
>m_tcctxt[user_id].trans_b,
0, 0|TPNOREPLY|TPNOTIME ); //
#else
rtn = tpacall( TPAPL, ( char *)srv-
>m_tcctxt[user_id].trans_b,
sizeof( delstruct ), 0| TPNOTIME |
TPNOREPLY );
#endif

DBGRR(tsp(1));

#else
dummy_delivery( bp );
rtn = 0;
#endif

sprintf ( S_WORK, h_del2);

/* ----- The execution result is
checke. */

if ( rtn == -1 ) {

/* Display messege */

#ifdef SCRTEST
char buf[1024];

switch ( tperno ) {
case TPELIMIT: /* キューが一杯になった
ため処理できない */
case TPETIME: /* トランザクションタイ
ムアウト発生 */
case TPGOTSIG: /* シグナルが発生した */

/* sprintf ( S_WORK, "tpacall : Retry in Delivery:
tperno = %d\n"
" svc = '%s' carrier = %d\n", tperno,
now_otlname, bp->delin.o_carrier_id );
*/

/* Because it is an executable again error,
processing is executed again. */
goto resend_delivery;
break;

case TPESVCFAIL:
sprintf ( S_WORK, "Oracle failed to process
Delivery Transaction.\n"
"tperno = %d svc = '%s' carrier = %d\n",
tperno,
TPAPL, bp-
>delin.o_carrier_id );

set_oraerr( s_buf, S_WORK, cookie );
return (-1);

default:
/* The error which was not able to be executed
again occurred */
sprintf( S_WORK, "tpacall failed in Delivery:
tperno = %d\n"
" svc = '%s' carrier = %d\n", tperno,
TPAPL, bp->delin.o_carrier_id );

set_tuxerr( s_buf, S_WORK, cookie);
return (-1);
}
#endif

} else {
/* 99.8.19 w_id 5 keta */
int2str ((S_WORK + delp[0]), 5, (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");
}

/* ----- The execution result data notified RTE is made
by the HTML form */

sprintf(s_buf, h_del1);
strcat (s_buf, S_WORK);

sprintf(S_WORK, h_del3, SOPATH, cookie);
strcat (s_buf, S_WORK);

return 0;
}

/*-----
-
stocklevel : this function processes the StockLevel
transaction.
-----
-*/
int stocklevel (char *s_buf, rte_input_data *in_data, int
cookie)
{
stestruct *bp;
long olen;
int loopc = 0;
int rtn = 0;
int user_id;

char S_WORK[WORK_S];
char TPAPL[12];

#ifdef USE_FML
stestruct tbuf;
int w_id;

bp = &tbuf;
user_id = cookie - Term_Base;
memset (bp, 0, sizeof(stestruct)); /*
98.7.29 */
#else
user_id = cookie - Term_Base;
bp = ( stestruct *)srv->m_tcctxt[user_id].trans_b;
#endif

bp->retval = TX_STOCKLVL;
bp->stout.terror = 0;

#ifdef USE_FML
sprintf (TPAPL, "TPCC"); // TP-Base
Application Name (View)
#else
sprintf (TPAPL, "STOCKLVL");
// sprintf (TPAPL, "TPCCs%d", (int)srv-
>m_tcctxt[user_id].trn_id[4] );
#endif

if ((rtn = ThrTpInit()) < 0){
sprintf( S_WORK, "Thread init abort STOCK (%d)
\n", rtn);
set_oraerr( s_buf, S_WORK, cookie );
return (-1);
}

/* ----- check the
Input data */
bp->stoin.w_id = (short)srv->m_tcctxt[user_id].w_id;
bp->stoin.d_id = (short)srv->m_tcctxt[user_id].d_id;

/* check threshold data : 98.6.29 範囲チェック強
化 */
bp->stoin.threshold = (long)str2short(in_data-
>threshold, 2);

```

```

if(bp->stoin.threshold < 10 || bp->stoin.threshold >
20)
    return set_errpage(s_buf, cookie, 3, (int)bp-
>stoin.threshold, 0, 0);

DBGP(sto_dsp(in_data, bp, srv-
>m_tcctxt[user_id].w_id, srv->m_tcctxt[user_id].d_id,
0));

/* ----- Execute Stock Level
transaction */
resend_stock:

#ifdef SCRTEST

    DBGR(tsp(0));

#ifdef USE_FML
    w_id = bp->stoin.w_id;

    rtn = Fchg( (Fbfr *)srv->m_tcctxt[user_id].trans_b,
FML_TERM, 0, (char *)&w_id, 0);
    rtn = Fchg( (Fbfr *)srv->m_tcctxt[user_id].trans_b,
FML_TRAN, 0, (char *)&bp->retval, 0);
    rtn = Fchg( (Fbfr *)srv->m_tcctxt[user_id].trans_b,
FML_DATA, 0, (char *)bp,
        (FLDLEN)sizeof( stostruct ) );

    if ( tpcall( TPAPL, ( char *)srv-
>m_tcctxt[user_id].trans_b, 0,
        ( char ** )&srv->m_tcctxt[user_id].trans_b,
        &olen, 0|TPNOTIME ) == -1 ){

        if ( tpermo == TPESVCFAIL ) {
            sprintf( S_WORK, "Oracle failed to process
StockLevel Transaction.\n"
                "tpermo = %d svc = '%s' threshold = %d\n",
tpermo,
                    TPAPL, bp->stoin.threshold );

            set_or Kerr( s_buf, S_WORK, cookie );
            return (-1);
        }

        sprintf( S_WORK, "stockLevel: tpermo
= %d\n"
                " svc = '%s' threshold = %d\n",
tpermo,
                    TPAPL, bp->stoin.threshold );

            set_tuxerr( s_buf, S_WORK, cookie );
            return (-1);
        }

        sprintf( S_WORK, "stockLevel: tpermo
= %d\n"
                " svc = '%s' threshold = %d\n",
tpermo,
                    TPAPL, bp->stoin.threshold );

            set_tuxerr( s_buf, S_WORK, cookie );
            return (-1);
        }

        DBGR(tsp(1));
        tbuf = *((stostruct *)Ffind( (Fbfr *)srv-
>m_tcctxt[user_id].trans_b, FML_DATA, 0, NULL));
        bp = &tbuf;

    #else

        if ( tpcall( TPAPL, ( char *)srv-
>m_tcctxt[user_id].trans_b,
                sizeof( stostruct ),
                ( char ** )&srv->m_tcctxt[user_id].trans_b, &olen,
0|TPNOTIME ) == -1 ){

            if ( tpermo == TPESVCFAIL ) {

                sprintf( S_WORK, "Oracle failed to process
StockLevel Transaction.\n"
                    "tpermo = %d svc = '%s' threshold = %d\n",
tpermo,
                        TPAPL, bp->stoin.threshold );

                set_or Kerr( s_buf, S_WORK, cookie );
                return (-1);
            }

            sprintf( S_WORK, "stockLevel: tpermo
= %d\n"
                    " svc = '%s' threshold = %d\n",
tpermo,
                        TPAPL, bp->stoin.threshold );

                set_tuxerr( s_buf, S_WORK, cookie );
                return (-1);
            }

            DBGR(tsp(1));
            tbuf = *((stostruct *)Ffind( (Fbfr *)srv-
>m_tcctxt[user_id].w_id, srv->m_tcctxt[user_id].d_id,
1));

            sprintf( S_WORK, h_stock2);
            /* 99.8.19 w_id 5 keta */
            int2str ((S_WORK + stockp[0]), 5, (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);

                sprintf( S_WORK, "Oracle failed to process
StockLevel Transaction.\n"
                    "tpermo = %d svc = '%s' threshold = %d\n",
tpermo,
                        TPAPL, bp->stoin.threshold );

                set_tuxerr( s_buf, S_WORK, cookie );
                return (-1);
            }

            DBGR(tsp(1));

        }

        bp = ( stostruct * )srv->m_tcctxt[user_id].trans_b;

    #endif

/* ----- Check the execution
result */
// if ( bp->stout.terror != 1 ){
//     if ( bp->stout.terror != NOERR || bp->stout.terror
== 1){

        #ifndef Symfo
            if ( bp->stout.terror == IRRECERR ) {
                sprintf( S_WORK, "Irrecoverable error in
stocklevel \n");
                set_tuxerr( s_buf, S_WORK, cookie );
                return (-1);
            }
            goto resend_stock; /* TP application
busy. Try again */
        #else
            set_symfoerr( s_buf, bp->errorpos, bp-
>sqlstate, cookie );
            return (-1);
        #endif
    }

    #else
        dummy_stocklvl ( bp );
        DBGP(sto_dsp(in_data, bp, srv-
>m_tcctxt[user_id].w_id, srv->m_tcctxt[user_id].d_id,
1));

        sprintf( S_WORK, h_stock2);
        /* 99.8.19 w_id 5 keta */
        int2str ((S_WORK + stockp[0]), 5, (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);

            select_trn:

            RTE から指定されたトランザクション毎の
処理を実行する
            復帰値: 正常終了時はゼロを通知する

            s_buf にトランザクションを処理した結果を
HTML フォームで設定する

            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 form
ParseFormData
            ptrs - a pointer to a raw_form_data structure with
pointers
                    to values in 'query'.

            有効な端末番号を受信した時にクリティカ
ルセクションを設定する
            ----- */

            int select_trn ( rte_input_data *in_data, char *s_buf, int
cookie ) {

                int length = 0;
                int user_id = cookie - Term_Base;
                int rtn = 0;

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

                    if (in_data->form[0] == 'I'){
                        /* send the transaction select screen
page */

                            rtn = fast_menu (s_buf, in_data,
cookie);

                            return rtn;
                        }

                    else{

                        EnterCriticalSection(&(srv-
>m_tcctxt[user_id].user)); // Make it thread safe.

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

```

```

        rtn = neworder (s_buf, in_data,
cookie);
        break;
    case 'D':
        rtn = delivery(s_buf, in_data, cookie);
        break;
    case 'P':
        rtn = payment (s_buf, in_data,
cookie);
        break;
    case 'S':
        rtn = stocklevel(s_buf, in_data,
cookie);
        break;
    case 'O':
        rtn = orderstatus (s_buf,
in_data, cookie);
        break;
    default:
        /* uninput transaction type */
        set_errpage(s_buf, cookie, 1, -
4, 0, 0);
        rtn = 1;
        break;
    }

    LeaveCriticalSection(&(srv-
>m_tcctx[user_id].user));
    return rtn;
}
else if(in_data->button) {

    EnterCriticalSection(&(srv-
>m_tcctx[user_id].user)); // Make it thread safe.

    /* send the data input screen page */
    switch(in_data->button[0]) {
    case 'N':

        sprintf(s_buf, in_newpage, SOPATH,
cookie, srv->m_tcctx[user_id].w_id);
        length = strlen (s_buf);

        sprintf(s_buf+length-1 ,
in_newpage2);

        break;

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

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

    case 'S':
        sprintf(s_buf, in_stkpage, SOPATH, cookie,
srv-
>m_tcctx[user_id].w_id, srv->m_tcctx[user_id].d_id);
        break;

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

    case 'Q':
        /* This value use WWW browser only. */
        if (in_data->cookie)
            srv->Termfree (in_data-
>cookie);

        sprintf (s_buf, loginpage , VLDATA, SOPATH);
        return rtn;

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

    LeaveCriticalSection(&(srv-
>m_tcctx[user_id].user));
    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;
}

////////////////////////////////////
// 唯一の CWinApp オブジェクト
// メモ: MFC を DLL 内で使用しないように変更
// する場合は、このオブジェクト
// を削除してもかまいません。

CWinApp theApp;

#ifdef WIZDEF
////////////////////////////////////
// コマンド解析マップ
BEGIN_PARSE_MAP(CTpapIExtension, CHttpServer)
// TODO: この位置にコマンドをフックす
// るために ON_PARSE_COMMAND()
// および ON_PARSE_COMMAND_PARAMS()
// を挿入してください。
// 例:
// ON_PARSE_COMMAND (Default,
// CTpapIExtension, ITS_PSTR)
// DEFAULT_PARSE_COMMAND (Default,
// CTpapIExtension)
END_PARSE_MAP(CTpapIExtension)
#endif

////////////////////////////////////
// 唯一の CTpapIExtension オブジェクト
CTpapIExtension theExtension;
////////////////////////////////////
////////////////////////////////////
// CTpapIExtension クラスのインプリメンテーシ
// ヨン
//
// constructor
//
CTpapIExtension::CTpapIExtension() : CHttpServer(?)
{
    DWORD kind;
    DWORD type;
    DWORD size;

    int user_id;
    int def_base = 1;
    int def_warehouse = 2000;
    int def_maxusers = 20000;
    int def_maxterm = 2000;
    int CONTROL_Flag = 0;

    int errcode = 0;

    union dtg {BYTE bit[32]; char chara[32]; DWORD
data[8];} reg_d;

    /* Open registry Area */
    RegOpenKeyEx(HKEY_LOCAL_MACHINE,
"SOFTWARE", 0,
    KEY_READ | KEY_WRITE,
&m_tpcpregkey);

    RegCreateKeyEx(m_tpcpregkey, "Fujitsu", 0,
NULL,
    REG_OPTION_NON_VOLATILE,
KEY_READ | KEY_WRITE, NULL,
&m_tpcpregkey, &kind);

    RegCreateKeyEx(m_tpcpregkey, "TPC-C ISAPI
Application", 0, NULL,
    REG_OPTION_NON_VOLATILE,
KEY_READ | KEY_WRITE, NULL,
&m_tpcpregkey, &kind);

    /* If the key is newly made, the default value is
set in the key. */
    if(kind == REG_CREATED_NEW_KEY) {
        RegSetValueEx(m_tpcpregkey,
"Term_Base",
    0, REG_DWORD,(const unsigned
char *)&def_base, 4);
        RegSetValueEx(m_tpcpregkey,
"NumWarehouses",
    0, REG_DWORD,(const unsigned
char *)&def_warehouse, 4);
        RegSetValueEx(m_tpcpregkey,
"MaxUsers",
    0, REG_DWORD,(const unsigned
char *)&def_maxusers, 4);
        RegSetValueEx(m_tpcpregkey, "MaxTerm
of Client",
    0, REG_DWORD,(const unsigned
char *)&def_maxterm, 4);
        RegSetValueEx(m_tpcpregkey,
"CONTROL_Flag",
    0, REG_DWORD,(const unsigned
char *)&CONTROL_Flag, 4);
    }

    /* Get registry data */

```

```

    RegQueryValueEx(m_tpccregkey, "Term_Base",
        0, &type, (unsigned char *)&reg_d.bit,
&size);
    Term_Base = reg_d.data[0];
    // Start terminal(user) number

    RegQueryValueEx(m_tpccregkey,
"NumWarehouses",
        0, &type, (unsigned char *)&reg_d.bit,
&size);
    maxwh = reg_d.data[0];
    // Max warehouse scale

    RegQueryValueEx(m_tpccregkey, "MaxUsers",
        0, &type, (unsigned char *)&reg_d.bit,
&size);
    maxconnect = reg_d.data[0];
    // Max terminal(user) number

    RegQueryValueEx(m_tpccregkey, "MaxTerm of
Client",
        0, &type, (unsigned char *)&reg_d.bit,
&size);
    maxterm = reg_d.data[0];
    // Max terminal(user) number of client

    RegQueryValueEx(m_tpccregkey,
"CONTROL_Flag",
        0, &type, (unsigned char *)&reg_d.bit,
&size);
    C_FLAG = reg_d.data[0];
    // debug flag

    InitializeCriticalSection (&crit);
    EnterCriticalSection (&crit);

#ifdef SCRTEST

    TLSIsTpnitedKey = TlsAlloc ();
#endif

    /* Initialize Working Area */
    for(user_id = 0; user_id < maxterm; user_id++) {
        m_tcctx[user_id].w_id = 0;
        m_tcctx[user_id].d_id = 0;
        m_tcctx[user_id].trans_b = 0;
    }

    LeaveCriticalSection (&crit);
    srv = this;
}

//
// deconstructor
//
CTpaplExtension::~CTpaplExtension()
{
    int x;

    EnterCriticalSection(&crit);

    for(x = Term_Base; x < Term_Base + maxterm;
x++){
        Termfree (x);
    }
}

```

```

    LeaveCriticalSection(&crit);

    TlsFree(TLSIsTpnitedKey);
}

BOOL
CTpaplExtension::GetExtensionVersion(HSE_VERSIO
N_INFO* pVer)
{
    // 初期化処理用にデフォルトのインプリ
メンテーションを呼び出します
    CHttpServer::GetExtensionVersion(pVer);

    // 説明の文字列を読み込みます
    TCHAR
sz[HSE_MAX_EXT_DLL_NAME_LEN+1];
    ISAPIVERIFY(::LoadString(AfxGetResourceHan
dle(),
        IDS_SERVER, sz,
HSE_MAX_EXT_DLL_NAME_LEN));
    _tcsncpy(pVer->lpszExtensionDesc, sz);
    return TRUE;
}

////////////////////////////////////
// HttpExtensionProc
////////////////////////////////////
// This method is called every time the user makes a
// request. We don't do anything here except catch a
successful
// return value and substitute a value that permits the
// network connection to be kept alive.

DWORD
CTpaplExtension::HttpExtensionProc(EXTENSION_CO
NTROL_BLOCK *pECB)
{
    char QueryString[1024];
    char *page;
    int cookie = -1;
    int len, rtn;

    char S_BUF [BUF_S];
    char S_WORK[WORK_S];

    rte_input_data in_data_area;

#ifdef DBPRT
    char dbg_query[1024];
    long s_pid;
    char path_w[96];
#endif

    char szHeader[128];
    char szHeader1[128];
    int isize;
    BOOL rtcode;

    strcpy(QueryString, (char *)pECB-
>lpszQueryString);

#ifdef DBPRT
    strcpy(dbg_query, (char *)pECB-
>lpszQueryString);
#endif

    strct_init (&in_data_area);
}

```

```

    cookie = anly_para ((char *)QueryString,
&in_data_area );

#ifdef DBPRT
    if ( cookie >= 0){
        sprintf (path_w, "c:\\tclg\\log%d", cookie);
        test_fp = fopen (path_w, "aw+");
    }
#endif

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

        sprintf (S_BUF, badterm, Term_Base,
Term_Base+maxterm-1, cookie);
        len = strlen (S_BUF);

#ifdef DBPRT
        if ( cookie >= 0){
            fprintf (test_fp, "--> QUERY: %s\n",
dbg_query);
            fprintf (test_fp, "%s%s\n", S_WORK,
S_BUF);
            fflush (test_fp);
            fclose (test_fp);
        }
#endif
        isize = sprintf (szHeader, "200 OK");
        sprintf (szHeader1,
            "Connection: keep-alive\nContent-
type: text/html\nContent-length: %d\n\n",
len );

        rtcode = (*pECB-
>ServerSupportFunction)(pECB->ConnID,
HSE_REQ_SEND_RESPONSE_HEADER,

            szHeader, (unsigned long *)&isize),
(LPDWORD)szHeader1);

        rtcode = (*pECB->WriteClient)(pECB-
>ConnID, &S_BUF[0], (unsigned long *)&len, 0);

        return
HSE_STATUS_SUCCESS_AND_KEEP_CONN;
    }

    // トランザクション要求を切り分け、各
トランザクションの処理を実行する
    rtn = select_tm (&in_data_area, S_BUF,
cookie );

    // 処理結果フォームを出力する
    len = strlen (S_BUF);

    isize = sprintf (szHeader, "200 OK");

    sprintf (szHeader1,
        "Connection: keep-alive\nContent-
type: text/html\nContent-length: %d\n\n",
len );

    rtcode = (*pECB-
>ServerSupportFunction)(pECB->ConnID,
HSE_REQ_SEND_RESPONSE_HEADER,

```

```

szHeader, (unsigned long *)&size),
(LPWORD)szHeader1);

rcode = (*pECB->WriteClient)(pECB->ConnID,
&S_BUF[0], (unsigned long *)&len, 0);

#ifdef DBPRT
if (cookie >= 0){
    fprintf (test_fp, "--> QUERY: %s\n",
dbg_query);
    fprintf (test_fp, "%s\n", S_WORK,
S_BUF);
    fflush (test_fp);
    fclose (test_fp);
}
#endif

// For Debug : output error message
if ((C_FLAG & PRT_ELOG) != 0)
if (strstr (S_BUF, "ERROR") != 0) {
    FILE *test_fp2;
    char w_path[96];

    sprintf (w_path, "c:\\tcllog\\log%d", cookie);
    test_fp2 = fopen (w_path, "aw+");
    fprintf (test_fp2, "%s\n", S_BUF);
    fflush(test_fp2);
    fclose(test_fp2);
}
}
}

return
HSE_STATUS_SUCCESS_AND_KEEP_CONN;
}

////////////////////////////////////
// Termit
////////////////////////////////////
// Called when a user logs in. This function gives the
// user a cookie and creates a user context in the
// server
// object associated with that cookie. The context
// contains
// working copies of the response forms with the cookie
// plugged into them, the warehouse and district ids
// provided by the user, and the database connection.
// This function also opens the connection to the
// database
// for this user.

int CTpaplExtension::Termit(int w_id, int d_id, int
cookie)
{
    int user_id;

#ifdef USE_FML
#define BUF_TYPE "FML"
#else
#define BUF_TYPE "CARRAY"
#endif

#ifdef LOG_ALL
char buffer[80];
#endif LOG_ALL

```

```

// Non-reentrant code.
// Must find the first free context slot.

EnterCriticalSection(&crit); // Make it thread
safe

user_id = cookie - Term_Base;

m_tcctx[user_id].w_id = w_id;
m_tcctx[user_id].d_id = d_id;

#ifdef SCRTEST
// Create Tuxedo buffer (Execute tpalloc)
if ((m_tcctx[user_id].trans_b =
(void *)tpalloc (BUF_TYPE, NULL,
trans_size)) == NULL) {

// tpalloc abnormal end. create message
form.

#ifdef DBPRT
fprintf (test_fp, "tpalloc() faild (%d) :
user_id=%d\n",
tperrno, cookie);
#endif

LeaveCriticalSection(&crit);
return (-1);
}
}

#else
// For Debug : create local buffer
if ((m_tcctx[user_id].trans_b =
(void *)malloc (trans_size)) == NULL){
// malloc abnormal end. create message
form.
#ifdef DBPRT
fprintf (test_fp, "tpalloc() faild : user_id
= %d\n", cookie);
#endif

LeaveCriticalSection(&crit);
return (-1);
}
}

#ifdef USE_FML
memset (m_tcctx[user_id].trans_b, 0,
(size_t)trans_size);
#endif

LeaveCriticalSection(&crit);

InitializeCriticalSection(&(m_tcctx[user_id].user));

return cookie;
}

////////////////////////////////////
// Termfree
////////////////////////////////////
// Given a cookie, determines whether there is a
// session
// associated with it, and if there is, closes the attached
// database session, frees the associated resources,
// and
// makes the slot available for future use.

```

```

void CTpaplExtension::Termfree (int Cookie) {

#ifdef LOG_ALL
char buffer[80];
#endif LOG_ALL

int idx = Cookie - Term_Base;

// Login されたことをチェックしたのち、
// alloc 領域を開放する
if(m_tcctx[idx].w_id != 0) {

#ifdef SCRTEST
tpfree ((char *)m_tcctx[idx].trans_b);
#else
free (m_tcctx[idx].trans_b);
#endif

// Deconstructor からの実行時、Enter
// していないけど大丈夫!!
LeaveCriticalSection(&(m_tcctx[idx].user));

// Login 情報をクリアする
m_tcctx[idx].d_id = 0;
m_tcctx[idx].w_id = 0;
}
}

// For convenience, an overloaded version of Termfree
// that understands the string representation of the
// cookie.
void CTpaplExtension::Termfree (char *Cookie) {

int x;

if((x = str2int (Cookie, 4)) < 0) return;
else Termfree (x);
}

// ClassWizard が必要とする以下の行を編集し
// ないでください。
#ifdef 0
BEGIN_MESSAGE_MAP(CTpaplExtension,
CHttpServer)
//{{AFX_MSG_MAP(CTpaplExtension)
//}}AFX_MSG_MAP
END_MESSAGE_MAP()
#endif // 0

////////////////////////////////////
// 拡張オブジェクトが MFC を使用しない場合、
// 拡張オブジェクトがモジュールに
// 対するリソース ハンドルを確実に見つける
// ために、このコードが必要となります。
// もし拡張オブジェクトが MFC を使用するよ
// うに変更する場合は、以下の
// AfxGetResourceHandle() および DllMain() 関数を
// g_hInstance global
// と同様に削除してください。
/****

static HINSTANCE g_hInstance;

```

```
HINSTANCE AfxGetResourceHandle()
{
    return g_hInstance;
}

BOOL WINAPI DllMain(HINSTANCE hInst, ULONG
ulReason,
                    LPVOID lpReserved)
{
    if (ulReason == DLL_PROCESS_ATTACH)
    {
        g_hInstance = hInst;
    }

    return TRUE;
}

****/
```


Appendix B: Server Source Code

```
===== tpc.h =====
```

```
/*
 * $Header: tpc.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
 | tpc.h
 | DESCRIPTION
 | Include file for TPC-C benchmark programs.
 +-----+
 =====*/

#ifndef TPCC_H
#define TPCC_H

#ifndef FALSE
#define FALSE 0
#endif

#ifndef TRUE
#define TRUE 1
#endif

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

#ifndef boolean
#define boolean int
#endif

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

typedef struct cda_def csrdef;
typedef struct cda_def ldadef;

/* TPC-C transaction functions */
```

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

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

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

#define DELRT 80.0

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

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

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

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

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

extern FILE *lfp;
extern FILE *fopen ();
extern int proc_no;
```

```
extern int doid[];

extern int execstatus;
extern int errcode;

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

/* for stock-level transaction */

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

/* for delivery transaction */

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

/* for order-status transaction */

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

/* for payment transaction */

extern int c_w_id;
extern int c_d_id;
extern int h_amount;
extern char w_street_1[21];
extern char w_street_2[21];
extern char w_city[21];
extern char w_zip[10];
extern char w_state[3];
extern char d_street_1[21];
extern char d_street_2[21];
extern char d_city[21];
extern char d_state[3];
extern char d_zip[10];
extern char c_street_1[21];
extern char c_street_2[21];
extern char c_city[21];
extern char c_state[3];
extern char c_zip[10];
extern char c_phone[17];
```

```

extern text c_since_d[11];
extern char c_credit[3];
extern int c_credit_lim;
extern float c_discount;
extern char c_data[201];
extern text h_date[20];

/* for new order transaction */

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

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

#ifdef DISCARD
# define DISCARD (void)
#endif

#ifdef sword
# define sword int
#endif

#define VER7      2

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

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

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

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

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

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

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

#define
OCIBNDRA(stmp,bndp,errp,sqlvar,progvl,progvl,ftype,in
dp,alen,arcode) \
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),(indp),(alen),(arcode),0,0,OCI_D
EFAULT));

#define
OCIBNDRAD(stmp,bndp,errp,sqlvar,progvl,ftype,indp,ct
xp,cbf_nodata,cbf_data) \
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)),0,(progvl),(ftype), \
indp,0,0,0,0,OCI_DATA_AT_EXEC)); \
ocierror(__FILE__,__LINE__,(errp), \
OCIBindDynamic((bndp),(errp),(ctxp),(cbf_nodat
a),(ctxp),(cbf_data)));

#define
OCIBNDR(stmp,bndp,errp,sqlvar,progvl,progvl,ftype,ind
p,alen,arcode) \
ocierror(__FILE__,__LINE__,(errp), \
OCIHandleAlloc((stmp),(dvoid*)&(bndp),OCI_HT
YPE_BIND,0,(dvoid**)0)); \
ocierror(__FILE__,__LINE__,(errp), \
OCIBindByName((stmp),&(bndp),(errp),(text
*)(sqlvar),strlen((sqlvar)), \
(progvl),(progvl),(ftype),(indp),(alen),(arcode),0,0,OCI_D
EFAULT));

#define
OCIBNDRAA(stmp,bndp,errp,sqlvar,progvl,progvl,ftype,i
ndp,alen,arcode,ms,cu) \
ocierror(__FILE__,__LINE__,(errp), \
OCIHandleAlloc((stmp),(dvoid*)&(bndp),OCI_HT
YPE_BIND,0,(dvoid**)0)); \
ocierror(__FILE__,__LINE__,(errp), \
OCIBindByName((stmp),&(bndp),(errp),(text
*)(sqlvar),strlen((sqlvar)), \
(progvl),(progvl),(ftype),(indp),(alen),(arcode),0,0,OCI_D
EFAULT)); \
ocierror(__FILE__,__LINE__,(errp), \
OCIBindDynamic((bndp),(errp),(ctxp),(cbf_nodat
a),(ctxp),(cbf_data)));

ocierror(__FILE__,__LINE__,(errp), \
OCIBindByName((stmp),&(bndp),(errp),(text
*)(sqlvar),strlen((sqlvar)), \
(progvl),(progvl),(ftype),(indp),(alen),(arcode),(ms),(
cu),OCI_DEFAULT));

#define
OCIDEF(stmp,dfnp,errp,pos,progvl,progvl,ftype)\
OCIDefineByPos((stmp),&(dfnp),(errp),(pos),(prog
v),(progvl),(ftype), \
0,0,0,OCI_DEFAULT);

#define
OCIDEF(stmp,dfnp,errp,pos,progvl,progvl,ftype)\
\
OCIHandleAlloc((stmp),(dvoid*)&(dfnp),OCI_HTY
PE_DEFINE,0, \
(dvoid**)0); \
OCIDefineByPos((stmp),&(dfnp),(errp),(pos),(prog
v),(progvl), \
(ftype),NULL,NULL,NULL,OCI_DEFA
ULT); \

#define
OCIDFNRA(stmp,dfnp,errp,pos,progvl,progvl,ftype,indp,
alen,arcode) \
OCIHandleAlloc((stmp),(dvoid*)&(dfnp),OCI_HTY
PE_DEFINE,0, \
(dvoid**)0); \
OCIDefineByPos((stmp),&(dfnp),(errp),(pos),(prog
v)), \
(progvl),(ftype),(indp),(alen), \
(arcode),OCI_DEFAULT);

#define
OCIDFNDR(stmp,dfnp,errp,pos,progvl,progvl,ftype,ind
p,ctxp,cbf_data) \
ocierror(__FILE__,__LINE__,(errp), \
OCIHandleAlloc((stmp),(dvoid*)&(dfnp),OCI_HTY
PE_DEFINE,0, \
(dvoid**)0); \
ocierror(__FILE__,__LINE__,(errp), \
OCIDefineByPos((stmp),&(dfnp),(errp),(pos),(prog
v),(progvl),(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_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;
};

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

===== 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;
    long startsec;
    long startusec;
    /* double qltime;
    int in_timing_int;*/
};

struct deloutstruct {
    int terror;
    int retry;
};

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

/* Stock level */

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

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

struct stostruct {
    int retval;
    struct stoinstruct stoin;
    struct stoostruct stoostruct;
};

/* 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

===== tpccapi.h =====

#ifndef TPCCAPI_H
#define TPCCAPI_H

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

//VERSION RESOURCE DEFINES
#define _APS_NEXT_RESOURCE_VALUE 101
#define _APS_NEXT_COMMAND_VALUE 40001
#define _APS_NEXT_CONTROL_VALUE 1000
#define _APS_NEXT_SYMED_VALUE 101

//note that the welcome form must be processed
first as terminal ids
//assigned here, once the terminal id is assigned
then the forms can
//be processed in any order.

#define WELCOME_FORM 1
//beginning form no term id assigned,
form id
#define MAIN_MENU_FORM 2
//term id assigned main menu form id
#define NEW_ORDER_FORM 3
//new order form id
#define PAYMENT_FORM 4

```

```

//payment form id
#define DELIVERY_FORM 5
//delivery form id
#define ORDER_STATUS_FORM 6
//order status id
#define STOCK_LEVEL_FORM 7
//stock level form id

//This macro is used to prevent the compiler error
unused formal parameter
#define UNUSEDPARAM(x) (x = x)

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

//This structure defines the data necessary to
keep distinct for each
//terminal or client connection.
typedef struct _CLIENTDATA
{
    int inUse; //in use flag allows client entries
    to be reused
    int w_id; //warehouse id assigned at
    welcome form
    int d_id; //district id assigned at welcome
    form
    int iProcessRequestCounter; //request
    counter
    int iSyncId; //synchronization id
    int iTickCount; //time of last access;
    int iTermId; //terminal id of http stream
    connection
    char szBuffer[4096]; //form buffer; HTML form
    built for client here
    //for TOPEND
    char type;
    int retval;
    int client;
    BOOL bFailed;
#ifdef TOPEND
    long *tp_structadr;
#else // assume tuxedo
    struct newstruct *newOrderDataPtr; //new
    order Tuxedo buffer
    struct paystruct *paymentDataPtr;
    //payment Tuxedo buffer
    struct ordstruct *orderStatusDataPtr; //order
    status Tuxedo buffer
    struct delstruct *deliveryDataPtr; //delivery
    Tuxedo buffer
    struct stostruct *stockLevelDataPtr; //stock
    level Tuxedo buffer
#endif
    union {
        struct newstruct newOrderData; //new order
        form data
        struct paystruct paymentData; //payment
        form data
        struct ordstruct orderStatusData; //order status
        form data
        struct delstruct deliveryData; //delivery form
        data
    }
};

struct stostruct stocklevelData; //stock level
form data
} trans;
} CLIENTDATA;

typedef CLIENTDATA *PCLIENTDATA; //pointer to
client structure
//This structure is used to define the operational
interface for
//terminal id support
typedef struct _TERM
{
    int iAvailable; //total allocated terminal array entries
    int iNext; //next available terminal array element
    int iMasterSyncId; //synchronization id
    BOOL bInit; //structure has been initialized flag
    CLIENTDATA *pClientData; //pointer to allocated
    client data
    void (*Init)(void); //API to initialize this structure
    int (*Allocate)(void); //API to allocate new terminal
    entry; array id returned
    void (*Restore)(void); //API to free terminal data
    int (*Add)(EXTENSION_CONTROL_BLOCK *pECB,
    char *pQueryString); //API to add a terminal id to
    array,
    //this context will be
    passed from the
    //browser to the tpc.dll
    //TERMID= key in the
    HTTP string.
    void (*Delete)(EXTENSION_CONTROL_BLOCK
    *pECB, int id); //API to free
    //resources
    used by a
    //terminal
    array
    //entry
} TERM;

typedef TERM *PTERM; //pointer to terminal structure
type
//this structure allows the EXTENSION CONTROL
BLOCK to be passed to the
//msg and error handlers.
typedef struct _ECBINFO
{
    int iTermId; //terminal id
    int iSyncId; //browser sync id
    BOOL bDeadlock; //deadlock condition flag
    BOOL bFailed; //cleared before sql
    transaction, set in err
    //handlers if an error occurs
    EXTENSION_CONTROL_BLOCK *pECB;
    //inetsrv current connection structure
    //information
} ECBINFO, *PECBINFO;

//function prototypes
#ifdef TOPEND // in tuxedo defined in atmi.h
extern int tpsvrint (int argc, char * argv[]);
// argv arguments are int pid, char *uid, char
*pwd, int txntype
extern int NEWORDER(CLIENTDATA *jobData,
NewOrderData *neword, int deadlock);
extern int PAYMENT(CLIENTDATA *jobData,
PaymentData *paydata, int deadlock);
extern int ORDERSTATUS(CLIENTDATA *jobData,
OrderStatusData *orddata, int deadlock);
extern int STOCKLEVEL(CLIENTDATA *jobData,
StockLevelData *stodata, int deadlock);
extern int DELIVERY(CLIENTDATA *jobData,
DeliveryData *deldata, int deadlock);
#endif

#ifdef TUX
extern void NEWORDER(TPSVCINFO *msg);
extern void PAYMENT(TPSVCINFO *msg);
//extern void ORDERSTATUS(TPSVCINFO *msg);
extern void ORDERSTATUS();
extern void STOCKLEVEL(TPSVCINFO *msg);
extern void DELIVERY(TPSVCINFO *msg);
#endif

BOOL APIENTRY DllMain(HANDLE hModule,
DWORD u_reason_for_call, LPVOID lpReserved);
static BOOL IsValidTermId(int TermId);
BOOL
ProcessQueryString(EXTENSION_CONTROL_BLOCK
*pECB, int *pCmd, int *pFormId, int *pTermId, int
*pSyncId);
void NewOrderForm(EXTENSION_CONTROL_BLOCK
*pECB, int iFormId, int iTermId, int iSyncId);
void PaymentForm(EXTENSION_CONTROL_BLOCK
*pECB, int iFormId, int iTermId, int iSyncId);
void DeliveryForm(EXTENSION_CONTROL_BLOCK
*pECB, int iFormId, int iTermId, int iSyncId);
void
OrderStatusForm(EXTENSION_CONTROL_BLOCK
*pECB, int iFormId, int iTermId, int iSyncId);
void
StockLevelForm(EXTENSION_CONTROL_BLOCK
*pECB, int iFormId, int iTermId, int iSyncId);
void ExitCmd(EXTENSION_CONTROL_BLOCK *pECB,
int iFormId, int iTermId, int iSyncId);
void SubmitCmd(EXTENSION_CONTROL_BLOCK
*pECB, int iFormId, int iTermId, int iSyncId);
void BeginCmd(EXTENSION_CONTROL_BLOCK
*pECB, int iFormId, int iTermId, int iSyncId);
void ProcessCmd(EXTENSION_CONTROL_BLOCK
*pECB, int iFormId, int iTermId, int iSyncId);
void ClearCmd(EXTENSION_CONTROL_BLOCK
*pECB, int iFormId, int iTermId, int iSyncId);
void MenuCmd(EXTENSION_CONTROL_BLOCK
*pECB, int iFormId, int iTermId, int iSyncId);
static void
WriteZString(EXTENSION_CONTROL_BLOCK *pECB,
char *szStr);
static void h_printf(EXTENSION_CONTROL_BLOCK
*pECB, char *format, ...);
void ErrorMessage(EXTENSION_CONTROL_BLOCK
*pECB, int iError, int iErrorType, char *szMsg, int
iTermId, int iSyncId);
static BOOL GetKeyValue(char *pQueryString, char
*pKey, char *pValue, int iMax);
static void TermInit(void);
static void TermRestore(void);
static int TermAllocate(void);
static int TermAdd(EXTENSION_CONTROL_BLOCK
*pECB, char *pQueryString);
static void
TermDelete(EXTENSION_CONTROL_BLOCK *pECB,
int id);

```

```

BOOL Init(EXTENSION_CONTROL_BLOCK *pECB,
int iTermId, int iSynclD, char *szServer, char *szUser,
char *szPassword, char *szDatabase);
static BOOL Close(EXTENSION_CONTROL_BLOCK
*pECB, int iTermId, int iSynclD);
static void FormatString(char *szDest, char *szPic, char
*szSrc);
static char *MakeStockLevelForm(int iTermId, int
iSynclD, BOOL bInPut);
static char *MakeMainMenuForm(int iTermId, int
iSynclD);
static char *MakeWelcomeForm(void);
static char *MakeNewOrderForm(int iTermId, int
iSynclD, BOOL bInPut, BOOL bValid, char
*execution_status);
static char *MakePaymentForm(int iTermId, int iSynclD,
BOOL bInPut);
static char *MakeOrderStatusForm(int iTermId, int
iSynclD, BOOL bInPut);
static char *MakeDeliveryForm(int iTermId, int iSynclD,
BOOL bInPut, char *execution_status);
void UtilStrCpy(char *pDest, char *pSrc, int n);
static void
ProcessNewOrderForm(EXTENSION_CONTROL_BL
OCK *pECB, int iTermId, int iSynclD);
static void
ProcessPaymentForm(EXTENSION_CONTROL_BLO
CK *pECB, int iTermId, int iSynclD);
static void
ProcessOrderStatusForm(EXTENSION_CONTROL_B
LOCK *pECB, int iTermId, int iSynclD);
static void
ProcessDeliveryForm(EXTENSION_CONTROL_BLOC
K *pECB, int iTermId, int iSynclD);
static void
ProcessStockLevelForm(EXTENSION_CONTROL_BL
OCK *pECB, int iTermId, int iSynclD);
static int GetNewOrderData(LPSTR lpszQueryString,
NewOrderData *pNewOrderData);
static int GetPaymentData(LPSTR lpszQueryString,
PaymentData *pPaymentData);
static int GetOrderStatusData(LPSTR lpszQueryString,
OrderStatusData *pOrderStatusData);
static BOOL ReadRegistrySettings(void);
static BOOL IsNumeric(char *ptr);
static void FormatHTMLString(char *szBuff, char *szStr,
int iLen);
static void Log(char *szType, char *szStr);
int do_neworder(NewOrderData *pNODData);
int do_payment(PaymentData *pPAYData);
int do_orderstatus(OrderStatusData *pOSData);
int do_delivery(pDeliveryData pDelivery);
int do_stocklevel(StockLevelData *pSTODData);
int end_neworder(NewOrderData *pNODData);
int end_payment(PaymentData *pPAYData);
int end_orderstatus(OrderStatusData *pOSData);
int end_stocklevel(StockLevelData *pSTODData);
int do_all_txns(int txn);
int do_disconnect();

#endif /* TPCCAPI_H */

===== tpccflags.h =====

#define PLSQLNO

#define DMLRETDL

===== checkerr.c =====

static void checkerr(errhp, status)
OCIError *errhp;
sword status;
{
    text errbuf[512];
    ub4 buflen;
    sb4 errcode;

    if (status == OCI_SUCCESS) return;

    switch (status)
    {
        case OCI_SUCCESS_WITH_INFO:
            printf("Error -
OCI_SUCCESS_WITH_INFO\n");
            break;
        case OCI_NEED_DATA:
            printf("Error - OCI_NEED_DATA\n");
            break;
        case OCI_NO_DATA:
            printf("Error - OCI_NO_DATA\n");
            break;
        case OCI_ERROR:
            while ( OCIErrorGet ((dvoid *) errhp, (ub4) 1,
(text *) NULL, &errcode,
errbuf, (ub4) sizeof(errbuf), (ub4)
OCI_HTYPE_ERROR) != OCI_NO_DATA)
                printf("Error - %d: %s\n", errcode, errbuf);
            break;
        case OCI_INVALID_HANDLE:
            printf("Error - OCI_INVALID_HANDLE\n");
            break;
        case OCI_STILL_EXECUTING:
            printf("Error - OCI_STILL_EXECUTE\n");
            break;
        case OCI_CONTINUE:
            printf("Error - OCI_CONTINUE\n");
            break;
        default:
            printf("Error - %d\n", status);
            break;
    }
}

===== errrpt.c =====

#include <stdio.h>
#include <time.h>
#include "tpcc.h"
#include "c_defs.h"
#include "glodec.h"

/*
#define ORA7
*/

#ifdef TUX
#include <userlog.h>
#endif

void logerr();

/* external declarations */
extern FILE *errfp;
extern int firstlog;
extern int proc_no;

extern void terminate();
#ifdef ORA7
int errrpt(lda, cda)
Lda_Def *lda;
Cda_Def *cda;
{
#define MAX_ERR_LEN 2048
char buff[MAX_ERR_LEN + 1];
char bigbuff[MAX_ERR_LEN+20];
sb2 err;

err = (sb2) cda->rc;
oerhms(lda, err, buf, (int)sizeof(buff));
sprintf(bigbuff, "ORA-%d ***\n%s\n", err, buf);
logerr(bigbuff, __FILE__, __LINE__);
if ((cda->rc == DEADLOCK) || (cda->rc ==
SNAPSHOT_TOO_OLD))
    return (RECOVERR);
else
    return (IRRECERR);
}
#else
void errrpt (lda, cur, fname, lineno)

csrdef *lda;
csrdef *cur;
char *fname;
int lineno;
{
    text msg[2048];

    if (cur->rc) {
        oerhms (lda, cur->rc, msg, 2048);
        printf ("TPC-C load error: %s\n", msg);
        printf ("Module: %s Line: %d\n", fname, lineno);
    }
}

#endif /* ORA7 */

/*
log error message in error file

*/
void logerr(errtxt, fname, lineno)
char *errtxt;
char *fname;
int lineno;
{
#ifdef TUX
    printf("%s", errtxt);
#else
    char error_filename[30];

    printf("%s", errtxt);
    if (firstlog) {
        sprintf(error_filename, "%s_%d.err", run_name, proc_
no);
        if ((errfp = fopen(error_filename, "w")) == NULL)
        {
            printf("fopen error on file '%s'\n", error_filename);
        }
    }
}

```

```

        terminate(1);
    };
    firstlog = FALSE;
};
fprintf(errfp, "%s\n", errtxt);
fprintf(errfp, "filename = %s, lineno
= %d\n", fname, lineno);
#endif
}

===== pldel.c =====

#ifdef RCSID
static char *RCSID =
"$Header: pldel.c 7030100.5 96/06/24 16:26:06 plai
Generic<base> $ Copyr (c) 1994 Oracle";
#endif /* RCSID */

/*=====
=====+
| Copyright (c) 1996 Oracle Corp, Redwood
| Shores, CA |
| OPEN SYSTEMS PERFORMANCE
GROUP |
| All Rights Reserved |
+=====
=====+
| FILENAME
| pldel.c
| DESCRIPTION
| OCI version of DELIVERY transaction in TPC-C
benchmark.
+=====
=====*/

#include "tpcc.h"
#include "tpcc_info.h"
#ifdef TUX
#include <userlog.h>
#endif

/*
extern int userlog();
*/

#include "tpccflags.h"

#ifdef ISO || defined(ISO5) || defined(ISO6) ||
defined(ISO8)
#define SQLTXT0 "SELECT substr(value,1,5) FROM
v$parameter \
WHERE name = 'instance_number'"
#endif

#ifdef PLSQLDEL
#define SQLTXT "BEGIN delivery.deliver
(:w_id, :carrier_id, :order_id, \
:retry); END;"
#else
#ifdef DMLRETDEL
#define SQLTXT1 "DELETE FROM nord WHERE
no_d_id = :d_id \
AND no_w_id = :w_id and rownum <= 1 \
RETURNING no_o_id into :o_id "
#else
#define SQLTXT1A "\
SELECT /*+ USE_NL(nord odr) ORDERED */ 1,
no_o_id, nord.rowid, o_c_id, odr.rowid \
FROM nord, odr \
WHERE no_w_id = :w_id AND no_d_id = 1 AND
o_w_id = :w_id AND o_d_id = 1 AND \
o_id = no_o_id AND rownum <= 1 UNION ALL \
"

#define SQLTXT1B "\
SELECT /*+ USE_NL(nord odr) ORDERED */ 2,
no_o_id, nord.rowid, o_c_id, odr.rowid \
FROM nord, odr \
WHERE no_w_id = :w_id AND no_d_id = 2 AND
o_w_id = :w_id AND o_d_id = 2 AND \
o_id = no_o_id AND rownum <= 1 UNION ALL \
"

#define SQLTXT1C "\
SELECT /*+ USE_NL(nord odr) ORDERED */ 3,
no_o_id, nord.rowid, o_c_id, odr.rowid \
FROM nord, odr \
WHERE no_w_id = :w_id AND no_d_id = 3 AND
o_w_id = :w_id AND o_d_id = 3 AND \
o_id = no_o_id AND rownum <= 1 UNION ALL \
"

#define SQLTXT1D "\
SELECT /*+ USE_NL(nord odr) ORDERED */ 4,
no_o_id, nord.rowid, o_c_id, odr.rowid \
FROM nord, odr \
WHERE no_w_id = :w_id AND no_d_id = 4 AND
o_w_id = :w_id AND o_d_id = 4 AND \
o_id = no_o_id AND rownum <= 1 UNION ALL \
"

#define SQLTXT1E "\
SELECT /*+ USE_NL(nord odr) ORDERED */ 5,
no_o_id, nord.rowid, o_c_id, odr.rowid \
FROM nord, odr \
WHERE no_w_id = :w_id AND no_d_id = 5 AND
o_w_id = :w_id AND o_d_id = 5 AND \
o_id = no_o_id AND rownum <= 1 UNION ALL \
"

#define SQLTXT1F "\
SELECT /*+ USE_NL(nord odr) ORDERED */ 6,
no_o_id, nord.rowid, o_c_id, odr.rowid \
FROM nord, odr \
WHERE no_w_id = :w_id AND no_d_id = 6 AND
o_w_id = :w_id AND o_d_id = 6 AND \
o_id = no_o_id AND rownum <= 1 UNION ALL \
"

#define SQLTXT1G "\
SELECT /*+ USE_NL(nord odr) ORDERED */ 7,
no_o_id, nord.rowid, o_c_id, odr.rowid \
FROM nord, odr \
WHERE no_w_id = :w_id AND no_d_id = 7 AND
o_w_id = :w_id AND o_d_id = 7 AND \
o_id = no_o_id AND rownum <= 1 UNION ALL \
"

#define SQLTXT1H "\
SELECT /*+ USE_NL(nord odr) ORDERED */ 8,
no_o_id, nord.rowid, o_c_id, odr.rowid \
FROM nord, odr \
WHERE no_w_id = :w_id AND no_d_id = 8 AND
o_w_id = :w_id AND o_d_id = 8 AND \
o_id = no_o_id AND rownum <= 1 UNION ALL \
"

"

#define SQLTXT11 "\
SELECT /*+ USE_NL(nord odr) ORDERED */ 9,
no_o_id, nord.rowid, o_c_id, odr.rowid \
FROM nord, odr \
WHERE no_w_id = :w_id AND no_d_id = 9 AND
o_w_id = :w_id AND o_d_id = 9 AND \
o_id = no_o_id AND rownum <= 1 UNION ALL \
"

#define SQLTXT1J "\
SELECT /*+ USE_NL(nord odr) ORDERED */ 10,
no_o_id, nord.rowid, o_c_id, odr.rowid \
FROM nord, odr \
WHERE no_w_id = :w_id AND no_d_id = 10 AND
o_w_id = :w_id AND o_d_id = 10 AND \
o_id = no_o_id AND rownum <= 1"

#define SQLTXT2 "DELETE FROM nord WHERE
rowid = :no_rowid"
#endif

#ifdef DMLRETDEL
#define SQLTXT3 "UPDATE odr SET o_carrier_id
=:carrier_id \
WHERE o_id = :o_id and o_d_id = :d_id and
o_w_id = :w_id \
returning o_c_id into :o_c_id"
#else
#define SQLTXT3 "UPDATE odr SET o_carrier_id
=:carrier_id \
WHERE rowid = :o_rowid"
#endif

#ifdef DMLRETDEL
#define SQLTXT4 "UPDATE /*+ buffer */ ordl SET
ol_delivery_d = :cr_date \
WHERE ol_w_id = :w_id AND ol_d_id = :d_id AND
ol_o_id = :o_id \
RETURNING ol_amount into :ol_amount "
#else
#define SQLTXT4 "UPDATE ordl SET ol_delivery_d
=:cr_date \
WHERE ol_w_id = :w_id AND ol_d_id = :d_id AND
ol_o_id = :o_id"

#define SQLTXT5A "\
SELECT :d_id1, SUM(ol_amount) FROM ordl WHERE
ol_w_id = :w_id AND \
ol_d_id = :d_id1 AND ol_o_id = :o_id1 UNION ALL \
SELECT :d_id2, SUM(ol_amount) FROM ordl WHERE
ol_w_id = :w_id AND \
ol_d_id = :d_id2 AND ol_o_id = :o_id2 UNION ALL \
"

#define SQLTXT5B "\
SELECT :d_id3, SUM(ol_amount) FROM ordl WHERE
ol_w_id = :w_id AND \
ol_d_id = :d_id3 AND ol_o_id = :o_id3 UNION ALL \
SELECT :d_id4, SUM(ol_amount) FROM ordl WHERE
ol_w_id = :w_id AND \
ol_d_id = :d_id4 AND ol_o_id = :o_id4 UNION ALL \
"

#define SQLTXT5C "\
SELECT :d_id5, SUM(ol_amount) FROM ordl WHERE
ol_w_id = :w_id AND \
ol_d_id = :d_id5 AND ol_o_id = :o_id5 UNION ALL \
"

```

```

SELECT :d_id6, SUM(ol_amount) FROM ordl WHERE
ol_w_id = :w_id AND \
ol_d_id = :d_id6 AND ol_o_id = :o_id6 UNION ALL \
"

#define SQLTXT5D "\
SELECT :d_id7, SUM(ol_amount) FROM ordl WHERE
ol_w_id = :w_id AND \
ol_d_id = :d_id7 AND ol_o_id = :o_id7 UNION ALL \
SELECT :d_id8, SUM(ol_amount) FROM ordl WHERE
ol_w_id = :w_id AND \
ol_d_id = :d_id8 AND ol_o_id = :o_id8 UNION ALL \
"

#define SQLTXT5E "\
SELECT :d_id9, SUM(ol_amount) FROM ordl WHERE
ol_w_id = :w_id AND \
ol_d_id = :d_id9 AND ol_o_id = :o_id9 UNION ALL \
SELECT :d_id10, SUM(ol_amount) FROM ordl
WHERE ol_w_id = :w_id AND \
ol_d_id = :d_id10 AND ol_o_id = :o_id10"

#endif
#endif /* PLSQLDEL */

#define SQLTXT6 "UPDATE cust SET c_balance =
c_balance + :amt, \
c_delivery_cnt = c_delivery_cnt + 1 WHERE c_w_id
= :w_id AND \
c_d_id = :d_id AND c_id = :c_id"

#define NDISTS 10
#define ROWIDLEN 20

struct delctx {
    sb2 del_o_id_ind[NDISTS];
    sb2 cons_ind[NDISTS];
    sb2 w_id_ind[NDISTS];
    sb2 d_id_ind[NDISTS];
    sb2 c_id_ind[NDISTS];
    sb2 del_date_ind[NDISTS];
    sb2 carrier_id_ind[NDISTS];
    sb2 amt_ind[NDISTS];
    sb2 no_rowid_ind[NDISTS];
    sb2 o_rowid_ind[NDISTS];
    #if defined(ISO) || defined(ISO5) || defined(ISO6) ||
    defined(ISO8)
        sb2 inum_ind;
    #endif
};

#ifdef DMLRETDEL
    ub4 del_o_id_len[NDISTS];
    ub4 c_id_len[NDISTS];
    int oid_ctx;
    int cid_ctx;
    OCIBind *olamt_bp;
#else
    ub2 del_o_id_len[NDISTS];
    ub2 c_id_len[NDISTS];
#endif

    ub2 cons_len[NDISTS];
    ub2 w_id_len[NDISTS];
    ub2 d_id_len[NDISTS];
    ub2 del_date_len[NDISTS];
    ub2 carrier_id_len[NDISTS];
    ub2 amt_len[NDISTS];
    ub2 no_rowid_len[NDISTS];
    ub2 no_rowid_ptr_len[NDISTS];
    ub2 o_rowid_len[NDISTS];
    ub2 o_rowid_ptr_len[NDISTS];
    #if defined(ISO) || defined(ISO5) || defined(ISO6) ||
    defined(ISO8)
        ub2 inum_len;
    #endif

    ub2 del_o_id_rcode[NDISTS];
    ub2 cons_rcode[NDISTS];
    ub2 w_id_rcode[NDISTS];
    ub2 d_id_rcode[NDISTS];
    ub2 c_id_rcode[NDISTS];
    ub2 del_date_rcode[NDISTS];
    ub2 carrier_id_rcode[NDISTS];
    ub2 amt_rcode[NDISTS];
    ub2 no_rowid_rcode[NDISTS];
    ub2 o_rowid_rcode[NDISTS];
    #if defined(ISO) || defined(ISO5) || defined(ISO6) ||
    defined(ISO8)
        ub2 inum_rcode;
    #endif

    int del_o_id[NDISTS];
    int cons[NDISTS];
    int w_id[NDISTS];
    int d_id[NDISTS];
    int c_id[NDISTS];
    int carrier_id[NDISTS];
    int amt[NDISTS];
    ub4 del_o_id_rcnt;
    int retry;
    OCIRowid *no_rowid_ptr[NDISTS];
    OCIRowid *o_rowid_ptr[NDISTS];
    OCIDate del_date[NDISTS];
    #if defined(ISO) || defined(ISO5) || defined(ISO6) ||
    defined(ISO8)
        char inum[10];
    #endif

    OCISmt *curd0;
    OCISmt *curd1;
    OCISmt *curd2;
    OCISmt *curd3;
    OCISmt *curd4;
    OCISmt *curd5;
    OCISmt *curd6;
    OCISmt *curdtest;

    OCIBind *w_id_bp;
    OCIBind *w_id_bp3;
    OCIBind *w_id_bp4;
    OCIBind *w_id_bp5;
    OCIBind *w_id_bp6;
    OCIBind *d_id_bp;
    OCIBind *d_id_bp3;
    OCIBind *d_id_bp4;
    OCIBind *d_id_bp5;
    OCIBind *d_id_bp6;
    OCIBind *o_id_bp;
    OCIBind *cr_date_bp;
    OCIBind *c_id_bp;
    OCIBind *c_id_bp3;
    OCIBind *no_rowid_bp;
    OCIBind *carrier_id_bp;
    OCIBind *o_rowid_bp;
    OCIBind *del_o_id_bp;
    OCIBind *del_o_id_bp3;
    OCIBind *amt_bp;
    OCIBind *bstr1_bp[10];
    OCIBind *bstr2_bp[10];

    OCIBind *retry_bp;
    OCIDefine *inum_dp;
    OCIDefine *d_id_dp;
    OCIDefine *del_o_id_dp;
    OCIDefine *no_rowid_dp;
    OCIDefine *c_id_dp;
    OCIDefine *o_rowid_dp;
    OCIDefine *cons_dp;
    OCIDefine *amt_dp;

    int norow;
};

typedef struct delctx delctx;

delctx *dctx;

#ifdef DMLRETDEL
    struct amtctx {
        int ol_amt[NDISTS][NITEMS];
        sb2 ol_amt_ind[NDISTS][NITEMS];
        ub4 ol_amt_len[NDISTS][NITEMS];
        ub2 ol_amt_rcode[NDISTS][NITEMS];
        int ol_cnt[NDISTS];
    };
    typedef struct amtctx amtctx;
    amtctx *actx;
#endif

#ifdef DMLRETDEL
    extern sb4 no_data();

    sb4 TPC_oid_data(dvoid *ctxp, OCIBind *bp, ub4 iter,
    ub4 index,
        dvoid **bufpp, ub4 **alenp, ub1 *piecep,
        dvoid **indpp, ub2 **rcodepp)
    {
        *bufpp = &dctx->del_o_id[indp];
        *indpp = &dctx->del_o_id_ind[indp];
        dctx->del_o_id_len[indp] = sizeof(dctx->del_o_id[indp]);
        *alenp = &dctx->del_o_id_len[indp];
        *rcodepp = &dctx->del_o_id_rcode[indp];
        *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[indp];
        *indpp = &dctx->c_id_ind[indp];
        dctx->c_id_len[indp] = sizeof(dctx->c_id[indp]);
        *alenp = &dctx->c_id_len[indp];
        *rcodepp = &dctx->c_id_rcode[indp];
        *piecep = OCI_ONE_PIECE;
        return (OCI_CONTINUE);
    }

    sb4 amt_data(dvoid *ctxp, OCIBind *bp, ub4 iter, ub4
    index,
        dvoid **bufpp, ub4 **alenp, ub1 *piecep,
        dvoid **indpp, ub2 **rcodepp)
    {
        amtctx *actx;
        actx = (amtctx *)ctxp;
    }
#endif

```



```

actx->ol_cnt[iter]=actx->ol_cnt[iter]+1;
*bufpp = &actx->ol_amt[iter][index];
*indpp= &actx->ol_amt_ind[iter][index];
actx->ol_amt_len[iter][index]=sizeof(actx-
>ol_amt[0][0]);
*alenp= &actx->ol_amt_len[iter][index];
*rcodepp = &actx->ol_amt_rcode[iter][index];
*piecep =OCI_ONE_PIECE;
return (OCI_CONTINUE);
}

#endif

tkvcldinit ()
{
# ifndef DMLRETDDEL
int i,j;
char bstr1[10];
char bstr2[10];
# endif /* !DMLRETDDEL */
text stmbuff[SQL_BUF_SIZE];

dctx = (delctx *) malloc (sizeof(delctx));
memset(dctx,(char)0,sizeof(delctx));
dctx->norow = 0;
#ifdef DMLRETDDEL
actx = (amtctx *) malloc (sizeof(amtctx));
memset(actx,(char)0,sizeof(amtctx));
#else
for(i=0;i<NDISTS;i++) {
OCIERROR(errhp,
OCIDescriptorAlloc(tpcenv,(dvoid*)&dctx-
>o_rowid_ptr[i],
OCI_DTYPE_ROWID,0,(dvoid**)0);
OCIERROR(errhp,
OCIDescriptorAlloc(tpcenv,(dvoid*)&dctx-
>no_rowid_ptr[i],
OCI_DTYPE_ROWID,0,(dvoid**)0);
}
#endif

#if defined(ISO) || defined(ISO5) || defined(ISO6) ||
defined(ISO8)
OCIHandleAlloc(tpcenv, (dvoid *)&(dctx->curd0),
OCI_HTYPE_STMT, 0, (dvoid**)0);
sprintf ((char *) stmbuff, SQLTXT0);
OCIStmtPrepare(dctx->curd0, errhp, stmbuff,
strlen((char *)stmbuff),OCI_NTV_SYNTAX,
OCI_DEFAULT);

OCIDFNRA(dctx->curd0, dctx->inum_dp,errhp,1,dctx-
>inum,SIZ(dctx->inum),SQLT_STR,
&(dctx->inum_ind),&(dctx->inum_len),&(dctx-
>inum_rcode));
#endif

/* If PLSQDEL and ISO? are both defined, then they
both try to use
curd0! This could cause a problem. Will try to fix later
- VMM 12/30/97 */

#ifdef PLSQDEL
OCIHandleAlloc(tpcenv, (dvoid *)&(dctx->curd0),
OCI_HTYPE_STMT,
0, (dvoid**)0);
sprintf ((char *) stmbuff, SQLTXT);
OCIStmtPrepare(dctx->curd0, errhp, stmbuff,
strlen((char *)stmbuff),
OCI_NTV_SYNTAX, OCI_DEFAULT);
OCIBND(dctx->curd0, dctx->w_id_bp ,
errhp,"w_id",ADR(w_id),SIZ(int),
SQLT_INT);
OCIBNDRA(dctx->curd0, dctx->carrier_id_bp ,
errhp,"carrier_id",
ADR(dctx->carrier_id), SIZ(int), SQLT_INT);

OCIBNDRAA(dctx->curd0, dctx->o_id_bp, errhp,
"order_id",
dctx->del_o_id,SIZ(int),SQLT_INT, dctx-
>del_o_id_ind,
dctx->del_o_id_len,dctx-
>del_o_id_rcode,NDISTS,
&dctx->del_o_id_rcnt);
OCIBND(dctx->curd0, dctx->retry_bp ,
errhp,"retry",ADR(dctx->retry),
SIZ(int),SQLT_INT);
#else
#ifdef DMLRETDDEL
OCIHandleAlloc(tpcenv, (dvoid *)&(dctx->curd1),
OCI_HTYPE_STMT, 0, (dvoid**)0);
sprintf ((char *) stmbuff, "%s", SQLTXT1);
OCIStmtPrepare(dctx->curd1, errhp, stmbuff,
strlen((char *)stmbuff),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);

OCIBNDRAA(dctx->curd1, dctx->del_o_id_bp, errhp,
"o_id",
SIZ(int),SQLT_INT,NULL,
&dctx->oid_ctx,no_data,TPC_oid_data);
#else
OCIHandleAlloc(tpcenv, (dvoid *)&(dctx->curd1),
OCI_HTYPE_STMT, 0, (dvoid**)0);
sprintf ((char *) stmbuff,
"%s%s%s%s%s%s%s%s%s", SQLTXT1A,
SQLTXT1B,
SQLTXT1C,
SQLTXT1D,
SQLTXT1E,
SQLTXT1F,
SQLTXT1G,
SQLTXT1H,
SQLTXT1I,
SQLTXT1J
);
OCIStmtPrepare(dctx->curd1, errhp, stmbuff,
strlen((char *)stmbuff),OCI_NTV_SYNTAX,
OCI_DEFAULT);

OCIERROR(errhp,
OCIAttrSet(dctx-
>curd1,OCI_HTYPE_STMT,(dvoid*)&dctx->norow,0,
OCI_ATTR_PREFETCH_ROWS,errhp));

/* bind variables */
OCIBND(dctx->curd1, dctx-
>w_id_bp,errhp,"w_id",ADR(w_id),SIZ(int),SQLT_INT);
OCIDFNRA(dctx->curd1, dctx->d_id_dp,errhp,1,dctx-
>d_id,SIZ(int),
SQLT_INT, dctx->d_id_ind,dctx->d_id_len,dctx-
>d_id_rcode);
OCIDFNRA(dctx->curd1, dctx-
>del_o_id_dp,errhp,2,dctx->del_o_id,
SIZ(int), SQLT_INT,dctx->del_o_id_ind,
dctx->del_o_id_len, dctx->del_o_id_rcode);
OCIDFNRA(dctx->curd1, dctx-
>no_rowid_dp,errhp,3,dctx->no_rowid_ptr,
SIZ(OCIRowid *), SQLT_RDD,dctx-
>no_rowid_ind,
dctx->no_rowid_len, dctx->no_rowid_rcode);
OCIDFNRA(dctx->curd1, dctx->c_id_dp,errhp,4,dctx-
>c_id,SIZ(dctx->c_id[0]),
SQLT_INT, dctx->c_id_ind,dctx->c_id_len,dctx-
>c_id_rcode);
OCIDFNRA(dctx->curd1, dctx-
>o_rowid_dp,errhp,5,dctx->o_rowid_ptr,
SIZ(OCIRowid *), SQLT_RDD,dctx-
>o_rowid_ind,
dctx->o_rowid_len, dctx->o_rowid_rcode);

/* open second cursor */
OCIHandleAlloc(tpcenv, (dvoid *)&(dctx->curd2),
OCI_HTYPE_STMT, 0, (dvoid**)0);
sprintf ((char *) stmbuff, SQLTXT2);
OCIStmtPrepare(dctx->curd2, errhp, stmbuff,
strlen((char *)stmbuff),
OCI_NTV_SYNTAX, OCI_DEFAULT);

/* bind variables */
OCIBNDRA(dctx->curd2, dctx-
>no_rowid_bp,errhp,"no_rowid",&(dctx-
>no_rowid_ptr[0]),
SIZ(dctx->no_rowid_ptr[0]),SQLT_RDD,dctx-
>no_rowid_ind,
dctx->no_rowid_len,dctx->no_rowid_rcode);
#endif /* DMLRETDDEL */

/* open third cursor */
OCIHandleAlloc(tpcenv, (dvoid *)&(dctx->curd3),
OCI_HTYPE_STMT, 0, (dvoid**)0);
sprintf ((char *) stmbuff, SQLTXT3);
OCIStmtPrepare(dctx->curd3, errhp, stmbuff,
strlen((char *)stmbuff),
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);
#endif DMLRETDDEL
OCIBNDRA(dctx->curd3, dctx->w_id_bp3, errhp,
"w_id", dctx->w_id,SIZ(int),
SQLT_INT, NULL, NULL, NULL);
OCIBNDRA(dctx->curd3, dctx->d_id_bp3, errhp,
"d_id", dctx->d_id,SIZ(int),

```

```

        SQLT_INT,NULL, NULL, NULL);
    OCIBNDRA(dctx->curd3, dctx->del_o_id_bp3, errhp,
"o_id", dctx->del_o_id,
        SIZ(int), SQLT_INT,NULL,NULL,NULL);
    OCIBNDRA(dctx->curd3, dctx->c_id_bp3, errhp,
"o_c_id", SIZ(int),
        SQLT_INT,NULL,&dctx->cid_ctx,no_data,
cid_data);
# else

    OCIBNDRA(dctx->curd3, dctx-
>o_rowid_bp,errhp,"o_rowid",&(dctx->o_rowid_ptr[0]),
        SIZ(dctx->o_rowid_ptr[0]),SQLT_RDD,dctx-
>o_rowid_ind,
        dctx->o_rowid_ptr_len,dctx->o_rowid_rcode);
#endif

    /* open fourth cursor */

    OCIHandleAlloc(tpcenv, (dvoid **)&(dctx->curd4),
OCI_HTYPE_STMT, 0, (dvoid**)0);
    sprintf((char *) stmbuf, SQLTXT4);
    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);
# ifdef DMLRETDEL
    OCIBNDRAD(dctx->curd4, dctx->olamt_bp, errhp,
"ol_amount",
        SIZ(int), SQLT_INT,NULL,
actx,no_data,amt_data);
# else

    /* open fifth cursor */

    OCIHandleAlloc(tpcenv, (dvoid **)&(dctx->curd5),
OCI_HTYPE_STMT, 0, (dvoid**)0);
    sprintf((char *) stmbuf, "%s%s%s%s%s",
        SQLTXT5A,
                                SQLTXT5B,
                                SQLTXT5C,
                                SQLTXT5D,
                                SQLTXT5E
);
    OCISmtPrepare(dctx->curd5, errhp, stmbuf,
strlen((char *)stmbuf),
        OCI_NTV_SYNTAX, OCI_DEFAULT);

    OCIERROR(errhp,
    OCIAttrSet(dctx-
>curd5,OCI_HTYPE_STMT,(dvoid*)&dctx->norow,0,
    OCI_ATTR_PREFETCH_ROWS,errhp));

    /* bind variables */

    OCIBND(dctx->curd5,dctx-
>w_id_bp,errhp,"w_id",ADR(w_id),SIZ(w_id),SQLT_IN
T);
    for (i = 0; i < NDISTS; i++) {
        sprintf (bstr1, "d_id%d", i + 1);
        sprintf (bstr2, "o_id%d", i + 1);
        OCIBNDRA(dctx->curd5,dctx-
>bstr1_bp[i],errhp,bstr1,ADR(dctx->d_id[i]),
            SIZ(dctx->d_id[0]),SQLT_INT, &(dctx-
>d_id_ind[i]),
            &(dctx->d_id_len[i]),&(dctx->d_id_rcode[i]));
        OCIBNDRA(dctx->curd5,dctx-
>bstr2_bp[i],errhp,bstr2,ADR(dctx->del_o_id[i]),
            SIZ(dctx->del_o_id[0]),SQLT_INT, &(dctx-
>del_o_id_ind[i]),
            &(dctx->del_o_id_len[i]),&(dctx-
>del_o_id_rcode[i]));
    }

    OCIDFNRA(dctx->curd5,dctx->cons_dp,errhp,1,dctx-
>cons,SIZ(dctx->cons[0]),SQLT_INT,
        dctx->cons_ind,dctx->cons_len,dctx-
>cons_rcode);
    OCIDFNRA(dctx->curd5,dctx->amt_dp,errhp,2,dctx-
>amt,SIZ(dctx->amt[0]),SQLT_INT,
        dctx->amt_ind,dctx->amt_len,dctx->amt_rcode);
#endif
    /* open sixth cursor */

    OCIHandleAlloc(tpcenv, (dvoid **)&(dctx->curd6),
OCI_HTYPE_STMT, 0, (dvoid**)0);
    sprintf((char *) stmbuf, SQLTXT6);
    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);
#endif
    return (0);
}

void shiftdata(from)
int from ;
{
    int i;
    for (i=from;i<NDISTS-1; i++)
    {
        dctx->del_o_id_ind[i] = dctx->del_o_id_ind[i+1];
        dctx->del_o_id[i] = dctx->del_o_id[i+1];
        dctx->w_id[i] = dctx->w_id[i+1];
        dctx->d_id[i] = dctx->d_id[i+1];
        dctx->carrier_id[i] = dctx->carrier_id[i+1];
    }
}

tkvcd ()
{
    int i, j;
    int rpc,rcount,count;
    int invalid;
# ifndef DMLRETDEL
    int tmp_id,v;
    int tmp_amt;
# endif /* !DMLRETDEL */

# if defined(ISO) || defined(ISO5) || defined(ISO6) ||
defined(ISO8)
    int hasno;
    int reread;
    char sdate[30];

    OCISmtExecute(tpcsvc,dctx-
>curd0,errhp,1,0,0,OCI_DEFAULT);
    sysdate (sdate);
    printf ("Delivery started at %s on node %s\n", sdate,
dctx->inum);
#endif
# ifdef PLSQDEL
    for (i = 0; i < NDISTS; i++)
    {
        dctx->del_o_id_ind[i] = TRUE;
        dctx->del_o_id_len[i] = sizeof(int);
    }

    OCIERROR(errhp,
    OCISmtExecute(tpcsvc,dctx-
>curd0,errhp,1,0,0,OCI_DEFAULT));

    for (i = 0; i < NDISTS; i++)
    {
        del_o_id[i] = 0;
        if (dctx->del_o_id_ind[i] == 0)
        {
            del_o_id[i] = dctx->del_o_id[i];
        }
    }
# else

    retry:

# if defined(ISO) || defined(ISO5) || defined(ISO6) ||
defined(ISO8)
        reread = 1;
# endif

# if defined(ISO) || defined(ISO5) || defined(ISO6) ||
defined(ISO8)
        iso:
# endif

        invalid = 0;

        /* initialization for array operations */

        for (i = 0; i < NDISTS; i++) {
            dctx->del_o_id_ind[i] = TRUE;
            dctx->cons_ind[i] = TRUE;

```

```

dctx->w_id_ind[i] = TRUE;
dctx->d_id_ind[i] = TRUE;
dctx->c_id_ind[i] = TRUE;
dctx->del_date_ind[i] = TRUE;
dctx->carrier_id_ind[i] = TRUE;
dctx->amt_ind[i] = TRUE;
dctx->no_rowid_ind[i] = TRUE;
dctx->o_rowid_ind[i] = TRUE;

dctx->del_o_id_len[i] = SIZ(dctx->del_o_id[0]);
dctx->cons_len[i] = SIZ(dctx->cons[0]);
dctx->w_id_len[i] = SIZ(dctx->w_id[0]);
dctx->d_id_len[i] = SIZ(dctx->d_id[0]);
dctx->c_id_len[i] = SIZ(dctx->c_id[0]);
dctx->del_date_len[i] = DEL_DATE_LEN;
dctx->carrier_id_len[i] = SIZ(dctx->carrier_id[0]);
dctx->amt_len[i] = SIZ(dctx->amt[0]);
dctx->no_rowid_len[i] = ROWIDLEN;
dctx->o_rowid_len[i] = ROWIDLEN;
dctx->o_rowid_ptr_len[i] = SIZ(dctx-
>o_rowid_ptr[0]);
dctx->no_rowid_ptr_len[i] = SIZ(dctx-
>no_rowid_ptr[0]);

dctx->w_id[i] = w_id;
dctx->d_id[i] = i+1;
dctx->carrier_id[i] = o_carrier_id;
memcpy(&dctx-
>del_date[i],&cr_date,sizeof(OCIDate));
}

#ifdef DMLRETDEL /* VMM 1/13/98 */
memset(acts,char)0,sizeof(amtctx);
#endif /* DMLRETDEL */
/* array select from new_order and orders tables */

execstatus=OCISmtExecute(tpcsvc,dctx-
>curd1,errhp,NDISTS,0,0,OCI_DEFAULT);
if((execstatus != OCI_SUCCESS) && (execstatus !=
OCI_NO_DATA)) {
OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
errcode = OCIERROR(errhp,execstatus);
if(errcode == NOT_SERIALIZABLE) {
retries++;
goto retry;
} else if (errcode == RECOVERR) {
retries++;
goto retry;
} else if (errcode == SNAPSHOT_TOO_OLD) {
retries++;
goto retry;
} else {
return -1;
}
}
/* mark districts with no new order */
OCIAttrGet(dctx-
>curd1,OCI_HTYPE_STMT,&rcount,0,OCI_ATTR_RO
W_COUNT,errhp);
rpc = rcount;
#ifdef DMLRETDEL /* we have to compress the array
here */
if (rcount != NDISTS)
{
int j = 0;
for (i=0; i < NDISTS; i++)
{
if (dctx->del_o_id_ind[j] == 0) /* there is data here */
j++;
else
shiftdata();
}
}
#else
invalid = NDISTS - rcount;
for (i = rpc; i < NDISTS; i++) {
dctx->del_o_id_ind[i] = NA;
dctx->w_id_ind[i] = NA;
dctx->d_id_ind[i] = NA;
dctx->c_id_ind[i] = NA;
dctx->carrier_id_ind[i] = NA;
dctx->no_rowid_ind[i] = NA;
dctx->o_rowid_ind[i] = NA;
}
#endif

#if defined(ISO) || defined(ISO5) || defined(ISO6) ||
defined(ISO8)
if (invalid) {
sysdate (sdate);
for (i = 1; i <= NDISTS; i++) {
hasno = 0;
for (j = 0; j < rpc; j++) {
if (dctx->d_id[j] == i) {
hasno = 1;
break;
}
}
if (!hasno)
printf ("Delivery [dist %d] found no new order
at %s\n", i, sdate);
}
}
if (reread) {
sleep (60);
sysdate (sdate);
printf ("Delivery wake up at %s\n", sdate);
reread = 0;
goto iso;
}
}
#endif

#ifdef DMLRETDEL
/* array delete of new_order table */
execstatus=OCISmtExecute(tpcsvc,dctx-
>curd2,errhp,rpc,0,0,OCI_DEFAULT);
if(execstatus != OCI_SUCCESS) {
OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
errcode = OCIERROR(errhp,execstatus);
if(errcode == NOT_SERIALIZABLE) {
retries++;
goto retry;
} else if (errcode == RECOVERR) {
retries++;
goto retry;
} else if (errcode == SNAPSHOT_TOO_OLD) {
retries++;
goto retry;
} else {
return -1;
}
}

/* mark districts with no new order */
OCIAttrGet(dctx-
>curd2,OCI_HTYPE_STMT,&rcount,0,OCI_ATTR_RO
W_COUNT,errhp);

```

```

    }
}
#endif DMLRETDL
OCIAttrGet(dctx-
>curd4,OCI_HTYPE_STMT,&rcount,NULL,OCI_ATTR
_ROW_COUNT,errhp);
/* add up amounts */
count=0;
for (i=0;i<rpc;i++)
{
dctx->amt[i]=0;
for (j=0;j<actx->ol_cnt[i];j++)
if (actx->ol_amt_rcode[i][j] == 0)
{
dctx->amt[i] = dctx->amt[i] + actx->ol_amt[i][j];
count = count+1;
}
}
if (rcount > rpc*NITEMS) {
printf ("Error in TPC-C server %d: %d ordns
updated, %d ordl updated\n",
proc_no, rpc, rcount);
}
#else
/* array select from order_line table */
execstatus=OCIStmtExecute(tpscvc,dctx-
>curd5,errhp,rcp,0,0,OCI_DEFAULT);
if((execstatus != OCI_SUCCESS) && (execstatus !=
OCI_NO_DATA)) {
OCITransRollback(tpscvc,errhp,OCI_DEFAULT);
errcode = OCIERROR(errhp,execstatus);
if(errcode == NOT_SERIALIZABLE) {
retries++;
goto retry;
} else if (errcode == RECOVER) {
retries++;
goto retry;
} else if (errcode == SNAPSHOT_TOO_OLD) {
retries++;
goto retry;
} else {
return -1;
}
}

OCIAttrGet(dctx-
>curd5,OCI_HTYPE_STMT,&rcount,OCI_ATTR_RO
W_COUNT,errhp);
if (rcount != rpc) {
#ifdef TUX
printf ("Error in TPC-C server %d: %d rows
selected, %d ordl selected\n",
proc_no, rpc, rcount);
#else
printf ("Error in TPC-C server %d: %d rows
selected, %d ordl selected\n",
proc_no, rpc, rcount);
#endif
OCITransRollback(tpscvc,errhp,OCI_DEFAULT);
return (-1);
}

/* reorder amount selected if necessary */

for (i = 0; i < rpc; i++) {
if (dctx->cons[i] != dctx->d_id[i]) {
#ifdef TUX
printf ("TPC-C server %d: reordering amount\n",
proc_no);
#else
printf ("TPC-C server %d: reordering amount\n",
proc_no);
#endif
for (j = i + 1; j < rpc; j++) {
if (dctx->cons[j] == dctx->d_id[i]) {
tmp_id = dctx->cons[j];
dctx->cons[j] = dctx->cons[i];
dctx->cons[i] = tmp_id;
tmp_amt = dctx->amt[i];
dctx->amt[i] = dctx->amt[j];
dctx->amt[j] = tmp_amt;
break;
}
}
if (j >= rpc) {
#ifdef TUX
printf ("Error in TPC-C server %d: missing
ordl?\n", proc_no);
#else
printf ("Error in TPC-C server %d: missing
ordl?\n", proc_no);
#endif
OCITransRollback(tpscvc,errhp,OCI_DEFAULT);
return (-1);
}
}
#ifdef TUX
printf ("Error in TPC-C server %d: missing
ordl?\n", proc_no);
#endif
OCITransRollback(tpscvc,errhp,OCI_DEFAULT);
return (-1);
}
}
#ifdef TUX
printf ("Error in TPC-C server %d: missing
ordl?\n", proc_no);
#endif
OCITransRollback(tpscvc,errhp,OCI_DEFAULT);
return (-1);
}
}

#ifdef ISO5 || defined(ISO6)
sysdate (sdate);
#endif ISO5
printf ("Delivery sleep before commit at %s\n", sdate);
#else
printf ("Delivery sleep before abort at %s\n", sdate);
#endif
sleep (60);
sysdate (sdate);
printf ("Delivery wake up at %s\n", sdate);
#endif

#ifdef ISO6
printf("Delivery ISO6 Rolling back.\n");
OCITransRollback(tpscvc,errhp,OCI_DEFAULT);
#endif

#ifdef ISO5
OCITransCommit(tpscvc, errhp, OCI_DEFAULT);
#endif

#ifdef ISO5 || defined(ISO6)
sysdate (sdate);
printf ("Delivery completed at: %s\n", sdate);
#endif

/* return o_id's in district id order */
for (i = 0; i < NDISTS; i++)
del_o_id[i] = 0;
for (i = 0; i < rpc; i++)
del_o_id[dctx->d_id[i] - 1] = dctx->del_o_id[i];
#endif

return (0);
}

void tkvcddone ()
{
if (dctx)
{
#ifdef ISO || defined(ISO5) || defined(ISO6) ||
defined(ISO8)
OCIHandleFree((dvoid *)dctx-
>curd0,OCI_HTYPE_STMT);
#endif
}
}

```

```

#ifdef PLSQLDEL
    OCIHandleFree((dvoid *)dctx-
>curd0,OCI_HTYPE_STMT);
#else
    /* Again the above will cause a problem if both
    PSQLDEL and ISO are
        defined - VMM 12/30/97 */
    OCIHandleFree((dvoid *)dctx-
>curd1,OCI_HTYPE_STMT);
    OCIHandleFree((dvoid *)dctx-
>curd2,OCI_HTYPE_STMT);
    OCIHandleFree((dvoid *)dctx-
>curd3,OCI_HTYPE_STMT);
    OCIHandleFree((dvoid *)dctx-
>curd4,OCI_HTYPE_STMT);
    OCIHandleFree((dvoid *)dctx-
>curd5,OCI_HTYPE_STMT);
    OCIHandleFree((dvoid *)dctx-
>curd6,OCI_HTYPE_STMT);
#endif
    free (dctx);
}

===== plnew.c =====

#ifdef RCSID
static char *RCSid =
    "$Header: 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 "tpcc.h"
#include "tpcc_info.h"
#ifdef TUX
#include <userlog.h>
#endif
#include "tpccflags.h"

extern void userlog();
extern int SellItemStk(rpc3);
extern int UpdStk2 (rpc3);

#ifdef PLSQLNO
#define SQLTXT2 "BEGIN initnew.new_init(:idx1arr);
END;"
#else
#define SQLTXT2 "UPDATE stok SET s_order_cnt =
s_order_cnt + 1, \
s_ytd = s_ytd + :ol_quantity, s_remote_cnt =
s_remote_cnt + :s_remote, \
s_quantity = :s_quantity \
WHERE rowid = :s_rowid"

/*
#define SQLTXT3 "\
SELECT
0,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_data,
s_quantity \
FROM item,stok WHERE i_id = :10 AND s_w_id = :30
AND s_i_id = i_id UNION ALL \
SELECT
1,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_data,
s_quantity \
FROM item,stok WHERE i_id = :11 AND s_w_id = :31
AND s_i_id = i_id UNION ALL \
SELECT
2,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_data,
s_quantity \
FROM item,stok WHERE i_id = :12 AND s_w_id = :32
AND s_i_id = i_id UNION ALL \
SELECT
3,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_data,
s_quantity \
FROM item,stok WHERE i_id = :13 AND s_w_id = :33
AND s_i_id = i_id UNION ALL \
SELECT
4,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_data,
s_quantity \
FROM item,stok WHERE i_id = :14 AND s_w_id = :34
AND s_i_id = i_id UNION ALL \
SELECT
5,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_data,
s_quantity \
FROM item,stok WHERE i_id = :15 AND s_w_id = :35
AND s_i_id = i_id UNION ALL \
SELECT
6,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_data,
s_quantity \
FROM item,stok WHERE i_id = :16 AND s_w_id = :36
AND s_i_id = i_id UNION ALL \
SELECT
7,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_data,
s_quantity \
FROM item,stok WHERE i_id = :17 AND s_w_id = :37
AND s_i_id = i_id UNION ALL \
SELECT
8,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_data,
s_quantity \
FROM item,stok WHERE i_id = :18 AND s_w_id = :38
AND s_i_id = i_id UNION ALL \
SELECT
9,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_data,
s_quantity \
FROM item,stok WHERE i_id = :19 AND s_w_id = :39
AND s_i_id = i_id UNION ALL \
SELECT
10,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_dat
a,s_quantity \
FROM item,stok WHERE i_id = :20 AND s_w_id = :40
AND s_i_id = i_id UNION ALL \
SELECT
11,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_dat
a,s_quantity \
FROM item,stok WHERE i_id = :21 AND s_w_id = :41
AND s_i_id = i_id UNION ALL
SELECT
12,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_dat
a,s_quantity \
FROM item,stok WHERE i_id = :22 AND s_w_id = :42
AND s_i_id = i_id UNION ALL \
SELECT
13,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_dat
a,s_quantity \
FROM item,stok WHERE i_id = :23 AND s_w_id = :43
AND s_i_id = i_id UNION ALL \
SELECT
14,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_dat
a,s_quantity \
FROM item,stok WHERE i_id = :24 AND s_w_id = :44
AND s_i_id = i_id"
*/

#define SQLTXT3_1 "\
SELECT
0,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_data,
s_quantity \
FROM item,stok WHERE i_id = :10 AND s_w_id = :30
AND s_i_id = i_id UNION ALL \
SELECT
1,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_data,
s_quantity \
FROM item,stok WHERE i_id = :11 AND s_w_id = :31
AND s_i_id = i_id UNION ALL \
SELECT
2,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_data,
s_quantity \
FROM item,stok WHERE i_id = :12 AND s_w_id = :32
AND s_i_id = i_id UNION ALL \
SELECT
3,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_data,
s_quantity \
FROM item,stok WHERE i_id = :13 AND s_w_id = :33
AND s_i_id = i_id UNION ALL \
SELECT
4,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_data,
s_quantity \
FROM item,stok WHERE i_id = :14 AND s_w_id = :34
AND s_i_id = i_id UNION ALL \
SELECT
5,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_data,
s_quantity \
FROM item,stok WHERE i_id = :15 AND s_w_id = :35
AND s_i_id = i_id UNION ALL \
SELECT
6,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_data,
s_quantity \
FROM item,stok WHERE i_id = :16 AND s_w_id = :36
AND s_i_id = i_id UNION ALL \
SELECT
7,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_data,
s_quantity \
FROM item,stok WHERE i_id = :17 AND s_w_id = :37
AND s_i_id = i_id UNION ALL \
SELECT
8,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_data,
s_quantity \
FROM item,stok WHERE i_id = :18 AND s_w_id = :38
AND s_i_id = i_id UNION ALL \
SELECT
9,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_data,
s_quantity \
FROM item,stok WHERE i_id = :19 AND s_w_id = :39
AND s_i_id = i_id UNION ALL \
SELECT
10,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_dat
a,s_quantity \
FROM item,stok WHERE i_id = :20 AND s_w_id = :40
AND s_i_id = i_id UNION ALL \
SELECT
11,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_dat
a,s_quantity \
FROM item,stok WHERE i_id = :21 AND s_w_id = :41
AND s_i_id = i_id UNION ALL \
SELECT
12,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_dat
a,s_quantity \
FROM item,stok WHERE i_id = :22 AND s_w_id = :42
AND s_i_id = i_id UNION ALL \
SELECT
13,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_dat
a,s_quantity \
FROM item,stok WHERE i_id = :23 AND s_w_id = :43
AND s_i_id = i_id UNION ALL \
SELECT
14,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_dat
a,s_quantity \
FROM item,stok WHERE i_id = :24 AND s_w_id = :44
AND s_i_id = i_id"
*/

#define SQLTXT3_2 "\
SELECT
8,stok.rowid,i_price,i_name,i_data,s_dist_%02d,s_data,
s_quantity \
FROM item,stok WHERE i_id = :18 AND s_w_id = :38
AND s_i_id = i_id UNION ALL \

```

```

SELECT
9,stk.rowid,i_price,i_name,i_data,s_dist_%02d,s_data,
s_quantity \
FROM item,stk WHERE i_id = :19 AND s_w_id = :39
AND s_i_id = i_id UNION ALL \
SELECT
10,stk.rowid,i_price,i_name,i_data,s_dist_%02d,s_data
a,s_quantity \
FROM item,stk WHERE i_id = :20 AND s_w_id = :40
AND s_i_id = i_id UNION ALL \
SELECT
11,stk.rowid,i_price,i_name,i_data,s_dist_%02d,s_data
a,s_quantity \
FROM item,stk WHERE i_id = :21 AND s_w_id = :41
AND s_i_id = i_id UNION ALL \
SELECT
12,stk.rowid,i_price,i_name,i_data,s_dist_%02d,s_data
a,s_quantity \
FROM item,stk WHERE i_id = :22 AND s_w_id = :42
AND s_i_id = i_id UNION ALL \
SELECT
13,stk.rowid,i_price,i_name,i_data,s_dist_%02d,s_data
a,s_quantity \
FROM item,stk WHERE i_id = :23 AND s_w_id = :43
AND s_i_id = i_id UNION ALL \
SELECT
14,stk.rowid,i_price,i_name,i_data,s_dist_%02d,s_data
a,s_quantity \
FROM item,stk WHERE i_id = :24 AND s_w_id = :44
AND s_i_id = i_id"

#define SQLTXT4 "INSERT INTO ordl \
(ol_o_id, ol_d_id, ol_w_id, ol_number, ol_delivery_d,
ol_i_id, \
ol_supply_w_id, ol_quantity, ol_amount, ol_dist_info)
VALUES (:ol_o_id, :ol_d_id, \
:ol_w_id, :ol_number, :null_date, :ol_i_id, :ol_supply_
w_id, :ol_quantity, \
:ol_amount, :ol_dist_info)"
#endif /* PLSQLNO */

#define NITEMS 15
#define ROWIDLEN 20
#define OCIROWLEN 20

sb4 no_data(dvoid *ctxp, OCIBind *bp, ub4 iter, ub4
index,
dvoid **bufpp, ub4 *alenp, ub1 *piecep,
dvoid **indpp)
{
*bufpp = (dvoid*)0;
*alenp = 0;
*indpp = (dvoid*)0;
*piecep = OCI_ONE_PIECE;
return (OCI_CONTINUE);
}

struct newctx {
sb2 nol_i_id_ind[NITEMS];
sb2 nol_supply_w_id_ind[NITEMS];
sb2 nol_quantity_ind[NITEMS];
sb2 nol_amount_ind[NITEMS];
sb2 i_name_ind[NITEMS];
sb2 s_quantity_ind[NITEMS];
sb2 i_price_ind[NITEMS];
sb2 ol_w_id_ind[NITEMS];
sb2 ol_d_id_ind[NITEMS];
sb2 ol_o_id_ind[NITEMS];

```

```

sb2 ol_number_ind[NITEMS];
sb2 cons_ind[NITEMS];
sb2 s_rowid_ind[NITEMS];
sb2 s_remote_ind[NITEMS];
sb2 s_quant_ind[NITEMS];
sb2 i_data_ind[NITEMS];
sb2 s_data_ind[NITEMS];
sb2 s_dist_info_ind[NITEMS];
sb2 ol_dist_info_ind[NITEMS];
sb2 null_date_ind[NITEMS];
#ifdef PLSQLNO
sb2 s_bg_ind[NITEMS];
#endif

ub2 nol_i_id_len[NITEMS];
ub2 nol_supply_w_id_len[NITEMS];
ub2 nol_quantity_len[NITEMS];
ub2 nol_amount_len[NITEMS];
ub2 s_quantity_len[NITEMS];
ub2 i_name_len[NITEMS];
ub2 i_price_len[NITEMS];
ub2 i_data_len[NITEMS];
ub2 s_dist_info_len[NITEMS];
ub2 s_data_len[NITEMS];
ub2 ol_w_id_len[NITEMS];
ub2 ol_d_id_len[NITEMS];
ub2 ol_o_id_len[NITEMS];
ub2 ol_number_len[NITEMS];
ub2 cons_len[NITEMS];
ub2 s_rowid_len[NITEMS];
ub2 s_remote_len[NITEMS];
ub2 s_quant_len[NITEMS];
ub2 ol_dist_info_len[NITEMS];
ub2 null_date_len[NITEMS];
ub2 s_bg_len[NITEMS];
#ifdef PLSQLNO
ub2 nol_i_id_rcode[NITEMS];
ub2 nol_supply_w_id_rcode[NITEMS];
ub2 nol_quantity_rcode[NITEMS];
ub2 nol_amount_rcode[NITEMS];
ub2 i_name_rcode[NITEMS];
ub2 s_quantity_rcode[NITEMS];
ub2 i_price_rcode[NITEMS];
ub2 ol_w_id_rcode[NITEMS];
ub2 ol_d_id_rcode[NITEMS];
ub2 ol_o_id_rcode[NITEMS];
ub2 ol_number_rcode[NITEMS];
ub2 cons_rcode[NITEMS];
ub2 s_rowid_rcode[NITEMS];
ub2 s_remote_rcode[NITEMS];
ub2 s_quant_rcode[NITEMS];
ub2 i_data_rcode[NITEMS];
ub2 s_data_rcode[NITEMS];
ub2 s_dist_info_rcode[NITEMS];
ub2 ol_dist_info_rcode[NITEMS];
ub2 null_date_rcode[NITEMS];
#endif
ub2 s_bg_rcode[NITEMS];
#endif

int ol_w_id[NITEMS];
int ol_d_id[NITEMS];
int ol_o_id[NITEMS];
int ol_number[NITEMS];
int cons[NITEMS];

OCIRowid *s_rowid_ptr[NITEMS];

```

```

int s_remote[NITEMS];
char i_data[NITEMS][51];
char s_data[NITEMS][51];
char s_dist_info[NITEMS][25];
OCIDate null_date[NITEMS]; /* base date for null date
entry */
OCISmt *curn1;
#ifdef PLSQLNO
OCIBind *ol_i_id_bp;
OCIBind *ol_supply_w_id_bp;
OCIBind *i_price_bp;
OCIBind *i_name_bp;
OCIBind *s_bg_bp;
OCIBind *s_data_bp;
OCIBind *i_data_bp;
ub4 nol_i_count;
ub4 nol_s_count;
ub4 nol_q_count;
ub4 nol_item_count;
ub4 nol_name_count;
ub4 nol_qty_count;
ub4 nol_bg_count;
ub4 nol_am_count;
ub4 s_remote_count;
ub4 s_data_count;
ub4 i_data_count;
#endif
OCISmt *curn2;
OCISmt *curn3[10];
OCIBind *ol_i_id_bp4;
OCIBind *ol_supply_w_id_bp4;
OCIBind *ol_quantity_bp;
OCIBind *ol_quantity_bp4;
OCIBind *s_remote_bp;
OCIBind *s_quantity_bp;
OCISmt *curn4;
OCIBind *w_id_bp;
OCIBind *d_id_bp;
OCIBind *c_id_bp;
OCIBind *o_all_local_bp;
OCIBind *o_all_cnt_bp;
OCIBind *w_tax_bp;
OCIBind *d_tax_bp;
OCIBind *o_id_bp;
OCIBind *c_discount_bp;
OCIBind *c_credit_bp;
OCIBind *c_last_bp;
OCIBind *retries_bp;
OCIBind *cr_date_bp;
OCIBind *s_rowid_bp;
OCIBind *id_bp[10][15];
OCIBind *sd_bp[10][15];
OCIDefine *Dcons[10];
OCIDefine *Ds_rowid[10];
OCIDefine *Di_price[10];
OCIDefine *Di_data[10];
OCIDefine *Ds_dist_info[10];
OCIDefine *Ds_data[10];
OCIDefine *Ds_quantity[10];
OCIDefine *Di_name[10];
OCIBind *ol_o_id_bp;
OCIBind *ol_d_id_bp;
OCIBind *ol_w_id_bp;
OCIBind *ol_number_bp;
OCIBind *ol_amount_bp;
OCIBind *ol_dist_info_bp;
OCIBind *null_date_bp;

```

```

sb2 w_id_ind;
ub2 w_id_len;
ub2 w_id_rc;

sb2 d_id_ind;
ub2 d_id_len;
ub2 d_id_rc;

sb2 c_id_ind;
ub2 c_id_len;
ub2 c_id_rc;

sb2 o_all_local_ind;
ub2 o_all_local_len;
ub2 o_all_local_rc;

sb2 o_ol_cnt_ind;
ub2 o_ol_cnt_len;
ub2 o_ol_cnt_rc;

sb2 w_tax_ind;
ub2 w_tax_len;
ub2 w_tax_rc;

sb2 d_tax_ind;
ub2 d_tax_len;
ub2 d_tax_rc;

sb2 o_id_ind;
ub2 o_id_len;
ub2 o_id_rc;

sb2 c_discount_ind;
ub2 c_discount_len;
ub2 c_discount_rc;

sb2 c_credit_ind;
ub2 c_credit_len;
ub2 c_credit_rc;

sb2 c_last_ind;
ub2 c_last_len;
ub2 c_last_rc;

sb2 retries_ind;
ub2 retries_len;
ub2 retries_rc;

sb2 cr_date_ind;
ub2 cr_date_len;
ub2 cr_date_rc;

int cs;
int norow;

/* context holders */
int i_name_ctx;
int i_data_ctx;
int i_price_ctx;
int s_data_ctx;
int s_dist_info_ctx;
int s_quantity_ctx;
};

typedef struct newctx newctx;

newctx *nctx;

tkvcninit ()
{
    int i,lr ;
    text stmbuf1[16*1024];
    text stmbuf1[8*1024];
    text stmbuf2[8*1024];

    #ifndef PLSQLNO
    char sd[4];
    char id[4];
    int j;
    #endif /* !PLSQLNO */

    nctx = (newctx *) malloc (sizeof(newctx));
    memset(nctx,(char)0,sizeof(newctx));
    nctx->cs = 1;
    nctx->norow=0;
    for(i=0;i<NITEMS;i++) {
        OCIERROR(errhp,
        OCIDescriptorAlloc(tpcenv,(dvoid**)&nctx-
        >s_rowid_ptr[i],
        OCI_DTYPE_ROWID,0,(dvoid**)0));
    }
    nctx->w_id_ind = TRUE;
    nctx->w_id_len = sizeof(w_id);
    nctx->d_id_ind = TRUE;
    nctx->d_id_len = sizeof(d_id);
    nctx->c_id_ind = TRUE;
    nctx->c_id_len = sizeof(c_id);
    nctx->o_all_local_ind = TRUE;
    nctx->o_all_local_len = sizeof(o_all_local);
    nctx->o_ol_cnt_ind = TRUE;
    nctx->o_ol_cnt_len = sizeof(o_ol_cnt);
    nctx->w_tax_ind = TRUE;
    nctx->w_tax_len = 0;
    nctx->d_tax_ind = TRUE;
    nctx->d_tax_len = 0;
    nctx->o_id_ind = TRUE;
    nctx->o_id_len = sizeof(o_id);
    nctx->c_discount_ind = TRUE;
    nctx->c_discount_len = 0;
    nctx->c_credit_ind = TRUE;
    nctx->c_credit_len = 0;
    nctx->c_last_ind = TRUE;
    nctx->c_last_len = 0;
    nctx->retries_ind = TRUE;
    nctx->retries_len = sizeof(retries);
    nctx->cr_date_ind = TRUE;
    nctx->cr_date_len = sizeof(cr_date);

    /* open first cursor */
    OCIERROR(errhp,OCIHandleAlloc(tpcenv,(dvoid
    **>(&nctx->cur1),
        OCI_HTYPE_STMT, 0, (dvoid**)0));
    #ifndef PLSQLNO
    sqlfile("../blocks/tkvcnnew.sql",stmbuf);
    #else
    sqlfile("../blocks/tkvcnnew.sql",stmbuf);
    #endif
    OCIERROR(errhp,OCIStmtPrepare(nctx->cur1,
    errhp, stmbuf, strlen((char *)stmbuf),
        OCI_NTV_SYNTAX,
        OCI_DEFAULT));

    /* bind variables */
    OCIBNDR(nctx->cur1, nctx->w_id_bp, errhp,
    ":w_id",ADR(w_id),SIZ(w_id),
        SQLT_INT, &nctx->w_id_ind, &nctx->w_id_len,
    &nctx->w_id_rc);
    OCIBNDR(nctx->cur1, nctx->d_id_bp, errhp,
    ":d_id",ADR(d_id),SIZ(d_id),
        SQLT_INT, &nctx->d_id_ind, &nctx->d_id_len,
    &nctx->d_id_rc);
    OCIBNDR(nctx->cur1, nctx->c_id_bp, errhp,
    ":c_id",ADR(c_id),SIZ(c_id),
        SQLT_INT, &nctx->c_id_ind, &nctx->c_id_len,
    &nctx->c_id_rc);
    OCIBNDR(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_ind,
        &nctx->o_all_local_len, &nctx->o_all_local_rc);
    OCIBNDR(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_ind,
    &nctx->o_ol_cnt_len,
        &nctx->o_ol_cnt_rc);
    OCIBNDR(nctx->cur1, nctx->w_tax_bp, errhp,
    ":w_tax",ADR(w_tax),SIZ(w_tax),
        SQLT_FLT, &nctx->w_tax_ind, &nctx->w_tax_len,
    &nctx->w_tax_rc);
    OCIBNDR(nctx->cur1, nctx->d_tax_bp, errhp,
    ":d_tax",ADR(d_tax),SIZ(d_tax),
        SQLT_FLT, &nctx->d_tax_ind, &nctx->d_tax_len,
    &nctx->d_tax_rc);
    OCIBNDR(nctx->cur1, nctx->o_id_bp, errhp,
    ":o_id",ADR(o_id),SIZ(o_id),
        SQLT_INT, &nctx->o_id_ind, &nctx->o_id_len,
    &nctx->o_id_rc);
    OCIBNDR(nctx->cur1, nctx->c_discount_bp, errhp,
    ":c_discount",
        ADR(c_discount), SIZ(c_discount),SQLT_FLT,
        &nctx->c_discount_ind, &nctx->c_discount_len,
    &nctx->c_discount_rc);
    OCIBNDR(nctx->cur1, nctx->c_credit_bp, errhp,
    ":c_credit",c_credit,
        SIZ(c_credit),SQLT_CHR,
        &nctx->c_credit_ind, &nctx->c_credit_len, &nctx-
    >c_credit_rc);
    OCIBNDR(nctx->cur1, nctx->c_last_bp, errhp,
    ":c_last",c_last,SIZ(c_last),
        SQLT_STR, &nctx->c_last_ind, &nctx-
    >c_last_len, &nctx->c_last_rc);
    OCIBNDR(nctx->cur1, nctx->retries_bp, errhp,
    ":retries",ADR(retries),
        SIZ(retries),SQLT_INT,
        &nctx->retries_ind, &nctx->retries_len, &nctx-
    >retries_rc);
    OCIBNDR(nctx->cur1, nctx->cr_date_bp, errhp,
    ":cr_date",&cr_date,SIZ(OCIDate),
        SQLT_ODT, &nctx->cr_date_ind, &nctx->cr_date_len,
    &nctx->cr_date_rc);

    #ifndef PLSQLNO
    OCIBNDRAA(nctx->cur1, nctx-
    >ol_i_id_bp,errhp,":ol_i_id",nol_i_id,
        SIZ(int), SQLT_INT, nctx->nol_i_id_ind,nctx-
    >nol_i_id_len,
        nctx->nol_i_id_rcode,NITEMS,&nctx-
    >nol_i_count);
    OCIBNDRAA(nctx->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_ind,

```

```

nctx->no_l_supply_w_id_len, nctx-
>no_l_supply_w_id_rcode,
NITEMS, &nctx->no_l_s_count);
OCIBNDRAA(nctx->curr1, nctx-
>ol_quantity_bp, errhp, ":ol_quantity", nol_quantity,
SIZ(int), SOLT_INT, nctx->no_l_quantity_ind, nctx-
>no_l_quantity_len,
nctx->no_l_quantity_rcode, NITEMS, &nctx-
>no_l_q_count);
OCIBNDRAA(nctx->curr1, nctx-
>i_price_bp, errhp, ":i_price", i_price, SIZ(int),
SOLT_INT, nctx->i_price_ind, nctx-
>i_price_len, nctx->i_price_rcode,
NITEMS, &nctx->no_l_item_count);
OCIBNDRAA(nctx->curr1, nctx-
>i_name_bp, errhp, ":i_name", i_name,
SIZ(i_name[0]), SOLT_STR, nctx-
>i_name_ind, nctx->i_name_len,
nctx->i_name_rcode, NITEMS, &nctx-
>no_l_name_count);
OCIBNDRAA(nctx->curr1, nctx-
>s_quantity_bp, errhp, ":s_quantity", s_quantity,
SIZ(int), SOLT_INT, nctx->s_quant_ind, nctx-
>s_quant_len,
nctx->s_quant_rcode, NITEMS, &nctx-
>no_l_qty_count);
OCIBNDRAA(nctx->curr1, nctx-
>s_bg_bp, errhp, ":brand_generic", brand_generic,
SIZ(char), SOLT_CHR, nctx->s_bg_ind, nctx-
>s_bg_len,
nctx->s_bg_rcode, NITEMS, &nctx-
>no_l_bg_count);
OCIBNDRAA(nctx->curr1, nctx-
>ol_amount_bp, errhp, ":ol_amount", ol_amount,
SIZ(int), SOLT_INT, nctx->ol_amount_ind, nctx-
>ol_amount_len,
nctx->ol_amount_rcode, NITEMS, &nctx-
>no_l_am_count);
OCIBNDRAA(nctx->curr1, nctx-
>s_remote_bp, errhp, ":s_remote", nctx->s_remote,
SIZ(int), SOLT_INT, nctx->s_remote_ind, nctx-
>s_remote_len,
nctx->s_remote_rcode, NITEMS, &nctx-
>s_remote_count);

/* open second cursor */
OCIERROR(errhp, OCIHandleAlloc(tpcenv, (dvoid
**)(&nctx->curr2), OCI_HTYPE_STMT,
0, (dvoid**)0));
sprintf((char *) stmbuf, SOLTXT2);
OCIERROR(errhp, OCIStmtPrepare(nctx->curr2,
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];
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_ol_cnt = NITEMS;

/* Bind array */
OCIBNDRAA(nctx->curr2,
idx1arr_bp, errhp, ":idx1arr", idx1arr,
SIZ(int), SOLT_INT, idx1arr_ind, idx1arr_len,
idx1arr_rcode, NITEMS, &idx1arr_count);

execstatus = OCIStmtExecute(tpcsvc, nctx-
>curr2, errhp, 1, 0, 0, OCI_DEFAULT);
if (execstatus != OCI_SUCCESS) {
OCITransRollback(tpcsvc, errhp, OCI_DEFAULT);
errcode = OCIERROR(errhp, execstatus);
return -1;
}
}
#else

/* open second cursor */
OCIERROR(errhp, OCIHandleAlloc(tpcenv, (dvoid
**)(&nctx->curr2), OCI_HTYPE_STMT,
0, (dvoid**)0));
sprintf((char *) stmbuf, SOLTXT2);
OCIERROR(errhp, OCIStmtPrepare(nctx->curr2,
errhp, stmbuf,
strlen((char *)stmbuf), OCI_NTV_SYNTAX,
OCI_DEFAULT));

/* bind variables */

OCIBNDRA(nctx->curr2, nctx-
>s_quantity_bp, errhp, ":s_quantity", s_quantity,
SIZ(int), SOLT_INT, nctx->s_quant_ind, nctx-
>s_quant_len,
nctx->s_quant_rcode);
OCIBNDRA(nctx->curr2, nctx->s_rowid_bp, errhp,
":s_rowid", nctx->s_rowid_ptr,
sizeof(nctx->s_rowid_ptr[0]), SOLT_RDD, nctx-
>s_rowid_ind,
nctx->s_rowid_len, nctx->s_rowid_rcode);
OCIBNDRA(nctx->curr2, nctx-
>ol_quantity_bp, errhp, ":ol_quantity", nol_quantity,
SIZ(int), SOLT_INT, nctx->no_l_quantity_ind, nctx-
>no_l_quantity_len,
nctx->no_l_quantity_rcode);
OCIBNDRA(nctx->curr2, nctx->s_remote_bp, errhp,
":s_remote", nctx->s_remote,
SIZ(int), SOLT_INT, nctx->s_remote_ind, nctx-
>s_remote_len,
nctx->s_remote_rcode);

/* open third cursor and bind variables */
for (i = 0; i < 10; i++)
{
j = i + 1;
OCIERROR(errhp, OCIHandleAlloc(tpcenv, (dvoid
**)(&nctx->curr3)[i]),
OCI_HTYPE_STMT, 0,
(dvoid**)0);

// kawabata add 11.20

// sprintf ((char *) stmbuf, SOLTXT3, j, j, j, j, j, j, j,
j, j);
// j, j, j);

/* sprintf ((char *) stmbuf1, SOLTXT3_1, j, j, j, j, j,
j, j);
strcat (stmbuf, stmbuf1);
sprintf ((char *) stmbuf2, SOLTXT3_2, j, j, j, j, j,
j);
strcat (stmbuf, stmbuf2);
*/

lr = sprintf ((char *) stmbuf, SOLTXT3_1, j, j, j, j, j,
j, j);
/* strcat (stmbuf, stmbuf1);
sprintf ((char *) stmbuf + lr, SOLTXT3_2, j, j, j, j, j,
j);
*/
strcat (stmbuf, stmbuf2);

// kawabata add
//printf("ORA=%s", (char *)stmbuf);

OCIERROR(errhp, OCIStmtPrepare((nctx-
>curr3)[i], errhp, stmbuf,
strlen((char
*)stmbuf), OCI_NTV_SYNTAX,
OCI_DEFAULT));
OCIERROR(errhp,
OCIAttrSet(nctx-
>curr3[i], OCI_HTYPE_STMT, (dvoid *)&nctx->norow, 0,
OCI_ATTR_PREFETCH_ROWS, errhp));
for (j = 0; j < NITEMS; j++)
{
sprintf (id, ":%d", j + 10);
sprintf (sd, ":%d", j + 30);
OCIBNDRA(nctx->curr3)[i], (nctx-
>id_bp)[i][j], errhp, id, ADR(nol_i_id[j]),
SIZ(int), SOLT_INT,
&nctx->no_l_i_id_ind[j], &nctx->no_l_i_id_len[j],
&nctx->no_l_i_id_rcode[j]);
OCIBNDRA(nctx->curr3)[i], (nctx-
>sd_bp)[i][j], errhp, sd,
ADR(nol_supply_w_id[j]), SIZ(int), SOLT_INT,
&nctx->no_l_supply_w_id_ind[j], &nctx-
>no_l_supply_w_id_len[j],
&nctx->no_l_supply_w_id_rcode[j]);
nctx->no_l_i_id_ind[j] = NA;
nctx->no_l_supply_w_id_ind[j] = NA;
nctx->no_l_i_id_len[j] = sizeof(int);
nctx->no_l_supply_w_id_len[j] = sizeof(int);
}

OCIDEF((nctx->curr3)[i], (nctx-
>Dcons)[i], errhp, 1, &(nctx->cons[0]),
SIZ(nctx->cons[0]), SOLT_INT);
OCIDEF((nctx->curr3)[i], (nctx->Ds_rowid)[i], errhp, 2,
nctx->s_rowid_ptr, sizeof(nctx->s_rowid_ptr[0]),
SOLT_RDD);
OCIDEF((nctx->curr3)[i], (nctx-
>Di_price)[i], errhp, 3, i_price, SIZ(int),
SOLT_INT);

OCIDFNRA((nctx->curr3)[i], (nctx-
>Di_name)[i], errhp, 4, i_name,
SIZ(i_name[0]), SOLT_STR, nctx-
>i_name_ind, nctx->i_name_len,
nctx->i_name_rcode);

```



```

OCIDFNRA((nctx->curm3)[i], (nctx-
>Di_data)[i],errhp,5,nctx->i_data,
    SIZ(nctx->i_data[0]), SOLT_STR, NULL,nctx-
>i_data_len,NULL);
OCIDFNRA((nctx->curm3)[i], (nctx-
>Ds_dist_info)[i],errhp,6,
    nctx->s_dist_info, SIZ(nctx-
>s_dist_info[0]),SOLT_STR,
    NULL,nctx->s_dist_info_len, NULL);
OCIDFNRA((nctx->curm3)[i],(nctx-
>Ds_data)[i],errhp,7,nctx->s_data,
    SIZ(nctx->s_data[0]),SOLT_STR,NULL,nctx-
>s_data_len,NULL);
OCIDEF((nctx->curm3)[i],(nctx-
>Ds_quantity)[i],errhp,8,s_quantity,
    SIZ(int),SOLT_INT);
}

/* open fourth cursor */
OCIHandleAlloc(tpcenv, (dvoid **)&nctx->curm4,
OCI_HTYPE_STMT, 0,
    (dvoid**)0);
sprintf((char *) stmbuf, SOLTXT4);
OCISmtPrepare(nctx->curm4, errhp, stmbuf,
strlen((char *)stmbuf),
    OCI_NTV_SYNTAX, OCI_DEFAULT);
/* bind variables */

OCIBNDRA(nctx->curm4, nctx-
>ol_o_id_bp,errhp,":ol_o_id",nctx->ol_o_id,
    SIZ(int),SOLT_INT, NULL,nctx->ol_o_id_len,
    NULL);
OCIBNDRA(nctx->curm4, nctx-
>ol_d_id_bp,errhp,":ol_d_id",nctx->ol_d_id,
    SIZ(int),SOLT_INT, NULL,nctx->ol_d_id_len,
    NULL);
OCIBNDRA(nctx->curm4, nctx-
>ol_w_id_bp,errhp,":ol_w_id",nctx->ol_w_id,
    SIZ(int),SOLT_INT, NULL,nctx->ol_w_id_len,
    NULL);
OCIBNDRA(nctx->curm4, nctx-
>ol_number_bp,errhp,":ol_number",nctx->ol_number,
    SIZ(int),SOLT_INT, NULL,nctx->ol_number_len,
    NULL);
OCIBNDRA(nctx->curm4, nctx-
>ol_i_id_bp4,errhp,":ol_i_id",nol_i_id,SIZ(int),
    SOLT_INT, NULL,nctx->nol_i_id_len, NULL);
OCIBNDRA(nctx->curm4, nctx-
>ol_supply_w_id_bp4,errhp,":ol_supply_w_id",
    nol_supply_w_id,SIZ(int),SOLT_INT, NULL,
    nctx->nol_supply_w_id_len, NULL);
OCIBNDRA(nctx->curm4, nctx-
>ol_quantity_bp4,errhp,":ol_quantity",nol_quantity,
    SIZ(int),SOLT_INT, NULL,nctx-
>nol_quantity_len,
    NULL);
OCIBNDRA(nctx->curm4, nctx-
>ol_amount_bp,errhp,":ol_amount",nol_amount,
    SIZ(int),SOLT_INT, NULL,nctx->nol_amount_len,
    NULL);
OCIBNDRA(nctx->curm4, nctx-
>ol_dist_info_bp,errhp,":ol_dist_info",
    nctx->s_dist_info, SIZ(nctx-
>s_dist_info[0]),SOLT_AFC,
    NULL, nctx->ol_dist_info_len,
    NULL);
OCIBNDRA(nctx->curm4, nctx-
>>null_date_bp,errhp,":null_date",nctx->>null_date,
    SIZ(OCIDate), SOLT_ODT,NULL,
    nctx->>null_date_len, NULL);

/* set up the null date Null date is 15-sep-11 */
for (i=0;i<NITEMS;i++)
{
    OCIDateSetDate(&nctx-
>>null_date[i],(sb2)1811,(ub1)9,(ub1)15);
}
#endif

return (0);
}

static void checkerr(errhp, status)
OCIError *errhp;
sword status;
{
    text errbuf[512];
    ub4 buflen;
    sb4 errcode;

    if (status == OCI_SUCCESS) return;

    switch (status)
    {
        case OCI_SUCCESS_WITH_INFO:
            printf("Error -
OCI_SUCCESS_WITH_INFO\n");
            break;
        case OCI_NEED_DATA:
            printf("Error - OCI_NEED_DATA\n");
            break;
        case OCI_NO_DATA:
            printf("Error - OCI_NO_DATA\n");
            break;
        case OCI_ERROR:
            printf("OCI_ERROR\n");
            while ( OCIErrorGet ((dvoid *) errhp, (ub4) 1,
(text *) NULL, &errcode,
                errbuf, (ub4) sizeof(errbuf), (ub4)
OCI_HTYPE_ERROR) != OCI_NO_DATA)
                printf("Error - %d: %s\n", errcode, errbuf);
            break;
        case OCI_INVALID_HANDLE:
            printf("Error - OCI_INVALID_HANDLE\n");
            break;
        case OCI_STILL_EXECUTING:
            printf("Error - OCI_STILL_EXECUTE\n");
            break;
        case OCI_CONTINUE:
            printf("Error - OCI_CONTINUE\n");
            break;
        default:
            printf("Error - %d\n", status);
            printf("default\n");
            break;
    }
}

tkvcn ()
{
    int i;
    int rcount;

    int rollcode;
    #ifndef PLSQLNO
        ub4 flags;
        int rpc,rpc3;
        // int rowoff,rpc,rpc3,ifers,j,k;
        #endif /* !PLSQLNO */
        int failed = 0;

    retry:

        status = 0;          /* number of invalid items */

        /* get number of order lines, and check if all are local
        */
        //printf("tkvcn start1 ");
        //printf("o_ol_cnt=%d",o_ol_cnt);
        //printf("status=%d",status);

        o_ol_cnt = NITEMS;
        o_all_local = 1;
        for (i = 0; i < NITEMS; i++) {
            //kawabata add
            // if((nol_i_id[i] == 0) && (nol_supply_w_id[i] == 0)
            && (nol_quantity[i] == 0)){
                if (nol_i_id[i] == 0) {
                    o_ol_cnt = i;
                    break;
                }
                if (nol_supply_w_id[i] != w_id) {
                    nctx->s_remote[i] = 1;
                    o_all_local = 0;
                }
                else
                    nctx->s_remote[i] = 0;
            }

            nctx->w_id_ind = TRUE;
            nctx->w_id_len = sizeof(w_id);
            nctx->d_id_ind = TRUE;
            nctx->d_id_len = sizeof(d_id);
            nctx->c_id_ind = TRUE;
            nctx->c_id_len = sizeof(c_id);
            nctx->o_all_local_ind = TRUE;
            nctx->o_all_local_len = sizeof(o_all_local);
            nctx->o_ol_cnt_ind = TRUE;
            nctx->o_ol_cnt_len = sizeof(o_ol_cnt);
            nctx->w_tax_ind = TRUE;
            nctx->w_tax_len = 0;
            nctx->d_tax_ind = TRUE;
            nctx->d_tax_len = 0;
            nctx->o_id_ind = TRUE;
            nctx->o_id_len = sizeof(o_id);
            nctx->c_discount_ind = TRUE;
            nctx->c_discount_len = 0;
            nctx->c_credit_ind = TRUE;
            nctx->c_credit_len = 0;
            nctx->c_last_ind = TRUE;
            nctx->c_last_len = 0;
            nctx->retries_ind = TRUE;
            nctx->retries_len = sizeof(retries);
            nctx->cr_date_ind = TRUE;
            nctx->cr_date_len = sizeof(cr_date);

            //printf("tkvcn start2 ");
            //printf("o_ol_cnt=%d",o_ol_cnt);
            //printf("status=%d",status);

        #ifndef PLSQLNO
    
```

```

/* this is the row count */
rcount = o_ol_cnt;
nctx->nol_i_count = o_ol_cnt;
nctx->nol_q_count = o_ol_cnt;
nctx->nol_s_count = o_ol_cnt;
nctx->s_remote_count = o_ol_cnt;

nctx->nol_qty_count = 0;
nctx->nol_bg_count = 0;
nctx->nol_item_count = 0;
nctx->nol_name_count = 0;
nctx->nol_am_count = 0;
/* following not relevant */
nctx->s_data_count = o_ol_cnt;
nctx->i_data_count = o_ol_cnt;

/* initialization for array operations */
for (i = 0; i < o_ol_cnt; i++) {
    nctx->ol_w_id[i] = w_id;
    nctx->ol_d_id[i] = d_id;
    nctx->ol_number[i] = i + 1;
    nctx->null_date_ind[i] = TRUE;
    nctx->nol_i_id_ind[i] = 0;
    nctx->nol_supply_w_id_ind[i] = TRUE;
    nctx->nol_quantity_ind[i] = TRUE;
    nctx->nol_amount_ind[i] = TRUE;
    nctx->ol_w_id_ind[i] = TRUE;
    nctx->ol_d_id_ind[i] = TRUE;
    nctx->ol_o_id_ind[i] = TRUE;
    nctx->ol_number_ind[i] = TRUE;
    nctx->ol_dist_info_ind[i] = TRUE;
    nctx->s_remote_ind[i] = TRUE;
    nctx->s_data_ind[i] = TRUE;
    nctx->i_data_ind[i] = TRUE;
    nctx->s_quant_ind[i] = TRUE;
    nctx->s_bg_ind[i] = TRUE;
    nctx->cons_ind[i] = TRUE;
    nctx->s_rowid_ind[i] = TRUE;
    nctx->nol_i_id_len[i] = sizeof(int);
    nctx->nol_supply_w_id_len[i] = sizeof(int);
    nctx->nol_quantity_len[i] = sizeof(int);
    nctx->nol_amount_len[i] = sizeof(int);
    nctx->ol_w_id_len[i] = sizeof(int);
    nctx->ol_d_id_len[i] = sizeof(int);
    nctx->ol_o_id_len[i] = sizeof(int);
    nctx->ol_number_len[i] = sizeof(int);
    nctx->ol_dist_info_len[i] = nctx->s_dist_info_len[i];
    nctx->null_date_len[i] = sizeof(OCIDate);
    nctx->s_remote_len[i] = sizeof(int);
    nctx->s_data_len[i] = sizeof(int);
    nctx->i_data_len[i] = sizeof(int);
    nctx->s_quant_len[i] = sizeof(int);
    nctx->s_rowid_len[i] = sizeof(nctx->s_rowid_ptr[0]);
    nctx->cons_len[i] = sizeof(int);
    nctx->i_name_len[i] = 0;
    nctx->s_bg_len[i] = 0;
}
for (i = o_ol_cnt; i < NITEMS; i++) {
    nctx->nol_i_id_ind[i] = NA;
    nctx->nol_supply_w_id_ind[i] = NA;
    nctx->nol_quantity_ind[i] = NA;
    nctx->nol_amount_ind[i] = NA;
    nctx->ol_w_id_ind[i] = NA;
    nctx->ol_d_id_ind[i] = NA;
    nctx->ol_o_id_ind[i] = NA;
    nctx->ol_number_ind[i] = NA;
    nctx->ol_dist_info_ind[i] = NA;
    nctx->null_date_ind[i] = NA;
    nctx->s_remote_ind[i] = NA;
    nctx->s_data_ind[i] = NA;
    nctx->i_data_ind[i] = NA;
    nctx->s_quant_ind[i] = NA;
    nctx->s_bg_ind[i] = NA;
    nctx->cons_ind[i] = NA;
    nctx->s_rowid_ind[i] = NA;

    nctx->s_data_ind[i] = NA;
    nctx->i_data_ind[i] = NA;
    nctx->s_quant_ind[i] = NA;
    nctx->s_bg_ind[i] = NA;
    nctx->cons_ind[i] = NA;
    nctx->s_rowid_ind[i] = NA;

    nctx->nol_i_id_len[i] = 0;
    nctx->nol_supply_w_id_len[i] = 0;
    nctx->nol_quantity_len[i] = 0;
    nctx->nol_amount_len[i] = 0;
    nctx->ol_w_id_len[i] = 0;
    nctx->ol_d_id_len[i] = 0;
    nctx->ol_o_id_len[i] = 0;
    nctx->ol_number_len[i] = 0;
    nctx->ol_dist_info_len[i] = 0;
    nctx->null_date_len[i] = 0;
    nctx->s_remote_len[i] = 0;
    nctx->i_data_len[i] = 0;
    nctx->s_data_len[i] = 0;
    nctx->s_quant_len[i] = 0;
    nctx->s_rowid_len[i] = 0;
    nctx->i_name_len[i] = 0;
    nctx->s_bg_len[i] = 0;
}

execstatus = OCISStmtExecute(tpscvc,nctx-
>curm1,errhp,1,0,0,
OCI_DEFAULT |
OCI_COMMIT_ON_SUCCESS);

#else
//printf("tkvcn start3 ");
// execstatus = OCISStmtExecute(tpscvc,nctx-
>curm1,errhp,1,0,0,OCI_DEFAULT);
execstatus = OCISStmtExecute(tpscvc,nctx-
>curm1,errhp,1,0,0,OCI_DEFAULT|OCI_COMMIT_0
N_SUCCESS);
//printf("tkvcn start4 ");
#endif

if(execstatus != OCI_SUCCESS) {
// rollcode =
OCITransRollback(tpscvc,errhp,OCI_DEFAULT);
OCITransRollback(tpscvc,errhp,OCI_DEFAULT);
//checkerr(errhp, execstatus);
errcode = OCIErr(errhp,execstatus);
if(errcode == NOT_SERIALIZABLE) {
    retries++;
    goto retry;
} else if (errcode == RECOVER) {
    retries++;
    goto retry;
}
}
/*else if (errcode == SNAPSHOT_TOO_OLD) {
    retries++;
printf("tkvcn start5.6 ");
    goto retry;
}*/
else {
    return -1;
}
}

#ifdef PLSQLNO
/* did the txn succeed ? */
if (rcount != o_ol_cnt)
{
    status = rcount - o_ol_cnt;
    o_ol_cnt = rcount;
}
#endif

#ifdef DEBUG
    printf("w_id = %d, d_id = %d, c_id = %d\n",w_id, d_id,
c_id);
#endif

#ifdef PLSQLNO
/* initialization for array operations */
for (i = 0; i < o_ol_cnt; i++) {
    nctx->ol_w_id[i] = w_id;
    nctx->ol_d_id[i] = d_id;
    nctx->ol_number[i] = i + 1;
    nctx->null_date_ind[i] = TRUE;
    nctx->nol_i_id_ind[i] = TRUE;
    nctx->nol_quantity_ind[i] = TRUE;
    nctx->nol_amount_ind[i] = TRUE;
    nctx->ol_w_id_ind[i] = TRUE;
    nctx->ol_d_id_ind[i] = TRUE;
    nctx->ol_o_id_ind[i] = TRUE;
    nctx->ol_number_ind[i] = TRUE;
    nctx->ol_dist_info_ind[i] = TRUE;
    nctx->s_remote_ind[i] = TRUE;
    nctx->s_data_ind[i] = TRUE;
    nctx->i_data_ind[i] = TRUE;
    nctx->s_quant_ind[i] = TRUE;
    nctx->cons_ind[i] = TRUE;
    nctx->s_rowid_ind[i] = TRUE;

    nctx->nol_i_id_len[i] = sizeof(int);
    nctx->nol_supply_w_id_len[i] = sizeof(int);
    nctx->nol_quantity_len[i] = sizeof(int);
    nctx->nol_amount_len[i] = sizeof(int);
    nctx->ol_w_id_len[i] = sizeof(int);
    nctx->ol_d_id_len[i] = sizeof(int);
    nctx->ol_o_id_len[i] = sizeof(int);
    nctx->ol_number_len[i] = sizeof(int);
    nctx->ol_dist_info_len[i] = nctx->s_dist_info_len[i];
    nctx->null_date_len[i] = sizeof(OCIDate);
    nctx->s_remote_len[i] = sizeof(int);
    nctx->s_data_len[i] = sizeof(int);
    nctx->i_data_len[i] = sizeof(int);
    nctx->s_quant_len[i] = sizeof(int);
    nctx->s_rowid_len[i] = sizeof(nctx->s_rowid_ptr[0]);
    nctx->cons_len[i] = sizeof(int);
}
for (i = o_ol_cnt; i < NITEMS; i++) {
    nctx->nol_i_id_ind[i] = NA;
    nctx->nol_supply_w_id_ind[i] = NA;
    nctx->nol_quantity_ind[i] = NA;
    nctx->nol_amount_ind[i] = NA;
    nctx->ol_w_id_ind[i] = NA;
    nctx->ol_d_id_ind[i] = NA;
    nctx->ol_o_id_ind[i] = NA;
    nctx->ol_number_ind[i] = NA;
    nctx->ol_dist_info_ind[i] = NA;
    nctx->null_date_ind[i] = NA;
    nctx->s_remote_ind[i] = NA;
    nctx->s_data_ind[i] = NA;
    nctx->i_data_ind[i] = NA;
    nctx->s_quant_ind[i] = NA;
    nctx->cons_ind[i] = NA;
    nctx->s_rowid_ind[i] = NA;

    nctx->nol_i_id_len[i] = 0;
    nctx->nol_supply_w_id_len[i] = 0;
    nctx->nol_quantity_len[i] = 0;
    nctx->nol_amount_len[i] = 0;
    nctx->ol_w_id_len[i] = 0;
}
}

```

```

nctx->ol_d_id_len[i] = 0;
nctx->ol_o_id_len[i] = 0;
nctx->ol_number_len[i] = 0;
nctx->ol_dist_info_len[i] = 0;
nctx->null_date_len[i] = 0;
nctx->s_remote_len[i] = 0;
nctx->s_quant_len[i] = 0;
nctx->s_rowid_len[i] = 0;
nctx->cons_len[i] = 0;
}

rpc3 = SellItemStk ();

if (rpc3 == -2)
    goto retry;
else if (rpc3 == -1)
    return (-1);

/* compute order line amounts, total amount and
stock quantities */

total_amount = 0.0;
for (i = 0; i < o_ol_cnt; i++)
{
    nctx->ol_o_id[i] = o_id;
    if (nctx->no_l_id_ind[i] != NA) {
        s_quantity[i] = nol_quantity[i];
        if (s_quantity[i] < 10)
            s_quantity[i] += 91;
        nol_amount[i] = (nol_quantity[i] * i_price[i]);
        total_amount += nol_amount[i];
        if (strstr (nctx->i_data[i], "ORIGINAL") &&
            strstr (nctx->s_data[i], "ORIGINAL"))
            brand_gen[i] = 'B';
        else
            brand_gen[i] = 'G';
    }
}

total_amount *= ((float)(1 - c_discount)) * (1.0 +
(float)d_tax) + ((float)w_tax));
total_amount = total_amount/100;

rpc = UpdStk2 ();

if (rpc == -2)
    goto retry;
else if (rpc == -1)
    return (-1);

/* error processing - will keep it separated for
readability */
/* number of items selected != number of stock
updated */

if (rpc3 != rpc) {
#ifdef TUX
    printf ("Error in TPC-C server %d: %d rows of item
read, ",
            proc_no, rpc3);
    printf ("          but %d rows of stock update\n",
rpc);
#endif
/* rollback */
    OCITransRollback(tpcsvc, errhp, OCI_DEFAULT);
    return (-1);
}

/* common code for insert into order_line */
for (i=0; i< o_ol_cnt; i++) /* move district info in place
*/
{
    nctx->ol_dist_info_len[i]=nctx->s_dist_info_len[i];
}

/* array insert into order line table */
flags= (status ? OCI_DEFAULT :
(OCI_DEFAULT|OCI_COMMIT_ON_SUCCESS));
if ((o_ol_cnt - status) > 0)
{
    execstatus = OCIStmtExecute(tpcsvc,nctx-
>curm4, errhp, o_ol_cnt - status,
                                0,0,0,flags);

    if(execstatus != OCI_SUCCESS) {
        OCITransRollback(tpcsvc, errhp, OCI_DEFAULT);
        errcode = OCIERROR(errhp,execstatus);
        if(errcode == NOT_SERIALIZABLE) {
            retries++;
            goto retry;
        } else if (errcode == RECOVER) {
            retries++;
            goto retry;
        } else if (errcode == SNAPSHOT_TOO_OLD) {
            retries++;
            goto retry;
        } else {
            return -1;
        }
    }

    OCIAttrGet(nctx-
>curm4,OCI_HTYPE_STMT,&rcount,NULL,
                OCI_ATTR_ROW_COUNT, errhp);

    if (rcount != (o_ol_cnt - status))
    {
#ifdef TUX
        printf ("Error in TPC-C server %d: array
insert failed\n",
                proc_no);
    #else
        printf ("Error in TPC-C server %d: array insert
failed\n",
                proc_no);
    #endif
}

/* rollback */
OCITransRollback(tpcsvc, errhp, OCI_DEFAULT);

return (-1);

}

/* commit if no invalid item */

if (status) {
    rollcode =
OCITransRollback(tpcsvc, errhp, OCI_DEFAULT);
//checkerr(errhp, execstatus);
#ifdef TUX
    // fflush(stdout);
#endif
}

#endif

total_amount = 0.0;
for (i = 0; i < o_ol_cnt; i++)
{
    if (nctx->no_l_amount_ind[i] != NA) {
        total_amount += nol_amount[i];
    }
}

total_amount *= ((float)(1 - c_discount)) * (float)(1.0 +
(float)d_tax) + ((float)w_tax));
total_amount = total_amount/100;

return (0);
}

void tkvcndone ()
{
    int i;

    if (nctx)
    {
        OCIHandleFree((dvoid *)nctx-
>curm1,OCI_HTYPE_STMT);
        OCIHandleFree((dvoid *)nctx-
>curm2,OCI_HTYPE_STMT);
        for (i = 0; i < 10; i++)
            OCIHandleFree((dvoid *) (nctx-
>curm3)[i],OCI_HTYPE_STMT);
        OCIHandleFree((dvoid *)nctx-
>curm4,OCI_HTYPE_STMT);
        free (nctx);
    }
}

/* the arrays are initialized based on a successful select
from */

```

```

/* stock/item. We need to shift the values in the
orderline array */
/* one position up to compensate when we have an
invalid item */

void shiftitemstock (i, j)

int i, j;

{
/* shift up the values for the stock table */
nctx->s_remote[i] = nctx->s_remote[j];

/* shift up the order_line values */

nctx->noI_i_id_ind[i]=nctx->noI_i_id_ind[j];
noI_i_id[i] = noI_i_id[j];

nctx->noI_quantity_ind[i] = nctx->noI_quantity_ind[j];
noI_quantity[i] = noI_quantity[j];

nctx->noI_supply_w_id_ind [i] = nctx-
>noI_supply_w_id_ind[j];
nol_supply_w_id[i] = nol_supply_w_id[j];
}

void swapitemstock (i, j)

int i, j;

{

int tempi;
int tempf;
char tempstr[52];
ub2 tempub2;
sb2 tempub2;
OCIRowid *tmprid;

tempub2 = nctx->cons_ind[i];
nctx->cons_ind[i] = nctx->cons_ind[j];
nctx->cons_ind[j] = tempub2;
tempub2 = nctx->cons_len[i];
nctx->cons_len[i] = nctx->cons_len[j];
nctx->cons_len[j] = tempub2;
tempub2 = nctx->cons_rcode[i];
nctx->cons_rcode[i] = nctx->cons_rcode[j];
nctx->cons_rcode[j] = tempub2;
tempi = nctx->cons[j];
nctx->cons[i] = nctx->cons[j];
nctx->cons[j] = tempi;

tempub2 = nctx->s_rowid_ind[i];
nctx->s_rowid_ind[i] = nctx->s_rowid_ind[j];
nctx->s_rowid_ind[j] = tempub2;
tempub2 = nctx->s_rowid_len[i];
nctx->s_rowid_len[i] = nctx->s_rowid_len[j];
nctx->s_rowid_len[j] = tempub2;
tempub2 = nctx->s_rowid_rcode[i];
nctx->s_rowid_rcode[i] = nctx->s_rowid_rcode[j];
nctx->s_rowid_rcode[j] = tempub2;
tmprid = nctx->s_rowid_ptr[i];
nctx->s_rowid_ptr[i] = nctx->s_rowid_ptr[j];
nctx->s_rowid_ptr[j]=tmprid;

tempub2 = nctx->i_price_ind[i];
nctx->i_price_ind[i] = nctx->i_price_ind[j];
nctx->i_price_ind[j] = tempub2;
tempub2 = nctx->i_price_len[i];
nctx->i_price_len[i] = nctx->i_price_len[j];
nctx->i_price_len[j] = tempub2;
tempub2 = nctx->i_price_rcode[i];
nctx->i_price_rcode[i] = nctx->i_price_rcode[j];
nctx->i_price_rcode[j] = tempub2;
tempf = i_price[i];
i_price[i] = i_price[j];
i_price[j] = tempf;

tempub2 = nctx->i_name_ind[i];
nctx->i_name_ind[i] = nctx->i_name_ind[j];
nctx->i_name_ind[j] = tempub2;
tempub2 = nctx->i_name_len[i];
nctx->i_name_len[i] = nctx->i_name_len[j];
nctx->i_name_len[j] = tempub2;
tempub2 = nctx->i_name_rcode[i];
nctx->i_name_rcode[i] = nctx->i_name_rcode[j];
nctx->i_name_rcode[j] = tempub2;
strncpy (tempstr, i_name[i], 25);
strncpy (i_name[i], i_name[j], 25);
strncpy (i_name[j], tempstr, 25);

tempub2 = nctx->i_data_ind[i];
nctx->i_data_ind[i] = nctx->i_data_ind[j];
nctx->i_data_ind[j] = tempub2;
tempub2 = nctx->i_data_len[i];
nctx->i_data_len[i] = nctx->i_data_len[j];
nctx->i_data_len[j] = tempub2;
tempub2 = nctx->i_data_rcode[i];
nctx->i_data_rcode[i] = nctx->i_data_rcode[j];
nctx->i_data_rcode[j] = tempub2;
strncpy (tempstr, nctx->i_data[i], 51);
strncpy (nctx->i_data[i], nctx->i_data[j], 51);
strncpy (nctx->i_data[j], tempstr, 51);

tempub2 = nctx->s_quantity_ind[i];
nctx->s_quantity_ind[i] = nctx->s_quantity_ind[j];
nctx->s_quantity_ind[j] = tempub2;
tempub2 = nctx->s_quantity_len[i];
nctx->s_quantity_len[i] = nctx->s_quantity_len[j];
nctx->s_quantity_len[j] = tempub2;
tempub2 = nctx->s_quantity_rcode[i];
nctx->s_quantity_rcode[i] = nctx->s_quantity_rcode[j];
nctx->s_quantity_rcode[j] = tempub2;
tempi = s_quantity[i];
s_quantity[i] = s_quantity[j];
s_quantity[j] = tempi;

tempub2 = nctx->s_dist_info_ind[i];
nctx->s_dist_info_ind[i] = nctx->s_dist_info_ind[j];
nctx->s_dist_info_ind[j] = tempub2;
tempub2 = nctx->s_dist_info_len[i];
nctx->s_dist_info_len[i] = nctx->s_dist_info_len[j];
nctx->s_dist_info_len[j] = tempub2;
tempub2 = nctx->s_dist_info_rcode[i];
nctx->s_dist_info_rcode[i] = nctx-
>s_dist_info_rcode[j];
nctx->s_dist_info_rcode[j] = tempub2;
strncpy (tempstr, nctx->s_dist_info[i], 25);
strncpy (nctx->s_dist_info[i], nctx->s_dist_info[j], 25);
strncpy (nctx->s_dist_info[j], tempstr, 25);

tempub2 = nctx->s_data_ind[i];
nctx->s_data_ind[i] = nctx->s_data_ind[j];
nctx->s_data_ind[j] = tempub2;
tempub2 = nctx->s_data_len[i];
nctx->s_data_len[i] = nctx->s_data_len[j];
nctx->s_data_len[j] = tempub2;
tempub2 = nctx->s_data_rcode[i];
nctx->s_data_rcode[i] = nctx->s_data_rcode[j];
nctx->s_data_rcode[j] = tempub2;
strncpy (tempstr, nctx->s_data[i], 51);
strncpy (nctx->s_data[i], nctx->s_data[j], 51);
strncpy (nctx->s_data[j], tempstr, 51);
}

SellItemStk ()
{
int i, j, rpc3,rcount;

/*printf("w_idt=%d",w_id);
/*printf("d_id=%d",d_id);
/*printf("c_id=%d",c_id);

/* array select from item and stock tables */
execstatus=OCIStmtExecute(tpcscv,(nctx-
>cur3)[d_id-1],errhp,o_of_cnt,
0,0,0,OCI_DEFAULT);

printf("execstatus=%d",execstatus);
/*printf("SellItemStk start2 ");
/*checkerr(errhp, execstatus);

if((execstatus != OCI_SUCCESS) && (execstatus !=
OCI_NO_DATA)) {
errcode = OCIERROR(errhp,execstatus);
checkerr(errhp, execstatus);
if(errcode == NOT_SERIALIZABLE) {
retries++;
OCITransRollback(tpcscv,errhp,OCI_DEFAULT);
return (-2);
} else if (errcode == RECOVER) {
/* In case of NO_DATA this should NOT return, but
simply fall through */
OCITransRollback(tpcscv,errhp,OCI_DEFAULT);
retries++;
return (-2);
} else if (errcode == SNAPSHOT_TOO_OLD) {
OCITransRollback(tpcscv,errhp,OCI_DEFAULT);
retries++;
return (-2);
} else {
OCITransRollback(tpcscv,errhp,OCI_DEFAULT);
return (-1);
}
}
/* mark invalid items */

OCIAttrGet((nctx->cur3)[d_id-1],
OCI_HTYPE_STMT,&rcount,NULL,
OCI_ATTR_ROW_COUNT, errhp);
rpc3 = rcount;
printf("rcount=%d",rcount);

/* the result is in order, so we have to shift up to fill
*/
/* the slot for the line with the invalid item. */
/* If more than one item is wrong, this is not an
simulated */
/* error and we'll blow off */

```

```

    if ((status = o_ol_cnt - rcount) >1)
    {
#ifdef TUX
printf("SellItemStk start3.5 ");
    printf ("TPC-C server %d: more than 1 invalid
item?\n", proc_no);
#else
    printf ("TPC-C server %d: more than 1 invalid
item?\n", proc_no)
;
#endif
//printf("SellItemStk start3.6 ");

        return (rpc3);
    }
    if (status == 0) return (rpc3);

    /* find the invalid item, transfer the rowid information
*/

    for (i = 0; i < o_ol_cnt; i++) {
//printf("SellItemStk start3.7 ");

        if (nctx->cons[i] != i) break; /* this item is
invalid */
    }
//printf("SellItemStk start3.8 ");

#ifdef TUX
    printf ("TPC-C server %d: reordering items and
stocks\n",
        proc_no);
#else
/*
    printf ("TPC-C server %d: reordering items and
stocks\n",
        proc_no); */
#endif

//printf("SellItemStk start4 ");
//printf("o_ol_cnt=%d",o_ol_cnt);
//printf("status=%d",status);

    /* not the last item - shift up */

    for (j = i; j < o_ol_cnt-1; j++)
    {
        shiftitemstock (j, j+1);
    }
    /* zero the last item */
    i = o_ol_cnt-1;
    nctx->no_l_i_id_ind[i] = NA;
    nctx->no_l_supply_w_id_ind[i] = NA;
    nctx->no_l_quantity_ind[i] = NA;
    nctx->no_l_amount_ind[i] = NA;
    nctx->ol_w_id_ind[i] = NA;
    nctx->ol_d_id_ind[i] = NA;
    nctx->ol_o_id_ind[i] = NA;
    nctx->null_date_ind[i] = NA;
    nctx->ol_number_ind[i] = NA;
    nctx->ol_dist_info_ind[i] = NA;
    nctx->s_remote_ind[i] = NA;
    nctx->s_quant_ind[i] = NA;

    nctx->no_l_i_id_len[i] = 0;
    nctx->no_l_supply_w_id_len[i] = 0;
    nctx->no_l_quantity_len[i] = 0;

```

```

    nctx->no_l_amount_len[i] = 0;
    nctx->ol_w_id_len[i] = 0;
    nctx->ol_d_id_len[i] = 0;
    nctx->ol_o_id_len[i] = 0;
    nctx->ol_number_len[i] = 0;
    nctx->ol_dist_info_len[i] = 0;
    nctx->null_date_ind[i] = 0;
    nctx->s_remote_len[i] = 0;
    nctx->s_quant_len[i] = 0;
//printf("SellItemStk end ");
//printf("o_ol_cnt=%d",o_ol_cnt);
//printf("status=%d",status);

    return (rpc3);
}

UpdStk2 ()
{
    int rpc, rcount;

    /* array update of stock table */

    execstatus = OCIStmtExecute(tpcsvc,nctx-
>curm2,errhp,o_ol_cnt-status,0,0,0,
        OCI_DEFAULT);
    if(execstatus != OCI_SUCCESS) {
        OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
        errcode = OCIERROR(errhp,execstatus);
        if(errcode == NOT_SERIALIZABLE) {
            retries++;
            return (-2);
        } else if (errcode == RECOVER) {
            retries++;
            return (-2);
        } else if (errcode == SNAPSHOT_TOO_OLD) {
            retries++;
            return (-2);
        } else {
            return -1;
        }
    }
    OCIAttrGet(nctx-
>curm2,OCI_HTYPE_STMT,&rcount,NULL,
        OCI_ATTR_ROW_COUNT, errhp);
    rpc = rcount;

    if (rpc != (o_ol_cnt - status)) {
#ifdef TUX
        printf ("Error in TPC-C server %d: array update
failed\n",
            proc_no);
#else
        printf ("Error in TPC-C server %d: array update
failed\n",
            proc_no);
#endif
    }
    OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
    return (-1);
}

    return (rpc);
}

```

```

===== plord.c =====

#ifdef RCSID
static char *RCSid =
    "$Header: plord.c 7030100.1 95/07/19 14:46:13 plai
Generic<base> $ Copyr (c) 1994 Oracle";
#endif /* RCSID */

/*=====
=====+
| Copyright (c) 1995 Oracle Corp, Redwood
Shores, CA |
| OPEN SYSTEMS PERFORMANCE
GROUP |
| All Rights Reserved |
=====+
| FILENAME
| plord.c
| DESCRIPTION
| OCI version (using PL/SQL anonymous block) of
| ORDER STATUS transaction in TPC-C benchmark.
=====+
=====*/

#include "tpcc.h"
#include "tpcc_info.h"
#include "tpccflags.h"

#ifdef PLSQLORD
#define SQLTXT "BEGIN orderstatus.getstatus
(:w_id, :d_id, :c_id, :byln, \
:c_last, :c_first, :c_middle, :c_balance, :o_id, :o_entry
_d, :o_cr_id, \
:o_ol_cnt, :ol_s_w_id, :ol_i_id, :ol_quantity, :ol_amou
nt, :ol_d_d); END;"
#else

#define SQLCUR0 "SELECT rowid FROM cust \
WHERE c_d_id = :d_id AND c_w_id = :w_id
AND c_last = :c_last \
ORDER BY c_last, c_d_id, c_w_id, c_first"

#define SQLCUR1 "SELECT c_id, c_balance, c_first,
c_middle, c_last, \
o_id, o_entry_d, o_carrier_id, o_ol_cnt \
FROM cust, order \
WHERE cust.rowid = :cust_rowid \
AND o_d_id = c_d_id AND o_w_id =
c_w_id AND o_c_id = c_id \
ORDER BY o_c_id, o_d_id, o_w_id, o_id
DESC"

#define SQLCUR2 "SELECT c_balance, c_first,
c_middle, c_last, \
o_id, o_entry_d, o_carrier_id, o_ol_cnt, c_id \
FROM cust, order \
WHERE c_id = :c_id AND c_d_id = :d_id
AND c_w_id = :w_id \
AND o_d_id = c_d_id AND o_w_id = c_w_id
AND o_c_id = c_id \
ORDER BY o_c_id, o_d_id, o_w_id, o_id
DESC"

```

```
#define SQLCUR3 "SELECT
ol_i_id,ol_supply_w_id,ol_quantity,ol_amount, \
    ol_delivery_d\
FROM ordl \
WHERE ol_d_id = :d_id AND ol_w_id = :w_id
AND ol_o_id = :o_id"

#define SQLCUR4 "SELECT count(c_last) FROM cust
\
WHERE c_d_id = :d_id AND c_w_id = :w_id
AND c_last = :c_last "
#endif

struct ordctx {
sb2 c_rowid_ind[100];
sb2 ol_supply_w_id_ind[NITEMS];
sb2 ol_i_id_ind[NITEMS];
sb2 ol_quantity_ind[NITEMS];
sb2 ol_amount_ind[NITEMS];
sb2 ol_delivery_d_ind[NITEMS];
sb2 ol_w_id_ind;
sb2 ol_d_id_ind;
sb2 ol_o_id_ind;
sb2 c_id_ind;
sb2 c_first_ind;
sb2 c_middle_ind;
sb2 c_balance_ind;
sb2 c_last_ind;
sb2 o_id_ind;
sb2 o_entry_d_ind;
sb2 o_carrier_id_ind;
sb2 o_ol_cnt_ind;

ub4 c_rowid_len[100];
ub2 ol_supply_w_id_len[NITEMS];
ub2 ol_i_id_len[NITEMS];
ub2 ol_quantity_len[NITEMS];
ub2 ol_amount_len[NITEMS];
ub2 ol_delivery_d_len[NITEMS];
ub2 ol_w_id_len;
ub2 ol_d_id_len;
ub2 ol_o_id_len;

ub2 c_rowid_rcode[100];
ub2 ol_supply_w_id_rcode[NITEMS];
ub2 ol_i_id_rcode[NITEMS];
ub2 ol_quantity_rcode[NITEMS];
ub2 ol_amount_rcode[NITEMS];
ub2 ol_delivery_d_rcode[NITEMS];
ub2 ol_w_id_rcode;
ub2 ol_d_id_rcode;
ub2 ol_o_id_rcode;

ub4 ol_supply_w_id_csize;
ub4 ol_i_id_csize;
ub4 ol_quantity_csize;
ub4 ol_amount_csize;
ub4 ol_delivery_d_csize;
ub4 ol_w_id_csize;
ub4 ol_d_id_csize;
ub4 ol_o_id_csize;

OCISlmt *curo0;
OCIBind *w_id_bp0;
OCIBind *d_id_bp0;
OCIBind *c_id_bp;
OCIBind *c_last_bp;
```

```
#ifndef PLSQLORD
OCIBind *byln_bp;
OCIBind *c_first_bp;
OCIBind *c_middle_bp;
OCIBind *c_balance_bp;
OCIBind *o_entry_d_bp;
OCIBind *o_cr_id_bp;
OCIBind *o_ol_cnt_bp;
OCIBind *ol_i_id_bp;
OCIBind *ol_supply_w_id_bp;
OCIBind *ol_quantity_bp;
OCIBind *ol_amount_bp;
OCIBind *ol_d_base_bp;
ub4 ol_i_id_cnt;
ub4 ol_sup_cnt;
ub4 ol_qty_cnt;
ub4 ol_amt_cnt;
ub4 ol_del_d_cnt;
#else
OCISlmt *curo1;
OCISlmt *curo2;
OCISlmt *curo3;
OCISlmt *curo4;
OCIBind *w_id_bp2;
OCIBind *w_id_bp3;
OCIBind *w_id_bp4;
OCIBind *d_id_bp2;
OCIBind *d_id_bp3;
OCIBind *d_id_bp4;
OCIBind *c_last_bp4;
OCIBind *o_id_bp;
OCIBind *c_rowid_bp;
OCIDefine *c_rowid_dp;
OCIDefine *c_last_dp;
OCIDefine *c_last_dp1;
OCIDefine *c_id_dp;
OCIDefine *c_id_dp1;
OCIDefine *c_first_dp1;
OCIDefine *c_first_dp2;
OCIDefine *c_middle_dp1;
OCIDefine *c_middle_dp2;
OCIDefine *c_balance_dp1;
OCIDefine *c_balance_dp2;
OCIDefine *o_id_dp1;
OCIDefine *o_id_dp2;
OCIDefine *o_entry_d_dp1;
OCIDefine *o_entry_d_dp2;
OCIDefine *o_cr_id_dp1;
OCIDefine *o_cr_id_dp2;
OCIDefine *o_ol_cnt_dp1;
OCIDefine *o_ol_cnt_dp2;
OCIDefine *ol_d_dp;
OCIDefine *ol_i_id_dp;
OCIDefine *ol_supply_w_id_dp;
OCIDefine *ol_quantity_dp;
OCIDefine *ol_amount_dp;
OCIDefine *ol_d_base_dp;
OCIDefine *c_count_dp;
OCIRowid *c_rowid_ptr[100];
OCIRowid *middle_cust;

int cs;
int cust_idx;
int norow;
int rcount;
int somerows;
#endif
};
```

```
typedef struct ordctx ordctx;

struct defctx
{
boolean reexec;
ub4 count;
};
typedef struct defctx defctx;

ordctx *octx;

defctx cbctx;

#ifndef PLSQLORD
sb4 rid_data(dvoid *ctxp, OCIDefine *dp, ub4 iter,
dvoid **bufpp, ub4 **alenp, ub1 *piecep,
dvoid **indpp, ub2 **rcodepp)
{
ub4 i;
if (((defctx*)ctxp)->reexec)* if this is the second
execute - use entry 0 */
{
i = 0;
((defctx*)ctxp)->count--; /* count down */
}
else
i = iter;
*bufpp = octx->c_rowid_ptr[i];
*indpp = &octx->c_rowid_ind[i];
*alenp = &octx->c_rowid_len[i];
*rcodepp = &octx->c_rowid_rcode[i];
*piecep = OCI_ONE_PIECE;
return (OCI_CONTINUE);
}
#endif

tkvcoint ()
{
int i;
text stmbuff[SQL_BUF_SIZE];

octx = (ordctx *) malloc (sizeof(ordctx));
memset(octx, (char)0, sizeof(ordctx));
#ifndef PLSQLORD
octx->cs = 1;
octx->norow = 0;
octx->somerows = 10;
/* get the rowid handles */
for(i=0;i<100;i++) {
OCIERROR(errhp,
OCIDescriptorAlloc(tpcenv, (dvoid**) &octx-
>c_rowid_ptr[i],
OCI_DTYPE_ROWID, 0, (dvoid**)0));
}
#endif

OCIERROR(errhp,
OCIHandleAlloc(tpcenv, (dvoid**) &octx-
>curo0, OCI_HTYPE_STMT, 0, (dvoid**)0));
OCIERROR(errhp,
OCIHandleAlloc(tpcenv, (dvoid**) &octx-
>curo1, OCI_HTYPE_STMT, 0, (dvoid**)0));
#endif
};
```

```

OCIERROR(errhp,
OCIHandleAlloc(tpcenv,(dvoid*)&octx-
>curo2,OCI_HTYPE_STMT,0,(dvoid**)0));
OCIERROR(errhp,
OCIHandleAlloc(tpcenv,(dvoid*)&octx-
>curo3,OCI_HTYPE_STMT,0,(dvoid**)0));
OCIERROR(errhp,
OCIHandleAlloc(tpcenv,(dvoid*)&octx-
>curo4,OCI_HTYPE_STMT,0,(dvoid**)0));
#endif

#ifdef PLSQLORD
sprintf((char *) stmbuf, SQLTXT);
OCIERROR(errhp,
OCIStmtPrepare(octx-
>curo0,errhp,stmbuf,strlen((char *)stmbuf),
OCI_NTV_SYNTAX,OCI_DEFAULT));
#else
/* c_id = 0, use find customer by lastname. Get an
array or rowid's back */
sprintf((char *) stmbuf, SQLCUR0);
OCIERROR(errhp,
OCIStmtPrepare(octx-
>curo0,errhp,stmbuf,strlen((char *)stmbuf),
OCI_NTV_SYNTAX,OCI_DEFAULT));
OCIERROR(errhp,
OCIAttrSet(octx-
>curo0,OCI_HTYPE_STMT,(dvoid*)&octx->norow,0,
OCI_ATTR_PREFETCH_ROWS,errhp));
/* get order/customer info back based on rowid */
sprintf((char *) stmbuf, SQLCUR1);
OCIERROR(errhp,
OCIStmtPrepare(octx-
>curo1,errhp,stmbuf,strlen((char *)stmbuf),
OCI_NTV_SYNTAX,OCI_DEFAULT));
OCIERROR(errhp,
OCIAttrSet(octx-
>curo1,OCI_HTYPE_STMT,(dvoid*)&octx->norow,0,
OCI_ATTR_PREFETCH_ROWS,errhp));

/* c_id == 0, use lastname to find customer */
sprintf((char *) stmbuf, SQLCUR2);
OCIERROR(errhp,
OCIStmtPrepare(octx-
>curo2,errhp,stmbuf,strlen((char *)stmbuf),
OCI_NTV_SYNTAX,OCI_DEFAULT));
OCIERROR(errhp,
OCIAttrSet(octx-
>curo2,OCI_HTYPE_STMT,(dvoid*)&octx->norow,0,
OCI_ATTR_PREFETCH_ROWS,errhp));

sprintf((char *) stmbuf, SQLCUR3);
OCIERROR(errhp,
OCIStmtPrepare(octx-
>curo3,errhp,stmbuf,strlen((char *)stmbuf),
OCI_NTV_SYNTAX,OCI_DEFAULT));
OCIERROR(errhp,
OCIAttrSet(octx-
>curo3,OCI_HTYPE_STMT,(dvoid*)&octx->norow,0,
OCI_ATTR_PREFETCH_ROWS,errhp));

sprintf((char *) stmbuf, SQLCUR4);
OCIERROR(errhp,
OCIStmtPrepare(octx-
>curo4,errhp,stmbuf,strlen((char *)stmbuf),
OCI_NTV_SYNTAX,OCI_DEFAULT));
OCIERROR(errhp,
OCIAttrSet(octx-
>curo4,OCI_HTYPE_STMT,(dvoid*)&octx->norow,0,
OCI_ATTR_PREFETCH_ROWS,errhp));
#endif

OCI_ATTR_PREFETCH_ROWS,errhp));
#endif

for (i = 0; i < NITEMS; i++) {
octx->ol_supply_w_id_ind[i] = TRUE;
octx->ol_i_id_ind[i] = TRUE;
octx->ol_quantity_ind[i] = TRUE;
octx->ol_amount_ind[i] = TRUE;
octx->ol_delivery_d_ind[i] = TRUE;

octx->ol_supply_w_id_len[i] = sizeof(int);
octx->ol_i_id_len[i] = sizeof(int);
octx->ol_quantity_len[i] = sizeof(int);
octx->ol_amount_len[i] = sizeof(int);
octx->ol_delivery_d_len[i] = sizeof(ol_d_base[0]);
}
octx->ol_supply_w_id_csize = NITEMS;
octx->ol_i_id_csize = NITEMS;
octx->ol_quantity_csize = NITEMS;
octx->ol_amount_csize = NITEMS;
octx->ol_delivery_d_csize = NITEMS;
octx->ol_w_id_csize = NITEMS;
octx->ol_o_id_csize = NITEMS;
octx->ol_d_id_csize = NITEMS;
octx->ol_w_id_ind = TRUE;
octx->ol_d_id_ind = TRUE;
octx->ol_o_id_ind = TRUE;
octx->ol_w_id_len = sizeof(int);
octx->ol_d_id_len = sizeof(int);
octx->ol_o_id_len = sizeof(int);

/* bind variables */
#ifdef PLSQLORD
OCIBND(octx->curo0, octx->w_id_bp0,
errhp, "w_id", ADR(w_id),
SIZ(int), SQLT_INT);
OCIBND(octx->curo0, octx->d_id_bp0,
errhp, "d_id", ADR(d_id),
SIZ(int), SQLT_INT);
OCIBND(octx->curo0, octx->c_id_bp ,
errhp, "c_id", ADR(c_id),
SIZ(c_id), SQLT_INT);
OCIBND(octx->curo0, octx->byln_bp ,
errhp, "byln", ADR(bylastname),
SIZ(int), SQLT_INT);
OCIBND(octx->curo0, octx->c_last_bp ,
errhp, "c_last", c_last,
SIZ(c_last), SQLT_STR);
OCIBND(octx->curo0, octx->c_first_bp ,
errhp, "c_first", c_first,
SIZ(c_first), SQLT_STR);
OCIBND(octx->curo0, octx->c_middle_bp ,
errhp, "c_middle", c_middle,
SIZ(c_middle), SQLT_STR);
OCIBND(octx->curo0, octx->c_balance_bp ,
errhp, "c_balance",
ADR(c_balance), SIZ(float), SQLT_FLT);
OCIBND(octx->curo0, octx->c_id_bp ,
errhp, "o_id", ADR(o_id),
SIZ(int), SQLT_INT);
OCIBND(octx->curo0, octx->o_entry_d_bp ,
errhp, "o_entry_d", o_entry_d,
SIZ(o_entry_d), SQLT_STR);
OCIBND(octx->curo0, octx->o_cr_id_bp ,
errhp, "o_cr_id", ADR(o_carrier_id),
SIZ(int), SQLT_INT);
OCIBND(octx->curo0, octx->o_ol_cnt_bp ,
errhp, "o_ol_cnt", ADR(o_ol_cnt),
SIZ(int), SQLT_INT);
#endif
}

SIZ(int), SQLT_INT);

OCIBNDRAA(octx->curo0, octx->ol_i_id_bp, errhp,
"ol_i_id",
ol_i_id, SIZ(int), SQLT_INT,
octx->ol_i_id_ind, octx->ol_i_id_len,
octx->ol_i_id_rcode, NITEMS, &octx-
>ol_i_id_cnt);
OCIBNDRAA(octx->curo0, octx-
>ol_supply_w_id_bp, errhp, "ol_s_w_id",
ol_supply_w_id, SIZ(int), SQLT_INT,
octx->ol_supply_w_id_ind, octx-
>ol_supply_w_id_len,
octx->ol_supply_w_id_rcode, NITEMS, &octx-
>ol_sup_cnt);
OCIBNDRAA(octx->curo0, octx-
>ol_quantity_bp, errhp, "ol_quantity",
ol_quantity, SIZ(int), SQLT_INT,
octx->ol_quantity_ind, octx->ol_quantity_len,
octx->ol_quantity_rcode, NITEMS, &octx-
>ol_qty_cnt);
OCIBNDRAA(octx->curo0, octx-
>ol_amount_bp, errhp, "ol_amount", ol_amount,
SIZ(float), SQLT_FLT, octx->ol_amount_ind,
octx->ol_amount_len, octx-
>ol_amount_rcode, NITEMS,
&octx->ol_amt_cnt);
OCIBNDRAA(octx->curo0, octx-
>ol_d_base_bp, errhp, "ol_d_d", ol_d_base,
SIZ(OCIDate), SQLT_ODT, octx-
>ol_delivery_d_ind,
octx->ol_delivery_d_len, octx-
>ol_delivery_d_rcode, NITEMS,
&octx->ol_del_cnt);
#endif

/* c_id (customer id) is not known */
OCIBND(octx->curo0, octx-
>w_id_bp0, errhp, "w_id", ADR(w_id), SIZ(int), SQLT_INT
);
OCIBND(octx->curo0, octx-
>d_id_bp0, errhp, "d_id", ADR(d_id), SIZ(int), SQLT_INT)
;
OCIBND(octx->curo0, octx-
>c_id_bp0, errhp, "c_id", ADR(c_id), SIZ(int), SQLT_INT)
;
OCIBND(octx->curo0, octx-
>c_last_bp, errhp, "c_last", c_last, SIZ(c_last),
SQLT_STR);
OCIFNDYN(octx->curo0, octx-
>c_rowid_dp, errhp, 1, octx->c_rowid_ptr,
SIZ(OCIRowid), SQLT_RDD, octx-
>c_rowid_ind, &cbctx, rid_data);

OCIBND(octx->curo1, octx-
>c_rowid_bp, errhp, "cust_rowid",
&octx->middle_cust, sizeof(octx-
>middle_cust), SQLT_RDD);

OCIDEF(octx->curo1, octx-
>c_id_dp, errhp, 1, ADR(c_id), SIZ(int), SQLT_INT);
OCIDEF(octx->curo1, octx-
>c_balance_dp1, errhp, 2, ADR(c_balance),
SIZ(double), SQLT_FLT);
OCIDEF(octx->curo1, octx-
>c_first_dp1, errhp, 3, c_first, SIZ(c_first)-1,
SQLT_CHR);
OCIDEF(octx->curo1, octx-
>c_middle_dp1, errhp, 4, c_middle,
SIZ(c_middle)-1, SQLT_AFC);

```

```

OCIDEF(octx->curo1,octx-
>c_last_dp1,errhp,5,c_last,SIZ(c_last)-1,
SQLT_CHR);
OCIDEF(octx->curo1,octx-
>o_id_dp1,errhp,6,ADR(o_id),SIZ(int),SQLT_INT);
OCIDEF(octx->curo1,octx->o_entry_d_dp1,errhp,7,
&o_entry_d_base,SIZ(OCIDate),SQLT_ODT);
OCIDEF(octx->curo1,octx-
>o_cr_id_dp1,errhp,8,ADR(o_carrier_id),
SIZ(int),SQLT_INT);
OCIDEF(octx->curo1,octx-
>o_ol_cnt_dp1,errhp,9,ADR(o_ol_cnt),
SIZ(int),SQLT_INT);

/* Bind for third cursor , no-zero customer id */
OCIBND(octx->curo2,octx-
>w_id_bp2,errhp,"w_id",ADR(w_id),SIZ(int),SQLT_INT
);
OCIBND(octx->curo2,octx-
>d_id_bp2,errhp,"d_id",ADR(d_id),SIZ(int),SQLT_INT)
;
OCIBND(octx->curo2,octx-
>c_id_bp,errhp,"c_id",ADR(c_id),SIZ(int),SQLT_INT);
OCIDEF(octx->curo2,octx-
>c_balance_dp2,errhp,1,ADR(c_balance),
SIZ(double),SQLT_FLT);
OCIDEF(octx->curo2,octx-
>c_first_dp2,errhp,2,c_first,SIZ(c_first)-1,
SQLT_CHR);
OCIDEF(octx->curo2,octx-
>c_middle_dp2,errhp,3,c_middle,
SIZ(c_middle)-1,SQLT_AFC);
OCIDEF(octx->curo2,octx-
>c_last_dp,errhp,4,c_last,SIZ(c_last)-1, SQLT_CHR);
OCIDEF(octx->curo2,octx-
>o_id_dp2,errhp,5,ADR(o_id),SIZ(int),SQLT_INT);
OCIDEF(octx->curo2,octx->o_entry_d_dp2,errhp,6,
&o_entry_d_base,
SIZ(OCIDate),SQLT_ODT);
OCIDEF(octx->curo2, octx-
>o_cr_id_dp2,errhp,7,ADR(o_carrier_id),
SIZ(int), SQLT_INT);
OCIDEF(octx->curo2,octx-
>o_ol_cnt_dp2,errhp,8,ADR(o_ol_cnt),
SIZ(int),SQLT_INT);
OCIDEF(octx->curo2,octx-
>c_id_dp1,errhp,9,ADR(c_id),SIZ(int),SQLT_INT);

/* Bind for last cursor */

OCIBND(octx->curo3,octx-
>w_id_bp3,errhp,"w_id",ADR(w_id),SIZ(int),SQLT_INT
);
OCIBND(octx->curo3,octx-
>d_id_bp3,errhp,"d_id",ADR(d_id),SIZ(int),SQLT_INT)
;
OCIBND(octx->curo3,octx-
>o_id_bp,errhp,"o_id",ADR(o_id),SIZ(int),SQLT_INT);

OCIDFNRA(octx->curo3, octx->ol_i_id_dp, errhp, 1,
ol_i_id,SIZ(int),SQLT_INT,
octx->ol_i_id_ind,octx->ol_i_id_len, octx-
>ol_i_id_rcode);
OCIDFNRA(octx->curo3,octx-
>ol_supply_w_id_dp,errhp,2, ol_supply_w_id,
SIZ(int),SQLT_INT, octx->ol_supply_w_id_ind,
octx->ol_supply_w_id_len, octx-
>ol_supply_w_id_rcode);

OCIDFNRA(octx->curo3, octx-
>ol_quantity_dp,errhp,3, ol_quantity,SIZ(int),
SQLT_INT, octx->ol_quantity_ind,octx-
>ol_quantity_len,
octx->ol_quantity_rcode);
OCIDFNRA(octx->curo3,octx-
>ol_amount_dp,errhp,4,ol_amount, SIZ(int),
SQLT_INT,octx->ol_amount_ind, octx-
>ol_amount_len,
octx->ol_amount_rcode);
OCIDFNRA(octx->curo3,octx-
>ol_d_base_dp,errhp,5,ol_d_base,SIZ(OCIDate),
SQLT_ODT, octx->ol_delivery_d_ind,octx-
>ol_delivery_d_len,
octx->ol_delivery_d_rcode);

OCIBND(octx->curo4,octx-
>w_id_bp4,errhp,"w_id",ADR(w_id),SIZ(int),SQLT_INT
);
OCIBND(octx->curo4,octx-
>d_id_bp4,errhp,"d_id",ADR(d_id),SIZ(int),SQLT_INT)
;
OCIBND(octx->curo4,octx-
>c_last_bp4,errhp,"c_last",c_last,SIZ(c_last),
SQLT_STR);
OCIDEF(octx->curo4,octx-
>c_count_dp,errhp,1,ADR(octx->rcount),SIZ(int),
SQLT_INT);

#endif
return (0);
}

static void checkerr(errhp, status)
OCIError *errhp;
sword status;
{
text errbuf[512];
ub4 buflen;
sb4 errcode;

if (status == OCI_SUCCESS) return;

switch (status)
{
case OCI_SUCCESS_WITH_INFO:
printf("Error -
OCI_SUCCESS_WITH_INFO\n");
break;
case OCI_NEED_DATA:
printf("Error - OCI_NEED_DATA\n");
break;
case OCI_NO_DATA:
printf("Error - OCI_NO_DATA\n");
break;
case OCI_ERROR:
printf("OCI_ERROR\n");
while ( OCIErrorGet ((dvoid *) errhp, (ub4) 1,
(text *) NULL, &errcode,
errbuf, (ub4) sizeof(errbuf), (ub4)
OCI_HTYPE_ERROR) != OCI_NO_DATA)
printf("Error - %d: %s\n", errcode, errbuf);
break;
case OCI_INVALID_HANDLE:
printf("Error - OCI_INVALID_HANDLE\n");
break;
case OCI_STILL_EXECUTING:
printf("Error - OCI_STILL_EXECUTING\n");
break;
case OCI_CONTINUE:
printf("Error - OCI_CONTINUE\n");
break;
default:
printf("Error - %d\n", status);
printf("default\n");
break;
}
}

tkvco ()
{
int i;
int rcount;

//kawabata add
//printf("tkvco start1 ");
//printf("w_id = %d",w_id);
//printf("d_id = %d",d_id);
//printf("c_id = %d",c_id);

for (i = 0; i < NITEMS; i++) {
octx->ol_supply_w_id_ind[i] = TRUE;
octx->ol_i_id_ind[i] = TRUE;
octx->ol_quantity_ind[i] = TRUE;
octx->ol_amount_ind[i] = TRUE;
octx->ol_delivery_d_ind[i] = TRUE;
octx->ol_supply_w_id_len[i] = sizeof(int);
octx->ol_i_id_len[i] = sizeof(int);
octx->ol_quantity_len[i] = sizeof(int);
octx->ol_amount_len[i] = sizeof(int);
octx->ol_delivery_d_len[i] = sizeof(OCIDate);
}

octx->ol_supply_w_id_csize = NITEMS;
octx->ol_i_id_csize = NITEMS;
octx->ol_quantity_csize = NITEMS;
octx->ol_amount_csize = NITEMS;
octx->ol_delivery_d_csize = NITEMS;
#ifdef PLSQLORD
octx->ol_i_id_cnt = 0;
octx->ol_sup_cnt = 0;
octx->ol_qly_cnt = 0;
octx->ol_ami_cnt = 0;
octx->ol_del_d_cnt = 0;
OCIERROR(errhp,
OCIStmtExecute(tpscvc,octx-
>curo0,errhp,1,0,0,0,OCI_DEFAULT));
#else
retry:

if(bylastname)
{
cbctx.reexec = FALSE;
execstatus=OCIStmtExecute(tpscvc,octx-
>curo0,errhp,100,0,0,0,OCI_DEFAULT);
/* will get OCI_NO_DATA if <100 found */
}
}
}

```



```

if ((execstatus != OCI_NO_DATA) && (execstatus !=
OCI_SUCCESS))
{
    errcode=OCIERROR(errhp, execstatus);
    if((errcode == NOT_SERIALIZABLE) || (errcode ==
RECOVERR)
        || (errcode == SNAPSHOT_TOO_OLD))
    {
        OCITransCommit(tpcsvc,errhp,OCI_DEFAULT);
        retries++;
        goto retry;
    } else {
        return -1;
    }
}
if (execstatus == OCI_NO_DATA) /* there are no
more rows */
{
    /* get rowcount, find middle one */
    OCIAttrGet(octx-
>curo0,OCI_HTYPE_STMT,&rcount,NULL,OCI_ATTR
_ROW_COUNT,errhp);
    if (rcount < 1)
    {
        printf("No Data Found\n");
        return (-1);
    }
    octx->cust_idx=(rcount-1)/2 ;
}
else
{
    /* count the number of rows */

    execstatus=OCISmtExecute(tpcsvc,octx-
>curo4,errhp,1,0,0,0,OCI_DEFAULT);
    if ((execstatus != OCI_NO_DATA) &&
(execstatus != OCI_SUCCESS))
    {
        if ((errcode == NOT_SERIALIZABLE) || (errcode
== RECOVERR)
            || (errcode == SNAPSHOT_TOO_OLD))
        {
            OCITransCommit(tpcsvc,errhp,OCI_DEFAULT);
            retries++;
            goto retry;
        } else {
            return -1;
        }
    }
    if (octx->rcount+1 < 200 )
        octx->cust_idx=(octx->rcount-1)/2 ;
    else /* */
    {
        cbctx.reexec = TRUE;
        cbctx.count = (octx->rcount+1)/2 ;
        execstatus=OCISmtExecute(tpcsvc,octx-
>curo0,errhp,cbctx.count,
            0,0,0,OCI_DEFAULT);
        /* will get OCI_NO_DATA if <100 found */
        if (cbctx.count > 0)
        {
            printf ("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)
                || (errcode == SNAPSHOT_TOO_OLD))
            {
                OCITransCommit(tpcsvc,errhp,OCI_DEFAULT
);
                retries++;
                goto retry;
            } else {
                return -1;
            }
        }
        octx->middle_cust = octx->c_rowid_ptr[octx-
>cust_idx];
        execstatus=OCISmtExecute(tpcsvc,octx-
>curo1,errhp,1,0,0,0,OCI_DEFAULT);

        if (execstatus != OCI_SUCCESS)
        {
            errcode=OCIERROR(errhp,execstatus);
            OCITransCommit(tpcsvc,errhp,OCI_DEFAULT);
            if((errcode == NOT_SERIALIZABLE) || (errcode ==
RECOVERR) ||
                (errcode == SNAPSHOT_TOO_OLD))
            {
                retries++;
                goto retry;
            } else {
                return -1;
            }
        }
        execstatus=OCISmtExecute(tpcsvc,octx-
>curo2,errhp,1,0,0,0,OCI_DEFAULT);
        if (execstatus != OCI_SUCCESS)
        {
            /*checkerr(errhp, execstatus);
            errcode=OCIERROR(errhp,execstatus);
            OCITransCommit(tpcsvc,errhp,OCI_DEFAULT);
            if((errcode == NOT_SERIALIZABLE) || (errcode ==
RECOVERR)
                || (errcode == SNAPSHOT_TOO_OLD))
            {
                retries++;
                goto retry;
            }
            else
            {
                return -1;
            }
        }
        octx->ol_w_id_ind = TRUE;
        octx->ol_d_id_ind = TRUE;
        octx->ol_o_id_ind = TRUE;
        octx->ol_w_id_len = sizeof(int);
        octx->ol_d_id_len = sizeof(int);
        octx->ol_o_id_len = sizeof(int);

        execstatus = OCISmtExecute(tpcsvc,octx-
>curo3,errhp,o_ol_cnt,0,0,0,
            OCI_DEFAULT |
OCI_COMMIT_ON_SUCCESS);
        if (execstatus != OCI_SUCCESS )
        {
            /*checkerr(errhp, execstatus);
            errcode=OCIERROR(errhp,execstatus);
            OCITransCommit(tpcsvc,errhp,OCI_DEFAULT);
            if((errcode == NOT_SERIALIZABLE) || (errcode ==
RECOVERR)
                || (errcode == SNAPSHOT_TOO_OLD))
            {
                retries++;
                goto retry;
            }
            else
            {
                return -1;
            }
        }
        #endif
        /* clean up and convert the delivery dates */
        for (i = 0; i < o_ol_cnt; i++)
        {
            if (octx->ol_delivery_d_ind[i] == -1) /* null date in
field */
                strncpy((char*)ol_delivery_d[i],"01-01-1811",10);
            else
            {
                ol_del_len[i]=sizeof(ol_delivery_d[i]);
                OCIERROR(errhp,OCIDateToText(errhp,&ol_d_b
ase[i],
                    (text*)SHORTDATE,strlen(SHORTDATE),(text*)
0,0,&ol_del_len[i],ol_delivery_d[i]));
            }
        }

        /*
        cvtdmy(ol_d_base[i],ol_delivery_d[i]);
        */

        return (0);
    }
}

void tkvcodone ()
{
}

if (octx)
    free (octx);
}

===== 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 "tpcc.h"
#include "tpcc_info.h"
#include "tpccflags.h"

#ifdef TUX
#include <userlog.h>
#endif

#define SQLTXT_INIT "BEGIN initpay.pay_init; END;"
#define SQLTXT_STP "begin
payment.dopayment(:w_id,:d_id,:c_w_id,:c_d_id, \
:c_id,:by_lname,:h_amount,:c_last,:w_stree
t_1,:w_street_1, \
:w_city,:w_state,:w_zip,:d_street_1,:d_stre
et_2,:d_city, \
:d_state,:d_zip,:c_first,:c_middle,:c_street_
1, \
:c_street_2,:c_city,:c_state,:c_zip,:c_phone
,:c_since, \
:c_credit,:c_credit_lim,:c_discount,:c_balan
ce,:c_data, \
:cr_date,:retry); end;"

struct payctx {
OCIStmt *curpi;
OCIStmt *curp0;
OCIStmt *curp1;
OCIBind *w_id_bp;
OCIBind *w_id_bp1;
sb2 w_id_ind;
ub2 w_id_len;
ub2 w_id_rc;

OCIBind *d_id_bp;
OCIBind *d_id_bp1;
sb2 d_id_ind;
ub2 d_id_len;
ub2 d_id_rc;

OCIBind *c_w_id_bp;
OCIBind *c_w_id_bp1;
sb2 c_w_id_ind;
ub2 c_w_id_len;
ub2 c_w_id_rc;

OCIBind *c_d_id_bp;
OCIBind *c_d_id_bp1;
sb2 c_d_id_ind;
ub2 c_d_id_len;

ub2 c_d_id_rc;

OCIBind *c_id_bp;
OCIBind *c_id_bp1;
sb2 c_id_ind;
ub2 c_id_len;
ub2 c_id_rc;

OCIBind *by_lname_bp;

OCIBind *h_amount_bp;
OCIBind *h_amount_bp1;
sb2 h_amount_ind;
ub2 h_amount_len;
ub2 h_amount_rc;

OCIBind *c_last_bp;
OCIBind *c_last_bp1;
sb2 c_last_ind;
ub2 c_last_len;
ub2 c_last_rc;

OCIBind *w_street_1_bp;
OCIBind *w_street_1_bp1;
sb2 w_street_1_ind;
ub2 w_street_1_len;
ub2 w_street_1_rc;

OCIBind *w_street_2_bp;
OCIBind *w_street_2_bp1;
sb2 w_street_2_ind;
ub2 w_street_2_len;
ub2 w_street_2_rc;

OCIBind *w_city_bp;
OCIBind *w_city_bp1;
sb2 w_city_ind;
ub2 w_city_len;
ub2 w_city_rc;

OCIBind *w_state_bp;
OCIBind *w_state_bp1;
sb2 w_state_ind;
ub2 w_state_len;
ub2 w_state_rc;

OCIBind *w_zip_bp;
OCIBind *w_zip_bp1;
sb2 w_zip_ind;
ub2 w_zip_len;
ub2 w_zip_rc;

OCIBind *d_street_1_bp;
OCIBind *d_street_1_bp1;
sb2 d_street_1_ind;
ub2 d_street_1_len;
ub2 d_street_1_rc;

OCIBind *d_street_2_bp;
OCIBind *d_street_2_bp1;
sb2 d_street_2_ind;
ub2 d_street_2_len;
ub2 d_street_2_rc;

OCIBind *d_city_bp;
OCIBind *d_city_bp1;
sb2 d_city_ind;
ub2 d_city_len;
ub2 d_city_rc;

OCIBind *d_state_bp;
OCIBind *d_state_bp1;
sb2 d_state_ind;
ub2 d_state_len;
ub2 d_state_rc;

OCIBind *d_zip_bp;
OCIBind *d_zip_bp1;
sb2 d_zip_ind;
ub2 d_zip_len;
ub2 d_zip_rc;

OCIBind *c_first_bp;
OCIBind *c_first_bp1;
sb2 c_first_ind;
ub2 c_first_len;
ub2 c_first_rc;

OCIBind *c_middle_bp;
OCIBind *c_middle_bp1;
sb2 c_middle_ind;
ub2 c_middle_len;
ub2 c_middle_rc;

OCIBind *c_street_1_bp;
OCIBind *c_street_1_bp1;
sb2 c_street_1_ind;
ub2 c_street_1_len;
ub2 c_street_1_rc;

OCIBind *c_street_2_bp;
OCIBind *c_street_2_bp1;
sb2 c_street_2_ind;
ub2 c_street_2_len;
ub2 c_street_2_rc;

OCIBind *c_city_bp;
OCIBind *c_city_bp1;
sb2 c_city_ind;
ub2 c_city_len;
ub2 c_city_rc;

OCIBind *c_state_bp;
OCIBind *c_state_bp1;
sb2 c_state_ind;
ub2 c_state_len;
ub2 c_state_rc;

OCIBind *c_zip_bp;
OCIBind *c_zip_bp1;
sb2 c_zip_ind;
ub2 c_zip_len;
ub2 c_zip_rc;

OCIBind *c_phone_bp;
OCIBind *c_phone_bp1;
sb2 c_phone_ind;
ub2 c_phone_len;
ub2 c_phone_rc;

OCIBind *c_since_bp;
OCIBind *c_since_bp1;
sb2 c_since_ind;
ub2 c_since_len;
ub2 c_since_rc;

OCIBind *c_credit_bp;
OCIBind *c_credit_bp1;
sb2 c_credit_ind;
ub2 c_credit_len;
ub2 c_credit_rc;

```

```

sb2 c_credit_ind;
ub2 c_credit_len;
ub2 c_credit_rc;

OCIBind *c_credit_lim_bp;
OCIBind *c_credit_lim_bp1;
sb2 c_credit_lim_ind;
ub2 c_credit_lim_len;
ub2 c_credit_lim_rc;

OCIBind *c_discount_bp;
OCIBind *c_discount_bp1;
sb2 c_discount_ind;
ub2 c_discount_len;
ub2 c_discount_rc;

OCIBind *c_balance_bp;
OCIBind *c_balance_bp1;
sb2 c_balance_ind;
ub2 c_balance_len;
ub2 c_balance_rc;

OCIBind *c_data_bp;
OCIBind *c_data_bp1;
sb2 c_data_ind;
ub2 c_data_len;
ub2 c_data_rc;

OCIBind *h_date_bp;
OCIBind *h_date_bp1;
sb2 h_date_ind;
ub2 h_date_len;
ub2 h_date_rc;

OCIBind *retries_bp;
OCIBind *retries_bp1;
sb2 retries_ind;
ub2 retries_len;
ub2 retries_rc;

OCIBind *cr_date_bp;
OCIBind *cr_date_bp1;
sb2 cr_date_ind;
ub2 cr_date_len;
ub2 cr_date_rc;

OCIBind *byln_bp;
sb2 byln_ind;
ub2 byln_len;
ub2 byln_rc;
};

typedef struct payctx payctx;

payctx *pctx;

int tkvcpininit (void)
{
    text stmbuff[SQL_BUF_SIZE];
    pctx = (payctx *)malloc(sizeof(payctx));
    memset(pctx, (char)0, sizeof(payctx));

    /* cursor for init */
    OCIERROR(errhp,OCIHandleAlloc(tpcenv, (dvoid
**)(&(pctx->curp1)),
        OCI_HTYPE_STMT,0,(dvoid**)0));

    OCIERROR(errhp,OCIHandleAlloc(tpcenv, (dvoid
**)(&(pctx->curp0)),
        OCI_HTYPE_STMT,0,(dvoid**)0));
    OCIERROR(errhp,OCIHandleAlloc(tpcenv, (dvoid
**)(&(pctx->curp1)),
        OCI_HTYPE_STMT,0,(dvoid**)0));

    /* build the init statement and execute it */

    sprintf ((char*)stmbuff, SQLTXT_INIT);
    OCIERROR(errhp,OCIStmtPrepare(pctx->curp1, errhp,
stmbuff,
        strlen((char *)stmbuff), OCI_NTV_SYNTAX,
OCI_DEFAULT));
    OCIERROR(errhp,
        OCISmExecute(tpcsvc,pctx-
>curp1,errhp,1,0,0,0,OCI_DEFAULT));
#ifdef PLSQLPAY
    /* prepare the stub for calling pls sql stored procedure
*/
    sprintf ((char*)stmbuff, SQLTXT_STP);
    OCIERROR(errhp,OCIStmtPrepare(pctx->curp0,
errhp, stmbuff,
        strlen((char *)stmbuff), OCI_NTV_SYNTAX,
OCI_DEFAULT));
#else

    /* customer id != 0, go by last name */

    sqlfile("../blocks/paynz.sql",stmbuff);
    OCIERROR(errhp,OCIStmtPrepare(pctx->curp0,
errhp, stmbuff,
        strlen((char *)stmbuff), OCI_NTV_SYNTAX,
OCI_DEFAULT));

    /* customer id == 0, go by last name */

    sqlfile("../blocks/payz.sql",stmbuff); /* sqlfile opens
$/bench/.../blocks/... */
    OCIERROR(errhp,OCIStmtPrepare(pctx->curp1,
errhp, stmbuff,
        strlen((char *)stmbuff), OCI_NTV_SYNTAX,
OCI_DEFAULT));
#endif
    pctx->w_id_ind = TRUE;
    pctx->w_id_len = SIZ(w_id);
    pctx->d_id_ind = TRUE;
    pctx->d_id_len = SIZ(d_id);
    pctx->c_w_id_ind = TRUE;
    pctx->c_w_id_len = SIZ(c_w_id);
    pctx->c_d_id_ind = TRUE;
    pctx->c_d_id_len = SIZ(c_d_id);
    pctx->c_id_ind = TRUE;
    pctx->c_id_len = 0;
    pctx->h_amount_len = SIZ(h_amount);
    pctx->h_amount_ind = TRUE;
    pctx->c_last_ind = TRUE;
    pctx->c_last_len = 0;
    pctx->w_street_1_ind = TRUE;
    pctx->w_street_1_len = 0;
    pctx->w_street_2_ind = TRUE;
    pctx->w_street_2_len = 0;
    pctx->w_city_ind = TRUE;
    pctx->w_city_len = 0;

    pctx->w_state_ind = TRUE;
    pctx->w_state_len = 0;
    pctx->w_zip_ind = TRUE;
    pctx->w_zip_len = 0;
    pctx->d_street_1_ind = TRUE;
    pctx->d_street_1_len = 0;
    pctx->d_street_2_ind = TRUE;
    pctx->d_street_2_len = 0;
    pctx->d_city_ind = TRUE;
    pctx->d_city_len = 0;
    pctx->d_state_ind = TRUE;
    pctx->d_state_len = 0;
    pctx->d_zip_ind = TRUE;
    pctx->d_zip_len = 0;
    pctx->c_first_ind = TRUE;
    pctx->c_first_len = 0;
    pctx->c_middle_ind = TRUE;
    pctx->c_middle_len = 0;
    pctx->c_street_1_ind = TRUE;
    pctx->c_street_1_len = 0;
    pctx->c_street_2_ind = TRUE;
    pctx->c_street_2_len = 0;
    pctx->c_city_ind = TRUE;
    pctx->c_city_len = 0;
    pctx->c_state_ind = TRUE;
    pctx->c_state_len = 0;
    pctx->c_zip_ind = TRUE;
    pctx->c_zip_len = 0;
    pctx->c_phone_ind = TRUE;
    pctx->c_phone_len = 0;
    pctx->c_since_ind = TRUE;
    pctx->c_since_len = 0;
    pctx->c_credit_ind = TRUE;
    pctx->c_credit_len = 0;
    pctx->c_credit_lim_ind = TRUE;
    pctx->c_credit_lim_len = 0;
    pctx->c_discount_ind = TRUE;
    pctx->c_discount_len = 0;
    pctx->c_balance_ind = TRUE;
    pctx->c_balance_len = sizeof(double);
    pctx->c_data_ind = TRUE;
    pctx->c_data_len = 0;
    pctx->h_date_ind = TRUE;
    pctx->h_date_len = 0;
    pctx->retries_ind = TRUE;
    pctx->retries_len = 0;
    pctx->cr_date_ind = TRUE;
    pctx->cr_date_len = 7;

    /* bind variables */

    OCIBNDR(pctx->curp0, pctx->w_id_bp,
errhp, "w_id",ADR(w_id),SIZ(int),
        SOLT_INT, &pctx->w_id_ind, NULL, NULL);
    OCIBNDR(pctx->curp0, pctx->d_id_bp,
errhp, "d_id",ADR(d_id),SIZ(int),
        SOLT_INT, &pctx->d_id_ind, NULL, NULL);
    OCIBND(pctx->curp0, pctx->c_w_id_bp,
errhp, "c_w_id",ADR(c_w_id),SIZ(int),
        SOLT_INT);
    OCIBND(pctx->curp0, pctx->c_d_id_bp,
errhp, "c_d_id",ADR(c_d_id),SIZ(int),
        SOLT_INT);
    OCIBND(pctx->curp0, pctx->c_id_bp,
errhp, "c_id",ADR(c_id),SIZ(int),
        SOLT_INT);

```

```
#ifndef PLSQLPAY
OCIBND(pctx->curp0, pctx->by_lname_bp,
errhp,":by_lname",ADR(bylastname),
SIZ(int), SOLT_INT);
#endif
OCIBNDR(pctx->curp0, pctx->h_amount_bp,
errhp,":h_amount",ADR(h_amount),
SIZ(int),SOLT_INT, &pctx->h_amount_ind,
&pctx->h_amount_len,
&pctx->h_amount_rc);
OCIBNDR(pctx->curp0, pctx->c_last_bp,
errhp,":c_last",c_last,SIZ(c_last),
SOLT_STR, &pctx->c_last_ind, &pctx-
>c_last_len, &pctx->c_last_rc);
OCIBNDR(pctx->curp0, pctx->w_street_1_bp,
errhp,":w_street_1",w_street_1,
SIZ(w_street_1),SOLT_STR, &pctx-
>w_street_1_ind,
&pctx->w_street_1_len, &pctx->w_street_1_rc);
OCIBNDR(pctx->curp0, pctx->w_street_2_bp,
errhp,":w_street_2",w_street_2,
SIZ(w_street_2),SOLT_STR, &pctx-
>w_street_2_ind,
&pctx->w_street_2_len, &pctx->w_street_2_rc);
OCIBNDR(pctx->curp0, pctx->w_city_bp,
errhp,":w_city",w_city,SIZ(w_city),
SOLT_STR, &pctx->w_city_ind, &pctx-
>w_city_len, &pctx->w_city_rc);
OCIBNDR(pctx->curp0, pctx->w_state_bp,
errhp,":w_state",w_state,SIZ(w_state),
SOLT_STR, &pctx->w_state_ind, &pctx-
>w_state_len, &pctx->w_state_rc);
OCIBNDR(pctx->curp0, pctx->w_zip_bp,
errhp,":w_zip",w_zip,SIZ(w_zip),
SOLT_STR, &pctx->w_zip_ind, &pctx-
>w_zip_len, &pctx->w_zip_rc);
OCIBNDR(pctx->curp0, pctx->d_street_1_bp,
errhp,":d_street_1",d_street_1,
SIZ(d_street_1),SOLT_STR, &pctx-
>d_street_1_ind,
&pctx->d_street_1_len, &pctx->d_street_1_rc);
OCIBNDR(pctx->curp0, pctx->d_street_2_bp,
errhp,":d_street_2",d_street_2,
SIZ(d_street_2),SOLT_STR, &pctx-
>d_street_2_ind,
&pctx->d_street_2_len, &pctx->d_street_2_rc);
OCIBNDR(pctx->curp0, pctx->d_city_bp,
errhp,":d_city",d_city,SIZ(d_city),
SOLT_STR, &pctx->d_city_ind, &pctx-
>d_city_len, &pctx->d_city_rc);
OCIBNDR(pctx->curp0, pctx->d_state_bp,
errhp,":d_state",d_state,SIZ(d_state),
SOLT_STR, &pctx->d_state_ind, &pctx-
>d_state_len, &pctx->d_state_rc);
OCIBNDR(pctx->curp0, pctx->d_zip_bp,
errhp,":d_zip",d_zip,SIZ(d_zip),
SOLT_STR, &pctx->d_zip_ind, &pctx->d_zip_len,
&pctx->d_zip_rc);
OCIBNDR(pctx->curp0, pctx->c_first_bp,
errhp,":c_first",c_first,SIZ(c_first),
SOLT_STR, &pctx->c_first_ind, &pctx-
>c_first_len, &pctx->c_first_rc);
OCIBNDR(pctx->curp0, pctx->c_middle_bp,
errhp,":c_middle",c_middle,2,
SOLT_AFC, &pctx->c_middle_ind, &pctx-
>c_middle_len,
&pctx->c_middle_rc);
OCIBNDR(pctx->curp0, pctx->c_street_1_bp,
errhp,":c_street_1",c_street_1,
```

```
SIZ(c_street_1),SOLT_STR, &pctx-
>c_street_1_ind,
&pctx->c_street_1_len, &pctx->c_street_1_rc);
OCIBNDR(pctx->curp0, pctx->c_street_2_bp,
errhp,":c_street_2",c_street_2,
SIZ(c_street_2),SOLT_STR, &pctx-
>c_street_2_ind,
&pctx->c_street_2_len, &pctx->c_street_2_rc);
OCIBNDR(pctx->curp0, pctx->c_city_bp,
errhp,":c_city",c_city,SIZ(c_city),
SOLT_STR, &pctx->c_city_ind, &pctx-
>c_city_len, &pctx->c_city_rc);
OCIBNDR(pctx->curp0, pctx->c_state_bp,
errhp,":c_state",c_state,SIZ(c_state),
SOLT_STR, &pctx->c_state_ind, &pctx-
>c_state_len, &pctx->c_state_rc);
OCIBNDR(pctx->curp0, pctx->c_zip_bp,
errhp,":c_zip",c_zip,SIZ(c_zip),
SOLT_STR, &pctx->c_zip_ind, &pctx->c_zip_len,
&pctx->c_zip_rc);
OCIBNDR(pctx->curp0, pctx->c_phone_bp,
errhp,":c_phone",c_phone,SIZ(c_phone),
SOLT_STR, &pctx->c_phone_ind, &pctx->c_phone_len,
&pctx->c_phone_rc);
OCIBNDR(pctx->curp0, pctx->c_since_bp,
errhp,":c_since",&c_since,
SIZ(OCIDate), SOLT_ODT, &pctx->c_since_ind,
&pctx->c_since_len,
&pctx->c_since_rc);
OCIBNDR(pctx->curp0, pctx->c_credit_bp,
errhp,":c_credit",c_credit,
SIZ(c_credit),SOLT_CHR, &pctx->c_credit_ind,
&pctx->c_credit_len,
&pctx->c_credit_rc);
OCIBNDR(pctx->curp0, pctx->c_credit_lim_bp,
errhp,":c_credit_lim",
ADR(c_credit_lim),SIZ(int), SOLT_INT, &pctx-
>c_credit_lim_ind,
&pctx->c_credit_lim_len, &pctx-
>c_credit_lim_rc);
OCIBNDR(pctx->curp0, pctx->c_discount_bp,
errhp,":c_discount",
ADR(c_discount),SIZ(c_discount), SOLT_FLT,
&pctx->c_discount_ind,
&pctx->c_discount_len, &pctx->c_discount_rc);
OCIBNDR(pctx->curp0, pctx->c_balance_bp,
errhp,":c_balance",ADR(c_balance),
SIZ(double),SOLT_FLT, &pctx->c_balance_ind,
&pctx->c_balance_len,
&pctx->c_balance_rc);
OCIBNDR(pctx->curp0, pctx->c_data_bp,
errhp,":c_data",c_data,SIZ(c_data),
SOLT_STR, &pctx->c_data_ind, &pctx-
>c_data_len, &pctx->c_data_rc);
/*
OCIBNDR(pctx->curp0, pctx->h_date_bp,
errhp,":h_date",h_date,SIZ(h_date),
SOLT_STR, &pctx->h_date_ind, &pctx-
>h_date_len, &pctx->h_date_rc);
*/
OCIBNDR(pctx->curp0, pctx->retries_bp,
errhp,":retry",ADR(retries),SIZ(int),
SOLT_INT, &pctx->retries_ind, &pctx-
>retries_len, &pctx->retries_rc);
OCIBNDR(pctx->curp0, pctx->cr_date_bp,
errhp,":cr_date",ADR(cr_date),
SIZ(OCIDate),SOLT_ODT, &pctx->cr_date_ind,
&pctx->cr_date_len,
&pctx->cr_date_rc);
```

```
#ifndef PLSQLPAY
/* ---- Binds for the second cursor */

OCIBNDR(pctx->curp1, pctx->w_id_bp1,
errhp,":w_id",ADR(w_id),SIZ(int),
SOLT_INT, &pctx->w_id_ind, &pctx->w_id_len,
&pctx->w_id_rc);
OCIBNDR(pctx->curp1, pctx->d_id_bp1,
errhp,":d_id",ADR(d_id),SIZ(int),
SOLT_INT, &pctx->d_id_ind, &pctx->d_id_len,
&pctx->d_id_rc);
OCIBND(pctx->curp1, pctx->c_w_id_bp1,
errhp,":c_w_id",ADR(c_w_id),SIZ(int),
SOLT_INT);
OCIBND(pctx->curp1, pctx->c_d_id_bp1,
errhp,":c_d_id",ADR(c_d_id),SIZ(int),
SOLT_INT);
OCIBNDR(pctx->curp1, pctx->c_id_bp1,
errhp,":c_id",ADR(c_id),SIZ(int),
SOLT_INT, &pctx->c_id_ind, &pctx->c_id_len,
&pctx->c_id_rc);
OCIBNDR(pctx->curp1, pctx->h_amount_bp1,
errhp,":h_amount",ADR(h_amount),
SIZ(int),SOLT_INT, &pctx->h_amount_ind,
&pctx->h_amount_len,
&pctx->h_amount_rc);
OCIBND(pctx->curp1, pctx->c_last_bp1,
errhp,":c_last",c_last,SIZ(c_last),
SOLT_STR);
OCIBNDR(pctx->curp1, pctx->w_street_1_bp1,
errhp,":w_street_1",w_street_1,
SIZ(w_street_1),SOLT_STR, &pctx-
>w_street_1_ind,
&pctx->w_street_1_len, &pctx->w_street_1_rc);
OCIBNDR(pctx->curp1, pctx->w_street_2_bp1,
errhp,":w_street_2",w_street_2,
SIZ(w_street_2),SOLT_STR, &pctx-
>w_street_2_ind,
&pctx->w_street_2_len, &pctx->w_street_2_rc);
OCIBNDR(pctx->curp1, pctx->w_city_bp1,
errhp,":w_city",w_city,SIZ(w_city),
SOLT_STR, &pctx->w_city_ind, &pctx-
>w_city_len, &pctx->w_city_rc);
OCIBNDR(pctx->curp1, pctx->w_state_bp1,
errhp,":w_state",w_state,SIZ(w_state),
SOLT_STR, &pctx->w_state_ind, &pctx-
>w_state_len, &pctx->w_state_rc);
OCIBNDR(pctx->curp1, pctx->w_zip_bp1,
errhp,":w_zip",w_zip,SIZ(w_zip),
SOLT_STR, &pctx->w_zip_ind, &pctx-
>w_zip_len, &pctx->w_zip_rc);
OCIBNDR(pctx->curp1, pctx->d_street_1_bp1,
errhp,":d_street_1",d_street_1,
SIZ(d_street_1),SOLT_STR, &pctx-
>d_street_1_ind,
&pctx->d_street_1_len, &pctx->d_street_1_rc);
OCIBNDR(pctx->curp1, pctx->d_street_2_bp1,
errhp,":d_street_2",d_street_2,
SIZ(d_street_2),SOLT_STR, &pctx-
>d_street_2_ind,
&pctx->d_street_2_len, &pctx->d_street_2_rc);
OCIBNDR(pctx->curp1, pctx->d_city_bp1,
errhp,":d_city",d_city,SIZ(d_city),
SOLT_STR, &pctx->d_city_ind, &pctx-
>d_city_len, &pctx->d_city_rc);
OCIBNDR(pctx->curp1, pctx->d_state_bp1,
errhp,":d_state",d_state,
```

```

        SIZ(d_state), SQLT_STR, &pctx->d_state_ind,
&pctx->d_state_len,
        &pctx->d_state_rc);
    OCIBNDR(pctx->curp1, pctx->d_zip_bp1,
errhp,":d_zip",d_zip,SIZ(d_zip),
        SQLT_STR, &pctx->d_zip_ind, &pctx->d_zip_len,
&pctx->d_zip_rc);
    OCIBNDR(pctx->curp1, pctx->c_first_bp1,
errhp,":c_first",c_first,
        SIZ(c_first), SQLT_STR, &pctx->c_first_ind,
&pctx->c_first_len,
        &pctx->c_first_rc);
    OCIBNDR(pctx->curp1, pctx->c_middle_bp1,
errhp,":c_middle",c_middle,2,
        SQLT_AFC, &pctx->c_middle_ind, &pctx-
>c_middle_len,
        &pctx->c_middle_rc);

    OCIBNDR(pctx->curp1, pctx->c_street_1_bp1,
errhp,":c_street_1",c_street_1,
        SIZ(c_street_1),SQLT_STR, &pctx-
>c_street_1_ind,
        &pctx->c_street_1_len, &pctx->c_street_1_rc);
    OCIBNDR(pctx->curp1, pctx->c_street_2_bp1,
errhp,":c_street_2",c_street_2,
        SIZ(c_street_2),SQLT_STR, &pctx-
>c_street_2_ind,
        &pctx->c_street_2_len, &pctx->c_street_2_rc);
    OCIBNDR(pctx->curp1, pctx->c_city_bp1,
errhp,":c_city",c_city,SIZ(c_city),SQLT_STR,
        &pctx->c_city_ind, &pctx->c_city_len, &pctx-
>c_city_rc);
    OCIBNDR(pctx->curp1, pctx->c_state_bp1,
errhp,":c_state",c_state,SIZ(c_state), SQLT_STR,
&pctx->c_state_ind, &pctx->c_state_len, &pctx-
>c_state_rc);
    OCIBNDR(pctx->curp1, pctx->c_zip_bp1,
errhp,":c_zip",c_zip,SIZ(c_zip),
        SQLT_STR, &pctx->c_zip_ind, &pctx->c_zip_len,
&pctx->c_zip_rc);
    OCIBNDR(pctx->curp1, pctx->c_phone_bp1,
errhp,":c_phone",c_phone,SIZ(c_phone),
        SQLT_STR, &pctx->c_phone_ind, &pctx-
>c_phone_len, &pctx->c_phone_rc);
    OCIBNDR(pctx->curp1, pctx->c_since_bp1,
errhp,":c_since",&c_since,
        SIZ(OCIDate), SQLT_ODT, &pctx->c_since_ind,
&pctx->c_since_len,
        &pctx->c_since_rc);
    OCIBNDR(pctx->curp1, pctx->c_credit_bp1,
errhp,":c_credit",c_credit,
        SIZ(c_credit),SQLT_CHR, &pctx->c_credit_ind,
&pctx->c_credit_len,
        &pctx->c_credit_rc);
    OCIBNDR(pctx->curp1, pctx->c_credit_lim_bp1,
errhp,":c_credit_lim",
        ADR(c_credit_lim),SIZ(int), SQLT_INT, &pctx-
>c_credit_lim_ind,
        &pctx->c_credit_lim_len, &pctx-
>c_credit_lim_rc);
    OCIBNDR(pctx->curp1, pctx->c_discount_bp1,
errhp,":c_discount",
        ADR(c_discount),SIZ(c_discount), SQLT_FLT,
&pctx->c_discount_ind,
        &pctx->c_discount_len, &pctx->c_discount_rc);
    OCIBNDR(pctx->curp1, pctx->c_balance_bp1,
errhp,":c_balance",ADR(c_balance),
        SIZ(double),SQLT_FLT, &pctx->c_balance_ind,
&pctx->c_balance_len,

```

```

        &pctx->c_balance_rc);
    OCIBNDR(pctx->curp1, pctx->c_data_bp1,
errhp,":c_data",c_data,SIZ(c_data),
        SQLT_STR, &pctx->c_data_ind, &pctx-
>c_data_len, &pctx->c_data_rc);
/*
    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);
*/
    OCIBNDR(pctx->curp1, pctx->retries_bp1,
errhp,":retry",ADR(retries),SIZ(int),
        SQLT_INT, &pctx->retries_ind, &pctx-
>retries_len, &pctx->retries_rc);
    OCIBNDR(pctx->curp1, pctx->cr_date_bp1,
errhp,":cr_date",ADR(cr_date),
        SIZ(OCIDate),SQLT_ODT, &pctx->cr_date_ind,
&pctx->cr_date_len,
        &pctx->cr_date_rc);
#endif

    return (0);
}

tkvcp ()
{
    retry:

    pctx->w_id_ind = TRUE;
    pctx->w_id_len = SIZ(w_id);
    pctx->d_id_ind = TRUE;
    pctx->d_id_len = SIZ(d_id);
    pctx->c_w_id_ind = TRUE;
    pctx->c_w_id_len = 0;
    pctx->c_d_id_ind = TRUE;
    pctx->c_d_id_len = 0;
    pctx->c_id_ind = TRUE;
    pctx->c_id_len = 0;
    pctx->h_amount_len = SIZ(h_amount);
    pctx->h_amount_ind = TRUE;
    pctx->c_last_ind = TRUE;
    pctx->c_last_len = SIZ(c_last);
    pctx->w_street_1_ind = TRUE;
    pctx->w_street_1_len = 0;
    pctx->w_street_2_ind = TRUE;
    pctx->w_street_2_len = 0;
    pctx->w_city_ind = TRUE;
    pctx->w_city_len = 0;
    pctx->w_state_ind = TRUE;
    pctx->w_state_len = 0;
    pctx->w_zip_ind = TRUE;
    pctx->w_zip_len = 0;
    pctx->d_street_1_ind = TRUE;
    pctx->d_street_1_len = 0;
    pctx->d_street_2_ind = TRUE;
    pctx->d_street_2_len = 0;
    pctx->d_city_ind = TRUE;
    pctx->d_city_len = 0;
    pctx->d_state_ind = TRUE;
    pctx->d_state_len = 0;
    pctx->d_zip_ind = TRUE;
    pctx->d_zip_len = 0;
    pctx->c_first_ind = TRUE;

```

```

    pctx->c_first_len = 0;
    pctx->c_middle_ind = TRUE;
    pctx->c_middle_len = 0;
    pctx->c_street_1_ind = TRUE;
    pctx->c_street_1_len = 0;
    pctx->c_street_2_ind = TRUE;
    pctx->c_street_2_len = 0;
    pctx->c_city_ind = TRUE;
    pctx->c_city_len = 0;
    pctx->c_state_ind = TRUE;
    pctx->c_state_len = 0;
    pctx->c_zip_ind = TRUE;
    pctx->c_zip_len = 0;
    pctx->c_phone_ind = TRUE;
    pctx->c_phone_len = 0;
    pctx->c_credit_ind = TRUE;
    pctx->c_credit_len = 0;
    pctx->c_credit_lim_ind = TRUE;
    pctx->c_credit_lim_len = 0;
    pctx->c_discount_ind = TRUE;
    pctx->c_discount_len = 0;
    pctx->c_balance_ind = sizeof(double);
    pctx->c_data_ind = TRUE;
    pctx->c_data_len = 0;
    pctx->h_date_ind = TRUE;
    pctx->h_date_len = 0;
    pctx->retries_ind = TRUE;
    pctx->retries_len = 0;
    pctx->cr_date_ind = TRUE;
    pctx->cr_date_len = 7;

#ifdef PLSQPLPAY
    execstatus=OCISmtExecute(tpcsvc,pctx-
>curp0,errhp,1,0,0,0,OCI_DEFAULT|OCI_COMMIT_O
N_SUCCESS);
#else
    if(bylastname) {
        execstatus=OCISmtExecute(tpcsvc,pctx-
>curp1,errhp,1,0,0,0,OCI_DEFAULT|OCI_COMMIT_O
N_SUCCESS);
    } else {
        execstatus=OCISmtExecute(tpcsvc,pctx-
>curp0,errhp,1,0,0,0,OCI_DEFAULT|OCI_COMMIT_O
N_SUCCESS);
    }
#endif

    if(execstatus != OCI_SUCCESS) {
        OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
        errcode = OCIERROR(errhp,execstatus);
        if(errcode == NOT_SERIALIZABLE) {
            retries++;
            goto retry;
        } else if (errcode == RECOVER) {
            retries++;
            goto retry;
        } else if (errcode == SNAPSHOT_TOO_OLD) {
            retries++;
            goto retry;
        } else {
            return -1;
        }
    }

```

```

}
}

return 0;
}

void tkvcpdone ()

{
  if(pctx) {
    free(pctx);
  }
}

===== plsto.c =====

#ifdef RCSID
static char *RCSid =
"$Header: plsto.c 7010000.3 95/02/14 12:48:03 plai
Generic-base> $ Copyr (c) 1994 Oracle";
#endif /* RCSID */

/*=====
=====+
| Copyright (c) 1994 Oracle Corp, Redwood
Shores, CA |
| OPEN SYSTEMS PERFORMANCE
GROUP |
| All Rights Reserved |
+=====
=====+
| FILENAME
| plsto.c
| DESCRIPTION
| OCI version of STOCK LEVEL transaction in TPC-C
benchmark.
+=====
=====*/

#include "tpcc.h"
#include "tpcc_info.h"
#include "tpccflags.h"

#ifdef PLSQLSTO
#define SQLTXT "BEGIN stocklevel.getstocklevel
(:w_id, :d_id, :threshold, \
:low_stock); END;"
#else
#define SQLTXT "SELECT count (DISTINCT s_i_id) \
FROM ordl, stok, dist \
WHERE d_id = :d_id AND d_w_id = :w_id AND \
d_id = ol_d_id AND d_w_id = ol_w_id AND \
ol_o_id = s_i_id AND ol_w_id = s_w_id AND \
s_quantity < :threshold AND \
ol_o_id BETWEEN (d_next_o_id - 20) AND \
(d_next_o_id - 1)"
/* query using functional index */
/*
#define SQLTXT "SELECT count (DISTINCT s_i_id) \
FROM ordl, stok, dist \
WHERE d_id = :d_id AND d_w_id = :w_id AND \
d_id = ol_d_id AND d_w_id = ol_w_id AND \
ol_o_id BETWEEN (d_next_o_id - 20) AND \
(d_next_o_id - 1) AND \
decode(SIGN(s_quantity - 21) , -1, \
s_w_id*100000 + s_i_id, NULL) \
= ol_w_id*100000 + \
ol_i_id AND \
s_quantity < :threshold;"
*/

#endif

struct stoctx {
  OCISmti *curs;
  OCIBind *w_id_bp;
  OCIBind *d_id_bp;
  OCIBind *threshold_bp;
#ifdef PLSQLSTO
  OCIBind *low_stock_bp;
#else
  OCIDefine *low_stock_bp;
#endif
  int norow;
};

typedef struct stoctx stoctx;

stoctx *sctx;

tkvcsinit ()
{
  text stmbuf[SQL_BUF_SIZE];
  sctx = (stoctx *)malloc(sizeof(stoctx));
  memset(sctx, (char)0, sizeof(stoctx));

  sctx->norow=0;

  OCIERROR(errhp,
    OCIHandleAlloc(tpcenv, (dvoid**)&sctx-
>curs, OCI_HTYPE_STMT, 0, (dvoid**)0));
  sprintf ((char *) stmbuf, SQLTXT);
  OCIERROR(errhp, OCISmtPrepare(sctx-
>curs, errhp, stmbuf, strlen((char *)stmbuf),
    OCI_NTV_SYNTAX, OCI_DEFAULT));
#ifdef PLSQLSTO
  OCIERROR(errhp,
    OCIAttrSet(sctx-
>curs, OCI_HTYPE_STMT, (dvoid*)&sctx->norow, 0,
    OCI_ATTR_PREFETCH_ROWS, errhp));
#endif

  /* bind variables */
  OCIBND(sctx->curs, sctx->w_id_bp, errhp, ":w_id",
ADR(w_id), sizeof(int),
  SQLT_INT);
  OCIBND(sctx->curs, sctx->d_id_bp, errhp, ":d_id",
ADR(d_id), sizeof(int),
  SQLT_INT);
  OCIBND(sctx->curs, sctx->threshold_bp, errhp,
":threshold", ADR(threshold),
  sizeof(int), SQLT_INT);
#ifdef PLSQLSTO
  OCIBND(sctx->curs, sctx-
>low_stock_bp, errhp, ":low_stock", ADR(low_stock),
  sizeof(int), SQLT_INT);
#else
  OCIDefine(sctx->curs, sctx->low_stock_bp,
ADR(low_stock),
  sizeof(int), SQLT_INT);
#endif
}

tkvcs ()
{
  retry:

  execstatus= OCISmtExecute(tpcsvc, sctx-
>curs, errhp, 1, 0, 0, 0,
    OCI_COMMIT_ON_SUCCESS |
    OCI_DEFAULT);

  if (execstatus != OCI_SUCCESS)
  {
    errcode=OCIERROR(errhp, execstatus);
    OCITransCommit(tpcsvc, errhp, OCI_DEFAULT);
    if((errcode == NOT_SERIALIZABLE) || (errcode ==
RECOVERR)
      || (errcode == SNAPSHOT_TOO_OLD))
    {
      retries++;
      goto retry;
    } else {
      return -1;
    }
  }

  return (0);
}

void tkvcsdone ()
{
  if(sctx) free(sctx);
}

===== sleep.c =====

/*=====
=====+
* Copyright (c) 1997 Oracle Corp, Redwood
Shores, CA |
* All Rights Reserved |
*=====
=====+
* FILENAME
* sleep.c
* DESCRIPTION
* Sleeps for argv[1] seconds
/*=====
=====*/
#ifdef ORANT

```

```

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

void main(int argc, char **argv)
{
    int s;

    if(--argc == 1)
    {
        s = atoi(argv[1]) * 1000;
#ifdef ORA_NT
        Sleep(s);
#else
        sleep(s/1000);
#endif
    }
    else printf("Usage: sleep seconds\n");
}

===== tpccpl.c =====

#ifdef RCSID
static char *RCSID =
    "$Header: tpccpl.c 7030100.2 96/04/02 17:51:34 plai
Generic<base> $ Copyr (c) 1994 Oracle";
#endif /* RCSID */

/*=====
+-----+
| Copyright (c) 1994 Oracle Corp, Redwood
Shores, CA |
| OPEN SYSTEMS PERFORMANCE
GROUP |
| All Rights Reserved |
+-----+
+-----+
| FILENAME
| tpccpl.c
| DESCRIPTION
| TPC-C transactions in PL/SQL.
+-----+
=====*/

#include <stdio.h>
#include <time.h>
#include "tpcc.h"
#include "tpcc_info.h"
#ifdef TUX
#include <userlog.h>
#else
#include <stdarg.h>
#endif
#include "SYSTEMEB.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"

FILE *lfp;
FILE *fopen ();
int iflg;
#ifdef ORA_NT
#undef boolean

```

```

int ocierror(fname, lineno, errhp, status)
char *fname;
int lineno;
OCIError *errhp;
sword status;
{
    text errbuf[512];
    sb4 errcode;
    sb4 lstat;
    ub4 recno=2;
//FILE *abcfp;
//char filename[40];

    switch (status)
    case OCI_SUCCESS:
        break;
    case OCI_SUCCESS_WITH_INFO:
        printf("Module %s Line %d\n", fname, lineno);
        printf("Error - OCI_SUCCESS_WITH_INFO\n");
        lstat = OCIErrorGet (errhp, recno++, (text *) NULL,
&errcode, errbuf,
            (ub4) sizeof(errbuf),
OCI_HTYPE_ERROR);
        printf("Error - %s\n", errbuf);
        break;
    case OCI_NEED_DATA:
        printf("Module %s Line %d\n", fname, lineno);
        printf("Error - OCI_NEED_DATA\n");
        return (IRRECERR);
    case OCI_NO_DATA:
//printf("OCI_NO_DATA1");
        printf("OCI_NO_DATA2");
        printf("Error - OCI_NO_DATA\n");
//printf("OCI_NO_DATA3");
        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)
        {
            printf("Module %s Line %d\n", fname, lineno);
            printf("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:
        printf("Module %s Line %d\n", fname, lineno);
        printf("Error - OCI_INVALID_HANDLE\n");
        TPCexit(1);
        exit(-1);
    case OCI_STILL_EXECUTING:
        printf("Module %s Line %d\n", fname, lineno);
        printf("Error - OCI_STILL_EXECUTE\n");
        return (IRRECERR);
    case OCI_CONTINUE:
        printf("Module %s Line %d\n", fname, lineno);
        printf("Error - OCI_CONTINUE\n");
        return (IRRECERR);

default:
    printf("Module %s Line %d\n", fname, lineno);
    printf("Status - %s\n", status);
    return (IRRECERR);
}
return (RECOVERR);
}

FILE *vopen(fnam,mode)
char *fnam;
char *mode;
{
    FILE *fd;

#ifdef DEBUG
    printf("tkvuopen() fnam: %s, mode: %s\n", fnam,
mode);
#endif

    fd = fopen((char *)fnam,(char *)mode);
    if (!fd){
        printf(" fopen on %s failed %d\n",fnam,fd);
        exit(-1);
    }
    return(fd);
}

int sqlfile(fnam,linebuf)
char *fnam;
text *linebuf;
{
    FILE *fd;
    int nulpt = 0;
    char realfile[512];

#ifdef DEBUG
    printf("sqlfile() fnam: %s, linebuf: %x\n", fnam,
linebuf);
#endif

/*
    sprintf(realfile,"%s/bench/tpc/tpcc/blocks/%s",oracle_
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 *oradt)
{
    struct tm *loctime;
    time_t int_time;

    struct ORADATE {
    unsigned char century;
    unsigned char year;
    unsigned char month;
    unsigned char day;
    unsigned char hour;
    unsigned char minute;
    unsigned char second;
    } Date;

    int century,month,year;

    memcpy(&Date,oradt,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 cvtdmyhms (unsigned char *oradt, 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,oradt,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)
{
    if (new_init) {
        tkvcndone();
        new_init = 0;
    }
    if (pay_init) {
        tkvcpdone();
        pay_init = 0;
    }
    if (ord_init) {
        tkvcddone();
        ord_init = 0;
    }
    if (del_init) {
        tkvcddone();
        del_init = 0;
    }
    if (sto_init) {
        tkvcddone();
        sto_init = 0;
    }

    OCIHandleFree((dvoid *)tpcusr,
OCI_HTYPE_SESSION);
    OCIHandleFree((dvoid *)tpcscv,
OCI_HTYPE_SVCCTX);
    OCIHandleFree((dvoid *)errhp,
OCI_HTYPE_ERROR);
    OCIHandleFree((dvoid *)tpcscv,
OCI_HTYPE_SERVER);
    OCIHandleFree((dvoid *)tpcenv, OCI_HTYPE_ENV);

    if (lfp) {
        fclose (lfp);
        lfp = NULL;
    }
}

TPCinit (id, uid, pwd)

int id;
char *uid;
char *pwd;

{
    // char filename[40];
    text stmbuf[100];
    //kawabata add 2000.5.9
    sword rval;
    //kawabata delete
    /*
    proc_no = id;
    sprintf (filename, "tpcc_%d.del", proc_no);
    if ((lfp = fopen (filename, "w")) == NULL) {
    #ifdef TUX
        printf ("Error in TPC-C server %d: Failed to
open %s\n",
            proc_no, filename);
    #else
        printf ("Error in TPC-C server %d: Failed to
open %s\n",
            proc_no, filename);
    #endif
    return (-1);
    }
    */
    // kawabata add
    //delivery flg init
    iflg =0;

    OCIInitialize(OCI_DEFAULT|OCI_OBJECT,(dvoid
*)0,0,0,0);
    OCIEnvInit(&tpcenv, OCI_DEFAULT, 0, (dvoid **)0);
    OCIHandleAlloc((dvoid *)tpcenv, (dvoid **)&tpcusr,
OCI_HTYPE_SERVER, 0, (dvoid **)0);
    OCIHandleAlloc((dvoid *)tpcenv, (dvoid **)&errhp,
OCI_HTYPE_ERROR, 0, (dvoid **)0);
    OCIHandleAlloc((dvoid *)tpcenv, (dvoid **)&tpcscv,
OCI_HTYPE_SVCCTX, 0, (dvoid **)0);

    // OCIServerAttach(tpcscv, errhp, (text
*)0,0,OCI_DEFAULT);
    // kawabata add 2000.5.9

    while (1) {
        rval = OCIServerAttach(tpcscv, errhp, (text
*)0,0,OCI_DEFAULT);
        if (rval == OCI_SUCCESS || rval ==
OCI_SUCCESS_WITH_INFO)
            break;
        OCIERROR(errhp, rval);
        Sleep(1);
    }

    OCIAttrSet((dvoid *)tpcscv, OCI_HTYPE_SVCCTX,
(dvoid *)tpcscv, (ub4)0,OCI_ATTR_SERVER, errhp);
    OCIHandleAlloc((dvoid *)tpcenv, (dvoid **)&tpcusr,
OCI_HTYPE_SESSION, 0, (dvoid **)0);
    OCIAttrSet((dvoid *)tpcusr, OCI_HTYPE_SESSION,
(dvoid *)uid, (ub4)strlen(uid),OCI_ATTR_USERNAME,
errhp);
    OCIAttrSet((dvoid *)tpcusr, OCI_HTYPE_SESSION,
(dvoid *)pwd, (ub4)strlen(pwd),
OCI_ATTR_PASSWORD, errhp);
    OCIERROR(errhp, OCISessionBegin(tpcscv, errhp,
tpcusr, OCI_CRED_RDBMS, OCI_DEFAULT));

    OCIAttrSet(tpcscv, 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, SQLTXMT);
    OCISmtPrepare(curi, errhp, stmbuf, strlen((char
*)stmbuf), OCI_NTV_SYNTAX, OCI_DEFAULT);
    OCIERROR(errhp,OCISmtExecute(tpcscv, curi,
errhp,1,0,0,OCI_DEFAULT));
    OCIHandleFree(curi, OCI_HTYPE_STMT);

    /*
    This is done in cvdrv.c
    if (tracelevel == 2) {
        OCIHandleAlloc(tpcenv, (dvoid **)&curi,
OCI_HTYPE_STMT, 0, (dvoid **)0);
        memset(stmbuf,0,100);
        sprintf ((char *) stmbuf, SQLTXTRC);
        OCISmtPrepare(curi, errhp, stmbuf, strlen((char
*)stmbuf),
            OCI_NTV_SYNTAX,
OCI_DEFAULT);
        OCIERROR(errhp, OCISmtExecute(tpcscv, curi,
errhp,1,0,0,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, SQLXTTIM);
        OCISmtPrepare(curi, errhp, stmbuf, strlen((char
*)stmbuf),
            OCI_NTV_SYNTAX,
OCI_DEFAULT);
        OCIERROR(errhp, OCISmtExecute(tpcscv, curi,
errhp,1,0,0,OCI_DEFAULT));
        OCIHandleFree((dvoid *)curi, OCI_HTYPE_STMT);
    }

    logon = 1;

    OCIERROR(errhp,OCIDateSysDate(errhp,&cr_date))
;

    if (tkvcninit 0) { /* new order */
        TPCexit 0;
        return (-1);
    }
}

```

```

}
else
    new_init = 1;

if (tkvcpininit ()) { /* payment */
    TPCexit ();
    return (-1);
}
else
    pay_init = 1;

if (tkvcoint ()) { /* order status */
    TPCexit ();
    return (-1);
}
else
    ord_init = 1;

if (tkvcdinit ()) { /* delivery */
    TPCexit ();
    return (-1);
}
else
    del_init = 1;

if (tkvcsinit ()) { /* stock level */
    TPCexit ();
    return (-1);
}
else
    sto_init = 1;

return (0);
}

TPCnew (str)
struct newstruct *str;
{
    int i;

    w_id = str->newin.w_id;
    d_id = str->newin.d_id;
    c_id = str->newin.c_id;

//kawabata add
    for (i = 0; i < 15; i++) {
        nol_i_id[i] = 0;
        nol_supply_w_id[i] = 0;
        nol_quantity[i] = 0;
    }

    for (i = 0; i < 15; i++) {
//kawabata add
        if((str->newin.ol_i_id[i] == 0) && (str-
>newin.ol_supply_w_id[i] == 0) && (str-
>newin.ol_quantity[i] == 0))
                break;
        nol_i_id[i] = str->newin.ol_i_id[i];
        nol_supply_w_id[i] = str->newin.ol_supply_w_id[i];
        nol_quantity[i] = str->newin.ol_quantity[i];

//printf("i_id=%d",nol_i_id[i]);
    }
    retries = 0;
/*
    vgetdate(cr_date); */
//printf("TPCnew start1");

    OCIERROR(errhp,OCIDateSysDate(errhp,&cr_date))
;
    if (str->newout.terror = tkvcn ()) {
        if (str->newout.terror != RECOVERR)
            str->newout.terror = IRRRECERR;
        return (-1);
    }
//printf("TPCnew end");

/* 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,
            &datelen,o_entry_d));

    str->newout.terror = NOERR;
    str->newout.o_id = o_id;
    str->newout.o_ol_cnt = o_ol_cnt;
    strncpy (str->newout.c_last, c_last, 17);
    strncpy (str->newout.c_credit, c_credit, 3);
    str->newout.c_discount = c_discount;
    str->newout.w_tax = (float)(w_tax);
    str->newout.d_tax = (float)(d_tax);
    strncpy (str->newout.o_entry_d, (char*)o_entry_d,
20);
    str->newout.total_amount = total_amount;
    for (i = 0; i < o_ol_cnt; i++) {
        strncpy (str->newout.i_name[i], i_name[i], 25);
        str->newout.s_quantity[i] = s_quantity[i];
        str->newout.brand_generic[i] = brand_generic[i][0];
        str->newout.i_price[i] = (float)(i_price[i])/100;
        str->newout.ol_amount[i] =
(float)(nol_amount[i])/100;
    }
    if (status)
        strcpy (str->newout.status, "Item number is not
valid");
    else
        str->newout.status[0] = '\0';
    str->newout.retry = retries;
# if defined(TOP) || defined(TUX) /* changed mjb 17
feb for tuxedo */
        return(1);
# else
        return (0);
# endif
    }
}

TPCpay (str)
struct paystruct *str;
{
    w_id = str->payin.w_id;
    d_id = str->payin.d_id;
    c_w_id = str->payin.c_w_id;
    c_d_id = str->payin.c_d_id;
    h_amount = str->payin.h_amount;
    bylastname = str->payin.bylastname;

/*
    vgetdate(cr_date); */
    OCIERROR(errhp,OCIDateSysDate(errhp,&cr_date))
;
    if (bylastname) {
        c_id = 0;
        strncpy (c_last, str->payin.c_last, 17);
    }
    else {
        c_id = str->payin.c_id;
        strcpy (c_last, "");
    }
    retries = 0;

//printf("TPCpay1\n");
//printf("clast=%s\n",c_last);

    if (str->payout.terror = tkvcp ()) {
//printf("TPCpay1.5\n");
        if (str->payout.terror != RECOVERR)
            str->payout.terror = IRRRECERR;
        return (-1);
    }
//printf("TPCpay2\n");

/*
    cvtdmyhms(cr_date,h_date);
*/
    hlen=SIZ(h_date);
    OCIERROR(errhp,OCIDateToText(errhp,&cr_date,
        (text*)FULLDATE,strlen(FULLDATE),(text*)0,
        0,&hlen,h_date));

/*
    cvtdmy(c_since,c_since_d);
*/
    sincelen=SIZ(c_since_d);
    OCIERROR(errhp,OCIDateToText(errhp,&c_since,
        (text*)SHORTDATE,strlen(SHORTDATE),(text*)
        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);
str->payout.c_credit_lim = (float)(c_credit_lim)/100;
str->payout.c_discount = c_discount;
str->payout.c_balance = (float)(c_balance)/100;
strncpy (str->payout.c_data, c_data, 201);
strncpy (str->payout.h_date, (char*)h_date, 20);
str->payout.retry = retries;
#if defined(TOP) || defined(TUX) /* changed mjb 17 Feb */
return(1);
#else
return (0);
#endif
}

TPCord (str)

struct ordstruct *str;

{
int i;

w_id = str->ordin.w_id;
d_id = str->ordin.d_id;
bylastname = str->ordin.bylastname;

//printf("TPCord start1 ");

if (bylastname) {
c_id = 0;
strncpy (c_last, str->ordin.c_last, 17);
}
else {
c_id = str->ordin.c_id;
strncpy (c_last, " ");
}
retries = 0;

if (str->ordout.error == tkvco ()) {
//printf("tkvco error ");
//printf("tkvco =%d", str->ordout.error);
if (str->ordout.error != RECOVERR)
str->ordout.error = IRRECERR;
return (-1);
}
//printf("tkvco success ");
//printf("tkvco =%d", str->ordout.error);

//printf("TPCord start2 ");
datelen = sizeof(o_entry_d);
OCIERROR(errhp,
OCIDateToText(errhp, &o_entry_d_base, (text*)FU
LLDATE, SIZ(FULLDATE), (text*)0, 0,
&datelen, o_entry_d));

str->ordout.error = NOERR;
str->ordout.c_id = c_id;
strncpy (str->ordout.c_last, c_last, 17);
strncpy (str->ordout.c_first, c_first, 17);
strncpy (str->ordout.c_middle, c_middle, 3);
str->ordout.c_balance = c_balance/100;
str->ordout.o_id = o_id;
strncpy (str->ordout.o_entry_d, (char*)o_entry_d, 20);
if (o_carrier_id == 11)
str->ordout.o_carrier_id = 0;
else
str->ordout.o_carrier_id = o_carrier_id;
str->ordout.o_ol_cnt = o_ol_cnt;
for (i = 0; i < o_ol_cnt; i++) {
ol_delivery_d[i][10] = '\0';
if (!strcmp((char*)ol_delivery_d[i], "15-09-1911"))
strncpy((char*)ol_delivery_d[i], "NOT
DELIVR", 10);
str->ordout.ol_supply_w_id[i] = ol_supply_w_id[i];
str->ordout.ol_i_id[i] = ol_i_id[i];
str->ordout.ol_quantity[i] = ol_quantity[i];
str->ordout.ol_amount[i] = (float)(ol_amount[i])/100;
strncpy (str->ordout.ol_delivery_d[i],
(char*)ol_delivery_d[i], 11);
}
str->ordout.retry = retries;
//printf("TPCord start3 ");

#if defined(TOP) || defined(TUX)
return(1);
#else
return (0);
#endif
}

TPCdel (str)

struct delstruct *str;

{
// double tr_end;
int i;
SYSTEMTIME systemTime;
struct tm times;
//kawabata add 2000.6.27
char filename[40];
int svrcnt;

//kawabata add
if(!iflg == 0)
{
srand ( (unsigned)time( NULL ) );
svrcnt = rand();
sprintf (filename, "tpcc_%d.del", svrcnt);
if (lfp = fopen (filename, "w") == NULL) {
#ifdef TUX
printf ("Error in TPC-C server %d:
Failed to open %s\n",
proc_no, filename);
#else
printf ("Error in TPC-C server %d:
Failed to open %s\n",
proc_no, filename);
#endif
return (-1);
}
iflg = 1;
}
//kawabata 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_date)
);

if (str->delout.error == tkvco ()) {
if(str->delout.error == DEL_ERROR)
return DEL_ERROR;
if (str->delout.error != RECOVERR)
str->delout.error = IRRECERR;
return (-1);
}

GetLocalTime(&systemTime);

// #ifdef TOOLKIT_ORIGINAL_STRUCTURE
/* 1996.08.07 */
// bp->delin.qtime = ( double )timeque.tv_sec
// + ( double )timeque.tv_usec / 1000000.0;
//
// #else !TOOLKIT_ORIGINAL_STRUCTURE

// GetLocalTime の結果を、mktime で数値デー
タに変換する
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;

// bp->delout.startsec = (long)mktime (&times);
// bp->delout.startusec =
(long)systemTime.wMilliseconds;

// tr_end = gettime ();
/* fprintf (lfp, "%d %d %f %f %d %d", str-
>delin.in_timing_int,
(tr_end - str->delin.qtime) <= DELRT ? 1 : 0,
str->delin.qtime, tr_end, w_id, o_carrier_id); */
/* fprintf (lfp, "%f %f %d %d", str->delin.qtime, tr_end,
w_id,
o_carrier_id);
*/

fprintf(lfp, "%09d%03d %09d%03d %d %d", str-
>delin.startsec,
str->delin.startusec, ((long)mktime
(&times)), (long)systemTime.wMilliseconds, w_id, o_carri
er_id);

for (i = 0; i < 10; i++) {

```

```

fprintf (lfp, "%d %d", i + 1, del_o_id[i]);
if (del_o_id[i] <= 0) {
#ifdef TUX
    printf ("DELIVERY: no new order for w_id: %d,
d_id %d\n",
        w_id, i + 1);
#else
    printf ("DELIVERY: no new order for w_id: %d,
d_id %d\n",
        w_id, i + 1);
#endif
}
}
fprintf (lfp, "%d\n", retries);
str->delout.terror = NOERR;
str->delout.retry = retries;
#if defined(TOP) || defined(TUX) /* changed mjb 17
feb */
return(1);
#else
return (0);
#endif
}

TPCsto (str)

struct stostruct *str;

{

w_id = str->stoin.w_id;
d_id = str->stoin.d_id;
threshold = str->stoin.threshold;
retries = 0;

if (str->stoout.terror = tkvcs ()) {
    if (str->stoout.terror != RECOVERR)
        str->stoout.terror = IRRECERR;
    return (-1);
}

str->stoout.terror = NOERR;
str->stoout.low_stock = low_stock;
str->stoout.retry = retries;
#if defined(TOP) || defined(TUX) /* changed mjb 17 feb
*/
return(1);
#else
return (0);
#endif
}

===== tpccload.c =====
#ifdef RCSID
static char *RCSID =
    "$Header: tpccload.c 7030100.1 96/05/13 16:20:36
plai Generic-base> $ Copyr (c) 1993 Oracle";
#endif /* RCSID */

/*=====
=====

```

```

| Copyright (c) 1994 Oracle Corp, Redwood
Shores, CA |
| OPEN SYSTEMS PERFORMANCE
GROUP |
| All Rights Reserved |
+-----+
| FILENAME
| tpccload.c
| DESCRIPTION
| Load or generate TPC-C database tables.
| Usage: tpccload -M <# of wares> [options]
| options: -A load all tables
| -w load ware table
| -d load dist table
| -c load cust table
| -i load item table
| -s load stok table (cluster around
s_w_id)
| -S load stok table (cluster around
s_i_id)
| -h load hist table
| -n load new-order table
| -o <oline file> load order and order-line
table
| -b <ware#> beginning ware number
| -e <ware#> ending ware number
| -j <item#> beginning item number (with
-S)
| -k <item#> ending item number (with -
S)
| -g generate rows to standard output
+-----+
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
/* #include <unistd.h> */
#include <time.h>
#include <systypes.h>
#include "tpcc.h"

#ifdef ORA_NT
#undef boolean
#include <process.h>
#include "dpcbcore.h"
# define gettime dpbtimef
# define getcpu dpbcpu
#define lrand48() ((long)rand() <<15 | rand())
#endif __STDC__
# define PROTO(args) args
#else
# define PROTO(args) ()
#endif
#endif

#define DISTARR 10 /* dist insert array size
*/
#define CUSTARR 100 /* cust insert array size
*/
#define STOCARR 100 /* stok insert array size
*/
#define ITEMARR 100 /* item insert array size
*/
#define HISTARR 100 /* hist insert array size
*/
#define ORDEARR 100 /* order insert array size
*/

```

```

#define NEWOARR 100 /* new order insert
array size */
#define DISTFAC 10 /* max. district id */
#define CUSTFAC 3000 /* max. cust id
*/
#define STOCFAC 100000 /* max. stok id
*/
#define ITEMFAC 100000 /* max. item id
*/
#define HISTFAC 30000 /* hist / ware */
#define ORDEFAC 3000 /* order / dist */
#define NEWOFAC 900 /* new order / dist */

#define C 0 /* constant in non-uniform dist.
eqt. */
#define CNUM1 1 /* first constant in non-
uniform dist. eqt. */
#define CNUM2 2 /* second constant in non-
uniform dist. eqt. */
#define CNUM3 3 /* third constant in non-
uniform dist. eqt. */

#define SEED 2 /* seed for random functions
*/

#define SQLTXTW "INSERT INTO ware (w_id, w_ytd,
w_tax, w_name, w_street_1, w_street_2, w_city,
w_state, w_zip) VALUES (w_id,
30000000, :w_tax, :w_name, :w_street_1, \
:w_street_2, :w_city, :w_state, :w_zip)"

#define SQLXTD "INSERT INTO dist (d_id, d_w_id,
d_ytd, d_tax, d_next_o_id, d_name, d_street_1,
d_street_2, d_city, d_state, d_zip) VALUES
(:d_id, :d_w_id, 30000000, :d_tax, \
3001, :d_name, :d_street_1, :d_street_2, :d_city, :d_
state, :d_zip)"

#define SQLXTC "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 SQLXTH "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 SQLXTS "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 SQLTXTI "INSERT INTO item
(L_ID,I_IM_ID,I_NAME,I_PRICE,I_DATA) VALUES
(:i_id, :i_im_id, :i_name, :i_price, \
 :i_data)"

#define SQLTXTO1 "INSERT INTO ordr (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, :o_d_id, :o_w_id, :o_c_id, \
SYSDATE, :o_carrier_id, :o_ol_cnt, 1)"

#define SQLTXTO2 "INSERT INTO ordr (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, :o_d_id, :o_w_id, :o_c_id, \
SYSDATE, 11, :o_ol_cnt, 1)"

#define SQLTXTOL1 "INSERT INTO ordl (OL_O_ID,
OL_D_ID, OL_W_ID, OL_NUMBER, OL_DELIVERY_D,
OL_I_ID, OL_SUPPLY_W_ID, OL_QUANTITY,
OL_AMOUNT, OL_DIST_INFO) \
VALUES (:ol_o_id, :ol_d_id, \
 :ol_w_id, :ol_number,
SYSDATE, :ol_i_id, :ol_supply_w_id, 5, 0, \
 :ol_dist_info)"

#define SQLTXTOL2 "INSERT INTO ordl (OL_O_ID,
OL_D_ID, OL_W_ID, OL_NUMBER, OL_DELIVERY_D,
OL_I_ID, OL_SUPPLY_W_ID, OL_QUANTITY,
OL_AMOUNT, OL_DIST_INFO) \
VALUES (:ol_o_id, :ol_d_id, \
 :ol_w_id, :ol_number, to_date('01-Jan-
1811'), :ol_i_id, :ol_supply_w_id, 5, :ol_amount, \
 :ol_dist_info)"

#define SQLTXTNO "INSERT INTO nord (no_o_id,
no_d_id, no_w_id) VALUES
(no_o_id, :no_d_id, :no_w_id)"

ldadef tpclda:
csrdef curw, curc, curc, curh, curs, curi, curo1, curo2,
curo1, curo2, curno;
unsigned long tpchda[256];

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();

void myusage()
{
printf (stderr, "\n");
printf (stderr, "Usage: ttpccload -M <multiplier>
[options]\n");
printf (stderr, "options:\n");
printf (stderr, "\t-A :tload all tables\n");
printf (stderr, "\t-w :tload ware table\n");
printf (stderr, "\t-d :tload dist table\n");
printf (stderr, "\t-c :tload cust table\n");
printf (stderr, "\t-i :tload item table\n");
printf (stderr, "\t-s :tload stok table (cluster around
s_w_id)\n");
printf (stderr, "\t-S :tload stok table (cluster around
s_i_id)\n");
printf (stderr, "\t-h :tload hist table\n");
printf (stderr, "\t-n :tload new-order table\n");
printf (stderr, "\t-o <oline file> :tload order and order-
line table\n");
printf (stderr, "\t-b <ware#> :tbeginning ware
number\n");
printf (stderr, "\t-e <ware#> :tending ware
number\n");
printf (stderr, "\t-j <item#> :tbeginning item number
(with -S)\n");
printf (stderr, "\t-k <item#> :tending item number
(with -S)\n");
printf (stderr, "\t-g :tgenerate rows to standard
output\n");
printf (stderr, "\n");
exit(1);
}

void errprt (lda, cur)

csrdef *lda;
csrdef *cur;

{
text msg[2048];

if (cur->rc) {
oerhms (lda, cur->rc, msg, 2048);
printf (stderr, "TPC-C load error: %s\n", msg);
}
}

void quit ()
{
if (oclose (&curw))
errprt (&tpclda, &curw);

if (oclose (&curd))
errprt (&tpclda, &curd);

if (oclose (&curc))
errprt (&tpclda, &curc);

if (oclose (&curh))
errprt (&tpclda, &curh);

if (oclose (&curs))
errprt (&tpclda, &curs);

if (oclose (&curi))
errprt (&tpclda, &curi);

if (oclose (&curo1))
errprt (&tpclda, &curo1);

if (oclose (&curo2))
errprt (&tpclda, &curo2);

if (oclose (&curo1))
errprt (&tpclda, &curo1);

if (oclose (&curo2))
errprt (&tpclda, &curo2);

if (oclose (&curno))
errprt (&tpclda, &curno);

if (ologof (&tpclda))
printf (stderr, "TPC-C load error: Error in logging
off\n");
}

void main (argc, argv)

int argc;
char *argv[];

{
char *uid="tpcc/tpcc";
text sqlbuf[1024];
int scale=0;
int i, j;
int loop;
int loopcount;
int cid;
int dwid;
int cdid;
int cwid;
int sid;
int swid;
int olcnt;
int nrows;
int row;

int w_id;
char w_name[11];
char w_street_1[21];
char w_street_2[21];
char w_city[21];
char w_state[2];
char w_zip[9];
float w_tax;

int d_id[10];
int d_w_id[10];

```

```
char d_name[10][11];
char d_street_1[10][21];
char d_street_2[10][21];
char d_city[10][21];
char d_state[10][2];
char d_zip[10][9];
float d_tax[10];
```

```
int c_id[100];
int c_d_id[100];
int c_w_id[100];
char c_first[100][17];
char c_last[100][17];
char c_street_1[100][21];
char c_street_2[100][21];
char c_city[100][21];
char c_state[100][2];
char c_zip[100][9];
char c_phone[100][16];
char c_credit[100][2];
float c_discount[100];
char c_data[100][501];
```

```
int i_id[100];
int i_im_id[100];
int i_price[100];
char i_name[100][25];
char i_data[100][51];
```

```
int s_i_id[100];
int s_w_id[100];
int s_quantity[100];
char s_dist_01[100][24];
char s_dist_02[100][24];
char s_dist_03[100][24];
char s_dist_04[100][24];
char s_dist_05[100][24];
char s_dist_06[100][24];
char s_dist_07[100][24];
char s_dist_08[100][24];
char s_dist_09[100][24];
char s_dist_10[100][24];
char s_data[100][51];
```

```
int h_w_id[100];
int h_d_id[100];
int h_c_id[100];
char h_data[100][25];
```

```
int o_id[100];
int o_d_id[100];
int o_w_id[100];
int o_c_id[100];
int o_carrier_id[100];
int o_ol_cnt[100];
```

```
int ol_o_id[15];
int ol_d_id[15];
int ol_w_id[15];
int ol_number[15];
int ol_i_id[15];
int ol_supply_w_id[15];
int ol_amount[15];
char ol_dist_info[15][24];
```

```
int no_o_id[100];
int no_d_id[100];
int no_w_id[100];
```

```
char sdate[30];
#ifdef ORA_NT
clock_t begin_time, end_time;
clock_t begin_cpu, end_cpu;

char *arg_ptr, **end_args;
#else
double begin_time, end_time;
double begin_cpu, end_cpu;
double gettime(), getcpu();

extern int getopt();
extern char *optarg;
extern int optind, opterr;

int opt;
#endif

char *argstr="M:AwdcisShno.b:e:j:k:g";
int do_A=0;
int do_w=0;
int do_d=0;
int do_i=0;
int do_c=0;
int do_s=0;
int do_S=0;
int do_h=0;
int do_o=0;
int do_n=0;
int gen=0;
int bware=1;
int eware=0;
int bitem=1;
int eitem=0;

FILE *olfp=NULL;
char olfname[100];
#ifdef ORA_NT
char fname[100];
FILE *logfile;
#endif /* ORA_NT */
```

```
/*-----+
| Parse command line -- look for scale factor.
|
+-----*/
```

```
if (argc == 1) {
    myusage 0;
}
#ifdef ORA_NT
end_args = argv + argc;

for (++argv; argv < end_args;)
{
    arg_ptr = *argv++;

    if (*arg_ptr != '-')
    {
        myusage 0;
    } else
    {
        switch (arg_ptr[1]) {
            case '?': myusage 0;
                    break;
            case 'M': scale = atoi (*argv++);
                    break;
            case 'A': do_A = 1;
                    break;

```

```
case 'w': do_w = 1;
        break;
case 'd': do_d = 1;
        break;
case 'c': do_c = 1;
        break;
case 'i': do_i = 1;
        break;
case 's': do_s = 1;
        break;
case 'S': do_S = 1;
        break;
case 'h': do_h = 1;
        break;
case 'n': do_n = 1;
        break;
case 'o': do_o = 1;
        strcpy (olfname, *argv++);
        break;
case 'b': bware = atoi (*argv++);
        break;
case 'e': eware = atoi (*argv++);
        break;
case 'j': bitem = atoi (*argv++);
        break;
case 'k': eitem = atoi (*argv++);
        break;
case 'g': gen = 1;
        strcpy (fname, *argv++);
        break;
case 'l': logfile=fopen(*argv++,"w");
        break;

        default: fprintf (stderr, "THIS SHOULD NEVER
HAPPEN!!!\n");
                fprintf (stderr, "(reached default case in
getopt (0))\n");
                myusage 0;
        }
    }
}
```

```
/*else
while ((opt = getopt (argc, argv, argstr)) != -1) {
    switch (opt) {
        case '?': myusage 0;
                break;
        case 'M': scale = atoi (optarg);
                break;
        case 'A': do_A = 1;
                break;
        case 'w': do_w = 1;
                break;
        case 'd': do_d = 1;
                break;
        case 'c': do_c = 1;
                break;
        case 'i': do_i = 1;
                break;
        case 's': do_s = 1;
                break;
        case 'S': do_S = 1;
                break;
        case 'h': do_h = 1;
                break;
        case 'n': do_n = 1;
                break;
        case 'o': do_o = 1;
```

```

        strcpy (olfname, optarg);
        break;
    case 'b': bware = atoi (optarg);
        break;
    case 'e': eware = atoi (optarg);
        break;
    case 'j': bitem = atoi (optarg);
        break;
    case 'k': eitem = atoi (optarg);
        break;
    case 'g': gen = 1;
        break;
    default: fprintf (stderr, "THIS SHOULD NEVER
HAPPEN!!!\n");
        fprintf (stderr, "(reached default case in
getopt ()\n");
        myusage ();
    }
}
# endif /* ORA_NT */

/*-----*/
| Rudimentary error checking |
/*-----*/

if (scale < 1) {
    fprintf (stderr, "Invalid scale factor: %d\n", scale);
    myusage ();
}

if (!(do_A || do_w || do_d || do_c || do_i || do_s || do_S
|| do_h || do_o ||
do_n)) {
    fprintf (stderr, "What should I load???\n");
    myusage ();
}

if (gen && (do_A || (do_w + do_d + do_c + do_i +
do_s + do_S + do_h + do_o +
do_n > 1))) {
    fprintf (stderr, "Can only generate table one at a
time\n");
    myusage ();
}

if (do_S && (do_A || do_s)) {
    fprintf (stderr, "Cluster stok table around s_w_id or
s_i_id?\n");
    myusage ();
}

if (eware <= 0)
    eware = scale;
if (eitem <= 0)
    eitem = STOCFAC;

if (do_S) {
    if ((bitem < 1) || (bitem > STOCFAC)) {
        fprintf (stderr, "Invalid beginning item number:
%d\n", bitem);
        myusage ();
    }

    if ((eitem < bitem) || (eitem > STOCFAC)) {
        fprintf (stderr, "Invalid ending item number: %d\n",
eitem);
        myusage ();
    }
}

if ((bware < 1) || (bware > scale)) {
    fprintf (stderr, "Invalid beginning ware number:
%d\n", bware);
    myusage ();
}

if ((eware < bware) || (eware > scale)) {
    fprintf (stderr, "Invalid ending ware number: %d\n",
eware);
    myusage ();
}

if (gen && do_o) {
    if ((olfp = fopen (olfname, "w")) == NULL) {
        fprintf (stderr, "Can't open '%s' for writing order
lines\n", olfname);
        myusage ();
    }
}

/*-----*/
| Prepare to insert into database. |
/*-----*/

sysdate (sdate);
if (!gen) {

    /* log on to Oracle */

    if (orlon (&tpclda, (ub1 *) tpchda, (text *) uid, -1, (text
*) 0, -1, 0)) {
        fprintf (stderr, "TPC-C load error: Error in logging
on\n");
        errrpt (&tpclda, &tpclda);
        exit (1);
    }

    fprintf (stderr, "\nConnected to Oracle userid
'%s'.\n", uid);

    /* turn off auto-commit */

    if (ocof (&tpclda) {
        errrpt (&tpclda, &tpclda);
        ologof (&tpclda);
        exit (1);
    }

    /* open cursors */

    if (oopen (&curw, &tpclda, (text *) 0, -1, -1, (text *)
uid, -1)) {
        errrpt (&tpclda, &curw);
        ologof (&tpclda);
        exit (1);
    }

    if (oopen (&curd, &tpclda, (text *) 0, -1, -1, (text *)
uid, -1)) {
        errrpt (&tpclda, &curd);
        oclose (&curw);
        ologof (&tpclda);
        exit (1);
    }

    if (oopen (&curc, &tpclda, (text *) 0, -1, -1, (text *)
uid, -1)) {
        errrpt (&tpclda, &curc);
        oclose (&curw);
        oclose (&curd);
        ologof (&tpclda);
        exit (1);
    }

    if (oopen (&curh, &tpclda, (text *) 0, -1, -1, (text *)
uid, -1)) {
        errrpt (&tpclda, &curh);
        oclose (&curw);
        oclose (&curd);
        oclose (&curc);
        ologof (&tpclda);
        exit (1);
    }

    if (oopen (&curi, &tpclda, (text *) 0, -1, -1, (text *)
uid, -1)) {
        errrpt (&tpclda, &curi);
        oclose (&curw);
        oclose (&curd);
        oclose (&curc);
        oclose (&curh);
        ologof (&tpclda);
        exit (1);
    }

    if (oopen (&curo1, &tpclda, (text *) 0, -1, -1, (text *)
uid, -1)) {
        errrpt (&tpclda, &curo1);
        oclose (&curw);
        oclose (&curd);
        oclose (&curc);
        oclose (&curh);
        oclose (&curi);
        ologof (&tpclda);
        exit (1);
    }

    if (oopen (&curo2, &tpclda, (text *) 0, -1, -1, (text *)
uid, -1)) {
        errrpt (&tpclda, &curo2);
        oclose (&curw);
        oclose (&curd);
        oclose (&curc);
        oclose (&curh);
        oclose (&curi);
        oclose (&curo1);
        ologof (&tpclda);
        exit (1);
    }

    if (oopen (&curo11, &tpclda, (text *) 0, -1, -1, (text *)
uid, -1)) {
        errrpt (&tpclda, &curo11);
        oclose (&curw);
        oclose (&curd);
        oclose (&curc);
        oclose (&curh);
        oclose (&curi);
        oclose (&curo1);
        ologof (&tpclda);
        exit (1);
    }
}
}

```

```

oclose (&curw);
oclose (&curd);
oclose (&curc);
oclose (&curh);
oclose (&curs);
oclose (&curi);
oclose (&curo1);
oclose (&curo2);
ologof (&tpclda);
exit (1);
}

if (opopen (&curo12, &tpclda, (text *) 0, -1, -1, (text *)
uid, -1)) {
    errrpt (&tpclda, &curo12);
    oclose (&curw);
    oclose (&curd);
    oclose (&curc);
    oclose (&curh);
    oclose (&curs);
    oclose (&curi);
    oclose (&curo1);
    oclose (&curo2);
    oclose (&curo1);
    ologof (&tpclda);
    exit (1);
}

if (opopen (&curno, &tpclda, (text *) 0, -1, -1, (text *)
uid, -1)) {
    errrpt (&tpclda, &curno);
    oclose (&curw);
    oclose (&curd);
    oclose (&curc);
    oclose (&curh);
    oclose (&curs);
    oclose (&curi);
    oclose (&curo1);
    oclose (&curo2);
    oclose (&curo1);
    oclose (&curo2);
    ologof (&tpclda);
    exit (1);
}

/* parse statements */

sprintf ((char *) sqlbuf, SQLTXTW);
if (oparse (&curw, sqlbuf, -1, 0, 1)) {
    errrpt (&tpclda, &curw);
    quit ();
    exit (1);
}

sprintf ((char *) sqlbuf, SQLXTXD);
if (oparse (&curd, sqlbuf, -1, 0, 1)) {
    errrpt (&tpclda, &curd);
    quit ();
    exit (1);
}

sprintf ((char *) sqlbuf, SQLXTXC);
if (oparse (&curc, sqlbuf, -1, 0, 1)) {
    errrpt (&tpclda, &curc);
    quit ();
    exit (1);
}

}

sprintf ((char *) sqlbuf, SQLXTXH);
if (oparse (&curh, sqlbuf, -1, 0, 1)) {
    errrpt (&tpclda, &curh);
    quit ();
    exit (1);
}

sprintf ((char *) sqlbuf, SQLXTXS);
if (oparse (&curs, sqlbuf, -1, 0, 1)) {
    errrpt (&tpclda, &curs);
    quit ();
    exit (1);
}

sprintf ((char *) sqlbuf, SQLXTXI);
if (oparse (&curi, sqlbuf, -1, 0, 1)) {
    errrpt (&tpclda, &curi);
    quit ();
    exit (1);
}

sprintf ((char *) sqlbuf, SQLXTXO1);
if (oparse (&curo1, sqlbuf, -1, 0, 1)) {
    errrpt (&tpclda, &curo1);
    quit ();
    exit (1);
}

}

sprintf ((char *) sqlbuf, SQLXTXO2);
if (oparse (&curo2, sqlbuf, -1, 0, 1)) {
    errrpt (&tpclda, &curo2);
    quit ();
    exit (1);
}

}

sprintf ((char *) sqlbuf, SQLXTXOL1);
if (oparse (&curo1, sqlbuf, -1, 0, 1)) {
    errrpt (&tpclda, &curo1);
    quit ();
    exit (1);
}

}

sprintf ((char *) sqlbuf, SQLXTXOL2);
if (oparse (&curo2, sqlbuf, -1, 0, 1)) {
    errrpt (&tpclda, &curo2);
    quit ();
    exit (1);
}

}

sprintf ((char *) sqlbuf, SQLXTXNO);
if (oparse (&curno, sqlbuf, -1, 0, 1)) {
    errrpt (&tpclda, &curno);
    quit ();
    exit (1);
}

}

/* bind variables */

/* ware */

if (obndrv (&curw, (text *) ":w_id", -1, (ub1 *) &w_id,
sizeof (w_id),
SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curw);
    quit ();
    exit (1);
}

if (obndrv (&curw, (text *) ":w_name", -1, (ub1 *)
w_name, 11,
SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curw);
    quit ();
    exit (1);
}

if (obndrv (&curw, (text *) ":w_street_1", -1, (ub1 *)
w_street_1, 21,
SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curw);
    quit ();
    exit (1);
}

if (obndrv (&curw, (text *) ":w_street_2", -1, (ub1 *)
w_street_2, 21,
SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curw);
    quit ();
    exit (1);
}

if (obndrv (&curw, (text *) ":w_city", -1, (ub1 *) w_city,
21,
SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curw);
    quit ();
    exit (1);
}

if (obndrv (&curw, (text *) ":w_state", -1, (ub1 *)
w_state, 2,
SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curw);
    quit ();
    exit (1);
}

if (obndrv (&curw, (text *) ":w_zip", -1, (ub1 *) w_zip,
9,
SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curw);
    quit ();
    exit (1);
}

}

if (obndrv (&curw, (text *) ":w_tax", -1, (ub1 *)
&w_tax, sizeof (w_tax),
SQLT_FLT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curw);
    quit ();
    exit (1);
}

}

/* dist */

if (obndrv (&curd, (text *) ":d_id", -1, (ub1 *) d_id,
sizeof (int),
SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curd);
    quit ();
    exit (1);
}

if (obndrv (&curd, (text *) ":d_w_id", -1, (ub1 *)
d_w_id, sizeof (int),
SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
}

```



```

errrpt (&tpclda, &curd);
quit ();
exit (1);
}

if (obndrv (&curd, (text *) ":d_name", -1, (ub1 *)
d_name, 11,
    SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curd);
    quit ();
    exit (1);
}

if (obndrv (&curd, (text *) ":d_street_1", -1, (ub1 *)
d_street_1, 21,
    SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curd);
    quit ();
    exit (1);
}

if (obndrv (&curd, (text *) ":d_street_2", -1, (ub1 *)
d_street_2, 21,
    SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curd);
    quit ();
    exit (1);
}

if (obndrv (&curd, (text *) ":d_city", -1, (ub1 *) d_city,
21,
    SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curd);
    quit ();
    exit (1);
}

if (obndrv (&curd, (text *) ":d_state", -1, (ub1 *)
d_state, 2,
    SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curd);
    quit ();
    exit (1);
}

if (obndrv (&curd, (text *) ":d_zip", -1, (ub1 *) d_zip,
9,
    SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curd);
    quit ();
    exit (1);
}

if (obndrv (&curd, (text *) ":d_tax", -1, (ub1 *) d_tax,
sizeof (float),
    SQLT_FLT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curd);
    quit ();
    exit (1);
}

/* cust */

if (obndrv (&curc, (text *) ":c_id", -1, (ub1 *) c_id,
sizeof (int),
    SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curc);
    quit ();
    exit (1);
}

}

if (obndrv (&curc, (text *) ":c_d_id", -1, (ub1 *)
c_d_id, sizeof (int),
    SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curc);
    quit ();
    exit (1);
}

if (obndrv (&curc, (text *) ":c_w_id", -1, (ub1 *)
c_w_id, sizeof (int),
    SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curc);
    quit ();
    exit (1);
}

if (obndrv (&curc, (text *) ":c_first", -1, (ub1 *) c_first,
17,
    SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curc);
    quit ();
    exit (1);
}

if (obndrv (&curc, (text *) ":c_last", -1, (ub1 *) c_last,
17,
    SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curc);
    quit ();
    exit (1);
}

if (obndrv (&curc, (text *) ":c_street_1", -1, (ub1 *)
c_street_1, 21,
    SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curc);
    quit ();
    exit (1);
}

if (obndrv (&curc, (text *) ":c_street_2", -1, (ub1 *)
c_street_2, 21,
    SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curc);
    quit ();
    exit (1);
}

if (obndrv (&curc, (text *) ":c_city", -1, (ub1 *) c_city,
21,
    SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curc);
    quit ();
    exit (1);
}

if (obndrv (&curc, (text *) ":c_state", -1, (ub1 *)
c_state, 2,
    SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curc);
    quit ();
    exit (1);
}

if (obndrv (&curc, (text *) ":c_zip", -1, (ub1 *) c_zip, 9,
    SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curc);
    quit ();
    exit (1);
}

quit ();
exit (1);
}

if (obndrv (&curc, (text *) ":c_phone", -1, (ub1 *)
c_phone, 16,
    SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curc);
    quit ();
    exit (1);
}

if (obndrv (&curc, (text *) ":c_credit", -1, (ub1 *)
c_credit, 2,
    SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curc);
    quit ();
    exit (1);
}

if (obndrv (&curc, (text *) ":c_discount", -1, (ub1 *)
c_discount,
    sizeof (float), SQLT_FLT, -1, (sb2 *) 0, (text
*) 0, -1,
    -1)) {
    errrpt (&tpclda, &curc);
    quit ();
    exit (1);
}

if (obndrv (&curc, (text *) ":c_data", -1, (ub1 *)
c_data, 501,
    SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curc);
    quit ();
    exit (1);
}

/* item */

if (obndrv (&curi, (text *) ":i_id", -1, (ub1 *) i_id,
sizeof (int),
    SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curi);
    quit ();
    exit (1);
}

if (obndrv (&curi, (text *) ":i_im_id", -1, (ub1 *)
i_im_id, sizeof (int),
    SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curi);
    quit ();
    exit (1);
}

if (obndrv (&curi, (text *) ":i_name", -1, (ub1 *)
i_name, 25,
    SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
    errrpt (&tpclda, &curi);
    quit ();
    exit (1);
}

if (obndrv (&curi, (text *) ":i_price", -1, (ub1 *) i_price,
sizeof (int), SQLT_INT, -1, (sb2 *) 0, (text *)
0, -1,
    -1)) {
    errrpt (&tpclda, &curi);
    quit ();
    exit (1);
}

```

```

quit ();
exit (1);
}

if (obndrv (&curi, (text *) "i_data", -1, (ub1 *) i_data,
51,
SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curi);
quit ();
exit (1);
}

/* stok */

if (obndrv (&curs, (text *) "s_i_id", -1, (ub1 *) s_i_id,
sizeof (int),
SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curs);
quit ();
exit (1);
}

if (obndrv (&curs, (text *) "s_w_id", -1, (ub1 *)
s_w_id, sizeof (int),
SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curs);
quit ();
exit (1);
}

if (obndrv (&curs, (text *) "s_quantity", -1, (ub1 *)
s_quantity,
sizeof (int), SQLT_INT, -1, (sb2 *) 0, (text *)
0, -1, -1)) {
errprt (&tpclda, &curs);
quit ();
exit (1);
}

if (obndrv (&curs, (text *) "s_dist_01", -1, (ub1 *)
s_dist_01, 24,
SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curs);
quit ();
exit (1);
}

if (obndrv (&curs, (text *) "s_dist_02", -1, (ub1 *)
s_dist_02, 24,
SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curs);
quit ();
exit (1);
}

if (obndrv (&curs, (text *) "s_dist_03", -1, (ub1 *)
s_dist_03, 24,
SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curs);
quit ();
exit (1);
}

if (obndrv (&curs, (text *) "s_dist_04", -1, (ub1 *)
s_dist_04, 24,
SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curs);
quit ();
exit (1);
}

}

if (obndrv (&curs, (text *) "s_dist_05", -1, (ub1 *)
s_dist_05, 24,
SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curs);
quit ();
exit (1);
}

if (obndrv (&curs, (text *) "s_dist_06", -1, (ub1 *)
s_dist_06, 24,
SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curs);
quit ();
exit (1);
}

if (obndrv (&curs, (text *) "s_dist_07", -1, (ub1 *)
s_dist_07, 24,
SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curs);
quit ();
exit (1);
}

if (obndrv (&curs, (text *) "s_dist_08", -1, (ub1 *)
s_dist_08, 24,
SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curs);
quit ();
exit (1);
}

if (obndrv (&curs, (text *) "s_dist_09", -1, (ub1 *)
s_dist_09, 24,
SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curs);
quit ();
exit (1);
}

if (obndrv (&curs, (text *) "s_dist_10", -1, (ub1 *)
s_dist_10, 24,
SQLT_CHR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curs);
quit ();
exit (1);
}

if (obndrv (&curs, (text *) "s_data", -1, (ub1 *)
s_data, 51,
SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curs);
quit ();
exit (1);
}

/* hist */

if (obndrv (&curh, (text *) "h_c_id", -1, (ub1 *)
h_c_id, sizeof (int),
SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curh);
quit ();
exit (1);
}

if (obndrv (&curh, (text *) "h_c_d_id", -1, (ub1 *)
h_d_id, sizeof (int),
SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curh);
quit ();
exit (1);
}

if (obndrv (&curh, (text *) "h_c_w_id", -1, (ub1 *)
h_w_id, sizeof (int),
SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curh);
quit ();
exit (1);
}

if (obndrv (&curh, (text *) "h_d_id", -1, (ub1 *)
h_d_id, sizeof (int),
SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curh);
quit ();
exit (1);
}

if (obndrv (&curh, (text *) "h_w_id", -1, (ub1 *)
h_w_id, sizeof (int),
SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curh);
quit ();
exit (1);
}

if (obndrv (&curh, (text *) "h_data", -1, (ub1 *)
h_data, 25,
SQLT_STR, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curh);
quit ();
exit (1);
}

/* ordl (delivered) */

if (obndrv (&curol1, (text *) "ol_o_id", -1, (ub1 *)
ol_o_id,
sizeof (int), SQLT_INT, -1, (sb2 *) 0, (text *)
0, -1, -1)) {
errprt (&tpclda, &curol1);
quit ();
exit (1);
}

if (obndrv (&curol1, (text *) "ol_d_id", -1, (ub1 *)
ol_d_id,
sizeof (int), SQLT_INT, -1, (sb2 *) 0, (text *)
0, -1, -1)) {
errprt (&tpclda, &curol1);
quit ();
exit (1);
}

if (obndrv (&curol1, (text *) "ol_w_id", -1, (ub1 *)
ol_w_id,
sizeof (int), SQLT_INT, -1, (sb2 *) 0, (text *)
0, -1, -1)) {
errprt (&tpclda, &curol1);
quit ();
exit (1);
}

```

```

if (obndrv (&curo1, (text *) ":ol_number", -1, (ub1 *)
ol_number,
sizeof (int), SSQL_INT, -1, (sb2 *) 0, (text *) 0, -1,
-1)) {
errprt (&tpclda, &curo1);
quit ();
exit (1);
}

if (obndrv (&curo1, (text *) ":ol_i_id", -1, (ub1 *)
ol_i_id,
sizeof (int), SSQL_INT, -1, (sb2 *) 0, (text *) 0, -1,
-1)) {
errprt (&tpclda, &curo1);
quit ();
exit (1);
}

if (obndrv (&curo1, (text *) ":ol_supply_w_id", -1,
SQLT_INT, -1,
(sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curo1);
quit ();
exit (1);
}

if (obndrv (&curo1, (text *) ":ol_dist_info", -1, (ub1 *)
ol_dist_info,
24, SSQL_CHR, -1, (sb2 *) 0, (text *) 0, -1, -
1)) {
errprt (&tpclda, &curo1);
quit ();
exit (1);
}

/* ordl (not delivered) */

if (obndrv (&curo2, (text *) ":ol_o_id", -1, (ub1 *)
ol_o_id,
sizeof (int), SSQL_INT, -1, (sb2 *) 0, (text *)
0, -1, -1)) {
errprt (&tpclda, &curo2);
quit ();
exit (1);
}

if (obndrv (&curo2, (text *) ":ol_d_id", -1, (ub1 *)
ol_d_id,
sizeof (int), SSQL_INT, -1, (sb2 *) 0, (text *)
0, -1, -1)) {
errprt (&tpclda, &curo2);
quit ();
exit (1);
}

if (obndrv (&curo2, (text *) ":ol_w_id", -1, (ub1 *)
ol_w_id,
sizeof (int), SSQL_INT, -1, (sb2 *) 0, (text *)
0, -1, -1)) {
errprt (&tpclda, &curo2);
quit ();
exit (1);
}

if (obndrv (&curo2, (text *) ":ol_number", -1, (ub1 *)
ol_number,
sizeof (int), SSQL_INT, -1, (sb2 *) 0, (text *) 0, -1,
-1)) {
errprt (&tpclda, &curo2);
quit ();
exit (1);
}

errprt (&tpclda, &curo2);
quit ();
exit (1);
}

if (obndrv (&curo2, (text *) ":ol_i_id", -1, (ub1 *)
ol_i_id,
sizeof (int), SSQL_INT, -1, (sb2 *) 0, (text *) 0, -1,
-1)) {
errprt (&tpclda, &curo2);
quit ();
exit (1);
}

if (obndrv (&curo2, (text *) ":ol_supply_w_id", -1,
(ub1 *) ol_supply_w_id, sizeof (int),
SQLT_INT, -1,
(sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curo2);
quit ();
exit (1);
}

if (obndrv (&curo2, (text *) ":ol_amount", -1, (ub1 *)
ol_amount,
sizeof (int), SSQL_INT, -1, (sb2 *) 0, (text *) 0, -1,
-1)) {
errprt (&tpclda, &curo2);
quit ();
exit (1);
}

if (obndrv (&curo2, (text *) ":ol_dist_info", -1, (ub1 *)
ol_dist_info,
24, SSQL_CHR, -1, (sb2 *) 0, (text *) 0, -1, -
1)) {
errprt (&tpclda, &curo2);
quit ();
exit (1);
}

/* ordl (delivered) */

if (obndrv (&curo1, (text *) ":o_id", -1, (ub1 *) o_id,
sizeof (int),
SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curo1);
quit ();
exit (1);
}

if (obndrv (&curo1, (text *) ":o_d_id", -1, (ub1 *)
o_d_id, sizeof (int),
SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curo1);
quit ();
exit (1);
}

if (obndrv (&curo1, (text *) ":o_w_id", -1, (ub1 *)
o_w_id, sizeof (int),
SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curo1);
quit ();
exit (1);
}

if (obndrv (&curo1, (text *) ":o_c_id", -1, (ub1 *)
o_c_id, sizeof (int),
SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curo1);
quit ();
exit (1);
}

SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curo1);
quit ();
exit (1);
}

if (obndrv (&curo1, (text *) ":o_carrier_id", -1, (ub1 *)
o_carrier_id,
sizeof (int), SSQL_INT, -1, (sb2 *) 0, (text *) 0, -1,
-1)) {
errprt (&tpclda, &curo1);
quit ();
exit (1);
}

if (obndrv (&curo1, (text *) ":o_ol_cnt", -1, (ub1 *)
o_ol_cnt,
sizeof (int), SSQL_INT, -1, (sb2 *) 0, (text *) 0, -1,
-1)) {
errprt (&tpclda, &curo1);
quit ();
exit (1);
}

/* ordr (not delivered) */

if (obndrv (&curo2, (text *) ":o_id", -1, (ub1 *) o_id,
sizeof (int),
SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curo2);
quit ();
exit (1);
}

if (obndrv (&curo2, (text *) ":o_d_id", -1, (ub1 *)
o_d_id, sizeof (int),
SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curo2);
quit ();
exit (1);
}

if (obndrv (&curo2, (text *) ":o_w_id", -1, (ub1 *)
o_w_id, sizeof (int),
SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curo2);
quit ();
exit (1);
}

if (obndrv (&curo2, (text *) ":o_c_id", -1, (ub1 *)
o_c_id, sizeof (int),
SQLT_INT, -1, (sb2 *) 0, (text *) 0, -1, -1)) {
errprt (&tpclda, &curo2);
quit ();
exit (1);
}

if (obndrv (&curo2, (text *) ":o_ol_cnt", -1, (ub1 *)
o_ol_cnt,
sizeof (int), SSQL_INT, -1, (sb2 *) 0, (text *)
0, -1, -1)) {
errprt (&tpclda, &curo2);
quit ();
exit (1);
}

/* new order */

```

```

if (obndrv (&curno, (text *) ":no_o_id", -1, (ub1 *)
no_o_id,
sizeof (int), SQLT_INT, -1, (sb2 *) 0, (text *)
0, -1, -1)) {
errprt (&tpclda, &curno);
quit ();
exit (1);
}

if (obndrv (&curno, (text *) ":no_d_id", -1, (ub1 *)
no_d_id,
sizeof (int), SQLT_INT, -1, (sb2 *) 0, (text *)
0, -1, -1)) {
errprt (&tpclda, &curno);
quit ();
exit (1);
}

if (obndrv (&curno, (text *) ":no_w_id", -1, (ub1 *)
no_w_id,
sizeof (int), SQLT_INT, -1, (sb2 *) 0, (text *)
0, -1, -1)) {
errprt (&tpclda, &curno);
quit ();
exit (1);
}

/*-----+
| Initialize random number generator
|
+-----*/

srand (SEED);
#ifdef ORA_NT
srand48 (SEED);
#endif
initperm ();

/*-----+
| Load the WAREHOUSE table.
|
+-----*/

if (do_A || do_w) {
nrows = aware - bware + 1;

fprintf (stderr, "Loading/generating ware: w%d -
w%d (%d rows)\n",
bware, aware, nrows);

begin_time = gettime ();
begin_cpu = getcpu ();

for (loop = bware; loop <= aware; loop++) {

w_tax = (float)((lrnd48 () % 2001) * 0.0001);
randstr (w_name, 6, 10);
randstr (w_street_1, 10, 20);
randstr (w_street_2, 10, 20);
randstr (w_city, 10, 20);
randstr (str2, 2, 2);
randnum (num9, 9);
num9[4] = num9[5] = num9[6] = num9[7] =
num9[8] = '1';

if (gen) {
printf ("%d
30000000 %f %s %s %s %s %s %s\n", loop, w_tax,
w_name, w_street_1, w_street_2, w_city,
str2, num9);
fflush (stdout);
}
else {
w_id = loop;
strncpy (w_state, str2, 2);
strncpy (w_zip, num9, 9);

if (oexec (&curw) {
errprt (&tpclda, &curw);
orol (&tpclda);
fprintf (stderr, "Aborted at ware %d\n", loop);
quit ();
exit (1);
}
}
else if (ocom (&tpclda) {
errprt (&tpclda, &tpclda);
orol (&tpclda);
fprintf (stderr, "Aborted at ware %d\n", loop);
quit ();
exit (1);
}
}
}

end_time = gettime ();
end_cpu = getcpu ();
fprintf (stderr, "Done. %d rows loaded/generated
in %10d sec. (%10d cpu)\n\n",
nrows, end_time - begin_time, end_cpu -
begin_cpu);
}

/*-----+
| Load the DISTRICT table.
|
+-----*/

if (do_A || do_d) {
nrows = (eware - bware + 1) * DISTFAC;

fprintf (stderr, "Loading/generating dist: w%d - w%d
(%d rows)\n",
bware, aware, nrows);

begin_time = gettime ();
begin_cpu = getcpu ();

dwid = bware - 1;

for (row = 0; row < nrows; ) {
dwid++;

for (i = 0; i < DISTARR; i++, row++) {
d_tax[i] = (float)((lrnd48 () % 2001) * 0.0001);
randstr (d_name[i], 6, 10);
randstr (d_street_1[i], 10, 20);
randstr (d_street_2[i], 10, 20);
randstr (d_city[i], 10, 20);
randstr (str2, 2, 2);
randnum (num9, 9);
num9[4] = num9[5] = num9[6] = num9[7] =
num9[8] = '1';

if (gen) {
printf ("%d %d %s %s %s %s %s %s %d
30000.0 3001\n",
i + 1, dwid, d_name[i], d_street_1[i],
d_street_2[i],
d_city[i], str2, num9, d_tax[i]); */
/* Reordered columns */
printf ("%d %d 30000000 %f
3001 %s %s %s %s %s\n",
i + 1, dwid, d_tax[i], d_name[i],
d_street_1[i],
d_street_2[i], d_city[i], str2, num9);
}
else {
d_id[i] = i + 1;
d_w_id[i] = dwid;
strncpy (d_state[i], str2, 2);
strncpy (d_zip[i], num9, 9);
}
}
}

if (gen) {
fflush (stdout);
}
else {
if (oexn (&curd, DISTARR, 0) {
errprt (&tpclda, &curd);
orol (&tpclda);
fprintf (stderr, "Aborted at ware %d, dist 1\n",
dwid);
quit ();
exit (1);
}
else if (ocom (&tpclda) {
errprt (&tpclda, &tpclda);
orol (&tpclda);
fprintf (stderr, "Aborted at ware %d, dist 1\n",
dwid);
quit ();
exit (1);
}
}
}

end_time = gettime ();
end_cpu = getcpu ();
fprintf (stderr, "Done. %d rows loaded/generated
in %10d sec. (%10d cpu)\n\n",
nrows, end_time - begin_time, end_cpu -
begin_cpu);
}

/*-----+
| Load the CUSTOMER table.
|
+-----*/

if (do_A || do_c) {
nrows = (eware - bware + 1) * CUSTFAC *
DISTFAC;

fprintf (stderr, "Loading/generating cust: w%d - w%d
(%d rows)\n ",
bware, 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 < CUSTARR; 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 %cC 5000000 %6.4f
-1000 1000 1 0 %s\n",
        cid, cdid, cwid, c_first[i], c_last[i],
        c_street_1[i], c_street_2[i], c_city[i], str2,
num9,
        num16, sdate, c_credit[i][0], c_discount[i],
c_data[i]);
    }
    else {
      strncpy (c_state[i], str2, 2);
      strncpy (c_zip[i], num9, 9);
      strncpy (c_phone[i], num16, 16);
    }
  }
}

if (gen) {
  fflush (stdout);
}
else {
  if (oexn (&curc, CUSTARR, 0)) {
    errprt (&tpclda, &curc);
    orol (&tpclda);
    fprintf (stderr, "Aborted at w_id %d, d_id %d,
c_id %d\n",
      c_w_id[0], c_d_id[0], c_id[0]);
    quit ();
    exit (1);
  }
  else if (ocom (&tpclda)) {
    errprt (&tpclda, &tpclda);

```

```

    orol (&tpclda);
    fprintf (stderr, "Aborted at w_id %d, d_id %d,
c_id %d\n",
      c_w_id[0], c_d_id[0], c_id[0]);
    quit ();
    exit (1);
  }
}

if ((++loopcount) % 50)
  fprintf (stderr, ".");
else
  fprintf (stderr, "%d rows committed\n ", row);
}

end_time = gettime ();
end_cpu = getcpu ();
fprintf (stderr, "Done. %d rows loaded/generated
in %10d sec. (%10d cpu)\n\n",
  nrows, end_time - begin_time, end_cpu -
begin_cpu);
}

/*-----+
| Load the ITEM table.
+-----*/

if (do_A || do_i) {
  nrows = ITEMFAC;

  fprintf (stderr, "Loading/generating item: (%d
rows)\n ", nrows);

  begin_time = gettime ();
  begin_cpu = getcpu ();

  loopcount = 0;

  for (row = 0; row < nrows; ) {
    for (i = 0; i < ITEMARR; i++, row++) {
      i_im_id[i] = (lrand48 () % 10000) + 1;
      i_price[i] = ((lrand48 () % 9901) + 100);
      randstr (i_name[i], 14, 24);
      randdatastr (i_data[i], 26, 50);

      if (gen) {
        printf ("%d %d %s %d %s\n", row + 1,
i_im_id[i], i_name[i],
          i_price[i], i_data[i]);
      }
      else {
        i_id[i] = row + 1;
      }
    }
  }

  if (gen) {
    fflush (stdout);
  }
  else {
    if (oexn (&curi, ITEMARR, 0)) {
      errprt (&tpclda, &curi);
      orol (&tpclda);
      fprintf (stderr, "Aborted at i_id %d\n", i_id[0]);
      quit ();
      exit (1);
    }
  }
  else if (ocom (&tpclda)) {
    errprt (&tpclda, &tpclda);

```

```

    orol (&tpclda);
    fprintf (stderr, "Aborted at i_id %d\n", i_id[0]);
    quit ();
    exit (1);
  }
}

if ((++loopcount) % 50)
  fprintf (stderr, ".");
else
  fprintf (stderr, "%d rows committed\n ", row);
}

end_time = gettime ();
end_cpu = getcpu ();
fprintf (stderr, "Done. %d rows loaded/generated
in %10d sec. (%10d cpu)\n\n",
  nrows, end_time - begin_time, end_cpu -
begin_cpu);
}

/*-----+
| Load the STOCK table.
+-----*/

if (do_A || do_s) {
  nrows = (eware - bware + 1) * STOCFAC;

  fprintf (stderr, "Loading/generating stok: w%d - w%d
(%d rows)\n ",
    bware, aware, nrows);

  begin_time = gettime ();
  begin_cpu = getcpu ();

  sid = 0;
  swid = bware;
  loopcount = 0;

  for (row = 0; row < nrows; ) {
    for (i = 0; i < STOCARR; i++, row++) {
      if (++sid > STOCFAC) { /* cheap mod */
        sid = 1;
        swid++;
      }
      s_quantity[i] = (lrand48 () % 91) + 10;
      randstr (str24[0], 24, 24);
      randstr (str24[1], 24, 24);
      randstr (str24[2], 24, 24);
      randstr (str24[3], 24, 24);
      randstr (str24[4], 24, 24);
      randstr (str24[5], 24, 24);
      randstr (str24[6], 24, 24);
      randstr (str24[7], 24, 24);
      randstr (str24[8], 24, 24);
      randstr (str24[9], 24, 24);
      randdatastr (s_data[i], 26, 50);

      if (gen) {
        printf
("%d %d %d %s %s %s %s %s %s %s %s %s %s %s 0
0 %s\n",
          sid, swid, s_quantity[i], str24[0], str24[1],
str24[2],
          str24[3], str24[4], str24[5], str24[6],
str24[7],
          str24[8], str24[9], s_data[i]);

```



```

        fprintf(stderr, "Aborted at w_id %d, d_id %d,
o_id %d\n ",
            cwid, cdid, cid);
        quit ();
        exit (1);
    }
}

if (++loopcount % 50)
    fprintf (stderr, ".");
else
    fprintf (stderr, "%d ordr committed\n ", row);
}

end_time = gettime ();
end_cpu = getcpu ();
fprintf (stderr, "Done. %d ordr loaded/generated
in %10d sec. (%10d cpu)\n\n",
        nrows, end_time - begin_time, end_cpu -
begin_cpu);
}

/*-----+
| Load the NEW-ORDER table.
|
+-----*/

if (do_A || do_n) {
    nrows = (eware - bware + 1) * NEWOFAC *
DISTFAC;

    fprintf (stderr, "Loading/generating new-order: w%d
- w%d (%d rows)\n ",
            bware, ewart, nrows);

    begin_time = gettime ();
    begin_cpu = getcpu ();

    cid = 0;
    cdid = 1;
    cwid = bware;
    loopcount = 0;

    for (row = 0; row < nrows;) {
        for (i = 0; i < NEWOARR; i++, row++) {
            cid++;
            if (cid > NEWOFAC) {
                cid = 1;
                cdid++;
                if (cdid > DISTFAC) {
                    cdid = 1;
                    cwid++;
                }
            }
        }

        if (gen) {
            printf ("%d %d %d\n", cid + 2100, cdid, cwid);
        }
        else {
            no_o_id[i] = cid + 2100;
            no_d_id[i] = cdid;
            no_w_id[i] = cwid;
        }
    }

    if (gen) {
        fflush (stdout);
    }
}

```

```

    else {
        if (oexn (&curno, NEWOARR, 0)) {
            errprt (&tpclda, &curno);
            orol (&tpclda);
            fprintf (stderr, "Aborted at w_id %d, d_id %d,
o_id %d\n ",
                cwid, cdid, cid + 2100);
            quit ();
            exit (1);
        }
        else if (ocom (&tpclda)) {
            errprt (&tpclda, &tpclda);
            orol (&tpclda);
            fprintf (stderr, "Aborted at w_id %d, d_id %d,
o_id %d\n ",
                cwid, cdid, cid + 2100);
            quit ();
            exit (1);
        }
    }

    if ((++loopcount) % 45)
        fprintf (stderr, ".");
    else
        fprintf (stderr, "%d rows committed\n ", row);
}

end_time = gettime ();
end_cpu = getcpu ();
fprintf (stderr, "Done. %d rows loaded/generated
in %10d sec. (%10d cpu)\n\n",
        nrows, end_time - begin_time, end_cpu -
begin_cpu);
}

/*-----+
| clean up and exit.
|
+-----*/

if (olfp)
    fclose (olfp);
if (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) (rand48 () % 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 = rand48 () % (A + 1);
    b = (rand48 () % (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);
}

===== buildsvr.bat =====

@rem      Copyright (c) 1998 Oracle Corp, Redwood
Shores, CA
@rem      All Rights Reserved

rem Build TM servers

cd /MD /Ox /G6 -!%TUXDIR%\include -!%CINC%\server
-!%CINC%\client -!c:\oracle\ora81\oci\include -DTUX -
Fegeneric BS-mb.c tpccsvr.c
c:\inetPub\wwwroot\common_files.lib %CINC%\lib\dpb
nt.lib
C:\Oracle\Ora81\OCIN\LIBMSVC\oci.lib %TUXDIR%\lib\
ibtux.lib %TUXDIR%\lib\libbuf.lib %TUXDIR%\lib\libtu
x2.lib %TUXDIR%\lib\libfml.lib %TUXDIR%\lib\libfml32.
lib %TUXDIR%\lib\libgp.lib wsock32.lib kernel32.lib
advapi32.lib user32.lib gdi32.lib winspool.lib -link -
implib:BS-mb.lib /nodefaultlib:libc.lib
/nodefaultlib:libcd.lib
del C:\sv-apl\ora\tpccfml.exe
del C:\sv-apl\ora\stdout
move generic.exe C:\sv-apl\ora\tpccfml.exe

cd /MD /Ox /G6 -!%TUXDIR%\include -!%CINC%\server
-!%CINC%\client -!c:\oracle\ora81\oci\include -DTUX -
Fegeneric BS-mbp.c tpccsvr.c
c:\inetPub\wwwroot\common_files.lib %CINC%\lib\dpb
nt.lib
C:\Oracle\Ora81\OCIN\LIBMSVC\oci.lib %TUXDIR%\lib\
ibtux.lib %TUXDIR%\lib\libbuf.lib %TUXDIR%\lib\libtu
x2.lib %TUXDIR%\lib\libfml.lib %TUXDIR%\lib\libfml32.
lib %TUXDIR%\lib\libgp.lib wsock32.lib kernel32.lib
advapi32.lib user32.lib gdi32.lib winspool.lib -link -
implib:BS-mb.lib /nodefaultlib:libc.lib
/nodefaultlib:libcd.lib
del C:\sv-apl\ora\tpccfmlp.exe
move generic.exe C:\sv-apl\ora\tpccfmlp.exe

cd /MD /Ox /G6 -!%TUXDIR%\include -!%CINC%\server
-!%CINC%\client -!c:\oracle\ora81\oci\include -DTUX -
Fegeneric BS-mbo.c tpccsvr.c
c:\inetPub\wwwroot\common_files.lib %CINC%\lib\dpb
nt.lib
C:\Oracle\Ora81\OCIN\LIBMSVC\oci.lib %TUXDIR%\lib\
ibtux.lib %TUXDIR%\lib\libbuf.lib %TUXDIR%\lib\libtu
x2.lib %TUXDIR%\lib\libfml.lib %TUXDIR%\lib\libfml32.
lib %TUXDIR%\lib\libgp.lib wsock32.lib kernel32.lib
advapi32.lib user32.lib gdi32.lib winspool.lib -link -
implib:BS-mb.lib /nodefaultlib:libc.lib
/nodefaultlib:libcd.lib
del C:\sv-apl\ora\tpccfmlo.exe
move generic.exe C:\sv-apl\ora\tpccfmlo.exe

cd /MD /Ox /G6 -!%TUXDIR%\include -!%CINC%\server
-!%CINC%\client -!c:\oracle\ora81\oci\include -DTUX -
Fegeneric BS-mbd.c tpccsvr.c
c:\inetPub\wwwroot\common_files.lib %CINC%\lib\dpb
nt.lib
C:\Oracle\Ora81\OCIN\LIBMSVC\oci.lib %TUXDIR%\lib\
ibtux.lib %TUXDIR%\lib\libbuf.lib %TUXDIR%\lib\libtu
x2.lib %TUXDIR%\lib\libfml.lib %TUXDIR%\lib\libfml32.
lib %TUXDIR%\lib\libgp.lib wsock32.lib kernel32.lib
advapi32.lib user32.lib gdi32.lib winspool.lib -link -
implib:BS-mb.lib /nodefaultlib:libc.lib
/nodefaultlib:libcd.lib
del C:\sv-apl\ora\tpccfmd.exe
move generic.exe C:\sv-apl\ora\tpccfmd.exe

cd /MD /Ox /G6 -!%TUXDIR%\include -!%CINC%\server
-!%CINC%\client -!c:\oracle\ora81\oci\include -DTUX -
Fegeneric BS-mbs.c tpccsvr.c
c:\inetPub\wwwroot\common_files.lib %CINC%\lib\dpb
nt.lib
C:\Oracle\Ora81\OCIN\LIBMSVC\oci.lib %TUXDIR%\lib\
ibtux.lib %TUXDIR%\lib\libbuf.lib %TUXDIR%\lib\libtu
x2.lib %TUXDIR%\lib\libfml.lib %TUXDIR%\lib\libfml32.
lib %TUXDIR%\lib\libgp.lib wsock32.lib kernel32.lib
advapi32.lib user32.lib gdi32.lib winspool.lib -link -
implib:BS-mb.lib /nodefaultlib:libc.lib
/nodefaultlib:libcd.lib
del C:\sv-apl\ora\tpccfmls.exe
move generic.exe C:\sv-apl\ora\tpccfmls.exe

rem mv.bat

rem cd /MD /Ox /G6 /Gf /Oy /W3 /D_X86_=1 -
!%TUXDIR%\include -!%CINC%\server -
!%CINC%\client -!c:\oracle\ora81\oci\include -DTUX -
Fegeneric BS-mb.c tpccsvr.c
c:\inetPub\wwwroot\common_files.lib %CINC%\lib\dpb
nt.lib
C:\Oracle\Ora81\OCIN\LIBMSVC\oci.lib %TUXDIR%\lib\
ibtux.lib %TUXDIR%\lib\libbuf.lib %TUXDIR%\lib\libtu
x2.lib %TUXDIR%\lib\libfml.lib %TUXDIR%\lib\libfml32.
lib %TUXDIR%\lib\libgp.lib wsock32.lib kernel32.lib
advapi32.lib user32.lib gdi32.lib winspool.lib -link -
implib:BS-mb.lib /nodefaultlib:libc.lib
/nodefaultlib:libcd.lib
rem mv.bat

===== mv.bat =====

del C:\sv-apl\ora\tpccfml.exe
del C:\sv-apl\ora\stdout
move generic.exe C:\sv-apl\ora\tpccfml.exe

===== fldtbl.h =====

/*      fname      flidid      */
/*      -----      -----      */
#define FML_TERM ((FLDID)10193) /*
number: 2001 type: long */
#define FML_TRAN((FLDID)10194) /* number:
2002 type: long */
#define FML_DATA((FLDID)51155) /* number:
2003 type: carry */

===== getval.h =====

/* MACRO: GetVal(char *pQueryString, char *pKey,
char *pValue, int iMax)
*
* PURPOSE: This function parses a http formatted
string for specific key values.
*
* ARGUMENTS: char *pQueryString
http string from client browser
* char *pKey
key value to look for
* char *pValue
character array into which to place key's
value
* int iMax
maximum length of key value array.
*
*/

```

```

* RETURNS:      BOOL    FALSE    key
value not found
*
*              TRUE     key
value found
*
*
* COMMENTS:   http keys are formatted either
KEY=value& or KEY=value0. This DLL formats
*             TPC-C input fields in such a
manner that the keys can be extracted in the
*             above manner.
*/

#define GetVal(s, k, v, m) \
{ int max = m; char * vp = v; \
  if ( (s==strstr(s, k)) ) \
  if ( (s==strchr(s, '=')) ) \
  { \
    s++; \
    max--; \
    while( *s && *s != '&' && max ) \
    { \
      *vp++ = *s++; \
      max--; \
    } \
    *vp = 0; \
    s--; \
  } \
}

===== httpext.h =====

/*****
*
* Copyright (c) 1995 Process Software Corporation
*
* Copyright (c) 1995 Microsoft Corporation
*
*
* Module Name : HttpExt.h
*
* Abstract :
*
* This module contains the structure definitions and
prototypes for the
* version 1.0 HTTP Server Extension interface.
*
*****/
#ifndef _HTTPEXT_H_
#define _HTTPEXT_H_
#include <windows.h>
#ifdef _cplusplus
extern "C" {
#endif
#define HSE_VERSION_MAJOR    1 // major
version of this spec
#define HSE_VERSION_MINOR    0 // minor
version of this spec
#define HSE_LOG_BUFFER_LEN    80
#define HSE_MAX_EXT_DLL_NAME_LEN 256
typedef LPVOID HCONN;
// the following are the status codes returned by the
Extension DLL
#define HSE_STATUS_SUCCESS    1
#define
HSE_STATUS_SUCCESS_AND_KEEP_CONN 2
#define HSE_STATUS_PENDING    3

#define HSE_STATUS_ERROR      4
// The following are the values to request services with
the ServerSupportFunction.
// Values from 0 to 1000 are reserved for future
versions of the interface
#define HSE_REQ_BASE          0
#define HSE_REQ_SEND_URL_REDIRECT_RESP
(HSE_REQ_BASE + 1)
#define HSE_REQ_SEND_URL
(HSE_REQ_BASE + 2)
#define HSE_REQ_SEND_RESPONSE_HEADER
(HSE_REQ_BASE + 3)
#define HSE_REQ_DONE_WITH_SESSION
(HSE_REQ_BASE + 4)
#define HSE_REQ_END_RESERVED
1000
//
// These are Microsoft specific extensions
//
#define HSE_REQ_MAP_URL_TO_PATH
(HSE_REQ_END_RESERVED+1)
#define HSE_REQ_GET_SSPI_INFO
(HSE_REQ_END_RESERVED+2)
//
// passed to GetExtensionVersion
//
typedef struct _HSE_VERSION_INFO
{
  DWORD dwExtensionVersion;
  CHAR
  lpszExtensionDesc[HSE_MAX_EXT_DLL_NAME_LEN
];
} HSE_VERSION_INFO, *LPHSE_VERSION_INFO;
//
// passed to extension procedure on a new request
//
typedef struct _EXTENSION_CONTROL_BLOCK
{
  DWORD cbSize; // size of this struct.
  DWORD dwVersion; // version info of
this spec
  HCONN ConnID; // Context number not to
be modified!
  DWORD dwHttpStatusCode; // HTTP
Status code
  CHAR lpszLogData[HSE_LOG_BUFFER_LEN]; //
null terminated log info specific to this Extension DLL
  LPSTR lpszMethod; //
REQUEST_METHOD
  LPSTR lpszQueryString; // QUERY_STRING
  LPSTR lpszPathInfo; // PATH_INFO
  LPSTR lpszPathTranslated; //
PATH_TRANSLATED
  DWORD cbTotalBytes; // Total bytes
indicated from client
  DWORD cbAvailable; // Available
number of bytes
  LPBYTE lpbData; // pointer to
cbAvailable bytes
  LPSTR lpszContentType; // Content type of
client data
  BOOL (WINAPI * GetServerVariable) ( HCONN
hConn,
LPSTR
lpszVariableName,
LPVOID lpvBuffer,
LPDWORD lpdwSize );
  BOOL (WINAPI * WriteClient) ( HCONN
ConnID,
LPVOID Buffer,
LPDWORD lpdwBytes,
DWORD dwReserved );
  BOOL (WINAPI * ReadClient) ( HCONN
ConnID,
LPVOID lpvBuffer,
LPDWORD lpdwSize );
  BOOL (WINAPI * ServerSupportFunction)( HCONN
hConn,
DWORD
dwHSERRequest,
LPVOID lpvBuffer,
LPDWORD lpdwSize,
LPDWORD
lpdwDataType );
} EXTENSION_CONTROL_BLOCK,
*LPEXTENSION_CONTROL_BLOCK;
//
// these are the prototypes that must be exported from
the extension DLL
//
_declspec (dllexport)
BOOL WINAPI
GetExtensionVersion( HSE_VERSION_INFO *pVer );
_declspec (dllexport)
DWORD WINAPI
HttpExtensionProc( EXTENSION_CONTROL_BLOCK
*pECB );
// the following type declarations are for the server side
typedef BOOL (WINAPI *
PFN_GETEXTENSIONVERSION)( HSE_VERSION_IN
FO *pVer );
typedef DWORD (WINAPI *
PFN_HTTPPEXTENSIONPROC)( EXTENSION_CONT
ROL_BLOCK *pECB );
#ifdef _cplusplus
}
#endif
#endif // end definition _HTTPEXT_H_

===== tpcc.h =====

/*
* $Header: tpcc.h 7030100.1 95/07/19 15:10:55 plai
Generic<base> $ Copy (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

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

typedef struct cda_def csrdef;
typedef struct cda_def ldadef;

/* TPC-C transaction functions */

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

/* Error codes */

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

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

#define DELRT 80.0

extern int tkvcninit ();
extern int tkvcpinit ();
extern int tkvcoint ();
extern int tkvcidinit ();
extern int tkvcsinit ();

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

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

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

extern void errprt ();
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 *tpcsusr;
extern OCISStmt *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;
extern int threshold;
extern int low_stock;

/* for delivery transaction */

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

/* for order-status transaction */

extern int bylastname;
extern char c_last[17];
extern char c_first[17];
extern char c_middle[3];
extern double c_balance;

extern int o_id;
extern text o_entry_d[20];
extern int o_carrier_id;
extern int o_ol_cnt;
extern int ol_supply_w_id[15];
extern int ol_i_id[15];
extern int ol_quantity[15];
extern int ol_amount[15];
ub4 ol_del_len[15];
extern text ol_delivery_d[15][11];

/* for payment transaction */

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

/* for new order transaction */

extern int no_l_id[15];
extern int no_supply_w_id[15];
extern int no_quantity[15];
extern int no_quant10[15];
extern int no_quant19[15];
extern int no_ytdqty[15];
extern int no_amount[15];
extern int o_all_local;
extern float w_tax;
extern float d_tax;
extern float total_amount;
extern char i_name[15][25];
extern int i_name_strlen[15];
extern ub2 i_name_strlen_len[15];
extern ub2 i_name_strlen_rcode[15];
extern ub4 i_name_strlen_csize;
extern int s_quantity[15];
extern char brand_gen[15];
extern ub2 brand_gen_len[15];
extern ub2 brand_gen_rcode[15];
extern ub4 brand_gen_csize;
extern int l_price[15];
extern char brand_generic[15][1];
extern int status;
extern int tracelevel;

/* Miscellaneous */
extern OCIDate cr_date;

```

```

extern OCIDate c_since;
extern OCIDate o_entry_d_base;
extern OCIDate ol_d_base[15];

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

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

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

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

#define OCIEERROR(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_
HTYPE_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));

#define
OCIBNDRA(stmp,bndp,errp,sqlvar,progvl,progvl,ftype,in
dp,alen,arcode) \
    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), (indp), (alen), (arcode), 0, 0, OCI_D
EFAULT));

#define
OCIBNDRAD(stmp,bndp,errp,sqlvar,progvl,ftype,indp,ct
xp,cbf_nodata,cbf_data) \
    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)), 0, (progvl), (ftype), \
            indp, 0, 0, 0, OCI_DATA_AT_EXEC)); \
    ocierror(__FILE__,__LINE__,(errp), \
        OCIBindDynamic((bndp), (errp), (ctxp), (cbf_nodat
a), (ctxp), (cbf_data)));

#define
OCIBNDR(stmp,bndp,errp,sqlvar,progvl,progvl,ftype,ind
p,alen,arcode) \
    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), (indp), (alen), (arcode), 0, 0, OCI_D
EFAULT));

#define
OCIBNDRAA(stmp,bndp,errp,sqlvar,progvl,progvl,ftype,i
ndp,alen,arcode,ms,cu) \
    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), (indp), (alen), (arcode), (ms), (
cu), OCI_DEFAULT));

#define
OCIDFNDYN(stmp,dfnp,errp,pos,progvl,ftype)\
    OCIDefineByPos((stmp), &(dfnp), (errp), (pos), (prog
v), (progvl), (ftype), \
        0, 0, 0, OCI_DEFAULT);

#define OCIDDEF(stmp,dfnp,errp,pos,progvl,progvl,ftype)
\
    OCIHandleAlloc((stmp),(dvoid**)&(dfnp),OCI_HTY
PE_DEFINE,0, \
        (dvoid**0)); \
    OCIDefineByPos((stmp), &(dfnp), (errp), (pos), (prog
v), (progvl), \
        (ftype), NULL, NULL, NULL, OCI_DEFA
ULT); \

#define
OCIDFNRA(stmp,dfnp,errp,pos,progvl,progvl,ftype,indp,
alen,arcode) \
    OCIHandleAlloc((stmp),(dvoid**)&(dfnp),OCI_HTY
PE_DEFINE,0, \
        (dvoid**0)); \
    OCIDefineByPos((stmp), &(dfnp), (errp), (pos), (prog
v), (progvl), \
        (ftype), (indp), (alen), \
        (arcode), OCI_DEFAULT);

```

```

#define
OCIDFNDYN(stmp,dfnp,errp,pos,progvl,progvl,ftype,ind
p,ctxp,cbf_data) \
    ocierror(__FILE__,__LINE__,(errp), \
        OCIHandleAlloc((stmp),(dvoid**)&(dfnp),OCI_HTY
PE_DEFINE,0, \
        (dvoid**0)); \
    ocierror(__FILE__,__LINE__,(errp), \
        OCIDefineByPos((stmp), &(dfnp), (errp), (pos), (prog
v), (progvl), (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_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 bylast_name;
    int h_amount;
    char c_last[17];
};

struct payoutstruct {

```

```

int terror;
char w_street_1[21];
char w_street_2[21];
char w_city[21];
char w_state[3];
char w_zip[10];
char d_street_1[21];
char d_street_2[21];
char d_city[21];
char d_state[3];
char d_zip[10];
int c_id;
char c_first[17];
char c_middle[3];
char c_last[17];
char c_street_1[21];
char c_street_2[21];
char c_city[21];
char c_state[3];
char c_zip[10];
char c_phone[17];
char c_since[11];
char c_credit[3];
double c_credit_lim;
float c_discount;
double c_balance;
char c_data[201];
char h_date[20];
int retry;
};

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

/* Order status */
/*
struct ordinstruct {
    int w_id;
    int d_id;
    int c_id;
    int bylastname;
    char c_last[17];
};

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

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

/* Delivery */
/*
struct delinstruct {
    int w_id;
    int o_carrier_id;
    double qtime;
    int in_timing_int;
};

struct deloutstruct {
    int terror;
    int retry;
};

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

/* Stock level */
/*
struct stoinstruct {
    int w_id;
    int d_id;
    int threshold;
};

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

struct stostruct {
    struct stoinstruct stoin;
    struct stoostruct stoostruct;
};
*/
#endif

===== tpcc_info.h =====

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

#endif TPCC_INFO_H

```

```

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 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 {
    int retval;
    struct ordinstruc ordin;
    struct ordoutstruct ordout;
};

```

```

/* Delivery */

struct delinstruc {
    int w_id;
    int o_carrier_id;
    long startsec;
    long startusec;
    /* double qtime;
    int in_timing_int;*/
};

struct deloutstruct {
    int terror;
    int retry;
};

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

/* Stock level */

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

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

struct stostruct {
    int retval;
    struct stoinstruct stoin;
    struct stoostruct stoo;
};

/* 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

===== tpccapi.h =====

#ifndef TPCCAPI_H
#define TPCCAPI_H

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

```

```

| OPEN SYSTEMS PERFORMANCE
GROUP |
| All Rights Reserved |
+-----+
| FILENAME
| tpccapi.h
| DESCRIPTION
| header file to tpcc.dll
+-----+
|=====*/

//VERSION RESOURCE DEFINES
#define _APS_NEXT_RESOURCE_VALUE 101
#define _APS_NEXT_COMMAND_VALUE 40001
#define _APS_NEXT_CONTROL_VALUE 1000
#define _APS_NEXT_SYMED_VALUE 101

//note that the welcome form must be processed
first as terminal ids
//assigned here, once the terminal id is assigned
then the forms can
//be processed in any order.

#define WELCOME_FORM 1
//beginning form no term id assigned,
form id
#define MAIN_MENU_FORM 2
//term id assigned main menu form id
#define NEW_ORDER_FORM 3
//new order form id
#define PAYMENT_FORM 4
//payment form id
#define DELIVERY_FORM 5
//delivery form id
#define ORDER_STATUS_FORM 6
//order status id
#define STOCK_LEVEL_FORM 7
//stock level form id

//This macro is used to prevent the compiler error
unused formal parameter
#define UNUSEDPARAM(x) (x = x)

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

//This structure defines the data necessary to
keep distinct for each
//terminal or client connection.
typedef struct _CLIENTDATA
{
    int inUse; //in use flag allows client entries
to be reused
    int w_id; //warehouse id assigned at
welcome form
    int d_id; //district id assigned at welcome
form
    int iProcessRequestCounter; //request
counter
    int iSynclId; //synchronization id
    int iTickCount; //time of last access;

```

```

int iTermId; //terminal id of http stream
connection
char szBuffer[4096]; //form buffer; HTML form
built for client here
//for TOPEND
char type;
int retval;
int client;
BOOL bFailed;
#ifdef TOPEND
long *tp_structadr;
#else // assume tuxedo
struct newstruct *newOrderDataPtr; //new
order Tuxedo buffer
struct paystruct *paymentDataPtr;
//payment Tuxedo buffer
struct ordstruct *orderStatusDataPtr; //order
status Tuxedo buffer
struct delstruct *deliveryDataPtr; //delivery
Tuxedo buffer
struct stostruct *stockLevelDataPtr; //stock
level Tuxedo buffer
#endif
union {
struct newstruct newOrderData; //new order
form data
struct paystruct paymentData; //payment
form data
struct ordstruct orderStatusData; //order status
form data
struct delstruct deliveryData; //delivery form
data
struct stostruct stocklevelData; //stock level
form data
} trans;
} CLIENTDATA;

typedef CLIENTDATA *PCLIENTDATA; //pointer to
client structure
//This structure is used to define the operational
interface for
//terminal id support
typedef struct _TERM
{
int iAvailable; //total allocated terminal array entries
int iNext; //next available terminal array element
int iMasterSyncId; //synchronization id
BOOL bInit; //structure has been initialized flag
CLIENTDATA *pClientData; //pointer to allocated
client data
void (*Init)(void); //API to initialize this structure
int (*Allocate)(void); //API to allocate new terminal
entry;array id returned
void (*Restore)(void); //API to free terminal data
int (*Add)(EXTENSION_CONTROL_BLOCK *pECB,
char *pQueryString); //API to add a terminal id to
array,
//this context will be
passed from the
//browser to the tpcc.dll
in the
//TERMID= key in the
HTTP string.
void (*Delete)(EXTENSION_CONTROL_BLOCK
*pECB, int id); //API to free
//resources
used by a
//terminal
array
} entry

typedef TERM *PTERM; //pointer to terminal structure
type

//this structure allows the EXTENSION CONTROL
BLOCK to be passed to the
//msg and error handlers.
typedef struct _ECBINFO
{
int iTermId; //terminal id
int iSyncId; //browser sync id
BOOL bDeadlock; //deadlock condition flag
BOOL bFailed; //cleared before sql
transaction, set in err
//handlers if an error occurs
EXTENSION_CONTROL_BLOCK *pECB;
//inetsrv current connection structure
//information
} ECBINFO, *PECBINFO;

//function prototypes

#ifdef TOPEND // in tuxedo defined in atmi.h
extern int tpsvrinit (int argc, char * argv[]);
// argv arguments are int pid, char *uid, char
*pwd, int txntype

extern int NEWORDER(CLIENTDATA *jobData,
NewOrderData *neword, int deadlock);
extern int PAYMENT(CLIENTDATA *jobData,
PaymentData *paydata, int deadlock);
extern int ORDERSTATUS(CLIENTDATA *jobData,
OrderStatusData *orddata, int deadlock);
extern int STOCKLEVEL(CLIENTDATA *jobData,
StockLevelData *stodata, int deadlock);
extern int DELIVERY(CLIENTDATA *jobData,
DeliveryData *deldata, int deadlock);
#endif

#ifdef TUX

extern void NEWORDER();
extern void PAYMENT();
extern void ORDERSTATUS();
extern void STOCKLEVEL();
extern void DELIVERY();
/*
extern void NEWORDER(TPSVCINFO *msg);
extern void PAYMENT(TPSVCINFO *msg);
extern void ORDERSTATUS(TPSVCINFO *msg);
extern void STOCKLEVEL(TPSVCINFO *msg);
extern void DELIVERY(TPSVCINFO *msg);*/
#endif

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

```

```

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

#endif /* TPCCAPI_H */

===== tpcerr.h =====

#ifndef TPCCERR_H
#define TPCCERR_H

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

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

#define ERR_DB_ERROR 1 /* A
database error has occurred */
#define ERR_TRANSPORT_ERROR 2 /* A
transport error has occurred */
#define ERR_DB_INTERFACE 3 /* An error
occured setting up */
/* connection to DB */

#define ERR_DB_DEADLOCK_LIMIT 4 /* The
DB deadlock retry limit was */
/* reached */
#define ERR_DB_NOT_COMMITTED 5 /*
Transaction not committed */
#define ERR_DB_DEAD 6 /* Database
connection is invalid */
#define ERR_SUCCESS 1000 /*Success,
no error.
#define ERR_COMMAND_UNDEFINED 1001
/*Command undefined.
#define ERR_NOT_IMPLEMENTED_YET 1002
/*Not Implemented Yet.
#define ERR_CANNOT_INIT_TERMINAL 1003
/*Cannot initialize client connection.
#define ERR_OUT_OF_MEMORY 1004
/*insufficient memory.
#define ERR_NEW_ORDER_NOT_PROCESSED
1005 /*Cannot process new Order form.
#define ERR_PAYMENT_NOT_PROCESSED
1006 /*Cannot process payment form.
#define ERR_NO_SERVER_SPECIFIED 1007
/*No Server name specified.
#define
ERR_ORDER_STATUS_NOT_PROCESSED
1008 /*Cannot process order status form.
#define ERR_W_ID_INVALID 1009 /*Invalid
Warehouse ID.
#define
ERR_CAN_NOT_SET_MAX_CONNECTIONS
1010 /*Insufficient memory to
/*allocate #
connections.
#define ERR_NOSUCH_CUSTOMER 1011 /*No
such customer.
#define ERR_D_ID_INVALID 1012 /*Invalid
District ID Must be 1 to 10.
#define ERR_MAX_CONNECT_PARAM 1013
/*Max client connections exceeded,
/*run install to increase.
#define ERR_INVALID_SYNC_CONNECTION
1014 /*Invalid Terminal Sync ID.
#define ERR_INVALID_TERMINAL 1015
/*Invalid Terminal ID.
#define ERR_PAYMENT_INVALID_CUSTOMER
1016 /*Payment Form, No such Customer.
#define ERR_SQL_OPEN_CONNECTION 1017
/*SQLOpenConnection API Failed.
#define
ERR_STOCKLEVEL_MISSING_THRESHOLD_
KEY 1018 /*Stock Level missing
/*Threshold key
"TT".
#define ERR_STOCKLEVEL_THRESHOLD_INVALID
1019 /*Stock Level Threshold invalid
/*data type range = 1
- 99.
#define
ERR_STOCKLEVEL_THRESHOLD_RANGE
1020 /*Stock Level Threshold out of
/*range, range must
be 1 - 99.
#define ERR_STOCKLEVEL_NOT_PROCESSED
1021 /*Stock Level not processed.
#define ERR_NEWORDER_FORM_MISSING_DID
1022 /*NewOrder miss District key "DID".
#define ERR_NEWORDER_DISTRICT_INVALID 1023
/*NewOrder District ID Invalid
/*range 1 - 10.

#define ERR_NEWORDER_DISTRICT_RANGE
1024 /*New Order District ID out of
/*Range. Range = 1 -
10.
#define ERR_NEWORDER_CUSTOMER_KEY
1025 /*New Order missing Customer key "CID".
#define
ERR_NEWORDER_CUSTOMER_INVALID 1026
/*NewOrder customer id invalid data
/*type, range = 1 to
3000.
#define ERR_NEWORDER_CUSTOMER_RANGE
1027 /*New Order customer id out of
/*range, range = 1 to
3000.
#define ERR_NEWORDER_MISSING_IID_KEY
1028 /*NewOrder missng ItemId key "IID".
#define ERR_NEWORDER_ITEM_BLANK_LINES
1029 /*NewOrder blank order lines all
/*orders must be
continuous.
#define ERR_NEWORDER_ITEMID_INVALID 1030
/*New Order Item Id is wrong data
/*type, must be
numeric.
#define ERR_NEWORDER_MISSING_SUPPW_KEY
1031 /*NewOrder missing Supp_W key
/*SP###.
#define ERR_NEWORDER_SUPPW_INVALID
1032 /*New Order Supp_W invalid data
/*type must be numeric.
#define ERR_NEWORDER_MISSING_QTY_KEY
1033 /*NewOrder Missing Qty key "Qty###".
#define ERR_NEWORDER_QTY_INVALID 1034
/*NewOrder Qty invalid must be
/*numeric range 1 - 99.
#define ERR_NEWORDER_SUPPW_RANGE
1035 /*New Order Supp_W value out of
/*range range = 1 -
Max Warehouses.
#define ERR_NEWORDER_ITEMID_RANGE 1036
/*New Order Item Id is out of range.
/*Range = 1 to 999999.
#define ERR_NEWORDER_QTY_RANGE 1037
/*New Order Qty is out of range.
/*Range = 1 to 99.
#define ERR_PAYMENT_DISTRICT_INVALID 1038
/*Payment District ID is invalid
/*must be 1 - 10.
#define
ERR_NEWORDER_SUPPW_WITHOUT_ITEMID 1039
/*NewOrder Supp_W field entered
/*without a
corresponding Item_Id.
#define
ERR_NEWORDER_QTY_WITHOUT_ITEMID
1040 /*NewOrder Qty entered without a
/*corresponding
Item_Id.
#define ERR_NEWORDER_NOITEMS_ENTERED
1041 /*NewOrder Blank Items between
/*items, items must
be continuous.
#define ERR_PAYMENT_MISSING_DID_KEY
1042 /*Payment missing District Key "DID".
#define ERR_PAYMENT_DISTRICT_RANGE
1043 /*Payment District Out of range,
/*range = 1 - 10.

```



```

#define ERR_PAYMENT_MISSING_CID_KEY
1044 /*Payment missing Customer Key "CID"*/
#define ERR_PAYMENT_CUSTOMER_INVALID
1045 /*PaymentCustomer data type invalid,
//must be numeric.
#define ERR_PAYMENT_MISSING_CLT 1046
//Payment missing Customer Last Name
//Key "CLT"*/
#define
ERR_PAYMENT_LAST_NAME_TO_LONG
1047 /*Payment Customer last name
// longer than 16
characters.
#define ERR_PAYMENT_CUSTOMER_RANGE
1048 /*Payment Customer ID out of range,
//must be 1 to 3000.
#define ERR_PAYMENT_CID_AND_CLT 1049
//Payment Customer ID and Last Name
//entered must be one or
other.
#define ERR_PAYMENT_MISSING_CDI_KEY 1050
//Payment missing Customer district
//key "CDI"*/
#define ERR_PAYMENT_CDI_INVALID 1051
//Payment Customer district invalid
//must be numeric.
#define ERR_PAYMENT_CDI_RANGE 1052
//Payment Customer district out of
//range must be 1 - 10.
#define ERR_PAYMENT_MISSING_CWI_KEY 1053
//Payment missing Customer Warehouse
//key "CWI"*/
#define ERR_PAYMENT_CWI_INVALID 1054
//Payment Customer Warehouse invalid
//must be numeric.
#define ERR_PAYMENT_CWI_RANGE 1055
//Payment Customer Warehouse out of
//range, 1 to Max
Warehouses.
#define ERR_PAYMENT_MISSING_HAM_KEY
1056 /*Payment missing Amount key "HAM"*/
#define ERR_PAYMENT_HAM_INVALID 1057
//Payment Amount invalid data type
//must be numeric.
#define ERR_PAYMENT_HAM_RANGE 1058
//Payment Amount out of range,
//0 - 9999.99.
#define
ERR_ORDERSTATUS_MISSING_DID_KEY
1059 /*OrderStatus missing District
//key "DID"*/
#define ERR_ORDERSTATUS_DID_INVALID
1060 /*Order Status District invalid,
//value must be numeric
1 - 10.
#define ERR_ORDERSTATUS_DID_RANGE 1061
//Order Status District out of range
//must be 1 - 10.
#define ERR_ORDERSTATUS_MISSING_CID_KEY
1062 /*OrderStatus missing Customer
//key "CID"*/
#define
ERR_ORDERSTATUS_MISSING_CLT_KEY
1063 /*OrderStatus missing Customer
//Last Name key
"CLT"*/
#define ERR_ORDERSTATUS_CLT_RANGE
1064 /*Order Status Customer last name

```

```

//longer than 16
characters.
#define ERR_ORDERSTATUS_CID_INVALID
1065 /*Order Status Customer ID invalid,
//range must be numeric
1 - 3000.
#define ERR_ORDERSTATUS_CID_RANGE 1066
//Order Status Customer ID out of
//range must be 1 - 3000.
#define ERR_ORDERSTATUS_CID_AND_CLT
1067 /*Order Status Customer ID and
//LastName entered
must be only one."
#define ERR_DELIVERY_MISSING_OCD_KEY
1068 /*Delivery missing Carrier ID key
//"OCD"*/
#define ERR_DELIVERY_CARRIER_INVALID
1069 /*Delivery Carrier ID invalid must
//be numeric 1 - 10.
#define ERR_DELIVERY_CARRIER_ID_RANGE
1070 /*Delivery Carrier ID out of range
//must be 1 - 10.
#define ERR_PAYMENT_MISSING_CLT_KEY 1071
//Payment missing Customer Last
//Name key "CLT"*/
#define TOPEND_SEND_ERROR 1072 /*TOPEND
client send error".
#define TOPEND_RECEIVE_ERROR 1073
//TOPEND client receive error".
#define ERR_TPINIT_BAD 5001 /*Tuxedo tpinit
Failed."
#define ERR_TPALLOC_BAD 5002 /*Tuxedo tpalloc
Failed."
#define ERR_TPCALL_BAD 5003 /*Tuxedo tpcall
Failed."
#endif /* TPCCERR_H */

===== tpcc.def =====

EXPORTS
GetExtensionVersion
HttpExtensionProc

===== bs-mb.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 }
};
#endif _TMDLLIMPORT
#define _TMDLLIMPORT
#endif
_TMDLLIMPORT extern struct xa_switch_t
tmnull_switch;

struct tmsvrgs_t tmsvrgs = {
NULL,
&_tmdspchtbl[0],
0,
tpsvrinit,
tpsvrdone,
_tmrunserver, /* PRIVATE */
NULL, /* RESERVED */
NULL, /* RESERVED */
NULL, /* RESERVED */
NULL /* RESERVED */
};

struct tmsvrgs_t *
#ifdef _TMPROTOTYPES
_tmgetsvrgs(void)
#else
_tmgetsvrgs()
#endif
{
tmsvrgs.xa_switch = &tmnull_switch;
return(&tmsvrgs);
}

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,
_tmgetsvrgs()));
}

===== bs-mbd.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 NEWORDER _((TPSVCINFO *));
#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 tmdsptchtbl_t_tmdsptchtbl[] = {
//  {"OPSTUXSERVER", "OPSTUXSERVER", (void
(*) _((TPSVCINFO *))) OPSTUXSERVER, 0, 0 },
//  { NULL, NULL, NULL, 0, 0 }
//};
static struct tmdsptchtbl_t_tmdsptchtbl[] = {
  {"DELIVERY", "DELIVERY", (void (*)
_((TPSVCINFO *))) DELIVERY, 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,
  &tmdsptchtbl[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 = &tmnull_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()));
}

===== bs-mbo.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 tmdsptchtbl_t_tmdsptchtbl[] = {
//  {"OPSTUXSERVER", "OPSTUXSERVER", (void
(*) _((TPSVCINFO *))) OPSTUXSERVER, 0, 0 },
//  { NULL, NULL, NULL, 0, 0 }
//};
static struct tmdsptchtbl_t_tmdsptchtbl[] = {
  {"ORDERSTATUS", "ORDERSTATUS", (void (*)
_((TPSVCINFO *))) ORDERSTATUS, 0, 0 },
  { NULL, NULL, NULL, 0, 0 }
};

#ifdef _TMDLLIMPORT
#define _TMDLLIMPORT
#endif

_TMDLLIMPORT extern struct xa_switch_t
tmnull_switch;

struct tmsvrargs_t tmsvrargs = {
  NULL,
  &tmdsptchtbl[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 = &tmnull_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()));
}

===== bs-mbp.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 tmdsptchtbl_t_tmdsptchtbl[] = {
//  {"OPSTUXSERVER", "OPSTUXSERVER", (void
(*) _((TPSVCINFO *))) OPSTUXSERVER, 0, 0 },
//  { NULL, NULL, NULL, 0, 0 }
//};
static struct tmdsptchtbl_t_tmdsptchtbl[] = {
  {"PAYMENT", "PAYMENT", (void (*)
_((TPSVCINFO *))) PAYMENT, 0, 0 },
  { NULL, NULL, NULL, 0, 0 }
};

#ifdef _TMDLLIMPORT
#define _TMDLLIMPORT
#endif

_TMDLLIMPORT extern struct xa_switch_t
tmnull_switch;

struct tmsvrargs_t tmsvrargs = {
  NULL,
  &tmdsptchtbl[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 = &tmnull_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()));
}

===== bs-mbs.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 tmdsptchtbl_t _tmdsptchtbl[] = {
// { "OPSTUXSERVER", "OPSTUXSERVER", (void (*)
// _((TPSVCINFO *))) OPSTUXSERVER, 0, 0 },
// { NULL, NULL, NULL, 0, 0 }
//};
static struct tmdsptchtbl_t _tmdsptchtbl[] = {
{ "STOCKLVL", "STOCKLVL", (void (*)
_((TPSVCINFO *))) STOCKLVL, 0, 0 },
{ NULL, NULL, NULL, 0, 0 }
};

#ifdef _TMDLLIMPORT
#define _TMDLLIMPORT
#endif

_TMDLLIMPORT extern struct xa_switch_t
tmnull_switch;

struct tmsvrargs_t tmsvrargs = {
    NULL,
    &_tmdsptchtbl[0],
    0,
    tpsvrinit,
    tpsvrdone,
    _tmrunserver, /* PRIVATE */
    NULL, /* RESERVED */
    NULL, /* RESERVED */
    NULL, /* RESERVED */
    NULL /* RESERVED */
};

struct tmsvrargs_t *
#ifdef _TMPROTOTYPES
_tmdsptchtbl(void)
#else
_tmdsptchtbl(void)
#endif
#endif

{
    tmsvrargs.xa_switch = &tmnull_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()));
}

===== tpcc.c =====

/*=====
=====+
| Copyright (c) 1997 Oracle Corp, Redwood
Shores, CA |
| All Rights Reserved |
+=====+
| FILE: TPCC.C |
| DESCRIPTION: Main module for TPCC.DLL |
| Master created: 15 Apr 97 |
*===== */

#ifdef TOPEND
#define TP_MT_SOURCE
#define STRICT
#endif

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

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

#include "tpccerr.h"
#include "tpcc_info.h"
#include "httpext.h"

#include "getval.h"

#ifdef TOPEND
#include <tp_csi.h>

```

```

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

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

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

//Terminal client id structure and interface definition
TERM Term = { 0, 0, 0, FALSE, NULL, TermInit,
TermAllocate, TermRestore, TermAdd, TermDelete };

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

"<HEAD><TITLE>Welcome To TPC-
C</TITLE></HEAD><BODY>"

"Please Identify your Warehouse and District for
this session.<BR>"

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

"<INPUT TYPE=\"hidden\" NAME=\"STATUSID\"
VALUE=\"0\">"

"<INPUT TYPE=\"hidden\" NAME=\"ERROR\"
VALUE=\"0\">"

"<INPUT TYPE=\"hidden\" NAME=\"FORMID\"
VALUE=\"1\">"

"<INPUT TYPE=\"hidden\" NAME=\"TERMIN\"
VALUE=\"-2\">"

"<INPUT TYPE=\"hidden\" NAME=\"SYNCD\"
VALUE=\"0\">"

"Warehouse ID <INPUT NAME=\"w_id\"
SIZE=4><BR>"

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

"<HR>"

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

"</FORM><BODY>"

"</HTML>";

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

static CRITICAL_SECTION CriticalSection;
static CRITICAL_SECTION
ErrorLogCriticalSection;

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

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

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

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

//can use for tuning, keep high
#define RECEIVE_THREAD_PRIORITY
THREAD_PRIORITY_TIME_CRITICAL
#define MT_LOG_ERROR {
char buf[160];\
sprintf(buf, "Problem at %d
in %s\n", __LINE__, __FILE__);
tp_system_log_text("TEthreads", buf);\
}

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

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

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

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

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

/* FUNCTION: BOOL APIENTRY DIIMain(HANDLE
hModule, DWORD ul_reason_for_call,
* LPVOID lpReserved)
*
* PURPOSE: This function is the entry point for the
DLL. This implementation
* is based on the fact that
DLL_PROCESS_ATTACH is only called from
* the inet service once.
* Connections are sent to this function as thread
attachments.
*
* ARGUMENTS: HANDLE hModule
module handle
* DWORD ul_reason_for_call reason for
call
* LPVOID lpReserved reserved for
future use
*
* RETURNS: BOOL FALSE errors
occured in initialization
* TRUE DLL successfully
initialized
*
* COMMENTS: None
*/
BOOL APIENTRY DIIMain(HANDLE hModule,
DWORD ul_reason_for_call, LPVOID lpReserved)
{
    int i;
    static SECURITY_ATTRIBUTES sa;
    static PSECURITY_DESCRIPTOR pSD;

#ifdef DEBUG_ENTRY
    DebugBreak();
#endif

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

```

```

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

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

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

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

    case DLL_THREAD_ATTACH:
        break;

    case DLL_THREAD_DETACH:
        break;

    case DLL_PROCESS_DETACH:

        if ( pSD )
            free( pSD );

        bTpccExit = TRUE;

        (*Term.Restore)();

        DeleteCriticalSection(&CriticalSection);

        DeleteCriticalSection(&ErrorLogCriticalSection);

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

        break;
    }
    return TRUE;
}

/* FUNCTION: BOOL WINAPI
GetExtensionVersion(HSE_VERSION_INFO *pVer)
*
* PURPOSE: This function is called by the inet service
when the DLL is first
    loaded.
* ARGUMENTS: HSE_VERSION_INFO *pVer
passed in structure in which to
    place expected version
    number.
*
* RETURNS: TRUE inet service
expected return value.
*
* COMMENTS: None
*/
_declspec (dllexport)
BOOL WINAPI
GetExtensionVersion(HSE_VERSION_INFO *pVer)
{
    pVer->dwExtensionVersion =
    MAKELONG(HSE_VERSION_MINOR,
    HSE_VERSION_MAJOR);
    lstrcpy(pVer->pszExtensionDesc, "TPC-C Server.",
    HSE_MAX_EXT_DLL_NAME_LEN);
    return TRUE;
}

/* FUNCTION: DWORD WINAPI
HttpExtensionProc(EXTENSION_CONTROL_BLOCK
*pECB)
*
* PURPOSE: This function is the main entry point for
the TPCC DLL. The
    inet service calls this function passing in the
    http string.
*
* ARGUMENTS: EXTENSION_CONTROL_BLOCK
*pECB structure pointer to passed in
    internet service
    information.
*
* RETURNS: DWORD HSE_STATUS_SUCCESS
connection can be dropped if error
*
HSE_STATUS_SUCCESS_AND_KEEP_CONN keep
connect valid comment sent
*
* COMMENTS: None
*/
_declspec (dllexport)
DWORD WINAPI
HttpExtensionProc(EXTENSION_CONTROL_BLOCK
*pECB)
{
    int iCmd, FormId, TermId, iSyncId;
    FILE *fp;

    static BOOL bReadRegistry = FALSE;

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

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

        fp = fopen(szTpccLogPath, "ab");

        GetLocalTime(&systemTime);

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

    //process http query
    if ( !ProcessQueryString(pECB, &iCmd, &FormId,
    &TermId, &iSyncId) )
    {
        if ( TermId < 0 )
            ErrorMessage(pECB,
            ERR_INVALID_TERMID, ERR_TYPE_WEBDLL,
            NULL, TermId, iSyncId);
        else
            ErrorMessage(pECB,
            ERR_COMMAND_UNDEFINED,
            ERR_TYPE_WEBDLL, NULL,
            TermId, iSyncId);
        return
        HSE_STATUS_SUCCESS_AND_KEEP_CONN;
    }

    if ( TermId != 0 )
    {
        if ( !IsValidTermId(TermId) )
        {
            ErrorMessage(pECB,
            ERR_INVALID_TERMID, ERR_TYPE_WEBDLL,
            NULL, TermId, iSyncId);
            return
            HSE_STATUS_SUCCESS_AND_KEEP_CONN;
        }

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

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

        //go execute http: command
    }
}

```

```

        (*DoCmd[iCmd])(pECB, FormId, TermId,
iSyncl);

        //finish up and keep connection
        return
HSE_STATUS_SUCCESS_AND_KEEP_CONN;
    }

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

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

/* FUNCTION: BOOL
ProcessQueryString(EXTENSION_CONTROL_BLOCK
*pECB, int *pCmd,
int *pFormId, int *pTermId, int
*pSyncl)
 *
 * PURPOSE: This function extracts the relevent
information out of the http
command passed in from the browser.
 *
 * ARGUMENTS: EXTENSION_CONTROL_BLOCK
*pECB structure pointer to passed in
internet service
information.
 * int *pCmd returned command
id
 * int *pFormId returned active form
client
 * int *pTermId browser is on
terminal id
 *
 * RETURNS: BOOL FALSE success
TRUE command passed in is invalid
 *
 * COMMENTS: If this is the initial connection i.e. client
is at welcome screen
 * then there will not be a terminal id or current
form id.
 * If this is the case then the pTermid and
pFormid return values are
 * undefined.
 */

BOOL
ProcessQueryString(EXTENSION_CONTROL_BLOCK
*pECB, int *pCmd, int *pFormId,

```

```

        int *pTermId, int *pSyncl)
{
    char *ptr;
    char szBuffer[25];
    char szTmp[25];
    char *dest = szBuffer;
    int i;

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

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

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

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

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

```

```

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

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

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

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

/* FUNCTION: void
PaymentForm(EXTENSION_CONTROL_BLOCK
*pECB, int iFormId, int
iTermId, int iSyncl)
 *
 * PURPOSE: This function wraps the functionality for
the TPC-C Payment Form.
 *
 * ARGUMENTS: int iFormId unused
int iTermId id of calling browser, i.e.
TERMID=
 * int iSyncl sync id of calling browser
 *
 * EXTENSION_CONTROL_BLOCK *pECB
structure pointer to passed in
 * internet service
information.

```

```

* RETURNS: None
*
* COMMENTS: None
*
*/

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

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

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

    return;
}

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

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

    return;
}

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

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

    return;
}

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

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

    return;
}

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

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

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

```

```

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

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

```

```

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

```

```

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

    WriteZString(pECB, MakeWelcomeForm());

    UNUSEDPARAM(iFormId);
    return;
}

```

```

/* FUNCTION: void
ProcessCmd(EXTENSION_CONTROL_BLOCK *pECB,
int iFormId, int
* iTermId, int iSyncl)
*
* PURPOSE: This function process the passed in
http command
*
* ARGUMENTS: int iFormId unused
* int iTermId id of calling browser, i.e.
TERMIID=
* from http command line
* int iSyncl sync id of calling browser
* EXTENSION_CONTROL_BLOCK *pECB
structure pointer to passed in
* internet service
information.
*
* RETURNS: None
*
* COMMENTS: None
*/
void ProcessCmd(EXTENSION_CONTROL_BLOCK
*pECB, int iFormId, int iTermId, int iSyncl)
{
    switch( iFormId )
    {
        case WELCOME_FORM:
            return;
        case MAIN_MENU_FORM:
            return;
        case NEW_ORDER_FORM:
            ProcessNewOrderForm(pECB, iTermId,
iSyncl);
            return;
        case PAYMENT_FORM:
            ProcessPaymentForm(pECB, iTermId,
iSyncl);
            return;
        case DELIVERY_FORM:
            ProcessDeliveryForm(pECB, iTermId,
iSyncl);
            return;
        case ORDER_STATUS_FORM:
            ProcessOrderStatusForm(pECB, iTermId,
iSyncl);
            return;
        case STOCK_LEVEL_FORM:
            ProcessStockLevelForm(pECB, iTermId,
iSyncl);
            return;
    }
}

```

```

/* FUNCTION: void
ClearCmd(EXTENSION_CONTROL_BLOCK *pECB,
int iFormId, int
* iTermId, int iSyncl)
*
* PURPOSE: This function frees all currently logged
in terminal ids.
*
* ARGUMENTS: int iFormId unused
* int iTermId id of calling browser, i.e.
TERMIID=
* from http command line
* int iSyncl sync id of calling browser

```

```

* EXTENSION_CONTROL_BLOCK *pECB
structure pointer to passed in
* internet service
information.
* RETURNS: None
*
* COMMENTS: Use this function with caution, it may
cause unpredictable
* results if existing browsers attempt to use the
web client
* without beginning at the login screen for each
client.
*/

```

```

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

```

```

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

Term.pClientData = NULL;
Term.bInIt = FALSE;

(*Term.Init)();

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

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

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

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

```

```

/* FUNCTION: void
MenuCmd(EXTENSION_CONTROL_BLOCK *pECB,
int iFormId,
* iTermId, int iSyncl)
*
* PURPOSE: This function causes an exit to the main
menu
*
* ARGUMENTS: int iFormId unused
* int iTermId id of calling browser, i.e.
TERMIID=
* from http command line
* int iSyncl sync id of calling browser

```



```

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

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

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

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

    lpbSize = strlen(szStr)+1;

    if ( bLog )
    {
        SYSTEMTIME systemTime;

        fp = fopen(szTpccLogPath, "ab");

        GetLocalTime(&systemTime);

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

        fclose(fp);
    }

#ifdef PERFORMIX

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

#else

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

#endif

    return;
}

/* FUNCTION: void
h_printf(EXTENSION_CONTROL_BLOCK *pECB,
char *format, ...)
*
* PURPOSE: This function forms a high level printf for
an HTML browser
*
* ARGUMENTS: EXTENSION_CONTROL_BLOCK
*pECB passed in structure pointer from
*
inetsrv.
*
char *format printf style format
string
*
... other arguments as required
by printf
*
style format string.
*
* RETURNS: None
*
* COMMENTS: This function is mainly used for
developmental support.
*/

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

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

    lpbSize = wsprintf(szBuff, "<HTML>%s</HTML>",
szTmp) + 1;

    (*pECB->WriteClient)(pECB->ConnID, szBuff,
&lpbSize, 0);

    return;
}

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

void ErrorMessage(EXTENSION_CONTROL_BLOCK
*pECB, int iError, int iErrorType, char *szMsg, int
iTermId, int iSyncId)
{
    int i;
    static SERRORMSG errorMsgs[] =
    {
        { ERR_SUCCESS, "Success, no error." },
        { ERR_COMMAND_UNDEFINED,
"Command undefined." },
        { ERR_NOT_IMPLEMENTED_YET, "Not
Implemented Yet." },
        { ERR_CANNOT_INIT_TERMINAL, "Cannot
initialize client connection." },
        { ERR_OUT_OF_MEMORY, "insufficient
memory." },
        { ERR_NEW_ORDER_NOT_PROCESSED,
"Cannot process new Order form." },
        { ERR_PAYMENT_NOT_PROCESSED,
"Cannot process payment form." },
        { ERR_NO_SERVER_SPECIFIED, "No
Server name specified." },
        { ERR_ORDER_STATUS_NOT_PROCESSE
D, "Cannot process order status form." },
    }
}

```

```

    { ERR_W_ID_INVALID, "Invalid Warehouse ID." },
    { ERR_CAN_NOT_SET_MAX_CONNECTIONS, "Insufficient memory to allocate # connections." },
    { ERR_NOSUCH_CUSTOMER, "No such customer." },
    { ERR_D_ID_INVALID, "Invalid District ID Must be 1 to 10." },
    { ERR_MAX_CONNECT_PARAM, "Max client connections exceeded, run install to increase." },
    { ERR_INVALID_SYNC_CONNECTION, "Invalid Terminal Sync ID." },
    { ERR_INVALID_TERMID, "Invalid Terminal ID." },
    { ERR_PAYMENT_INVALID_CUSTOMER, "Payment Form, No such Customer." },
    { ERR_SQL_OPEN_CONNECTION, "SQLOpenConnection API Failed." },
    { ERR_STOCKLEVEL_MISSING_THRESHOLD_KEY, "Stock Level missing Threshold key \"TT\"." },
    { ERR_STOCKLEVEL_THRESHOLD_INVALID_ID, "Stock Level Threshold invalid data type range = 1 - 99." },
    { ERR_STOCKLEVEL_THRESHOLD_RANGE, "Stock Level Threshold out of range, range must be 1 - 99." },
    { ERR_STOCKLEVEL_NOT_PROCESSED, "Stock Level not processed." },
    { ERR_NEWORDER_FORM_MISSING_DID, "New Order missing District key \"DID\"." },
    { ERR_NEWORDER_DISTRICT_INVALID, "New Order District ID Invalid range 1 - 10." },
    { ERR_NEWORDER_DISTRICT_RANGE, "New Order District ID out of Range. Range = 1 - 10." },
    { ERR_NEWORDER_CUSTOMER_KEY, "New Order missing Customer key \"CID\"." },
    { ERR_NEWORDER_CUSTOMER_INVALID, "New Order customer id invalid data type, range = 1 to 3000." },
    { ERR_NEWORDER_CUSTOMER_RANGE, "New Order customer id out of range, range = 1 to 3000." },
    { ERR_NEWORDER_MISSING_IID_KEY, "New Order missing Item Id key \"IID\"." },
    { ERR_NEWORDER_ITEM_BLANK_LINES, "New Order blank order lines all orders must be continuous." },
    { ERR_NEWORDER_ITEMID_INVALID, "New Order Item Id is wrong data type, must be numeric." },
    { ERR_NEWORDER_MISSING_SUPPW_KEY, "New Order missing Supp_W key \"SP##\"." },
    { ERR_NEWORDER_SUPPW_INVALID, "New Order Supp_W invalid data type must be numeric." },
    { ERR_NEWORDER_MISSING_QTY_KEY, "New Order Missing Qty key \"Qty##\"." },
    { ERR_NEWORDER_QTY_INVALID, "New Order Qty invalid must be numeric range 1 - 99." },
    { ERR_NEWORDER_SUPPW_RANGE, "New Order Supp_W value out of range - 1 to MaxWarehouses." },
    { ERR_NEWORDER_ITEMID_RANGE, "New Order Item Id is out of range. Range = 1 to 999999." },
    { ERR_NEWORDER_QTY_RANGE, "New Order Qty is out of range. Range = 1 to 99." },
    { ERR_PAYMENT_DISTRICT_INVALID, "Payment District ID is invalid must be 1 - 10." },
    { ERR_NEWORDER_SUPPW_WITHOUT_ITEMID, "New Order Supp_W field entered without a Item_Id." },
    { ERR_NEWORDER_QTY_WITHOUT_ITEMID, "New Order Qty entered without a corresponding Item_Id." },
    { ERR_NEWORDER_NOITEMS_ENTERED, "New Order Blank Items between items, items must be continuous." },
    { ERR_PAYMENT_MISSING_DID_KEY, "Payment missing district Key \"DID\"." },
    { ERR_PAYMENT_DISTRICT_RANGE, "Payment District Out of range, range = 1 - 10." },
    { ERR_PAYMENT_MISSING_CID_KEY, "Payment missing Customer Key \"CID\"." },
    { ERR_PAYMENT_CUSTOMER_INVALID, "Payment Customer data type invalid, must be numeric." },
    { ERR_PAYMENT_MISSING_CLT, "Payment missing Customer Last Name Key \"CLT\"." },
    { ERR_PAYMENT_LAST_NAME_TO_LONG, "Payment Customer last name longer than 16 characters." },
    { ERR_PAYMENT_CUSTOMER_RANGE, "Payment Customer ID out of range, must be 1 to 3000." },
    { ERR_PAYMENT_CID_AND_CLT, "Payment Customer ID and Last Name entered must be one or other." },
    { ERR_PAYMENT_MISSING_CDI_KEY, "Payment missing Customer district key \"CDI\"." },
    { ERR_PAYMENT_CDI_INVALID, "Payment Customer district invalid must be numeric." },
    { ERR_PAYMENT_CDI_RANGE, "Payment Customer district out of range must be 1 - 10." },
    { ERR_PAYMENT_MISSING_CWI_KEY, "Payment missing Customer Warehouse key \"CWI\"." },
    { ERR_PAYMENT_CWI_INVALID, "Payment Customer Warehouse invalid must be numeric." },
    { ERR_PAYMENT_CWI_RANGE, "Payment Customer Warehouse out of range, 1 to Max Warehouses." },
    { ERR_PAYMENT_MISSING_HAM_KEY, "Payment missing Amount key \"HAM\"." },
    { ERR_PAYMENT_HAM_INVALID, "Payment Amount invalid data type must be numeric." },
    { ERR_PAYMENT_HAM_RANGE, "Payment Amount out of range, 0 - 9999.99." },
    { ERR_ORDERSTATUS_MISSING_DID_KEY, "Order Status missing District key \"DID\"." },
    { ERR_ORDERSTATUS_DID_INVALID, "Order Status District invalid, value must be numeric 1 - 10." },
    { ERR_ORDERSTATUS_DID_RANGE, "Order Status District out of range must be 1 - 10." },
    { ERR_ORDERSTATUS_MISSING_CID_KEY, "Order Status missing Customer key \"CID\"." },
    { ERR_ORDERSTATUS_MISSING_CLT_KEY, "Order Status missing Customer Last Name key \"CLT\"." },
    { ERR_ORDERSTATUS_CLT_RANGE, "Order Status Customer last name longer than 16 characters." },
    { ERR_ORDERSTATUS_CID_INVALID, "Order Status Customer ID invalid, range must be numeric 1 - 3000." },
    { ERR_ORDERSTATUS_CID_RANGE, "Order Status Customer ID out of range must be 1 - 3000." },
    { ERR_ORDERSTATUS_CID_AND_CLT, "Order Status Customer ID and LastName entered must be only one." },
    { ERR_DELIVERY_MISSING_OCD_KEY, "Delivery missing Carrier ID key \"OCD\"." },
    { ERR_DELIVERY_CARRIER_INVALID, "Delivery Carrier ID invalid must be numeric 1 - 10." },
    { ERR_DELIVERY_CARRIER_ID_RANGE, "Delivery Carrier ID out of range must be 1 - 10." },
    { ERR_PAYMENT_MISSING_CLT_KEY, "Payment missing Customer Last Name key \"CLT\"." },
    { ERR_TPINIT_BAD, "Tuxedo tpinit Failed." },
    { ERR_TPALLOC_BAD, "Tuxedo tpalloc Failed." },
    { ERR_TPCALL_BAD, "Tuxedo tpcall Failed." },
    { TOPEND_SEND_ERROR, "TOPEND client send error." },
    { TOPEND_RECEIVE_ERROR, "TOPEND client receive error." },
    { 0, "" }
};

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

if (!szMsg)
    szMsg = szNoMsg;

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

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

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

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

```

```

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

    case ERR_TYPE_SQL:

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

    case ERR_TYPE_OCI:

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

    case ERR_TYPE_TUXEDO:

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

/* FUNCTION: BOOL GetKeyValue(char *pQueryString,
char *pKey, char *pValue,
* int iMax, char **pBeyondKey)
*
* PURPOSE: This function parses http formatted string
for specific key values.
*
* ARGUMENTS: char *pQueryString http string
from client browser
* char *pKey key value to look for
* char *pValue character array into which to
place key's value
* int iMax maximum length of key
value array.
* char **pBeyondKey Pointer to location
beyond key value
*
* RETURNS: BOOL FALSE key value not
found
* TRUE key value found
*
* COMMENTS: http keys are formatted either
KEY=value& or KEY=value0. This
* DLL formats TPC-C input fields in such a manner that
the keys can be
* extracted in the above manner.
*/

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

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

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

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

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

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

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

static int TermAllocate(void)
{

```

```

    Term.iAvailable += 32;
    if (!Term.pClientData)
        Term.pClientData =
(PCLIENTDATA)malloc(Term.iAvailable *
sizeof(CLIENTDATA));
    else
        Term.pClientData =
(PCLIENTDATA)realloc(Term.pClientData,
Term.iAvailable * sizeof(CLIENTDATA));

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

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

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

    int i, iCurrent, iTotConnections, iTickCount;

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

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

}
else
{
    for(i=0; i<Term.iAvailable; i++)
    {
        if (!Term.pClientData[i].inUse)
            break;
    }

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

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

    iCurrent = Term.iNext;
}

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

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

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

LeaveCriticalSection(&CriticalSection);
return iCurrent;

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

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

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

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

}

return;
}

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

BOOL Init(EXTENSION_CONTROL_BLOCK *pECB,
int iTermId, int iSynclId, char *szServer, char *szUser,
char*szPassword, char *szDatabase)
{
    char szApp[32];

```

```

#ifdef LOCAL_ALLOC
    char    buf[40];
    int     iRc;
#endif

#ifdef TOPEND

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

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

        if((i=(int)tp_ChangeToGroup((LPSTR)getenv("TP
_SYSTEM"),
        LOGON32_LOGON_INTERACTIVE,
        LOGON32_PROVIDER_DEFAULT))!=0)
        {
            if (bLog)
            {
                fp1 = fopen(szTpccLogPath,"ab");
                fprintf(fp1,"Bad status from
tp_ChangeToGroup,status=%d*\n",i);
                fclose(fp1);
            }
            MessageBox(NULL, "Failed in
tp_ChangeToGroup.",
            "Init", MB_OK | MB_ICONSTOP);
            return TRUE; //changed DMA
        }
        //initialize TOPEND

        if ((i=tp_mt_initialize(NULL,NULL,0L))!=TP_OK)
        {
            if (bLog)
            {
                fp1 = fopen(szTpccLogPath,"ab");
                fprintf(fp1,"Bad status from
tp_mt_initialize,status=%d*\n",i);
                fclose(fp1);
            }

            return TRUE; //changed DMA

        }
    } //if init
    else
    LeaveCriticalSection(&CriticalSection);

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

    if((i=client_dif= tp_csi_alloc(TP_DIF_ALL))==
NULL)
    {
        if (bLog)
        {
            // SYSTEMTIME systemTime;
            fp1 = fopen(szTpccLogPath,"ab");
            fprintf(fp1,"Bad status from
tp_csi_alloc,status=%d*\n",i);
            fclose(fp1);
        }

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

        return TRUE;
    }

    (tp_dif_structs_t*)Term.pClientData[iTermId].tp_stru
ctadr = client_dif;
    tclient_dif =
(tp_dif_structs_t*)Term.pClientData[iTermId].tp_structa
dr;

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

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

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

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

    if ((i = tp_mt_client_signon(tclient_dif-
>info,tclient_dif->client,
        inactivity_time,NULL,NULL,0,NULL)) !=
TP_OK)
    {
        if (bLog)
        {
            fp1 = fopen(szTpccLogPath,"ab");
            fprintf(fp1,"client signon
failed,status=%d,iTermId=%d,iSyncId=%d\n",
            i,iTermId,iSyncId);
            fclose(fp1);
        }

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

        return TRUE;
    }

    if ((i = tp_mt_client_receive(tclient_dif-
>info,wait_time,
        tclient_dif->output_format,tclient_dif-
>service,
        tclient_dif->location,&msglen,NULL)) !=
TP_OK)
    {
        if (bLog)
        {
            fp1 = fopen(szTpccLogPath,"ab");
            fprintf(fp1,"client receive from signon
failed,status=%d\n",i);
            fclose(fp1);
        }

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

        return TRUE;
    }

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

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

    // NEWORDER

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

        {
            FILE *fp;

            fp = fopen(szTpccLogPath,
"ab");

            fprintf(fp, ">>>> Init
Thread %d : NewOrder tpalloc failed: iRc = %d \r\n",
GetCurrentThreadId(),
iRc);

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

        return TRUE;
    }

    if (dLog)

```

```

    {
        FILE *fp;

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

//    PAYMENT

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

            fp = fopen(szTpccLogPath,
"ab");

            fprintf(fp, ">>>> Init
Thread %d : Payment tpalloc failed: iRc = %d \n",
                GetCurrentThreadId(),
iRc);

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

        return TRUE;
    }

    if (dLog)
    {
        FILE *fp;

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

//    ORDERSTATUS

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

            fp = fopen(szTpccLogPath,
"ab");

            fprintf(fp, ">>>> Init
Thread %d : OrderStatus tpalloc failed: iRc = %d \n",
                GetCurrentThreadId(),
iRc);

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

        return TRUE;
    }

    if (dLog)
    {
        FILE *fp;

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

//    DELIVERY

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

            fp = fopen(szTpccLogPath,
"ab");

            fprintf(fp, ">>>> Init
Thread %d : Delivery tpalloc failed: iRc = %d \n",
                GetCurrentThreadId(),
iRc);

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

        return TRUE;
    }

    if (dLog)
    {
        FILE *fp;

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

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

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

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

```

```

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

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

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

szPic++;
}

*szDest = 0;
return;
}

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

```

```

*
* input else FALSE
*
* RETURNS: char * A pointer to buffer
inside client
* structure where HTML form is
built.
*
* COMMENTS: The internal client buffer is created
when the terminal id is
* assigned and should not be freed except
when the client
* terminal id is no longer needed.
*/

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

if ( bInput )
{
strcpy(szForm,
"<HTML><HEAD><TITLE>TPC-C Stock
Level</TITLE></HEAD>\n
<FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">\n
<INPUT TYPE=\"hidden\" NAME=\"PI\" VALUE=\"\">\n
<INPUT TYPE=\"hidden\" NAME=\"STATUSID\"
VALUE=\"0\">\n
<INPUT TYPE=\"hidden\" NAME=\"ERROR\"
VALUE=\"0\">\n
<INPUT TYPE=\"hidden\" NAME=\"FORMID\"
VALUE=\"7\">); // STOCK_LEVEL_FORM = 7

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

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

Term.pClientData[iTermId].trans.stocklevelData.s
toin.w_id,

Term.pClientData[iTermId].trans.stocklevelData.s
toin.d_id,

Term.pClientData[iTermId].trans.stocklevelData.s
toin.threshold,

Term.pClientData[iTermId].trans.stocklevelData.s
toout.low_stock);

strcat(szForm, "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..NewOrder..\">\n
<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..Payment..\">\n
<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..Delivery..\">\n
<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..Order-Status..\">\n
<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..Stock-Level..\">\n
<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..Exit..\">\n
</FORM></HTML>");
}

return szForm;
}

strcat(szForm, "Stock Level Threshold:
<INPUT NAME=\"TT\" SIZE=2><BR><BR>\n
low stock: <BR><HR><INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"Process\">\n
<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"Menu\"></FORM></HTML>");
}
else // Not bInput
{

```

```

strcpy(szForm,
"<HTML><HEAD><TITLE>TPC-C Stock
Level</TITLE></HEAD>\n
<FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">\n
<INPUT TYPE=\"hidden\" NAME=\"STATUSID\"
VALUE=\"0\">\n
<INPUT TYPE=\"hidden\" NAME=\"ERROR\"
VALUE=\"0\">\n
<INPUT TYPE=\"hidden\" NAME=\"FORMID\"
VALUE=\"7\">); // STOCK_LEVEL_FORM = 7

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

strcat(szForm, "<PRE>
Stock-Level<BR>");

wsprintf(szForm+strlen(szForm),
"Warehouse: %4.4d District: %2.2d<BR><BR>\n
Stock Level Threshold: %2.2d<BR><BR>low
stock: %3.3d</PRE><BR><HR>",

Term.pClientData[iTermId].trans.stocklevelData.s
toin.w_id,

Term.pClientData[iTermId].trans.stocklevelData.s
toin.d_id,

Term.pClientData[iTermId].trans.stocklevelData.s
toin.threshold,

Term.pClientData[iTermId].trans.stocklevelData.s
toout.low_stock);

strcat(szForm, "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..NewOrder..\">\n
<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..Payment..\">\n
<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..Delivery..\">\n
<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..Order-Status..\">\n
<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..Stock-Level..\">\n
<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..Exit..\">\n
</FORM></HTML>");
}

return szForm;
}

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

```

```

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

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

    strcat(szForm, "<INPUT TYPE='hidden'"
NAME='STATUSID' VALUE='0'>");
    strcat(szForm, "<INPUT TYPE='hidden'"
NAME='ERROR' VALUE='0'>");
    sprintf(szForm+strlen(szForm), "<INPUT
TYPE='hidden' NAME='TERMID' VALUE='%d'>",
iTermId);
    sprintf(szForm+strlen(szForm), "<INPUT
TYPE='hidden' NAME='SYNCID' VALUE='%d'>",
iSyncId);
    sprintf(szForm+strlen(szForm), "<INPUT
TYPE='hidden' NAME='FORMID' VALUE='%d'>",
MAIN_MENU_FORM);
    strcat(szForm, "<INPUT TYPE='submit'"
NAME='CMD' VALUE='..NewOrder..'>"
        "<INPUT TYPE='submit'"
NAME='CMD'VALUE='..Payment..'>"
        "<INPUT TYPE='submit'"
NAME='CMD'VALUE='..Delivery..'>"
        "<INPUT TYPE='submit'"
NAME='CMD'VALUE='..Order-Status..'>"
        "<INPUT TYPE='submit'"
NAME='CMD'VALUE='..Stock-Level..'>"
        "<INPUT TYPE='submit'"
NAME='CMD'VALUE='..Exit..'>"
        "</FORM>"
        "</HTML>");
    return szForm;
}

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

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

```

```

/* FUNCTION: char *MakeNewOrderForm(int iTermId,
BOOL blnput, BOOL bValid, char *execution_status)
*
* PURPOSE: This function makes/presents the HTML
based New Order Form
*
* ARGUMENTS: int iTermId client browser
terminal id
* int iSyncId client browser sync id
* BOOL blnput TRUE if form is being
constructed for
* BOOL bValid TRUE if NeworderData
valid, ELSE FALSE
* effects output only
*
* RETURNS: char * A pointer to buffer
inside client
* structure where HTML form is
built.
*
* COMMENTS: The internal client buffer is created
when the terminal id is
* assigned and should not be freed except
when the client
* terminal id is no longer needed.
*/

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

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

    Term.pClientData[iTermId].trans.newOrderData.ne
win.w_id = Term.pClientData[iTermId].w_id;

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

        sprintf(szForm+strlen(szForm), "<INPUT
TYPE='hidden' NAME='TERMID' VALUE='%d'>"
        "<INPUT TYPE='hidden' NAME='SYNCID'"
        "VALUE='%d'>"
        "<PRE>
New Order<BR>"
        "Warehouse: %4.4d District: <INPUT NAME='DID'"
        "SIZE=1>
Date:<BR>"
        "Customer: <INPUT NAME='CID'" SIZE=4> Name:
Credit: %Disc:<BR>"

```

```

Order Number:      Number of Lines:      W_tax:
D_tax:<BR><BR>\
Supp_W Item_Id Item Name      Qty Stock
B/G Price Amount<BR>,"
                                iTermId,
                                iSyncId,

    Term.pClientData[iTermId].trans.newOrderData.n
ewin.w_id);

    strcat(szForm, " <INPUT
NAME='SP00'" SIZE=4> <INPUT NAME='IID00'"
SIZE=6>
                                <INPUT NAME='Qty00'"
SIZE=1><BR>\
                                <INPUT NAME='SP01'" SIZE=4> <INPUT
NAME='IID01'" SIZE=6>
                                <INPUT
NAME='Qty01'" SIZE=1><BR>\
                                <INPUT NAME='SP02'" SIZE=4> <INPUT
NAME='IID02'" SIZE=6>
                                <INPUT
NAME='Qty02'" SIZE=1><BR>\
                                <INPUT NAME='SP03'" SIZE=4> <INPUT
NAME='IID03'" SIZE=6>
                                <INPUT
NAME='Qty03'" SIZE=1><BR>\
                                <INPUT NAME='SP04'" SIZE=4> <INPUT
NAME='IID04'" SIZE=6>
                                <INPUT
NAME='Qty04'" SIZE=1><BR>\
                                <INPUT NAME='SP05'" SIZE=4> <INPUT
NAME='IID05'" SIZE=6>
                                <INPUT
NAME='Qty05'" SIZE=1><BR>\
                                <INPUT NAME='SP06'" SIZE=4> <INPUT
NAME='IID06'" SIZE=6>
                                <INPUT
NAME='Qty06'" SIZE=1><BR>\
                                <INPUT NAME='SP07'" SIZE=4> <INPUT
NAME='IID07'" SIZE=6>
                                <INPUT
NAME='Qty07'" SIZE=1><BR>\
                                <INPUT NAME='SP08'" SIZE=4> <INPUT
NAME='IID08'" SIZE=6>
                                <INPUT
NAME='Qty08'" SIZE=1><BR>\
                                <INPUT NAME='SP09'" SIZE=4> <INPUT
NAME='IID09'" SIZE=6>
                                <INPUT
NAME='Qty09'" SIZE=1><BR>\
                                <INPUT NAME='SP10'" SIZE=4> <INPUT
NAME='IID10'" SIZE=6>
                                <INPUT
NAME='Qty10'" SIZE=1><BR>\
                                <INPUT NAME='SP11'" SIZE=4> <INPUT
NAME='IID11'" SIZE=6>
                                <INPUT
NAME='Qty11'" SIZE=1><BR>\
                                <INPUT NAME='SP12'" SIZE=4> <INPUT
NAME='IID12'" SIZE=6>
                                <INPUT
NAME='Qty12'" SIZE=1><BR>\
                                <INPUT NAME='SP13'" SIZE=4> <INPUT
NAME='IID13'" SIZE=6>
                                <INPUT
NAME='Qty13'" SIZE=1><BR>\
                                <INPUT NAME='SP14'" SIZE=4> <INPUT
NAME='IID14'" SIZE=6>
                                <INPUT
NAME='Qty14'" SIZE=1><BR>\
Execution Status:
Total:<BR><HR>\
                                <INPUT TYPE='submit'" NAME='CMD'"
VALUE='Process'>\
                                <INPUT TYPE='submit'" NAME='CMD'"
VALUE='Menu'>\
                                </FORM></HTML>");
}
else // not blnput
{
    if (bValid)

```



```

    {
//      FormatHTMLString(szName,
Term.pClientData[iTermId].NewOrderData.c_last, 16),
//      FormatHTMLString(szCredit,
Term.pClientData[iTermId].NewOrderData.c_credit, 2);

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

sprintf(szForm+strlen(szForm), "<INPUT
TYPE='hidden' NAME='TERMD' VALUE='%d'>
<INPUT TYPE='hidden' NAME='SYNCID'
VALUE='%d'>
<PRE>
        New Order<BR>
Warehouse: %4.4d District: %2.2d
Date: %s <BR>
Customer: %4.4d Name: %s
Credit: %2.2s %%Disc: %5.2f <BR>
Order Number: %8.8d Number of Lines: %2.2d
W_tax: %5.2f D_tax: %5.2f <BR>
Supp_W Item_Id Item Name Qty Stock
B/G Price Amount<BR>",
        iTermId,
        iSyncl,

        Term.pClientData[iTermId].trans.newOrderData.n
ewin.w_id,

        Term.pClientData[iTermId].trans.newOrderData.n
ewin.d_id,

        Term.pClientData[iTermId].trans.newOrderData.n
ewout.o_entry_d,

        Term.pClientData[iTermId].trans.newOrderData.n
ewin.c_id,

        Term.pClientData[iTermId].trans.newOrderData.n
ewout.c_last,

        Term.pClientData[iTermId].trans.newOrderData.n
ewout.c_credit,

        Term.pClientData[iTermId].trans.newOrderData.n
ewout.c_discount * 100,

        Term.pClientData[iTermId].trans.newOrderData.n
ewout.o_id,

        Term.pClientData[iTermId].trans.newOrderData.n
ewout.o_ol_cnt,

        Term.pClientData[iTermId].trans.newOrderData.n
ewout.w_tax * 100,

        Term.pClientData[iTermId].trans.newOrderData.n
ewout.d_tax * 100);

// We could make 0-4 be hard coded in one sprintf,
since we know that there is at least 5

// order lines, or is this a benchmark special??
// etw
        for(i=0;
i<Term.pClientData[iTermId].trans.newOrderData.newo
ut.o_ol_cnt; i++)
        {
//      FormatHTMLString(szName,
Term.pClientData[iTermId].NewOrderData.Ol[i].ol_i_na
me, 24);

                sprintf(szForm+strlen(szForm),
" %4.4d %6.6d %24.24s %2.2d %3.3d %1c
$%6.2f $%7.2f <BR>",

                Term.pClientData[iTermId].trans.newOrderData.n
ewin.ol_supply_w_id[i],

                Term.pClientData[iTermId].trans.newOrderData.n
ewin.ol_i_id[i],

                Term.pClientData[iTermId].trans.newOrderData.n
ewout.i_name[i],

                Term.pClientData[iTermId].trans.newOrderData.o
ld_quantity[i],

                Term.pClientData[iTermId].trans.newOrderData.n
ewout.s_quantity[i],

                Term.pClientData[iTermId].trans.newOrderData.n
ewout.brand_generic[i],

                Term.pClientData[iTermId].trans.newOrderData.n
ewout.i_price[i],

                Term.pClientData[iTermId].trans.newOrderData.n
ewout.ol_amount[i] );
        }
        for(; i<15; i++)
                strcat(szForm, "<BR>");

                sprintf(szForm+strlen(szForm),
"Execution Status: %24.24s Total: $%8.2f ",
                execution_status,

                Term.pClientData[iTermId].trans.newOrderData.n
ewout.total_amount);

                strcat(szForm, "</PRE><HR><BR>
<INPUT TYPE='submit' NAME='CMD'
VALUE='..NewOrder..'>
<INPUT TYPE='submit' NAME='CMD'
VALUE='..Payment..'>
<INPUT TYPE='submit' NAME='CMD'
VALUE='..Delivery..'>
<INPUT TYPE='submit' NAME='CMD'
VALUE='..Order-Status..'>
<INPUT TYPE='submit' NAME='CMD'
VALUE='..Stock-Level..'>
<INPUT TYPE='submit' NAME='CMD'
VALUE='..Exit..'>
</FORM></HTML>");

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

                strcpy(szForm, "<HTML><HEAD><TITLE>TPC-C
New Order</TITLE>
</HEAD><BODY><FORM ACTION='tpcc.dll'
METHOD='GET'>
<INPUT TYPE='hidden' NAME='STATUSID'
VALUE='1'>
<INPUT TYPE='hidden' NAME='ERROR'
VALUE='0'>
<INPUT TYPE='hidden' NAME='FORMID'
VALUE='3'>"; // NEW_ORDER_FORM = 3

                sprintf(szForm+strlen(szForm), "<INPUT
TYPE='hidden' NAME='TERMD' VALUE='%d'>
<INPUT TYPE='hidden' NAME='SYNCID'
VALUE='%d'>
<PRE>
        New Order<BR>
Warehouse: %4.4d District: %2.2d
Date: <BR>
Customer: %4.4d Name: %s
Credit: %s %%Disc: <BR>
Order Number: %8.8d Number of Lines:
W_tax: D_tax:<BR><BR>",
                iTermId,
                iSyncl,

                Term.pClientData[iTermId].trans.newOrderData.n
ewin.w_id,

                Term.pClientData[iTermId].trans.newOrderData.n
ewin.d_id,

                Term.pClientData[iTermId].trans.newOrderData.n
ewin.c_id,

                Term.pClientData[iTermId].trans.newOrderData.n
ewout.c_last,

                Term.pClientData[iTermId].trans.newOrderData.n
ewout.c_credit,

                Term.pClientData[iTermId].trans.newOrderData.n
ewout.c_discount * 100,

                Term.pClientData[iTermId].trans.newOrderData.n
ewout.o_id,

                Term.pClientData[iTermId].trans.newOrderData.n
ewout.o_ol_cnt,

                Term.pClientData[iTermId].trans.newOrderData.n
ewout.w_tax * 100,

                Term.pClientData[iTermId].trans.newOrderData.n
ewout.d_tax * 100);

                sprintf(szForm+strlen(szForm), "Execution
Status: %24.24s Total:",
                execution_status);

                strcat(szForm, "</PRE><HR><BR>
<INPUT TYPE='submit' NAME='CMD'
VALUE='..NewOrder..'>
<INPUT TYPE='submit' NAME='CMD'
VALUE='..Payment..'>
<INPUT TYPE='submit' NAME='CMD'
VALUE='..Delivery..'>
<INPUT TYPE='submit' NAME='CMD'
VALUE='..Order-Status..'>
<INPUT TYPE='submit' NAME='CMD'
VALUE='..Stock-Level..'>
");
        }
    }
}

```

```

<INPUT TYPE="submit" NAME="CMD"
VALUE="Exit.">
</FORM></HTML>";
    }
}
return szForm;
}

/* FUNCTION: char *MakePaymentForm(int iTermId, int
iSyncl, BOOL bInput)
*
* PURPOSE: This function has the Make Payment
HTTP form
*
* ARGUMENTS: int iTermId client browser terminal
id
* int iSyncl client browser sync id
* BOOL bInput TRUE if form is being
constructed for input
* else FALSE
*
* RETURNS: char * A pointer to buffer inside client
structure where HTML
form is built.
*
* COMMENTS: The internal client buffer is created
when the terminal id is
* assigned and should not be freed except when
the client terminal
* id is no longer needed.
*/
static char *MakePaymentForm(int iTermId, int iSyncl,
BOOL bInput)
{
    char *szForm;
    char *ptr;
    char szTmp[64];
    char szW_Zip[26];
    char szD_Zip[26];
    char szC_Zip[26];
    char szC_Phone[26];
    // char szTmpStr1[122];
    // char szTmpStr2[122];
    // char szTmpStr3[122];
    // char szTmpStr4[122];
    int i;
    int l;
    char *szZipPic = "XXXXX-XXXX";

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

    Term.pClientData[iTermId].trans.paymentData.payi
n.w_id = Term.pClientData[iTermId].w_id;

    if ( bInput )
    {
        strcpy(szForm,
"<HTML><HEAD><TITLE>TPC-C
Payment</TITLE></HEAD><BODY>\
<FORM ACTION="tpcc.dll" METHOD="GET">\
<INPUT TYPE="hidden" NAME="PI" VALUE="1">\
<INPUT TYPE="hidden" NAME="STATUSID"
VALUE="0">\
<INPUT TYPE="hidden" NAME="ERROR"
VALUE="0">

```

```

<INPUT TYPE="hidden" NAME="FORMID"
VALUE="4">"); // PAYMENT_FORM = 4

        sprintf(szForm+strlen(szForm), "<INPUT
TYPE="hidden" NAME="TERMD" VALUE="%d">\
<INPUT TYPE="hidden" NAME="SYNCID"
VALUE="%d">\
<PRE>
                Payment<BR>\
Date:<BR><BR>\
Warehouse: %4.4d",
                iTermId,
                iSyncl,
                Term.pClientData[iTermId].trans.paymentData.pa
yin.w_id);

        strcat(szForm,
"
District: <INPUT NAME="DID"
SIZE=1><BR><BR><BR><BR><BR>\
Customer: <INPUT NAME="CID"
SIZE=4>\
Cust-Warehouse: <INPUT NAME="CWI"
SIZE=4> \
Cust-District: <INPUT NAME="CDI"
SIZE=1><BR>\
Name: <INPUT NAME="CLT"
SIZE=16>
                Since:<BR>\
                Credit:<BR>\
                Disc:<BR>\
Phone:<BR><BR>\
Amount Paid: $<INPUT NAME="HAM"
SIZE=7>
New Cust Balance:<BR>\
Credit Limit:<BR><BR>Cust-Data:
<BR><BR><BR><BR></PRE><HR>\
<INPUT TYPE="submit" NAME="CMD"
VALUE="Process"><INPUT TYPE="submit"
NAME="CMD" VALUE="Menu">\
</BODY></FORM></HTML>");
    }
    else // Not bInput
    {
        // FormatHTMLString(szTmpStr1,
Term.pClientData[iTermId].PaymentData.w_street_1,
20);
        // FormatHTMLString(szTmpStr2,
Term.pClientData[iTermId].PaymentData.d_street_1,
20);
        // FormatHTMLString(szTmpStr3,
Term.pClientData[iTermId].PaymentData.w_street_2,
20);
        // FormatHTMLString(szTmpStr4,
Term.pClientData[iTermId].PaymentData.d_street_2,
20);

        FormatString(szW_Zip, szZipPic,
Term.pClientData[iTermId].trans.paymentData.payout.
w_zip);
        FormatString(szD_Zip, szZipPic,
Term.pClientData[iTermId].trans.paymentData.payout.d
_zip);
        // FormatHTMLString(szTmpStr5,
Term.pClientData[iTermId].PaymentData.w_city, 20);
        // FormatHTMLString(szTmpStr6,
Term.pClientData[iTermId].PaymentData.w_state, 2);
        // FormatHTMLString(szTmpStr7,
Term.pClientData[iTermId].PaymentData.d_city, 20);

```

```

        // FormatHTMLString(szTmpStr8,
Term.pClientData[iTermId].PaymentData.d_state, 2);
        // FormatHTMLString(szTmpStr9,
Term.pClientData[iTermId].PaymentData.c_first, 16);
        // FormatHTMLString(szTmpStr10,
Term.pClientData[iTermId].PaymentData.c_middle, 2);
        // FormatHTMLString(szTmpStr11,
Term.pClientData[iTermId].PaymentData.c_last, 16);
        // FormatHTMLString(szTmpStr12,
Term.pClientData[iTermId].PaymentData.c_street_1,
20);
        // FormatHTMLString(szTmpStr13,
Term.pClientData[iTermId].PaymentData.c_credit, 2);
        // FormatHTMLString(szTmpStr14,
Term.pClientData[iTermId].PaymentData.d_street_2,
20);
        FormatString(szC_Zip, szZipPic,
Term.pClientData[iTermId].trans.paymentData.payout.c
_zip);
        FormatString(szC_Phone, "XXXXXX-XXX-
XXX-
XXXX", Term.pClientData[iTermId].trans.paymentData.p
ayout.c_phone);
        // FormatHTMLString(szTmpStr15,
Term.pClientData[iTermId].PaymentData.c_city, 20);
        // FormatHTMLString(szTmpStr16,
Term.pClientData[iTermId].PaymentData.c_state, 2);

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

        sprintf(szForm+strlen(szForm),
"<INPUT TYPE="hidden" NAME="TERMD"
VALUE="%d">\
<INPUT TYPE="hidden" NAME="SYNCID"
VALUE="%d">\
<PRE>
                Payment<BR>\
Date: %19s<BR><BR>\
Warehouse: %4.4d
District: %2.2d<BR>\
%20s %20s<BR>\
%20s %20s<BR>\
%20s %2s %10.10s %20s %-
2s %10.10s<BR><BR>\
Customer: %4.4d Cust-Warehouse: %4.4d Cust-
District: %2.2d<BR>\
Name: %16s %2s %16s Since: %10s<BR>\
%20s Credit: %s<BR>\
%20s %Disc: %5.2f<BR>\
%20s %2s %10.10s Phone: %-
19.19s<BR><BR>\
Amount Paid: $%7.2f
New Cust Balance:
$%14.2f<BR>\
Credit Limit: $%13.2f<BR><BR>",
                iTermId,

```

<pre> iSyncId, Term.pClientData[iTermId].trans.paymentData.pa yout.h_date, Term.pClientData[iTermId].trans.paymentData.pa yin.w_id, Term.pClientData[iTermId].trans.paymentData.pa yin.d_id, Term.pClientData[iTermId].trans.paymentData.pa yout.w_street_1, Term.pClientData[iTermId].trans.paymentData.pa yout.d_street_1, Term.pClientData[iTermId].trans.paymentData.pa yout.w_street_2, Term.pClientData[iTermId].trans.paymentData.pa yout.d_street_2, Term.pClientData[iTermId].trans.paymentData.pa yout.w_city, Term.pClientData[iTermId].trans.paymentData.pa yout.w_state, szW_Zip, Term.pClientData[iTermId].trans.paymentData.pa yout.d_city, Term.pClientData[iTermId].trans.paymentData.pa yout.d_state, szD_Zip, Term.pClientData[iTermId].trans.paymentData.pa yout.c_id, Term.pClientData[iTermId].trans.paymentData.pa yin.c_w_id, Term.pClientData[iTermId].trans.paymentData.pa yin.c_d_id, Term.pClientData[iTermId].trans.paymentData.pa yout.c_first, Term.pClientData[iTermId].trans.paymentData.pa yout.c_middle, Term.pClientData[iTermId].trans.paymentData.pa yout.c_last, Term.pClientData[iTermId].trans.paymentData.pa yout.c_since, Term.pClientData[iTermId].trans.paymentData.pa yout.c_street_1, Term.pClientData[iTermId].trans.paymentData.pa yout.c_credit, Term.pClientData[iTermId].trans.paymentData.pa yout.c_street_2, Term.pClientData[iTermId].trans.paymentData.pa yout.c_discount*100, </pre>	<pre> Term.pClientData[iTermId].trans.paymentData.pa yout.c_city, Term.pClientData[iTermId].trans.paymentData.pa yout.c_state, szC_Zip, szC_Phone, (float)(Term.pClientData[iTermId].trans.payment Data.payin.h_amount)/100.0, Term.pClientData[iTermId].trans.paymentData.pa yout.c_balance, Term.pClientData[iTermId].trans.paymentData.pa yout.c_credit_lim); // This part can probably be made a little more efficient //etw ptr = Term.pClientData[iTermId].trans.paymentData.payout.c _credit; if (*ptr == 'B' && *(ptr+1) == 'C') { ptr = Term.pClientData[iTermId].trans.paymentData.payout.c _data; l = strlen(ptr) / 50; for(j=0; j<4; j++, ptr += 50) { if (i <= l) UtilStrCpy(szTmp, ptr, 50); else szTmp[0] = 0; if (lj) FormatHTMLString(szTmpStr1, szTmp, 50); wsprintf(szForm+strlen(szForm), "Cust-Data: %- 50s
", szTmp); } else { FormatHTMLString(szTmpStr1, szTmp, 50); wsprintf(szForm+strlen(szForm), " %- 50s
", szTmp); } } else strcat(szForm, "Cust-Data:

"); strcat(szForm, "</PRE><HR>
\n <INPUT TYPE='submit' NAME='CMD1' VALUE='..NewOrder..'>\n <INPUT TYPE='submit' NAME='CMD1' VALUE='..Payment..'>\n <INPUT TYPE='submit' NAME='CMD1' VALUE='..Delivery..'>\n <INPUT TYPE='submit' NAME='CMD1' VALUE='..Order-Status..'>\n <INPUT TYPE='submit' NAME='CMD1' VALUE='..Stock-Level..'>\n </pre>	<pre> <INPUT TYPE='submit' NAME='CMD1' VALUE='..Exit..'>\n </BODY></FORM></HTML>"); } return szForm; } /* FUNCTION: char *MakeOrderStatusForm(int iTermId, int iSyncId, BOOL bInput) * * PURPOSE: This function makes the HTML based Order Status Form * * ARGUMENTS: int iTermId client browser terminal id * int iSyncId client browser sync id * BOOL bInput TRUE if form is being constructed for input * else FALSE * * RETURNS: char * A pointer to buffer inside client structure * where HTML form is built. * * COMMENTS: The internal client buffer is created when the terminal id is * assigned and should not be freed except when the client * terminal id is no longer needed. */ static char *MakeOrderStatusForm(int iTermId, int iSyncId, BOOL bInput) { char *szForm; // char c_first[98]; // char c_middle[14]; // char c_last[98]; int i; szForm = (char *)Term.pClientData[iTermId].szBuffer; Term.pClientData[iTermId].trans.orderStatusData.or din.w_id = Term.pClientData[iTermId].w_id; if (bInput) { strcpy(szForm, "<HTML><HEAD><TITLE>TPC-C Order- Status</TITLE></HEAD><BODY>\n <FORM ACTION='tpcc.dll' METHOD='GET'>\n <INPUT TYPE='hidden' NAME='PI*' VALUE=''\n>\n <INPUT TYPE='hidden' NAME='STATUSID'\n VALUE='0'\n>\n <INPUT TYPE='hidden' NAME='ERROR'\n VALUE='0'\n>\n <INPUT TYPE='hidden' NAME='FORMID'\n VALUE='6'\n>"; // ORDER_STATUS_FORM = 6 wsprintf(szForm+strlen(szForm), "<INPUT TYPE='hidden' NAME='TERMINID' VALUE='"%d">\n <INPUT TYPE='hidden' NAME='SYNCID'\n VALUE='"%d">";", iTermId, iSyncId); </pre>
---	---	--

```

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

        strcat(szForm, "District: <INPUT
NAME=\"DID\" SIZE=1><BR>\
Customer: <INPUT NAME=\"CID\" SIZE=4> Name:
<INPUT NAME=\"CLT\" SIZE=23><BR>\
Cust-Balance:<BR><BR>\
Order-Number:      Entry-Date:      Carrier-
Number:<BR>\
Supply-W Item-Id Qty Amount Delivery-
Date<BR></PRE>\
<HR><INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"Process\"><INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"Menu\">\
</BODY></FORM></HTML>");
    }
    else // Not bInput
    {
        strcpy(szForm,
"<HTML><HEAD><TITLE>TPC-C Order-
Status</TITLE></HEAD><BODY>\
<FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">\
<INPUT TYPE=\"hidden\" NAME=\"STATUSID\"
VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"ERROR\"
VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"FORMID\"
VALUE=\"6\">); // ORDER_STATUS_FORM = 6

        wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\" NAME=\"TER MID\" VALUE=\"%d\">\
<INPUT TYPE=\"hidden\" NAME=\"SY NCID\"
VALUE=\"%d\">",
                iTermId,
                iSyncId);

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

        sprintf(szForm+strlen(szForm),
"Warehouse: %4.4d District: %2.2d<BR>\
Customer: %4.4d Name: %-16s %-2s %-16s<BR>\
Cust-Balance: %9.2f<BR><BR>\
Order-Number: %8.8d Entry-Date: %19s
Carrier-Number: %2.2d<BR>",
                Term.pClientData[iTermId].trans.orderStatusData.
ordin.w_id,
                Term.pClientData[iTermId].trans.orderStatusData.
ordin.d_id,
                Term.pClientData[iTermId].trans.orderStatusData.
ordout.c_id,
                Term.pClientData[iTermId].trans.orderStatusData.
ordout.c_first,
                Term.pClientData[iTermId].trans.orderStatusData.
ordout.c_middle,
                Term.pClientData[iTermId].trans.orderStatusData.
ordout.c_last);

        Term.pClientData[iTermId].trans.orderStatusData.
ordout.c_balance,
                Term.pClientData[iTermId].trans.orderStatusData.
ordout.o_id,
                Term.pClientData[iTermId].trans.orderStatusData.
ordout.o_entry_d,
                Term.pClientData[iTermId].trans.orderStatusData.
ordout.o_carrier_id);

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

        for(i=0;
i<Term.pClientData[iTermId].trans.orderStatusData.ord
out.o_o_cnt; i++)
        {
            sprintf(szForm+strlen(szForm),
"%4.4d %6.6d %2.2d %9.2f %10s
<BR>",
                Term.pClientData[iTermId].trans.orderStatusData.
ordout.ol_supply_w_id[i],
                Term.pClientData[iTermId].trans.orderStatusData.
ordout.ol_i_id[i],
                Term.pClientData[iTermId].trans.orderStatusData.
ordout.ol_quantity[i],
                Term.pClientData[iTermId].trans.orderStatusData.
ordout.ol_amount[i],
                Term.pClientData[iTermId].trans.orderStatusData.
ordout.ol_delivery_d[i]);
        }

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

    return szForm;
}

/* FUNCTION: char *MakeDeliveryForm(int iTermId, int
iSyncId, BOOL bInput, char *execution_status)
*
* PURPOSE: This function puts out the HTML for
the delivery form
*
* ARGUMENTS: int iTermId client browser terminal
id
*             int iSyncId client browser sync id
*             BOOL bInput TRUE if form is being
constructed for input
*             else FALSE
*
* RETURNS: char * A pointer to buffer inside client
structure where HTML
form is built.
*
* COMMENTS: The internal client buffer is created
when the terminal id is
assigned and should not be freed except when
the client terminal
id is no longer needed.
*/
static char *MakeDeliveryForm(int iTermId, int iSyncId,
BOOL bInput, char *execution_status)
{
    char *szForm;
    szForm = (char
*)Term.pClientData[iTermId].szBuffer;
    Term.pClientData[iTermId].trans.deliveryData.delin.
w_id = Term.pClientData[iTermId].w_id;

    if ( bInput )
    {
        strcpy( szForm,
"<HTML><HEAD><TITLE>TPC-C
Delivery</TITLE></HEAD><BODY>\
<FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">\
<INPUT TYPE=\"hidden\" NAME=\"PI\" VALUE=\"1\">\
<INPUT TYPE=\"hidden\" NAME=\"STATUSID\"
VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"ERROR\"
VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"FORMID\"
VALUE=\"5\">); // DELIVERY_FORM = 5

        wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\" NAME=\"TER MID\" VALUE=\"%d\">\
<INPUT TYPE=\"hidden\" NAME=\"SY NCID\"
VALUE=\"%d\">",
                iTermId,
                iSyncId);

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

        wsprintf(szForm+strlen(szForm),
"Warehouse: %4.4d<BR><BR>",
                Term.pClientData[iTermId].trans.deliveryData.delin.w_
id);

        sprintf(szForm+strlen(szForm), "Carrier Number: <INPUT
NAME=\"OCD\" SIZE=1><BR><BR>\
Execution Status:<BR></PRE>\
<HR><INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"Process\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"Menu\">\
</BODY></FORM></HTML>");
    }
    else // Not bInput
    {
        strcpy( szForm,
"<HTML><HEAD><TITLE>TPC-C
Delivery</TITLE></HEAD><BODY>\
");
    }
}

```

```

<FORM ACTION="tpcc.dll" METHOD="GET">\
<INPUT TYPE="hidden" NAME="STATUSID"
VALUE="0">\
<INPUT TYPE="hidden" NAME="ERROR"
VALUE="0">\
<INPUT TYPE="hidden" NAME="FORMID"
VALUE="5">); // DELIVERY_FORM = 5

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

            iTermd,

            iSyncl);

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

        wsprintf(szForm+strlen(szForm),
"Warehouse: %4.4d<BR><BR>",
Term.pClientData[iTermd].trans.deliveryData.delin.w_i
d);

        wsprintf(szForm+strlen(szForm), "Carrier
Number: %2.2d<BR><BR>\
Execution Status: %25.25s<BR></PRE>",

            Term.pClientData[iTermd].trans.deliveryData.delin.o_carrier_id,

            execution_status);

        strcat(szForm, "<HR><INPUT
TYPE="submit" NAME="CMD"
VALUE="..NewOrder..">\
<INPUT TYPE="submit" NAME="CMD"
VALUE="..Payment..">\
<INPUT TYPE="submit" NAME="CMD"
VALUE="..Delivery..">\
<INPUT TYPE="submit" NAME="CMD"
VALUE="..Order-Status..">\
<INPUT TYPE="submit" NAME="CMD"
VALUE="..Stock-Level..">\
<INPUT TYPE="submit" NAME="CMD"
VALUE="..Exit..">\
</BODY></FORM></HTML>");

    }

    return szForm;
}

/* FUNCTION: void UtilStrCpy(char * pDest, char * pSrc,
int n)
*
* PURPOSE: This function copies n characters from
string pSrc to pDst and
* places a null character at the end of the
destination string.
*
* ARGUMENTS: char *pDest destination string
pointer
* char *pSrc source string pointer
* int n number of characters to copy

```

```

*
* RETURNS: None
*
* COMMENTS: Unlike strncpy this function ensures
that the result string is
* always null terminated.
*
*/

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

    return;
}

/* FUNCTION: void
ProcessNewOrderForm(EXTENSION_CONTROL_BLOCK *pECB, int
*
* iTermd, int iSyncl)
*
* PURPOSE: This function gets and validates the input
data from the new order
* form filling in the required input variables. it then calls
the SQLNewOrder
* transaction, constructs the output form and writes it
back to client browser.
*
* ARGUMENTS: EXTENSION_CONTROL_BLOCK
*pECB passed in structure pointer
* from inetsrv.
* int iTermd client browser terminal id
* int iSyncl client browser sync id
*
* RETURNS: None
*
* COMMENTS: None
*/

static void
ProcessNewOrderForm(EXTENSION_CONTROL_BLOCK *pECB, int iTermd, int iSyncl)
{
    #ifdef TOPEND
        int iRc;
        int msglen;
        tp_dif_structs_t *tclient_dif;
        void space_fill();
    #else // assume tuxedo at this point
        int iRc, iError;
        long ilen, *olen;
        char buff[40];
    #endif

    #ifdef LOCAL_ALLOC
        NewOrderData *newOrderDataPtr;
    #endif

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

```

```

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "" ProcessNewOrder
Thread %d iTermd %d Failed ThrTpnit \n",
            GetCurrentThreadId(),
            iTermd);
        fclose(fp);
    }
    sprintf(buf, "NewOrder ThrTpnit <0 >>
Thread %d", GetCurrentThreadId());
    ErrorMessage(pECB, ERR_TPINIT_BAD,
ERR_TYPE_TUXEDO, buf, 0, 0);
}

#endif

    memset(&Term.pClientData[iTermd].trans.newOrd
erData, 0, sizeof(NewOrderData));
    Term.pClientData[iTermd].trans.newOrderData.ne
win.w_id = Term.pClientData[iTermd].w_id;

    if ((iError=GetNewOrderData(pECB-
>lpSzQueryString,
&Term.pClientData[iTermd].trans.newOrderData)) !=
ERR_SUCCESS)
    {
        ErrorMessage(pECB, iError,
ERR_TYPE_WEBDLL, NULL, iTermd, iSyncl);
        return;
    }

    #ifdef TOPEND
        tclient_dif = (tp_dif_structs_t
*)Term.pClientData[iTermd].tp_structadr;

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

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

        Term.pClientData[iTermd].type = 'N';
        if ((i = tp_mt_client_send(tclient_dif->info, tclient_dif-
>service, tclient_dif->input_format, msglen, (char
*)&Term.pClientData[iTermd].type)) != TP_OK)
        {
            if (bLog)
            {
                FILE *fp1;
                fp1 = fopen(szTpccLogPath, "ab");

```

```

        fprintf(fp1, "tp_mt_client_send returned error
status = %d\n, i");
        fclose(fp1);
    }

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

msglen = sizeof(CLIENTDATA) -
TOPEND_STRUCT_OFFSET;

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

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

iRc = Term.pClientData[iTermId].retval

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

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

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "*** ProcessNewOrder
Thread %d iTermId %d tpalloc tperno %d \n",
GetCurrentThreadId(), iTermId,
iRc);
            fclose(fp);
        }
        sprintf(buf, "NewOrder Tmalloc %d", iRc);
        ErrorMessage(pECB,
ERR_TPALLOC_BAD, ERR_TYPE_TUXEDO, buf, 0,
0);
        tpfree((char *)newOrderDataPtr);
        return;
    }

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

    if (dLog)
    {
        FILE *fp;

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

    if (dLog)
    {
        FILE *fp;

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

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

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

    if (dLog)
    {
        FILE *fp;

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

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

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

    #endif
    #endif

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

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

```

```

}

/* FUNCTION: void
ProcessPaymentForm(EXTENSION_CONTROL_BLOCK *pECB, int iTermId,
*
* int iSynclD)
*
* PURPOSE: This function gets and validates the input
data from the payment
* form filling in the required input variables. It then
calls the
* SQLPayment transaction, constructs the output
form and writes it
* back to client browser.
*
* ARGUMENTS: EXTENSION_CONTROL_BLOCK
*pECB passed in structure pointer
*
* int iTermId client browser
terminal id
* int iSynclD client browser sync id
*
* RETURNS: None
*
* COMMENTS: None
*/

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

#ifdef LOCAL_ALLOC
    PaymentData *paymentDataPtr;
#endif

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

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

    memset(&Term.pClientData[iTermId].trans.payment
Data, 0, sizeof(PaymentData));
    Term.pClientData[iTermId].trans.paymentData.payi
n.w_id = Term.pClientData[iTermId].w_id;

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

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

    // Input topend information for call to process
payment form

    tclient_dif->info->tp_user_dialogue_id = iTermId;

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

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

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

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

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

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

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

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

    iRc = Term.pClientData[iTermId].retval;

#else // assume tuxedo running here

#ifdef LOCAL_ALLOC
    if ((paymentDataPtr = (PaymentData *)
tpalloc("CARRAY", NULL, sizeof(PaymentData))) ==
NULL)
    {
        iRc = tperrno;
        {
            FILE *fp;

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

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

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

    if (dLog)
    {
        FILE *fp;

```

```

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

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

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

        if ( dLog )
        {
            FILE *fp;

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

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

        iRc = paymentDataPtr->retval;

        tpfree((char *)paymentDataPtr);

    #else

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

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

        if ( dLog )
        {
            FILE *fp;

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

```

```

        fclose(fp);
    }

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

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

        if ( dLog )
        {
            FILE *fp;

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

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

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

        #endif
    #endif

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

    return;
}

```

```

/* FUNCTION: void
ProcessOrderStatusForm(EXTENSION_CONTROL_B
LOCK *pECB, int
*
iTermId, int iSyncId)
*
* PURPOSE: This function gets and validates the input
data from the Order Status
* form filling in the required input variables. It then
calls the
* SQLOrderStatus transaction, constructs the
output form and writes it
* back to client browser.
*
* ARGUMENTS: EXTENSION_CONTROL_BLOCK
*pECB passed in structure pointer from
inetsrv.
* int iTermId client browser
terminal id
* int iSyncId client browser sync id
*
* RETURNS: None
*
* COMMENTS: None
*/
static void
ProcessOrderStatusForm(EXTENSION_CONTROL_B
LOCK *pECB, int iTermId, int iSyncId)
{
    #ifdef TOPEND
        int i, iRc;
        int iError;
        void space_fill();
        tp_dif_structs_t *tclient_dif;
        int msglen;
    #else // assume tuxedo
        int iRc, iError;
        long ilen, *olen;
        // PECBINFO pECbInfo;
        char buf[40];
    #endif

    #ifdef LOCAL_ALLOC
        OrderStatusData *orderStatusDataPtr;
    #endif

    #ifdef TUX

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

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

```



```

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

#ifdef TOPEND

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

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

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

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

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

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

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

        msglen = sizeof(CLIENTDATA) -
TOPEND_STRUCT_OFFSET;

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

            printf("Received error status = %d from client
receive for order status transaction\n", i);
            ErrorMessage(pECB,
TOPEND_RECEIVE_ERROR, ERR_TYPE_WEBDLL,
NULL, iTermId, iSynclD);
            return;
        }

        iRc = Term.pClientData[iTermId].retval;

#ifdef LOCAL_ALLOC

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

                fp = fopen(szTpccLogPath, "ab");
                fprintf(fp, "*** ProcessOrderStatus
Thread %d iTermId %d tpalloc tperno %d \n",
GetCurrentThreadId(), iTermId, iRc);
                fclose(fp);

                sprintf(buf, "OrderStatus Tpcalloc %d",
iRc);
                ErrorMessage(NULL,
ERR_TPALLOC_BAD, ERR_TYPE_TUXEDO, buf, 0,
0);
                tpfree((char *)orderStatusDataPtr);
            }

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

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

            if (dLog)
            {
                FILE *fp;

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

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

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

                sprintf(buf, "OrderStatus tpccall %d",
iRc);
                ErrorMessage(pECB, ERR_TPCALL_BAD,
ERR_TYPE_TUXEDO, buf, iTermId, iSynclD);
                tpfree((char *)orderStatusDataPtr);
                return;
            }

            if (dLog)
            {
                FILE *fp;

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

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

            iRc = orderStatusDataPtr->retval;

            tpfree((char *)orderStatusDataPtr);

        }

        #else

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

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

        if (dLog)
        {
            FILE *fp;

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

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

```

```

    {
        FILE *fp;

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

if ( dLog )
{
    FILE *fp;

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

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

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

#endif
#endif

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

return;
}

/* FUNCTION: void
ProcessDeliveryForm(EXTENSION_CONTROL_BLOCK *pECB,
*
* int iTermId, int iSyncId)
*
* PURPOSE: This function gets and validates the input
data from delivery form
* filling in the required input variables. It then calls the
PostDeliveryInfo
* Api, The client is then informed that the transaction
has been posted.
*

```

```

* ARGUMENTS: EXTENSION_CONTROL_BLOCK
*pECB passed in structure pointer from
*
* int iTermId client browser
terminal id
*
* int iSyncId client browser sync id
*
* RETURNS: None
*
* COMMENTS: None
*
*/

static void
ProcessDeliveryForm(EXTENSION_CONTROL_BLOCK *pECB, int iTermId, int iSyncId)
{
#ifdef TOPEND
    char szTmp[26];
    int i;
    void space_fill();
    tp_dif_structs_t *tclient_dif;
    int msglen;
#endif

#ifdef LOCAL_ALLOC
    DeliveryData *deliveryDataPtr;
#endif

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

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

#endif

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

if ( !GetKeyValue(pECB->lpzQueryString, "OCD*",
szTmp, sizeof(szTmp)) )

```

```

{
    ErrorMessage(pECB,
ERR_DELIVERY_MISSING_OCD_KEY, ERR_TYPE_
WEBDLL, NULL, iTermId, iSyncId);
    return;
}

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

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

//post delivery info

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

/* Initialize Top End for tpcc transactions */
tclient_dif->info->tp_user_dialogue_id = iTermId;
tclient_dif->info->tp_flags = TP_NO_RESPONSE;
space_fill(tclient_dif->service->tp_function_name,
"del_bat", TP_FUNC_NAME_LEN);
space_fill(tclient_dif->service->tp_product_name,
"tpcc", TP_PROD_NAME_LEN);
space_fill(tclient_dif->input_format-
>tp_format_name, "", TP_FMT_NAME_LEN);
space_fill(tclient_dif->input_format-
>tp_format_language, "", TP_LANG_LEN);
space_fill(tclient_dif->input_format->tp_format_type,
"", TP_FMTTYPE_LEN);
tclient_dif->service->tp_function_qualifier = 0;
msglen = sizeof(CLIENTDATA) -
TOPEND_STRUCT_OFFSET;

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

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

if ((i = tp_mt_client_send(tclient_dif->info, tclient_dif-
>service,
tclient_dif->input_format, msglen, (char
*)&Term.pClientData[iTermId].type)) != TP_OK)
{
    if (bLog)
    {
        FILE *fp1;

```

```

        fp1 = fopen(szTpccLogPath, "ab");
        fprintf(fp1, "Received error status = %d on
client send for delivery transaction\n", i);
        fclose(fp1);
    }

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

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

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

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

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

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

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

        sprintf(buf, "Delivery tpalloc %d", iRc);
        ErrorMessage(pECB, ERR_TPALLOC_BAD,
ERR_TPALLOC_BAD, ERR_TYPE_TUXEDO, buf, 0,
0);

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

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

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

    if (dLog)
    {
        FILE *fp;

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

    if (dLog)
    {
        FILE *fp;

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

    tpfree((char *)deliveryDataPtr);
}

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

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

if (dLog)
{
    FILE *fp;

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

    if (dLog)
    {
        FILE *fp;

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

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

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

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

            sprintf(buf, "Delivery tpcall %d", iRc);
            ErrorMessage(pECB, ERR_TPCALL_BAD,
ERR_TYPE_TUXEDO, buf, iTermId, iSynclD);
            // tpfree((char
*)Term.pClientData[iTermId].deliveryDataPtr);
            return;
        }

        if (dLog)
        {
            FILE *fp;

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

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

    }

    #endif
    #endif

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

/* FUNCTION: void
ProcessStockLevelForm(EXTENSION_CONTROL_BL
OCK *pECB, int
* iTermId, int iSynclD)
*
* PURPOSE: This function gets and validates the input
data from the Stock Level
* form filling in the required input variables. It then calls
the
* SQLStockLevel transaction, constructs the output
form and writes it

```

```

* back to client browser.
*
* ARGUMENTS: EXTENSION_CONTROL_BLOCK
*pECB passed in structure pointer from
*
*          int          iTermId client browser
terminal id
*          int          iSynclId client browser sync id
*
* RETURNS: None
*
* COMMENTS: None
*
*/

static void
ProcessStockLevelForm(EXTENSION_CONTROL_BLOCK *pECB, int iTermId, int iSynclId)
{
#ifdef TOPEND
    char  szTmp[26];
    int   iIRc;
    int   msgLen;
    void  space_fill();
    tp_dif_structs_t *tclient_dif;
#else
    int   iRc;
//    int   iError;
    char  szTmp[26];
//    BOOL   bRc;
    long  iLen, *olen;
//    PECBINFO  pECBInfo;
    char  buf[40];
#endif

#ifdef LOCAL_ALLOC
    StockLevelData *stockLevelDataPtr;
#endif

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

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

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

#ifdef TOPEND

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

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

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

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

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

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

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

    msgLen = sizeof(CLIENTDATA) - TOPEND_STRUCT_OFFSET;

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

    iRc = Term.pClientData[iTermId].retVal;

#else

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

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

            sprintf(buf, "Tpalloc %d", iRc);
            ErrorMessage(NULL, ERR_TPALLOC_BAD, ERR_TYPE_TUXEDO, buf, 0, 0);
            tpfree((char *)stockLevelDataPtr);
            return;
        }
    }

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

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

    if (dLog)
    {
        FILE *fp;

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

```

```

        GetCurrentThreadId(), iTermId,
&stockLevelDataPtr);
        fclose(fp);
    }

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

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

        if (dLog)
        {
            FILE *fp;

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

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

        iRc = stockLevelDataPtr->retval;

        tpfree((char *)stockLevelDataPtr);

    #else

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

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

        if (dLog)
        {
            FILE *fp;

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

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

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

            if (dLog)
            {
                FILE *fp;

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

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

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

            #endif
            #endif

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

        return;
    }

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

    s=lpszQueryString;

    GetVal(s, "DID*", szTmp, sizeof(szTmp))
    if(! *s)
        return
ERR_NEWORDER_FORM_MISSING_DID;

    if (!IsNumeric(szTmp) )
        return
ERR_NEWORDER_DISTRICT_INVALID;

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

    GetVal(s, "CID*", szTmp, sizeof(szTmp))
    if(! *s)
        return
ERR_NEWORDER_CUSTOMER_KEY;

    if (!IsNumeric(szTmp) )
        return
ERR_NEWORDER_CUSTOMER_INVALID;

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

    bCheck = FALSE;
    for(i=0, items=0; i<15; i++)
    {
        GetVal(s, szIID[i], szTmp, sizeof(szTmp))
        if(! *s)
            return
ERR_NEWORDER_MISSING_IID_KEY;

        s=s-24;
        if ( szTmp[0])
        {
            //if blank lines between item ids
            if ( bCheck )
                return
ERR_NEWORDER_ITEM_BLANK_LINES;
            if ( !IsNumeric(szTmp) )
                return
ERR_NEWORDER_ITEMID_INVALID;
            pNewOrderData->newin.ol_i_id[i] =
atoi(szTmp);

            GetVal(s, szSP[i], szTmp,
sizeof(szTmp))
            if(! *s)

```

```

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

        GetVal(s, szQty[i], szTmp,
sizeof(szTmp))
        if(! *s)
            return
ERR_NEWORDER_MISSING_QTY_KEY;

        if (!IsNumeric(szTmp))
            return
ERR_NEWORDER_QTY_INVALID;

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

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

        for (j = i-1; j >= 0; j--)
            if (pNewOrderData->newin.ol_i_id[i]
== pNewOrderData->newin.ol_i_id[j]){
                pNewOrderData-
>newin.ol_quantity[i] += pNewOrderData-
>newin.ol_quantity[j];
                j = -1;
            }
        else
        {
            GetVal(s, szSP[i], szTmp,
sizeof(szTmp))
            if(! *s)
                return
ERR_NEWORDER_MISSING_QTY_KEY;

            if ( szTmp[0] )
                return
ERR_NEWORDER_SUPPW_WITHOUT_ITEMID;

            GetVal(s, szQty[i], szTmp,
sizeof(szTmp))
            if(! *s)
                return
ERR_NEWORDER_MISSING_QTY_KEY;

            if ( szTmp[0] )
                return
ERR_NEWORDER_QTY_WITHOUT_ITEMID;

```

```

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

// pNewOrderData->newout.o_ol_cnt = items;

return ERR_SUCCESS;
}

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

static int GetPaymentData(LPSTR lpszQueryString,
PaymentData *pPaymentData)
{
    char szTmp[26];
    char *ptr;
    char *s;
    double fTmp;

    s=lpszQueryString;

    GetVal(s, "DID*", szTmp, sizeof(szTmp))
    if(! *s)
        return
ERR_PAYMENT_MISSING_DID_KEY;

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

    GetVal(s, "CID*", szTmp, sizeof(szTmp))
    if(! *s)
        return
ERR_PAYMENT_MISSING_CID_KEY;

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

    pPaymentData->payin.c_id = atoi(szTmp);

    if ( szTmp[0] == 0 )
    {
        GetVal(s, "CLT*", szTmp, sizeof(szTmp))
        if(! *s)

```

```

        return
ERR_PAYMENT_MISSING_CLT;
        _strupr( szTmp );

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

    s=lpszQueryString;

    GetVal(s, "CWI*", szTmp, sizeof(szTmp))
    if(! *s)
        return
ERR_PAYMENT_MISSING_CWI_KEY;

    if ( !IsNumeric(szTmp) )
        return ERR_PAYMENT_CWI_INVALID;

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

    GetVal(s, "HAM*", szTmp, sizeof(szTmp))
    if(! *s)
        return
ERR_PAYMENT_MISSING_HAM_KEY;

    ptr = szTmp;

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

```

```

        else if (*ptr < '0' || *ptr > '9')
            return
ERR_PAYMENT_HAM_INVALID;
        ptr++;
    }

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

    return ERR_SUCCESS;
}

/* FUNCTION: int GetOrderStatusData(LPSTR
lpszQueryString,
*
    OrderStatusData
*pOrderStatusData)
*
* PURPOSE: This function extracts and validates the
payment form data from an
*
    http command string.
*
* ARGUMENTS: LPSTR    lpszQueryString
client browser http command
*
    string
*
    OrderStatusData *pOrderStatusData pointer
to order status
*
    data structure
*
* RETURNS: int        error code indicating reason
for failure
*
    ERR_SUCCESS    successfully parsed all
required input data
*
* COMMENTS: None
*/
static int GetOrderStatusData(LPSTR lpszQueryString,
OrderStatusData *pOrderStatusData)
{
    char szTmp[26];
    char *s;
    s=lpszQueryString;

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

    GetVal(s, "CID*", szTmp, sizeof(szTmp))
    if(! *s)
        return
ERR_ORDERSTATUS_MISSING_CID_KEY;

    if (szTmp[0] == 0)
    {
        pOrderStatusData->ordin.c_id = 0;
        GetVal(s, "CLT*", szTmp, sizeof(szTmp))
        if(! *s)
            return
ERR_ORDERSTATUS_MISSING_CLT_KEY;

```

```

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

    return ERR_SUCCESS;
}

/* FUNCTION: BOOL ReadRegistrySettings(void)
*
* PURPOSE: This function reads the NT registry for
startup parameters. There
*
    parameters are under the TPCC key.
*
* ARGUMENTS: None
*
* RETURNS: None
*
* COMMENTS: This function also sets up required
operation variables to
*
    their default value so if registry is not setup
the default
*
    values will be used.
*/
static BOOL ReadRegistrySettings(void)
{
    HKEY hKey;
    DWORD size;
    DWORD type;
    char szTmp[256];

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

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

```

```

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

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

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

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

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

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

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

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

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

```

```

    bGeneric = TRUE;
}

RegCloseKey(hKey);
return FALSE;
}

/* FUNCTION: BOOL IsNumeric(char *ptr)
 *
 * PURPOSE: This function determines if a string is
numeric. It fails if any
 * characters other than numeric and null
terminator are present.
 *
 * ARGUMENTS: char *ptr pointer to string to
check.
 *
 * RETURNS:  BOOL  FALSE  if string is not all
numeric
 *           TRUE   if string contains all numeric
characters i.e. '0' - '9'
 *
 * COMMENTS: None
 */

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

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

static void FormatHTMLString(char *szBuff, char *szStr,
int iLen)
{

```

```

while( iLen && *szStr )
{
    switch( *szStr )
    {
        case '>':
            *szBuff++ = '&';
            *szBuff++ = 'g';
            *szBuff++ = 't';
            *szBuff++ = ';';
            szStr++;
            break;
        case '<':
            *szBuff++ = '&';
            *szBuff++ = 'l';
            *szBuff++ = 't';
            *szBuff++ = ';';
            szStr++;
            break;
        case '&':
            *szBuff++ = '&';
            *szBuff++ = 'a';
            *szBuff++ = 'm';
            *szBuff++ = 'p';
            *szBuff++ = ';';
            szStr++;
            break;
        case '\\':
            *szBuff++ = '&';
            *szBuff++ = 'q';
            *szBuff++ = 'u';
            *szBuff++ = 'o';
            *szBuff++ = 't';
            *szBuff++ = ';';
            szStr++;
            break;
        default:
            *szBuff++ = *szStr++;
            break;
    }
    iLen--;
}
while( iLen-- )
*szBuff++ = ' ';
*szBuff = 0;
return;
}

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

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

```

```

        if ( dLog )
        {
            FILE *fp;

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

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

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

                        fp =
fopen(szTpccLogPath, "ab");
                        fprintf(fp, ">>>>
ThrTpnit %d : ttpalloc of tpinit failed: %d \\n",
GetCurrentThreadId(), TpRc);
                        fclose(fp);
                    }
                    tpinf = NULL;
                    retry++;

                    LeaveCriticalSection(&TpCriticalSection);
                    Sleep(50); // Relinquish thread
timeslice

                    continue;
                }

                tpinf-
>flags|=TPMULTICONTEXTS;
            }

            // Do the TPINIT
            itoa(++num_tpnits, tpinf->ctlname,
10);

            iRc = tpinit(tpinf);

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

                if ( dLog )
                {
                    FILE *fp;

                    fp =
fopen(szTpccLogPath, "ab");
                    fprintf(fp, "" ThrTpnit
Thread %d Retry : count %d * \\n",
GetCurrentThreadId(), retry);
                    fclose(fp);

```



```

    }
  }
  else
  {
    Success = TRUE;
    tpfree((char *) tpinf);
    tpinf=NULL;
    break;
  }
  --num_tpinit;

  LeaveCriticalSection(&TpCriticalSection);
  Sleep(50); // Relinquish
thread timeslice

  } // retry the tpinit if it failed the first time

  LeaveCriticalSection(&TpCriticalSection);

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

      fp = fopen(szTpccLogPath,
"ab");
      fprintf(fp, ">>>> ThrTpnit
Thread %d Failed tperno= %d Last Err= %d \r\n",
GetCurrentThreadId(),
TpRc, lasterr);
      fclose(fp);
    }
    sprintf(buf, "Thread %d: In
ThrTpnit iRc=%d Tperno=%d",

    GetCurrentThreadId(), iRc, TpRc);
    ErrorMessage(pECB,
ERR_TPINIT_BAD, ERR_TYPE_TUXEDO, buf, 0, 0);

    return -1;
  }
  if (Success == TRUE)
  {
    if (dLog)
    {
      FILE *fp;

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

    if ((iRc=TLsSetValue(TLSIsTpnitedKey, &x)) == 0)
    {
      FILE *fp;

      fp =
fopen(szTpccLogPath, "ab");
      fprintf(fp, ">>>>
ThrTpnit %d : TLsSetValue Failed iRc: %d \r\n",
GetCurrentThreadId(),
iRc);

```

```

    fclose(fp);
  }
  sprintf(buf, "TLsSetValue %d", iRc);
  ErrorMessage(pECB,
ERR_TPINIT_BAD, ERR_TYPE_TUXEDO, buf, 0, 0);
  return -1;
}
}
else
{
  if (dLog)
  {
    FILE *fp;

    fp = fopen(szTpccLogPath, "ab");
    fprintf(fp, "** ThrTpnit Thread %d
already tpnited * \r\n", GetCurrentThreadId());
    fclose(fp);
  }
}
return 0;
}

#ifdef TOPEND
void space_fill(s1, s2, i)
char *s1, *s2;
int i;
{
  int ii;

  strncpy(s1, s2, i);

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

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

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

```

```

    fclose(fp1);
  }
  MT_LOG_ERROR;
  return;
}

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

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

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

/* receive loop. Should there be a way to shut this
down? */
for (;;)
{
  if (tp_WaitForSingleObject (NULL, INFINITE) ==
WAIT_TIMEOUT)
  {
    MT_LOG_ERROR;
    return;
  }
  dif_struct->info->tp_user_dialogue_id =
TP_ANY_DIALOGUE;
  dif_struct->info->tp_flags = TP_NOFLAGS;
  buffer_length = TP_MAX_BUF_LEN;

  rc = tp_client_receive (dif_struct-
>info, TP_NOBLOCK,

    dif_struct->output_format,
    dif_struct->service,
    dif_struct->location,
    &buffer_length,
    message_buffer);

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

      flags = TP_TRUNCATE;
      buf_len = 1;
      tp_system_admin (&flags, &buf_len,
buffer);
      break;
    }
    case TP DISSOLVED:
    case TP_OK:
    case TP_RESET:

```

```

case TP_SERVICE:
case TP_SIGNON_INHIBITED:
case TP_USER:
{
    DialogTableEntry_t *DTE;
    void *ptr;

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

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

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

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

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

    DTE->buffer_length = buffer_length;

    if (DTE->message_buffer == NULL)
    {
        if ((DTE->message_buffer = malloc
(buffer_length)) == NULL)
        {
            MT_LOG_ERROR;
            return;
        }
    }
    else
    {
        /* NONPORTABLE */
        if (_msize (DTE->message_buffer) <
(size_t) buffer_length)
        {
            if ((DTE->message_buffer = realloc
(DTE->message_buffer,
buffer_length)) == NULL)
            {
                MT_LOG_ERROR;
                return;
            }
        }
    }
    memcpy (DTE->message_buffer,
message_buffer, buffer_length);
    DTE->rc = rc;
    /* Signal that the data is ready. */
    SetEvent (DTE->hSync);
    break;
}

}
case TP_SHUTDOWN:
{
    DialogTableEntry_t *DTE;
    /* tell everyone */
    if (WaitForSingleObject
(DTActiveList.hSync, INFINITE) == WAIT_TIMEOUT)
    {
        MT_LOG_ERROR;
        return;
    }
    DTE = DTActiveList.next;
    while (DTE != NULL)
    {
        DTE->rc = TP_SHUTDOWN;
        SetEvent (DTE->hSync);
        DTE = DTE->next;
    }
    if (!ReleaseMutex(DTActiveList.hSync))
    {
        MT_LOG_ERROR;
        return;
    }
    break;
}
default:
{
    if (bLog)
    {
        fp1 = fopen(szTpccLogPath, "ab");
        fprintf(fp1, "TEReceiveThread received
error status = %d\n", rc);
        fclose(fp1);
    }
    MT_LOG_ERROR;
    return;
}
}

}

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

    rc = tp_initialize(application_info, function_array,
number_functions);

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

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

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

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

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

    switch (rc)
    {
        case TP DISSOLVING:
        case TP_OK:
        {
            DTE->usr_dlg_storage = info-
>tp_user_dialogue_id;
            DTE->usr_dlg_ptr = &(DTE->usr_dlg_storage);
        }
    }
}

```

```

        break;
    }
    default:
    {
        /* there is nothing to receive */
        RemoveFromList (DTE, &DTActiveList);
        AddToList (DTE, &DTFreeList);
    }
}
return rc;
}

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

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

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

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

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

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

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

    RemoveFromList (DTE, &DTActiveList);

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

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

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

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

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

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

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

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

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

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

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

```

```

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

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

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

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

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

#endif

===== tpccsvr.c =====

#ifdef RCSID
static char *RCSid =
"$Header: tpccsvr.c 7030100.1 95/07/19 15:39:28
plai Generic-base> $ Copyr (c) 1995 Oracle";
#endif /* RCSID */

/*=====
=====+
| Copyright (c) 1995 Oracle Corp, Redwood
Shores, CA |
| OPEN SYSTEMS PERFORMANCE
GROUP |
| All Rights Reserved |
=====+
| FILENAME
| tpccsvr.c
| DESCRIPTION
| Tuxedo server for TPC-C. use a #define TUX
| TOPEND server for TPC-C. use a #define TOP
=====+
=====*/

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

#ifdef TUX
#include <atmi.h> // must occur prior to include of
tpccapi.h
#include <stdlib.h> // for generation of random seed
for server id
#include <time.h> // for generation of random seed
for server id
#include <fml.h>
#include <fldtbl.h>
#endif

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

#ifdef TUX

#include <tmenv.h>
#include <xa.h>
#include <userlog.h>
// SymfoWare Use Only
#define TX_NEWORDER 1
#define TX_PAYMENT 2
#define TX_ORDERSTAT 3
#define TX_DELIVERY 4
#define TX_STOCKLVL 5
/* set up pointers for type casting */
//kawabata change
struct newstruct *newinfo;
struct paystruct *payinfo;
struct ordstruct *ordinfo;
struct delstruct *delinfo;
struct stostruct *stoinfo;

struct newstruct nbuf;
struct paystruct pbuf;
struct ordstruct obuf;
struct delstruct dbuf;
struct stostruct sbuf;

struct newstruct *bpn;
struct paystruct *bpb;

```

```

struct ordstruct *bpo;
struct delstruct *bpd;
struct stostruct *bps;

extern void TMlog();

#endif

// Lifted from HP FDR since they did such a nice job
void TMlog(char *format, ...)
{
    va_list args;
    char buf[4096];
    int len;
    va_start(args, format);
    _strtime(buf);
    strcat(buf, " ");
    len = strlen(buf);
    (void)_vsnprintf(buf + len, sizeof(buf) - len - 1,
format, args);
    buf[sizeof(buf) - 1] = '\0';
    va_end(args);
    userlog(buf);
}

/* FUNCTION: int tpsvinit (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 tpsvinit (int argc, char *argv[])
{
    int svrid, txntype;
    char *uid, *pwd;
    int svrcnt;

    /* pull out the values from argv */
    svrid = atoi(argv[0]);
    uid = argv[1];
    pwd = argv[2];
    txntype = atoi(argv[3]);

#ifdef TUX
    srand ((unsigned)time(NULL));
    svrcnt = rand();

```

```

/* send 6 for all txns to be initd */
/* fix uid and pwd for now, pull out later */
/* not passing parameters through TUX yet */
if (TPCinit (svrcnt, "tpcc", "tpcc", 6)) {
    TMLog(" FAILED to init all txns types");
    return (-1);
}

return 0;

#else // for topend
if (TPCinit (svrid, uid, pwd, txntype)) {
    printf("Failed in TPCinit (probably
connecting.);");
    exit (1);
}

return (1);
#endif

}

void tpsvrdone ()
{
    TPCexit (0);
}

void TPCC (TPSVCINFO *msg)
{
    int mix;

    mix = Fvall( ( FBFR *)msg->data, FML_TRAN,
0); //symfo

    switch(mix)
    {
        case TX_NEWORDER:
            nbuf = *((struct newstruct *)Ffind((FBFR
*)msg->data, FML_DATA, 0, NULL));
            bpn = &nbuf;
            //printf("NEWORDER start ");
            TPCnew(bpn);
            //printf("NEWORDER end ");
            // NEWORDER(bpn);
            Fchg((FBFR *)msg->data, FML_DATA, 0,
(char *)bpn, sizeof(struct newstruct));
            // newinfo = (struct newstruct *)info;
            break;

            case TX_PAYMENT:
                pbuf = *((struct paystruct *)Ffind((FBFR
*)msg->data, FML_DATA, 0, NULL));
                bpp = &pbuf;
                TPCpay(bpp);
                // PAYMENT(bpp);
                Fchg((FBFR *)msg->data, FML_DATA, 0,
(char *)bpp, sizeof(struct paystruct));
                // payinfo = (struct paystruct *)info;
                break;

            case TX_ORDERSTAT:
                obuf = *((struct ordstruct *)Ffind((FBFR
*)msg->data, FML_DATA, 0, NULL));
                bpo = &obuf;
                TPCord(bpo);
                // ORDERSTATUS();
                Fchg((FBFR *)msg->data, FML_DATA, 0,
(char *)bpo, sizeof(struct ordstruct));
                //printf(" call ord2 \n"); /* Debug */
                //printf(" sizeof Orderstruct =%d \n",sizeof(struct
ordstruct)); /* Debug */

                // ordinfo = (struct ordstruct
*)info;
                break;

                case TX_DELIVERY:
                    delinfo = (struct delstruct *)info;
                    dbuf = *((struct delstruct *)Ffind((FBFR
*)msg->data, FML_DATA, 0, NULL));
                    bpd = &dbuf;
                    TPCdel(bpd);
                    // DELIVERY(bpd);
                    tpreturn(TPSUCCESS, 0, (char
*)NULL, 0, 0);
                    // break;

                    case TX_STOCKLVL:
                        sbuf = *((struct stostruct *)Ffind((FBFR
*)msg->data, FML_DATA, 0, NULL));
                        bps = &sbuf;
                        TPCsto(bps);
                        // STOCKLEVEL(bps);

                        Fchg((FBFR *)msg->data, FML_DATA, 0,
(char *)bps, sizeof(struct stostruct));
                        // stoinfo = (struct stostruct *)info;
                        break;
                    }
                    if(mix != TX_DELIVERY)
                    {
                        //printf(" call ord3 \n"); /* Debug */
                        tpreturn(TPSUCCESS, 0, msg->data, 0L,
0);
                    }
                }

/* FUNCTION: int NEWORDER(CLIENTDATA *jobData,
NewOrderData *neword, int deadlock)
*
* PURPOSE: This function handles the new order
transaction.
*
* ARGUMENTS: deadlock : count of deadlocks
encountered during txn
* jobData: pointer to entire block of user data
* neword: pointer to datastructure in jobData
that contains the new order data
* RETURNS: int TRUE transaction
committed
* FALSE item number not valid
* -1 deadlock max retry
reached
*
* COMMENTS: None
*/

/* 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 NEWORDER(CLIENTDATA *jobData,
NewOrderData *neword, int deadlock)
#else
void NEWORDER (TPSVCINFO *msg)
#endif
{
#ifdef TOP
    int result;

    result = TPCnew(neword);

    return result;
#else // for Tuxedo

    newinfo = (struct newstruct *) msg->data;
    newinfo->retval = TPCnew (newinfo); // set return
value to 0 or -1

    // TPCnew (msg); // set return value to 0 or -1
    // return;

    // always return tpreturn success - let client side poll
retval for actual error
    tpreturn (TPSUCCESS, 0, (char *) newinfo, sizeof
(struct newstruct), 0);
#endif
}

/* FUNCTION: int PAYMENT(CLIENTDATA *jobData,
PaymentData *paydata, int deadlock)
*
* PURPOSE: This function handles the new order
transaction.
*
* ARGUMENTS: deadlock : count of deadlocks
encountered during txn
* jobData: pointer to entire block of user data
* paydata: pointer to datastructure in
jobData that contains the new order data
* RETURNS: int TRUE transaction
committed
* FALSE item number not valid
* -1 deadlock max retry
reached
*
* COMMENTS: None
*/

#ifdef TOP
int PAYMENT(CLIENTDATA *jobData, PaymentData
*paydata, int deadlock)
#else
void PAYMENT (TPSVCINFO *msg)
#endif
{
#ifdef TOP
    int result;

```

```

    result = TPCpay(paydata);

    return result;
#else

    payinfo = (struct paystruct *) msg->data;
    payinfo->retval = TPCpay (payinfo); // set return value
to 1 or 0 or -1
// TPCpay (msg); // set return value to 0 or -1
// return;

    // always return tpreturn success - let client side poll
retval for actual error
    tpreturn (TPSUCCESS, 0, (char *) payinfo, sizeof
(struct paystruct), 0);

#endif

}

/* FUNCTION: int ORDERSTATUS(CLIENTDATA
*jobData, OrderStatusData *orddata, int deadlock)
*
* PURPOSE: This function handles the new order
transaction.
*
* ARGUMENTS: deadlock : count of deadlocks
encountered during txn
*      jobData: pointer to entire block of user data
*      orddata: pointer to datastructure in jobData
that contains the new order data
* RETURNS: int TRUE transaction
committed
*          FALSE item number not valid
*          -1     deadlock max retry
reached
*
* COMMENTS: None
*/

#ifdef TOP
int ORDERSTATUS(CLIENTDATA *jobData,
OrderStatusData *orddata, int deadlock)
#else
void ORDERSTATUS (TPSVCINFO *msg)
//void ORDERSTATUS ()
#endif

{

#ifdef TOP
    int result;

    result = TPCord(orddata);

    return result;

#else

    //printf("w_id=%d",(struct ordstruct *)bpo->ordin.w_id);
    //printf("w_id=%d",(struct ordstruct *)bpo->ordin.d_id);

    //printf("w_id=%d",(struct ordstruct *)msg->ordin.w_id);
    printf("d_id=%d",(struct ordstruct *)msg->ordin.d_id);*/

//printf("TPCord before data");

    ordinfo = (struct ordstruct *) msg->data;
    ordinfo->retval = TPCord (ordinfo); // set return value
to 0 or -1
// TPCord (bpo); // set return value to 0 or -1
// return;

    // always return tpreturn success - let client side poll
retval for actual error
    tpreturn (TPSUCCESS, 0, (char *) ordinfo, sizeof
(struct ordstruct), 0);

#endif

}

/* FUNCTION: int DELIVERY(CLIENTDATA *jobData,
DeliveryData *deldata, int deadlock)
*
* PURPOSE: This function handles the new order
transaction.
*
* ARGUMENTS: deadlock : count of deadlocks
encountered during txn
*      jobData: pointer to entire block of user data
*      stodata: pointer to datastructure in jobData
that contains the new order data
* RETURNS: int TRUE transaction
committed
*          FALSE item number not valid
*          -1     deadlock max retry
reached
*
* COMMENTS: None
*/

#ifdef TOP
int DELIVERY(CLIENTDATA *jobData, DeliveryData
*deldata, int deadlock)
#else
void DELIVERY (TPSVCINFO *msg)
#endif

{

#ifdef TOP
    int result;

    result = TPCdel(deldata);

    return result;

#else

    delinfo = (struct delstruct *) msg->data;
    delinfo->retval = TPCdel (delinfo); // set return value
to 0 or -1
// TPCdel (msg); // set return value to 0 or -1
// return;

    // always return tpreturn success - let client side poll
retval for actual error
    tpreturn (TPSUCCESS, 0, (char *) delinfo, sizeof
(struct delstruct), 0);

#endif

}

#endif

}

/* FUNCTION: int STOCKLEVEL(CLIENTDATA
*jobData, StockLevelData *stodata, int deadlock)
*
* PURPOSE: This function handles the new order
transaction.
*
* ARGUMENTS: deadlock : count of deadlocks
encountered during txn
*      jobData: pointer to entire block of user data
*      stodata: pointer to datastructure in jobData
that contains the new order data
* RETURNS: int TRUE transaction
committed
*          FALSE item number not valid
*          -1     deadlock max retry
reached
*
* COMMENTS: None
*/

#ifdef TOP
int STOCKLEVEL(CLIENTDATA *jobData,
StockLevelData *stodata, int deadlock)
#else
void STOCKLVL (TPSVCINFO *msg)
#endif

{

#ifdef TOP
    int result;

    result = TPCsto(stodata);

    return result;

#else

    stoinfo = (struct stostruct *) msg->data;
    stoinfo->retval = TPCsto (stoinfo); // set return value
to 0 or -1
// TPCsto (msg); // set return value to 0 or -1
// return;

    // always return tpreturn success - let client side poll
retval for actual error
    tpreturn (TPSUCCESS, 0, (char *) stoinfo, sizeof
(struct stostruct), 0);

#endif

}

/* FUNCTION: int OPSTUXSERVER(CLIENTDATA
*jobData, NewOrderData *neword, int deadlock)
*
* PURPOSE: This function handles all transactions.
*
* ARGUMENTS: deadlock : count of deadlocks
encountered during txn

```

```

*      jobData: pointer to entire block of user data
*      neword:  pointer to datastructure in jobData
that contains the new order data
* RETURNS:    int TRUE  transaction
committed
*              FALSE  item number not valid
*              -1      deadlock max retry
reached
*
*
* COMMENTS:   None
*/

#ifdef TOP
int OPSTUXSERVER(CLIENTDATA *jobData,
NewOrderData *neword, int deadlock)
#else
void OPSTUXSERVER (TPSVCINFO *msg)
#endif

{
return;
}
/*
#ifdef TOP
int result;

result = TPCnew(neword);

return result;

#else          // for Tuxedo

if (msg->len == 928) { // len for neworder
newinfo = (struct newstruct *) msg->data;
newinfo->retval = TPCnew (newinfo); // set
return value to 0 or -1

// always return tpreturn success - let client side
poll retval for actual error
tpreturn (TPSUCCESS, 0, (char *) newinfo,
sizeof (struct newstruct), 0);
}
else
if (msg->len == 616) { // len for payment
payinfo = (struct paystruct *) msg->data;
payinfo->retval = TPCpay (payinfo); // set
return value to 1 or 0 or -1

// always return tpreturn success - let client side
poll retval for actual error
tpreturn (TPSUCCESS, 0, (char *) payinfo,
sizeof (struct paystruct), 0);
}
else
if (msg->len == 544) { // len for order
status
ordinfo = (struct ordstruct *) msg->data;
ordinfo->retval = TPCord (ordinfo); // set
return value to 0 or -1

// always return tpreturn success - let client
side poll retval for actual error
tpreturn (TPSUCCESS, 0, (char *) ordinfo,
sizeof (struct ordstruct), 0);
}
else
if (msg->len == 40) { // len for delivery
delinfo = (struct delstruct *) msg->data;
delinfo->retval = TPCdel (delinfo); // set
return value to 0 or -1

// always return tpreturn success - let
client side poll retval for actual error
tpreturn (TPSUCCESS, 0, (char *)
delinfo, sizeof (struct delstruct), 0);
}
else { // assume rest is stock level
stoinfo = (struct stostruct *) msg-
>data;
stoinfo->retval = TPCsto (stoinfo); //
set return value to 0 or -1

// always return tpreturn success - let
client side poll retval for actual error
tpreturn (TPSUCCESS, 0, (char *)
stoinfo, sizeof (struct stostruct), 0);
}
}

#endif
}*/

===== initnew.sql =====
-- The initnew package for storing variables used in the
-- New Order anonymous block

CREATE OR REPLACE PACKAGE initnew
AS
TYPE intarray IS TABLE OF INTEGER index by
binary_integer;
TYPE distarray IS TABLE OF VARCHAR(24) index by
binary_integer;
nulldate DATE;
s_dist distarray;
idx1arr intarray;
s_remote intarray;
PROCEDURE new_init(idxarr intarray);
END initnew;
/
show errors;

CREATE OR REPLACE PACKAGE BODY initnew AS
PROCEDURE new_init (idxarr intarray)
IS
BEGIN
-- initialize null date
nulldate := TO_DATE('01-01-1811', 'MM-DD-YYYY');
idx1arr := idxarr;
END new_init;
END initnew;
/
show errors
exit

===== initpay.sql =====

CREATE OR REPLACE PACKAGE initpay
AS
TYPE rowidarray IS TABLE OF ROWID INDEX BY
BINARY_INTEGER;
row_id rowidarray;
cust_rowid ROWID;
dist_name VARCHAR2(11);
ware_name VARCHAR2(11);
c_num BINARY_INTEGER;
PROCEDURE pay_init;
END initpay;
/

CREATE OR REPLACE PACKAGE BODY initpay AS
PROCEDURE pay_init IS
BEGIN
NULL;
END pay_init;
END initpay;
/

exit;

===== paynz.sql =====

DECLARE /* paynz */
-- cust_rowid ROWID;
-- dist_name VARCHAR2(11);
-- ware_name VARCHAR2(11);
not_serializable EXCEPTION;
PRAGMA EXCEPTION_INIT(not_serializable,-
8177);
deadlock EXCEPTION;
PRAGMA EXCEPTION_INIT(deadlock,-60);
snapshot_too_old EXCEPTION;
PRAGMA EXCEPTION_INIT(snapshot_too_old,-
1555);
BEGIN
LOOP BEGIN
UPDATE ware
SET w_ytd = w_ytd + :h_amount
WHERE w_id = :w_id
RETURNING w_name, w_street_1, w_street_2,
w_city, w_state, w_zip
INTO
initpay.ware_name, :w_street_1, :w_street_2, :w_city,
:w_state, :w_zip;

UPDATE cust
SET c_balance = c_balance - :h_amount,
c_ytd_payment = c_ytd_payment
+ :h_amount,
c_payment_cnt = c_payment_cnt+1
WHERE c_id = :c_id AND c_d_id = :c_d_id
AND
c_w_id = :c_w_id
RETURNING rowid, c_first, c_middle, c_last,
c_street_1,
c_street_2, c_city, c_state, c_zip, c_phone,
c_since, c_credit, c_credit_lim,
c_discount, c_balance
INTO
initpay.cust_rowid, :c_first, :c_middle, :c_last, :c_street_
1,
:c_street_2, :c_city, :c_state, :c_zip, :c_pho
ne,
:c_since, :c_credit, :c_credit_lim,
:c_discount, :c_balance;
IF SQL%NOTFOUND THEN
raise NO_DATA_FOUND;

```

```

END IF;
--      insert into dummy values
(rowidtochar(initpay.cust_rowid));

--      :c_data := '';

IF :c_credit = 'BC' THEN
  UPDATE cust
    SET c_data= substr ((to_char (:c_id) || ' ' ||
      to_char (:c_d_id) || ' ' ||
      to_char (:c_w_id) || ' ' ||
      to_char (:d_id) || ' ' ||
      to_char (:w_id) || ' ' ||
      to_char (:h_amount, '9999.99') || '
|)
      || c_data, 1, 500)
  WHERE rowid = initpay.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
initpay.dist_name,:d_street_1,:d_street_2,:d_city,:d_sta
te,
      :d_zip;
IF SQL%NOTFOUND THEN
  raise NO_DATA_FOUND;
END IF;

INSERT INTO hist (h_c_id, h_c_d_id, h_c_w_id,
h_d_id, h_w_id,
      h_amount, h_date, h_data)
VALUES
(:c_id, :c_d_id, :c_w_id, :d_id, :w_id, :h_amount,
:cr_date, initpay.ware_name || ' ' ||
initpay.dist_name);
--      COMMIT;
--      :h_date := to_char (:cr_date, 'DD-MM-
YYYY.HH24:MI:SS');
EXIT;

EXCEPTION
  WHEN not_serializable OR deadlock OR
snapshot_too_old THEN
  ROLLBACK;
  :retry := :retry + 1;
END;

END LOOP;
END;

===== payz.sql =====
DECLARE /* payz */
  not_serializable EXCEPTION;
  PRAGMA EXCEPTION_INIT(not_serializable,-
8177);
  deadlock EXCEPTION;
  PRAGMA EXCEPTION_INIT(deadlock,-60);

```

```

snapshot_too_old EXCEPTION;
  PRAGMA EXCEPTION_INIT(snapshot_too_old,-
1555);
BEGIN
  LOOP BEGIN
    UPDATE ware
      SET w_ytd = w_ytd+ :h_amount
      WHERE w_id = :w_id
      RETURNING w_name,
w_street_1, w_street_2, w_city, w_state,
w_zip
  INTO initpay.ware_name,
:w_street_1, :w_street_2, :w_city, :w_state,
:w_zip;

  SELECT rowid
  BULK COLLECT INTO initpay.row_id
  FROM cust
  WHERE c_d_id = :c_d_id AND c_w_id = :c_w_id
  AND c_last = :c_last
  ORDER BY c_last, c_d_id, c_w_id, c_first;

  initpay.c_num := sql%rowcount;
  initpay.cust_rowid := initpay.row_id((initpay.c_num)
/2);

  UPDATE cust
    SET c_balance = c_balance - :h_amount,
c_ytd_payment = c_ytd_payment+ :h_amount,
c_payment_cnt = c_payment_cnt+1
  WHERE rowid = initpay.cust_rowid
  RETURNING
c_id, c_first, c_middle, c_last, c_street_1,
c_street_2,
c_city, c_state, c_zip, c_phone,
c_since, c_credit, c_credit_lim,
c_discount, c_balance
  INTO :c_id, :c_first, :c_middle, :c_last,
:c_street_1, :c_street_2, :c_city, :c_state,
:c_zip, :c_phone, :c_since, :c_credit,
:c_credit_lim, :c_discount, :c_balance;

:c_data := '';
IF :c_credit = 'BC' THEN
  UPDATE cust
    SET c_data = substr ((to_char (:c_id) || ' ' ||
      to_char (:c_d_id) || ' ' ||
      to_char (:c_w_id) || ' ' ||
      to_char (:d_id) || ' ' ||
      to_char (:w_id) || ' ' ||
      to_char (:h_amount/100,
'9999.99') || ' | ')
      || c_data, 1, 500)
  WHERE rowid = initpay.cust_rowid
  RETURNING substr(c_data,1, 200)
  INTO :c_data;

END IF;

UPDATE dist
  SET d_ytd = d_ytd+ :h_amount
  WHERE d_id = :d_id
  AND d_w_id = :w_id
  RETURNING d_name, d_street_1, d_street_2,
d_city,
d_state, d_zip
  INTO
initpay.dist_name, :d_street_1, :d_street_2, :d_city,

```

```

:d_state, :d_zip;

IF SQL%NOTFOUND
THEN
  raise NO_DATA_FOUND;
END IF;

INSERT INTO hist (h_c_id, h_c_d_id, h_c_w_id,
h_d_id, h_w_id,
      h_amount, h_date, h_data)
VALUES
(:c_id, :c_d_id, :c_w_id, :d_id, :w_id, :h_amount,
:cr_date, initpay.ware_name || ' ' ||
initpay.dist_name);

EXIT;

EXCEPTION
  WHEN not_serializable OR deadlock OR
snapshot_too_old THEN
  ROLLBACK;
  :retry := :retry + 1;
END;

END LOOP;
END;

===== tkvcbnew.sql =====

DECLARE /* new order */
  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
    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;

  UPDATE wh_dist SET d_next_o_id =
d_next_o_id + 1, d_tax=d_tax+0
  WHERE d_id = :d_id
  AND w_id = :w_id
  RETURNING d_tax, d_next_o_id-1, w_tax
  INTO :d_tax, :o_id, :w_tax;

  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_w_id, o_d_id, o_c_id,
o_carrier_id,
      o_ol_cnt, o_all_local, o_entry_id)
  VALUES (:o_id, :w_id, :d_id, :c_id, 11,
:o_ol_cnt, :o_all_local, :cr_date);
EXIT;

EXCEPTION
  WHEN not_serializable OR deadlock OR
snapshot_too_old THEN
  ROLLBACK;

```



```

        :retry := :retry + 1;
    END;
END LOOP;
END;

===== tkvcpnew.sql =====

-- New Order Anonymous block

DECLARE
    idx          BINARY_INTEGER;
    dummy_local  BINARY_INTEGER;
    not_serializable EXCEPTION;
    PRAGMA EXCEPTION_INIT(not_serializable,-
8177);
    deadlock     EXCEPTION;
    PRAGMA EXCEPTION_INIT(deadlock,-60);
    snapshot_too_old EXCEPTION;
    PRAGMA EXCEPTION_INIT(snapshot_too_old,-
1555);
PROCEDURE u1 IS
BEGIN
    FORALL idx IN 1 .. :o_ol_cnt
        UPDATE stock_item
            SET s_order_cnt = s_order_cnt + 1,
                s_ytd = s_ytd + :ol_quantity(idx),
                s_remote_cnt = s_remote_cnt + :s_remote(idx),
                s_quantity = s_quantity - :ol_quantity(idx) +
                    DECODE(sign(s_quantity
- :ol_quantity(idx) -
10),-1,91,0)
                WHERE i_id = :ol_i_id(idx)
                AND s_w_id = :ol_supply_w_id(idx)
                RETURNING i_price, i_name, s_quantity,
s_dist_01,
                    DECODE(instr(i_data,'ORIGINAL'), 0, 'G',
                    DECODE(instr(s_data,'ORIGINAL'), 0, 'G',
'B'))
        BULK COLLECT
INTO :i_price, :i_name, :s_quantity, initnew.s_dist,
:brand_generic;

    END u1;

PROCEDURE u2 IS
BEGIN
    FORALL idx IN 1 .. :o_ol_cnt
        UPDATE stock_item
            SET s_order_cnt = s_order_cnt + 1,
                s_ytd = s_ytd + :ol_quantity(idx),
                s_remote_cnt = s_remote_cnt + :s_remote(idx),
                s_quantity = s_quantity - :ol_quantity(idx) +
                    DECODE(sign(s_quantity
- :ol_quantity(idx) -
10),-1,91,0)
                WHERE i_id = :ol_i_id(idx)
                AND s_w_id = :ol_supply_w_id(idx)
                RETURNING i_price, i_name, s_quantity,
s_dist_02,
                    DECODE(instr(i_data,'ORIGINAL'), 0, 'G',
                    DECODE(instr(s_data,'ORIGINAL'), 0, 'G',
'B'))
        BULK COLLECT
INTO :i_price, :i_name, :s_quantity, initnew.s_dist,
:brand_generic;

```

```

    END u2;

PROCEDURE u3 IS
BEGIN
    FORALL idx IN 1 .. :o_ol_cnt
        UPDATE stock_item
            SET s_order_cnt = s_order_cnt + 1,
                s_ytd = s_ytd + :ol_quantity(idx),
                s_remote_cnt = s_remote_cnt + :s_remote(idx),
                s_quantity = s_quantity - :ol_quantity(idx) +
                    DECODE(sign(s_quantity
- :ol_quantity(idx) -
10),-1,91,0)
                WHERE i_id = :ol_i_id(idx)
                AND s_w_id = :ol_supply_w_id(idx)
                RETURNING i_price, i_name, s_quantity,
s_dist_03,
                    DECODE(instr(i_data,'ORIGINAL'), 0, 'G',
                    DECODE(instr(s_data,'ORIGINAL'), 0, 'G',
'B'))
        BULK COLLECT
INTO :i_price, :i_name, :s_quantity, initnew.s_dist,
:brand_generic;

    END u3;

PROCEDURE u4 IS
BEGIN
    FORALL idx IN 1 .. :o_ol_cnt
        UPDATE stock_item
            SET s_order_cnt = s_order_cnt + 1,
                s_ytd = s_ytd + :ol_quantity(idx),
                s_remote_cnt = s_remote_cnt + :s_remote(idx),
                s_quantity = s_quantity - :ol_quantity(idx) +
                    DECODE(sign(s_quantity
- :ol_quantity(idx) -
10),-1,91,0)
                WHERE i_id = :ol_i_id(idx)
                AND s_w_id = :ol_supply_w_id(idx)
                RETURNING i_price, i_name, s_quantity,
s_dist_04,
                    DECODE(instr(i_data,'ORIGINAL'), 0, 'G',
                    DECODE(instr(s_data,'ORIGINAL'), 0, 'G',
'B'))
        BULK COLLECT
INTO :i_price, :i_name, :s_quantity, initnew.s_dist,
:brand_generic;

    END u4;

PROCEDURE u5 IS
BEGIN
    FORALL idx IN 1 .. :o_ol_cnt
        UPDATE stock_item
            SET s_order_cnt = s_order_cnt + 1,
                s_ytd = s_ytd + :ol_quantity(idx),
                s_remote_cnt = s_remote_cnt + :s_remote(idx),
                s_quantity = s_quantity - :ol_quantity(idx) +
                    DECODE(sign(s_quantity
- :ol_quantity(idx) -
10),-1,91,0)
                WHERE i_id = :ol_i_id(idx)
                AND s_w_id = :ol_supply_w_id(idx)
                RETURNING i_price, i_name, s_quantity,
s_dist_05,
                    DECODE(instr(i_data,'ORIGINAL'), 0, 'G',
                    DECODE(instr(s_data,'ORIGINAL'), 0, 'G',
'B'))
        BULK COLLECT
INTO :i_price, :i_name, :s_quantity, initnew.s_dist,
:brand_generic;

```

```

    END u5;

PROCEDURE u6 IS
BEGIN
    FORALL idx IN 1 .. :o_ol_cnt
        UPDATE stock_item
            SET s_order_cnt = s_order_cnt + 1,
                s_ytd = s_ytd + :ol_quantity(idx),
                s_remote_cnt = s_remote_cnt + :s_remote(idx),
                s_quantity = s_quantity - :ol_quantity(idx) +
                    DECODE(sign(s_quantity
- :ol_quantity(idx) -
10),-1,91,0)
                WHERE i_id = :ol_i_id(idx)
                AND s_w_id = :ol_supply_w_id(idx)
                RETURNING i_price, i_name, s_quantity,
s_dist_06,
                    DECODE(instr(i_data,'ORIGINAL'), 0, 'G',
                    DECODE(instr(s_data,'ORIGINAL'), 0, 'G',
'B'))
        BULK COLLECT
INTO :i_price, :i_name, :s_quantity, initnew.s_dist,
:brand_generic;

    END u6;

PROCEDURE u7 IS
BEGIN
    FORALL idx IN 1 .. :o_ol_cnt
        UPDATE stock_item
            SET s_order_cnt = s_order_cnt + 1,
                s_ytd = s_ytd + :ol_quantity(idx),
                s_remote_cnt = s_remote_cnt + :s_remote(idx),
                s_quantity = s_quantity - :ol_quantity(idx) +
                    DECODE(sign(s_quantity
- :ol_quantity(idx) -
10),-1,91,0)
                WHERE i_id = :ol_i_id(idx)
                AND s_w_id = :ol_supply_w_id(idx)
                RETURNING i_price, i_name, s_quantity,
s_dist_07,
                    DECODE(instr(i_data,'ORIGINAL'), 0, 'G',
                    DECODE(instr(s_data,'ORIGINAL'), 0, 'G',
'B'))
        BULK COLLECT
INTO :i_price, :i_name, :s_quantity, initnew.s_dist,
:brand_generic;

    END u7;

PROCEDURE u8 IS
BEGIN
    FORALL idx IN 1 .. :o_ol_cnt
        UPDATE stock_item
            SET s_order_cnt = s_order_cnt + 1,
                s_ytd = s_ytd + :ol_quantity(idx),
                s_remote_cnt = s_remote_cnt + :s_remote(idx),
                s_quantity = s_quantity - :ol_quantity(idx) +
                    DECODE(sign(s_quantity
- :ol_quantity(idx) -
10),-1,91,0)
                WHERE i_id = :ol_i_id(idx)
                AND s_w_id = :ol_supply_w_id(idx)
                RETURNING i_price, i_name, s_quantity,
s_dist_08,
                    DECODE(instr(i_data,'ORIGINAL'), 0, 'G',
                    DECODE(instr(s_data,'ORIGINAL'), 0, 'G',
'B'))
        BULK COLLECT
INTO :i_price, :i_name, :s_quantity, initnew.s_dist,
:brand_generic;

```

```

END u8;

PROCEDURE u9 IS
BEGIN
  FORALL idx IN 1 .. :o_ol_cnt
    UPDATE stock_item
    SET s_order_cnt = s_order_cnt + 1,
    s_ytd = s_ytd + :ol_quantity(idx),
    s_remote_cnt = s_remote_cnt + :s_remote(idx),
    s_quantity = s_quantity - :ol_quantity(idx) +
      DECODE(sign(s_quantity
- :ol_quantity(idx) -
10),-1,91,0)
    WHERE i_id = :ol_i_id(idx)
    AND s_w_id = :ol_supply_w_id(idx)
    RETURNING i_price, i_name, s_quantity,
s_dist_09,
      DECODE(instr(i_data,'ORIGINAL'), 0, 'G',
      DECODE(instr(s_data,'ORIGINAL'), 0, 'G',
'B'))
    BULK COLLECT
INTO :i_price, :i_name, :s_quantity, initnew.s_dist,
      :brand_generic;
  END u9;

PROCEDURE u10 IS
BEGIN
  FORALL idx IN 1 .. :o_ol_cnt
    UPDATE stock_item
    SET s_order_cnt = s_order_cnt + 1,
    s_ytd = s_ytd + :ol_quantity(idx),
    s_remote_cnt = s_remote_cnt + :s_remote(idx),
    s_quantity = s_quantity - :ol_quantity(idx) +
      DECODE(sign(s_quantity
- :ol_quantity(idx) -
10),-1,91,0)
    WHERE i_id = :ol_i_id(idx)
    AND s_w_id = :ol_supply_w_id(idx)
    RETURNING i_price, i_name, s_quantity,
s_dist_10,
      DECODE(instr(i_data,'ORIGINAL'), 0, 'G',
      DECODE(instr(s_data,'ORIGINAL'), 0, 'G',
'B'))
    BULK COLLECT
INTO :i_price, :i_name, :s_quantity, initnew.s_dist,
      :brand_generic;
  END u10;

PROCEDURE fix_items IS
  rows_lost BINARY_INTEGER;
  max_index BINARY_INTEGER;
  temp_index BINARY_INTEGER;
BEGIN
  -- gotta shift price, name, s_quantity, brand_generic,
s_dist, ol_amount
  idx := 1;
  -- found 0 bad rows
  rows_lost := 0;
  -- so many rows in out array to begin with
  max_index := sql%rowcount;

  WHILE (max_index != :o_ol_cnt) LOOP

  -- find item where item ids dont match
  WHILE (idx <= sql%rowcount AND
    sql%bulk_rowcount(idx + rows_lost) =
1)
    LOOP
      idx := idx + 1;
  END LOOP;

  -- shift the items please
  temp_index := max_index;
  WHILE (temp_index >= idx + rows_lost) LOOP
    :i_price(temp_index +
1) := :i_price(temp_index);
    :i_name(temp_index +
1) := :i_name(temp_index);
    :s_quantity(temp_index +
1) := :s_quantity(temp_index) +
    initnew.s_dist(temp_index + 1) :=
    initnew.s_dist(temp_index);
    :brand_generic(temp_index +
1) := :brand_generic(temp_index);
    temp_index := temp_index - 1;
  END LOOP;

  -- values for the non-existent items if not at end
  IF (idx + rows_lost <= :o_ol_cnt) THEN
    :i_price(idx + rows_lost) := 0;
    :i_name(idx + rows_lost) := NULL;
    :s_quantity(idx + rows_lost) := 0;
    initnew.s_dist(idx + rows_lost) := NULL;
    :brand_generic(idx + rows_lost) := NULL;

  -- one more bad row
  rows_lost := rows_lost + 1;
  max_index := max_index + 1;
  END IF;

  END LOOP;
  END fix_items;

BEGIN
  LOOP BEGIN
    UPDATE dist SET d_next_o_id = d_next_o_id + 1
    WHERE d_id = :d_id AND d_w_id = :w_id
    RETURNING d_tax, d_next_o_id-1
    INTO :d_tax, :o_id;

    SELECT c_discount, c_last, c_credit, w_tax
    INTO :c_discount, :c_last, :c_credit, :w_tax
    FROM cust, ware
    WHERE c_id = :c_id AND c_d_id = :d_id AND
c_w_id = :w_id
    AND w_id = :w_id;

    INSERT INTO nord (no_o_id, no_d_id, no_w_id)
    VALUES (:o_id, :d_id, :w_id);
    INSERT INTO ordr (o_id,o_d_id, o_w_id, o_c_id,
o_entry_d,
      o_carrier_id, o_ol_cnt, o_all_local)
    VALUES (:o_id, :d_id, :w_id, :c_id,
      :cr_date, 11, :o_ol_cnt, :o_all_local);

  -- copying :d_id in local variable is important - lots
of instr.
  dummy_local := :d_id;

  IF (dummy_local = 1) THEN u1; END IF;

  IF (dummy_local = 2) THEN u2; END IF;

  IF (dummy_local = 3) THEN u3; END IF;

  IF (dummy_local = 4) THEN u4; END IF;

  IF (dummy_local = 5) THEN u5; END IF;

  IF (dummy_local = 6) THEN u6; END IF;

  IF (dummy_local = 7) THEN u7; END IF;

  IF (dummy_local = 8) THEN u8; END IF;

  IF (dummy_local = 9) THEN u9; END IF;

  IF (dummy_local = 10) THEN u10; END IF;

  -- cache the no of rows processed
  dummy_local := sql%rowcount;

  -- fix the rows if necessary
  IF (dummy_local != :o_ol_cnt) THEN fix_items;
  END IF;

  -- calculate ol_amount

  FOR idx IN 1 ..:o_ol_cnt LOOP
    :ol_amount(idx) := :ol_quantity(idx)*:i_price(idx);
  END LOOP;

  FORALL idx IN 1..:o_ol_cnt
    INSERT INTO ordl
      (ol_o_id, ol_d_id, ol_w_id, ol_number,
ol_delivery_d, ol_i_id,
      ol_supply_w_id,
ol_quantity,ol_amount,ol_dist_info)
    VALUES (:o_id, :d_id, :w_id, initnew.idx1arr(idx),
    initnew.nulldate,
      :ol_i_id(idx), :ol_supply_w_id(idx),
      :ol_quantity(idx), :ol_amount(idx),
    initnew.s_dist(idx));

  IF (dummy_local != :o_ol_cnt) THEN
    :o_ol_cnt := dummy_local;
    ROLLBACK;
  END IF;

  EXIT;

  EXCEPTION
  WHEN not_serializable OR deadlock OR
snapshot_too_old THEN
    ROLLBACK;
    :retry := :retry + 1;
  END;
  END LOOP;
  END;

===== views.sql =====

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 stock s, item i  
 where i.i_id = s.s_i_id  
/
```


Appendix C: RTE Scripts

```

.....
rte01.conf
.....
#
# tpcC.conf : configuration file for TPC-C
#
#

STARTGROUP
STARTRTE
  RTEHOST = rte01
  STARTSUT
    SUTHOST = cl01a
    SUTTERM = 750
    SUTPASS = /w00
    SUTLOG = 0
  ENDSUT
  STARTSUT
    SUTHOST = cl01a
    SUTTERM = 500
    SUTPASS = /w01
    SUTLOG = 0
  ENDSUT
  STARTSUT
    SUTHOST = cl01b
    SUTTERM = 500
    SUTPASS = /w02
    SUTLOG = 0
  ENDSUT
  STARTSUT
    SUTHOST = cl01b
    SUTTERM = 500
    SUTPASS = /w03
    SUTLOG = 0
  ENDSUT
  STARTSUT
    SUTHOST = cl01c
    SUTTERM = 500
    SUTPASS = /w04
    SUTLOG = 0
  ENDSUT
  STARTSUT
    SUTHOST = cl01c
    SUTTERM = 500
    SUTPASS = /w05
    SUTLOG = 0
  ENDSUT
  STARTSUT
    SUTHOST = cl01d
    SUTTERM = 500
    SUTPASS = /w06
    SUTLOG = 0
  ENDSUT
  STARTSUT
    SUTHOST = cl01e
    SUTTERM = 500
    SUTPASS = /w07
    SUTLOG = 0
  ENDSUT
  STARTSUT
    SUTHOST = cl01f

```

```

    SUTTERM = 500
    SUTPASS = /w08
    SUTLOG = 0
  ENDSUT
  ENDRTE
    SYNC = 0
    PORT = 80
  #WAREHOUSE SCALE
    VAL = U11 = 11400
  #RAMP-UP TIME
    VAL = U21 = 0
  #MEASUREMENT TIME
    VAL = U31 = 5400
  #RAMP-DOWN TIME
    VAL = U41 = 0
  #NEW THINKTIME (msec)
    VAL = U51 = 12030
  #PAY THINKTIME (msec)
    VAL = U61 = 12030
  #ORD THINKTIME (msec)
    VAL = U101 = 10080
  #DEL THINKTIME (msec)
    VAL = U111 = 5040
  #STK THINKTIME (msec)
    VAL = U121 = 5040
  #NURAND CONSTANT c_id
    VAL = U131 = 1023
  #NURAND CONSTANT c_last
    VAL = U141 = 255
  #NURAND CONSTANT ol_i_id
    VAL = U151 = 8191
  #MSG OFF:0, Each Term:1, Field:2
    VAL = U161 = 0
  #NEW KEYING-TIME (msec)
    VAL = U171 = 18060
  #PAY KEYING-TIME (msec)
    VAL = U181 = 3050
  #ORD KEYING-TIME (msec)
    VAL = U191 = 2050
  #DEL KEYING-TIME (msec)
    VAL = U201 = 2050
  #STK KEYING-TIME (msec)
    VAL = U211 = 2050
  ENDGROUP
  .....
  rte02.conf.diff
  .....
  8c8
  < RTEHOST = rte01
  ---
  > RTEHOST = rte02
  10c10
  < SUTHOST = cl01a
  ---
  > SUTHOST = cl02a
  16c16
  < SUTHOST = cl01a
  ---
  > SUTHOST = cl02a
  22c22
  < SUTHOST = cl01b
  ---
  > SUTHOST = cl02b
  28c28
  < SUTHOST = cl01b
  ---
  > SUTHOST = cl02b
  34c34
  < SUTHOST = cl01c

```

```

  ---
  > SUTHOST = cl02c
  40c40
  < SUTHOST = cl01c
  ---
  > SUTHOST = cl02c
  46c46
  < SUTHOST = cl01d
  ---
  > SUTHOST = cl02d
  52c52
  < SUTHOST = cl01e
  ---
  > SUTHOST = cl02e
  58c58
  < SUTHOST = cl01f
  ---
  > SUTHOST = cl02f
  .....
  rte03.conf.diff
  .....
  8c8
  < RTEHOST = rte01
  ---
  > RTEHOST = rte03
  10c10
  < SUTHOST = cl01a
  ---
  > SUTHOST = cl03a
  16c16
  < SUTHOST = cl01a
  ---
  > SUTHOST = cl03a
  22c22
  < SUTHOST = cl01b
  ---
  > SUTHOST = cl03b
  28c28
  < SUTHOST = cl01b
  ---
  > SUTHOST = cl03b
  34c34
  < SUTHOST = cl01c
  ---
  > SUTHOST = cl03c
  40c40
  < SUTHOST = cl01c
  ---
  > SUTHOST = cl03c
  46c46
  < SUTHOST = cl01d
  ---
  > SUTHOST = cl03d
  52c52
  < SUTHOST = cl01e
  ---
  > SUTHOST = cl03e
  58c58
  < SUTHOST = cl01f
  ---
  > SUTHOST = cl03f
  .....
  rte04.conf.diff
  .....
  8c8
  < RTEHOST = rte01
  ---
  > RTEHOST = rte04
  10c10

```

```

< SUTHOST = cl01a
---
> SUTHOST = cl04a
16c16
< SUTHOST = cl01a
---
> SUTHOST = cl04a
22c22
< SUTHOST = cl01b
---
> SUTHOST = cl04b
28c28
< SUTHOST = cl01b
---
> SUTHOST = cl04b
34c34
< SUTHOST = cl01c
---
> SUTHOST = cl04c
40c40
< SUTHOST = cl01c
---
> SUTHOST = cl04c
46c46
< SUTHOST = cl01d
---
> SUTHOST = cl04d
52c52
< SUTHOST = cl01e
---
> SUTHOST = cl04e
58c58
< SUTHOST = cl01f
---
> SUTHOST = cl04f
79a80
>
.....
rte05.conf.diff
.....
8c8
< RTEHOST = rte01
---
> RTEHOST = rte05
10c10
< SUTHOST = cl01a
---
> SUTHOST = cl05a
16c16
< SUTHOST = cl01a
---
> SUTHOST = cl05a
22c22
< SUTHOST = cl01b
---
> SUTHOST = cl05b
28c28
< SUTHOST = cl01b
---
> SUTHOST = cl05b
34c34
< SUTHOST = cl01c
---
> SUTHOST = cl05c
40c40
< SUTHOST = cl01c
---
> SUTHOST = cl05c
46c46

```

```

< SUTHOST = cl01d
---
> SUTHOST = cl05d
52c52
< SUTHOST = cl01e
---
> SUTHOST = cl05e
58c58
< SUTHOST = cl01f
---
> SUTHOST = cl05f
.....
rte06.conf.diff
.....
8c8
< RTEHOST = rte01
---
> RTEHOST = rte06
10c10
< SUTHOST = cl01a
---
> SUTHOST = cl06a
16c16
< SUTHOST = cl01a
---
> SUTHOST = cl06a
22c22
< SUTHOST = cl01b
---
> SUTHOST = cl06b
28c28
< SUTHOST = cl01b
---
> SUTHOST = cl06b
34c34
< SUTHOST = cl01c
---
> SUTHOST = cl06c
40c40
< SUTHOST = cl01c
---
> SUTHOST = cl06c
46c46
< SUTHOST = cl01d
---
> SUTHOST = cl06d
52c52
< SUTHOST = cl01e
---
> SUTHOST = cl06e
58c58
< SUTHOST = cl01f
---
> SUTHOST = cl06f
.....
rte07.conf.diff
.....
8c8
< RTEHOST = rte01
---
> RTEHOST = rte07
10c10
< SUTHOST = cl01a
---
> SUTHOST = cl07a
16c16
< SUTHOST = cl01a
---
> SUTHOST = cl07a

```

```

22c22
< SUTHOST = cl01b
---
> SUTHOST = cl07b
28c28
< SUTHOST = cl01b
---
> SUTHOST = cl07b
34c34
< SUTHOST = cl01c
---
> SUTHOST = cl07c
40c40
< SUTHOST = cl01c
---
> SUTHOST = cl07c
46c46
< SUTHOST = cl01d
---
> SUTHOST = cl07d
52c52
< SUTHOST = cl01e
---
> SUTHOST = cl07e
58c58
< SUTHOST = cl01f
---
> SUTHOST = cl07f
.....
rte08.conf.diff
.....
8c8
< RTEHOST = rte01
---
> RTEHOST = rte08
10c10
< SUTHOST = cl01a
---
> SUTHOST = cl08a
16c16
< SUTHOST = cl01a
---
> SUTHOST = cl08a
22c22
< SUTHOST = cl01b
---
> SUTHOST = cl08b
28c28
< SUTHOST = cl01b
---
> SUTHOST = cl08b
34c34
< SUTHOST = cl01c
---
> SUTHOST = cl08c
40c40
< SUTHOST = cl01c
---
> SUTHOST = cl08c
46c46
< SUTHOST = cl01d
---
> SUTHOST = cl08d
52c52
< SUTHOST = cl01e
---
> SUTHOST = cl08e
58c58
< SUTHOST = cl01f

```

```

---
>          SUTHOST = cl08f
.....
rte09.conf.diff
.....
8c8
<          RTEHOST  = rte01
---
>          RTEHOST  = rte09
>
10c10
<          SUTHOST = cl01a
---
>          SUTHOST = cl09a
16c16
<          SUTHOST = cl01a
---
>          SUTHOST = cl09a
22c22
<          SUTHOST = cl01b
---
>          SUTHOST = cl09b
28c28
<          SUTHOST = cl01b
---
>          SUTHOST = cl09b
34c34
<          SUTHOST = cl01c
---
>          SUTHOST = cl09c
40c40
<          SUTHOST = cl01c
---
>          SUTHOST = cl09c
46c46
<          SUTHOST = cl01d
---
>          SUTHOST = cl09d
52c52
<          SUTHOST = cl01e
---
>          SUTHOST = cl09e
58c58
<          SUTHOST = cl01f
---
>          SUTHOST = cl09f
.....
rte10.conf.diff
.....
8c8
<          RTEHOST  = rte01
---
>          RTEHOST  = rte10
10c10
<          SUTHOST = cl01a
---
>          SUTHOST = cl10a
16c16
<          SUTHOST = cl01a
---
>          SUTHOST = cl10a
22c22
<          SUTHOST = cl01b
---
>          SUTHOST = cl10b
28c28
<          SUTHOST = cl01b
---
>          SUTHOST = cl10b
34c34
<          SUTHOST = cl01c
---
>          SUTHOST = cl10c
40c40
<          SUTHOST = cl01c
---
>          SUTHOST = cl10c
46c46
<          SUTHOST = cl01d
---
>          SUTHOST = cl10d
52c52
<          SUTHOST = cl01e
---
>          SUTHOST = cl10e
58c58
<          SUTHOST = cl01f
---
>          SUTHOST = cl10f
.....
rte11.conf.diff
.....
8c8
<          RTEHOST  = rte01
---
>          RTEHOST  = rte11
10c10
<          SUTHOST = cl01a
---
>          SUTHOST = cl11a
16c16
<          SUTHOST = cl01a
---
>          SUTHOST = cl11a
22c22
<          SUTHOST = cl01b
---
>          SUTHOST = cl11b
28c28
<          SUTHOST = cl01b
---
>          SUTHOST = cl11b
34c34
<          SUTHOST = cl01c
---
>          SUTHOST = cl11c
40c40
<          SUTHOST = cl01c
---
>          SUTHOST = cl11c
46c46
<          SUTHOST = cl01d
---
>          SUTHOST = cl11d
52c52
<          SUTHOST = cl01e
---
>          SUTHOST = cl11e
58c58
<          SUTHOST = cl01f
---
>          SUTHOST = cl11f
.....
rte12.conf.diff
.....
8c8
<          RTEHOST  = rte01
---
>          RTEHOST  = rte12
10c10
<          SUTHOST = cl01a
---
>          SUTHOST = cl12a
16c16
<          SUTHOST = cl01a
---
>          SUTHOST = cl12a
22c22
<          SUTHOST = cl01b
---
>          SUTHOST = cl12b
28c28
<          SUTHOST = cl01b
---
>          SUTHOST = cl12b
34c34
<          SUTHOST = cl01c
---
>          SUTHOST = cl12c
40c40
<          SUTHOST = cl01c
---
>          SUTHOST = cl12c
46c46
<          SUTHOST = cl01d
---
>          SUTHOST = cl12d
52c52
<          SUTHOST = cl01e
---
>          SUTHOST = cl12e
58c58
<          SUTHOST = cl01f
---
>          SUTHOST = cl12f
.....
rte13.conf.diff
.....
8c8
<          RTEHOST  = rte01
---
>          RTEHOST  = rte13
10c10
<          SUTHOST = cl01a
---
>          SUTHOST = cl13a
16c16
<          SUTHOST = cl01a
---
>          SUTHOST = cl13a
22c22
<          SUTHOST = cl01b
---
>          SUTHOST = cl13b
28c28
<          SUTHOST = cl01b
---
>          SUTHOST = cl13b
34c34
<          SUTHOST = cl01c
---
>          SUTHOST = cl13c
40c40
<          SUTHOST = cl01c
---
>          SUTHOST = cl13c
46c46
<          SUTHOST = cl01d

```

```

---
> SUTHOST = cl13d
52c52
< SUTHOST = cl01e
---
> SUTHOST = cl13e
58c58
< SUTHOST = cl01f
---
> SUTHOST = cl13f
.....
rte14.conf.diff
.....
8c8
< RTEHOST = rte01
---
> RTEHOST = rte14
10c10
< SUTHOST = cl01a
---
> SUTHOST = cl14a
16c16
< SUTHOST = cl01a
---
> SUTHOST = cl14a
22c22
< SUTHOST = cl01b
---
> SUTHOST = cl14b
28c28
< SUTHOST = cl01b
---
> SUTHOST = cl14b
34c34
< SUTHOST = cl01c
---
> SUTHOST = cl14c
40c40
< SUTHOST = cl01c
---
> SUTHOST = cl14c
46c46
< SUTHOST = cl01d
---
> SUTHOST = cl14d
52c52
< SUTHOST = cl01e
---
> SUTHOST = cl14e
58c58
< SUTHOST = cl01f
---
> SUTHOST = cl14f
.....
rte15.conf.diff
.....
8c8
< RTEHOST = rte01
---
> RTEHOST = rte15
10c10
< SUTHOST = cl01a
---
> SUTHOST = cl15a
16c16
< SUTHOST = cl01a
---
> SUTHOST = cl15a
22c22
< SUTHOST = cl15a
---
< SUTHOST = cl01b
---
> SUTHOST = cl15b
28c28
< SUTHOST = cl01b
---
> SUTHOST = cl15b
34c34
< SUTHOST = cl01c
---
> SUTHOST = cl15c
40c40
< SUTHOST = cl01c
---
> SUTHOST = cl15c
46c46
< SUTHOST = cl01d
---
> SUTHOST = cl15d
52c52
< SUTHOST = cl01e
---
> SUTHOST = cl15e
58c58
< SUTHOST = cl01f
---
> SUTHOST = cl15f
.....
rte16.conf.diff
.....
8c8
< RTEHOST = rte01
---
> RTEHOST = rte16
10c10
< SUTHOST = cl01a
---
> SUTHOST = cl16a
16c16
< SUTHOST = cl01a
---
> SUTHOST = cl16a
22c22
< SUTHOST = cl01b
---
> SUTHOST = cl16b
28c28
< SUTHOST = cl01b
---
> SUTHOST = cl16b
34c34
< SUTHOST = cl01c
---
> SUTHOST = cl16c
40c40
< SUTHOST = cl01c
---
> SUTHOST = cl16c
46c46
< SUTHOST = cl01d
---
> SUTHOST = cl16d
52c52
< SUTHOST = cl01e
---
> SUTHOST = cl16e
58c58
< SUTHOST = cl01f
---
> SUTHOST = cl16f
.....
rte17.conf.diff
.....
8c8
< RTEHOST = rte01
---
> RTEHOST = rte17
10c10
< SUTHOST = cl01a
---
> SUTHOST = cl17a
16c16
< SUTHOST = cl01a
---
> SUTHOST = cl17a
22c22
< SUTHOST = cl01b
---
> SUTHOST = cl17b
28c28
< SUTHOST = cl01b
---
> SUTHOST = cl17b
34c34
< SUTHOST = cl01c
---
> SUTHOST = cl17c
40c40
< SUTHOST = cl01c
---
> SUTHOST = cl17c
46c46
< SUTHOST = cl01d
---
> SUTHOST = cl17d
52c52
< SUTHOST = cl01e
---
> SUTHOST = cl17e
58c58
< SUTHOST = cl01f
---
> SUTHOST = cl17f
63a64
> ENDRTE
.....
rte18.conf.diff
.....
8c8
< RTEHOST = rte01
---
> RTEHOST = rte18
10c10
< SUTHOST = cl01a
---
> SUTHOST = cl18a
16c16
< SUTHOST = cl01a
---
> SUTHOST = cl18a
22c22
< SUTHOST = cl01b
---
> SUTHOST = cl18b
28c28
< SUTHOST = cl01b
---
> SUTHOST = cl18b

```



```

34c34
<      SUTHOST = cl01c
---
>      SUTHOST = cl18c
40c40
<      SUTHOST = cl01c
---
>      SUTHOST = cl18c
46c46
<      SUTHOST = cl01d
---
>      SUTHOST = cl18d
52c52
<      SUTHOST = cl01e
---
>      SUTHOST = cl18e
58c58
<      SUTHOST = cl01f
---
>      SUTHOST = cl18f
.....
rte19.conf.diff
.....
8c8
<      RTEHOST = rte01
---
>      RTEHOST = rte19
10c10
<      SUTHOST = cl01a
---
>      SUTHOST = cl19a
16c16
<      SUTHOST = cl01a
---
>      SUTHOST = cl19a
22c22
<      SUTHOST = cl01b
---
>      SUTHOST = cl19b
28c28
<      SUTHOST = cl01b
---
>      SUTHOST = cl19b
34c34
<      SUTHOST = cl01c
---
>      SUTHOST = cl19c
40c40
<      SUTHOST = cl01c
---
>      SUTHOST = cl19c
46c46
<      SUTHOST = cl01d
---
>      SUTHOST = cl19d
52c52
<      SUTHOST = cl01e
---
>      SUTHOST = cl19e
58c58
<      SUTHOST = cl01f
---
>      SUTHOST = cl19f
.....
rte20.conf.diff
.....
8c8
<      RTEHOST = rte01
---

```

```

>      RTEHOST = rte20
10c10
<      SUTHOST = cl01a
---
>      SUTHOST = cl20a
16c16
<      SUTHOST = cl01a
---
>      SUTHOST = cl20a
22c22
<      SUTHOST = cl01b
---
>      SUTHOST = cl20b
28c28
<      SUTHOST = cl01b
---
>      SUTHOST = cl20b
34c34
<      SUTHOST = cl01c
---
>      SUTHOST = cl20c
40c40
<      SUTHOST = cl01c
---
>      SUTHOST = cl20c
46c46
<      SUTHOST = cl01d
---
>      SUTHOST = cl20d
52c52
<      SUTHOST = cl01e
---
>      SUTHOST = cl20e
58c58
<      SUTHOST = cl01f
---
>      SUTHOST = cl20f
.....
rte21.conf.diff
.....
8c8
<      RTEHOST = rte01
---
>      RTEHOST = rte21
10c10
<      SUTHOST = cl01a
---
>      SUTHOST = cl21a
16c16
<      SUTHOST = cl01a
---
>      SUTHOST = cl21a
22c22
<      SUTHOST = cl01b
---
>      SUTHOST = cl21b
28c28
<      SUTHOST = cl01b
---
>      SUTHOST = cl21b
34c34
<      SUTHOST = cl01c
---
>      SUTHOST = cl21c
40c40
<      SUTHOST = cl01c
---
>      SUTHOST = cl21c
46c46

```

```

<      SUTHOST = cl01d
---
>      SUTHOST = cl21d
52c52
<      SUTHOST = cl01e
---
>      SUTHOST = cl21e
58c58
<      SUTHOST = cl01f
---
>      SUTHOST = cl21f
.....
rte22.conf.diff
.....
8c8
<      RTEHOST = rte01
---
>      RTEHOST = rte22
10c10
<      SUTHOST = cl01a
---
>      SUTHOST = cl22a
16c16
<      SUTHOST = cl01a
---
>      SUTHOST = cl22a
22c22
<      SUTHOST = cl01b
---
>      SUTHOST = cl22b
28c28
<      SUTHOST = cl01b
---
>      SUTHOST = cl22b
34c34
<      SUTHOST = cl01c
---
>      SUTHOST = cl22c
40c40
<      SUTHOST = cl01c
---
>      SUTHOST = cl22c
46c46
<      SUTHOST = cl01d
---
>      SUTHOST = cl22d
52c52
<      SUTHOST = cl01e
---
>      SUTHOST = cl22e
58c58
<      SUTHOST = cl01f
---
>      SUTHOST = cl22f
.....
rte23.conf.diff
.....
8c8
<      RTEHOST = rte01
---
>      RTEHOST = rte23
10c10
<      SUTHOST = cl01a
---
>      SUTHOST = cl23a
16c16
<      SUTHOST = cl01a
---
>      SUTHOST = cl23a

```

```

22c22      SUTHOST = cl01b
<
---
>          SUTHOST = cl23b
28c28      SUTHOST = cl01b
<
---
>          SUTHOST = cl23b
34c34      SUTHOST = cl01c
<
---
>          SUTHOST = cl23c
40c40      SUTHOST = cl01c
<
---
>          SUTHOST = cl23c
46c46      SUTHOST = cl01d
<
---
>          SUTHOST = cl23d
52c52      SUTHOST = cl01e
<
---
>          SUTHOST = cl23e
58c58      SUTHOST = cl01f
<
---
>          SUTHOST = cl23f
.....
rte24.conf.diff
.....
8c8        RTEHOST  = rte01
<
---
>          RTEHOST  = rte24
10c10      SUTHOST = cl01a
<
---
>          SUTHOST = cl24a
16c16      SUTHOST = cl01a
<
---
>          SUTHOST = cl24a
22c22      SUTHOST = cl01b
<
---
>          SUTHOST = cl24b
28c28      SUTHOST = cl01b
<
---
>          SUTHOST = cl24b
34c34      SUTHOST = cl01c
<
---
>          SUTHOST = cl24c
40c40      SUTHOST = cl01c
<
---
>          SUTHOST = cl24c
46c46      SUTHOST = cl01d
<
---
>          SUTHOST = cl24d
52c52      SUTHOST = cl01e
<
---
>          SUTHOST = cl24e
58c58      SUTHOST = cl01f
<
---
>          SUTHOST = cl24f
.....
rte01.tpcc
.....
#!/usr/bin/sh
sh rmlog /w00
sh rmlog /w01
sh rmlog /w02
sh rmlog /w03
sh rmlog /w04
sh rmlog /w05
sh rmlog /w06
sh rmlog /w07
sh rmlog /w08
./psinit -f ../data/conf -t 1
#./psinit -f ../data/conf2 -t 1
.....
rte02.tpcc
.....
#!/usr/bin/sh
sh rmlog /w00
sh rmlog /w01
sh rmlog /w02
sh rmlog /w03
sh rmlog /w04
sh rmlog /w05
sh rmlog /w06
sh rmlog /w07
sh rmlog /w08
./psinit -f ../data/conf -t 4751
.....
rte03.tpcc
.....
#!/usr/bin/sh
sh rmlog /w00
sh rmlog /w01
sh rmlog /w02
sh rmlog /w03
sh rmlog /w04
sh rmlog /w05
sh rmlog /w06
sh rmlog /w07
sh rmlog /w08
./psinit -f ../data/conf -t 9501
.....
rte04.tpcc
.....
#!/usr/bin/sh
sh rmlog /w00
sh rmlog /w01
sh rmlog /w02
sh rmlog /w03
sh rmlog /w04
sh rmlog /w05
sh rmlog /w06
sh rmlog /w07
sh rmlog /w08
./psinit -f ../data/conf -t 14251
.....
rte05.tpcc
.....
#!/usr/bin/sh
sh rmlog /w00
sh rmlog /w01
sh rmlog /w02
sh rmlog /w03
sh rmlog /w04
sh rmlog /w05
sh rmlog /w06
sh rmlog /w07
sh rmlog /w08
./psinit -f ../data/conf -t 19001
.....
rte06.tpcc
.....
#!/usr/bin/sh
sh rmlog /w00
sh rmlog /w01
sh rmlog /w02
sh rmlog /w03
sh rmlog /w04
sh rmlog /w05
sh rmlog /w06
sh rmlog /w07
sh rmlog /w08
./psinit -f ../data/conf -t 23751
.....
rte07.tpcc
.....
#!/usr/bin/sh
sh rmlog /w00
sh rmlog /w01
sh rmlog /w02
sh rmlog /w03
sh rmlog /w04
sh rmlog /w05
sh rmlog /w06
sh rmlog /w07
sh rmlog /w08
./psinit -f ../data/conf -t 28501
.....
rte08.tpcc
.....
#!/usr/bin/sh
sh rmlog /w00
sh rmlog /w01
sh rmlog /w02
sh rmlog /w03
sh rmlog /w04
sh rmlog /w05
sh rmlog /w06
sh rmlog /w07
sh rmlog /w08
./psinit -f ../data/conf -t 33251
.....
rte09.tpcc
.....
#!/usr/bin/sh
sh rmlog /w00
sh rmlog /w01
sh rmlog /w02
sh rmlog /w03
sh rmlog /w04
sh rmlog /w05
sh rmlog /w06
sh rmlog /w07
sh rmlog /w08
./psinit -f ../data/conf -t 38001
.....
rte10.tpcc
.....
#!/usr/bin/sh
sh rmlog /w00
sh rmlog /w01
sh rmlog /w02
sh rmlog /w03
sh rmlog /w04

```


Appendix D: System Tunables

===== system =====

```
*ident"@(#)system      1.18  97/06/27 SMI" /*
SVR4 1.5 */
```

* SYSTEM SPECIFICATION FILE

* moddir:

* Set the search path for modules. This has a format similar to the csh path variable. If the module isn't found in the first directory it tries the second and so on. The default is /kernel /usr/kernel

* Example:
* moddir: /kernel /usr/kernel /other/modules

* root device and root filesystem configuration:

* The following may be used to override the defaults provided by the boot program:

* rootfs: Set the filesystem type of the root.

* rootdev: Set the root device. This should be a fully expanded physical pathname.

The default is the physical pathname of the device where the boot program resides. The physical pathname is highly platform and configuration dependent.

* Example:
* rootfs:ufs

* rootdev:/sbin@1,f8000000/esp@0,800000/sd@3,0:a

* (Swap device configuration should be specified in /etc/vfstab.)

* exclude:

* Modules appearing in the moddir path which are NOT to be loaded,

* even if referenced. Note that 'exclude' accepts either a module name, or a filename which includes the directory.

* Examples:
* exclude: win
* exclude: sys/shmsys

* forceload:

* Cause these modules to be loaded at boot time, (just before mounting the root filesystem) rather than at first reference. Note that forceload expects a filename which includes the directory. Also note that loading a module does not necessarily imply that it will be installed.

* Example:
* forceload: drv/foo

* set:

* Set an integer variable in the kernel or a module to a new value. This facility should be used with caution. See system(4).

* Examples:

* To set variables in 'unix':

```
set nautopush=32
set maxusers=40
```

* To set a variable named 'debug' in the module named 'test_module'

```
set test_module:debug = 0x13
```

```
* set phymem=8388608
set semsys:msginfo_msgmap=800
set semsys:msginfo_msgmni=100
set semsys:msginfo_msgtql=160
set semsys:msginfo_msgseg=4096
#set shmsys:shminfo_shmmax=63000000000
set shmsys:shminfo_shmmax=80000000000
#set shmsys:shminfo_shmmax= 4294967295
set shmsys:shminfo_shmseg=600
set semsys:seminfo_semmmap=400
set semsys:seminfo_semmni=1000
set semsys:seminfo_semmns=6000
set semsys:seminfo_semmnu=1200
set semsys:seminfo_semmnl=1024
set semsys:seminfo_semume=200
set hddv:hddvnoretry=1
```

```
* set vol_maxvol = 4096
```

```
* forceload: sys/perfmon
set sd:sd_max_throttle=32
set maxphys = 1048576
```

```
* set hddv:hddv_max_throttle=42
set hddv:hddv_max_throttle=32
```

```
* vxvm_START (do not remove)
forceload: drv/atf
forceload: drv/pln
forceload: drv/ses
forceload: drv/vxdmp
forceload: drv/vxio
forceload: drv/vxspec
* vxvm_END (do not remove)
set vxio:vol_maxio=8192
set vxio:vol_maxkiocount=8192
set vxio:voliomem_chunk_size=1048576
set vxio:voliomem_kvmap_size=15728640
```

* SPEC configuration
* set consistent_coloring=1
set dosynctodr=0

===== init.ora =====

```
#_use_ism=false
_bump_highwater_mark_count = 100
_db_aging_stay_count = 1
_db_block_cache_protect = FALSE
_db_block_hash_buckets = 52000000
_db_file_noncontig_mblock_read_count = 1
_enable_list_io = FALSE
_log_simultaneous_copies = 128
_spin_count = 20000
background_core_dump = PARTIAL
max_dump_file_size = 10K
buffer_pool_keep = (buffers:22500000,
lru_latches:40)
buffer_pool_recycle = (buffers:300000,
lru_latches:40)
compatible = 8.1.5.0.0
control_files = /oracle/dbs/control_001
cpu_count = 64
cursor_space_for_time = TRUE
db_block_buffers = 30000000
db_block_lru_latches = 128
#db_block_max_dirty_target = 0
db_block_max_dirty_target = 13000000
db_files = 1000
db_file_multiblock_read_count = 1
db_name = tpcc
db_writer_processes = 10
distributed_transactions = 0
dml_locks = 2300
enqueue_resources = 35000
fast_start_io_target = 0
hash_join_enabled = FALSE
log_archive_start = FALSE
log_buffer = 31457280
log_checkpoint_interval = 100000000
log_checkpoint_timeout = 0
log_checkpoints_to_alert = TRUE
max_rollback_segments = 600
#open_cursors = 500
open_cursors = 1000
parallel_max_servers = 300
processes = 1500
recovery_parallelism = 64
replication_dependency_tracking = FALSE
rollback_segments = (t1, t2, t3, t4, t5)
rollback_segments = (t6, t7, t8, t9, t10)
```

```
sessions = 1500
shared_pool_size = 60000000
#timed_statistics = TRUE
#timed_os_statistics = 30
transaction_auditing = FALSE
transactions_per_rollback_segment = 1
transactions = 1500
```

```
===== client 1 =====
REGEDIT4
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Parameters]
"BandwidthLevel"=dword:ffffff
"ListenBackLog"=dword:00000019
"MemoryCacheSize"=dword:00000000
"ObjectCacheTTL"=dword:ffffff
"PoolThreadLimit"=dword:0000020d
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Parameters\Filter]
"FilterType"=dword:00000000
"NumGrantSites"=dword:00000000
"NumDenySites"=dword:00000000
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Parameters\MimeMap]
"text/html,html,h"=""
"image/gif,gif,g"=""
"image/jpeg,jpg,:"=""
"text/plain,txt,0"=""
"text/html,html,h"=""
"image/jpeg,jpeg,:"=""
"image/jpeg,jpe,:"=""
"image/bmp,bmp,:"=""
"application/octet-stream,*"=""
"application/pdf,pdf,5"=""
"application/octet-stream,bin,5"=""
"application/oda,oda,5"=""
"application/zip,zip,9"=""
"application/rtf,rtf,5"=""
"application/postscript,ps,5"=""
"application/postscript,ai,5"=""
"application/postscript,eps,5"=""
"application/mac-binhex40,hqx,4"=""
"application/msword,doc,5"=""
"application/msword,dot,5"=""
"application/winhelp,help,5"=""
"video/mpeg,mpeg,:"=""
"video/mpeg,mpg,:"=""
"video/mpeg,mpe,:"=""
"video/x-msvideo,avi,<"=""
"video/quicktime,qt,:"=""
"video/quicktime,mov,:"=""
"video/x-sgi-movie,movie,<"=""
"x-world/x-vmrl,wrl,5"=""
"x-world/x-vmrl,xaf,5"=""
"x-world/x-vmrl,xof,5"=""
"x-world/x-vmrl,flr,5"=""
"x-world/x-vmrl,wrz,5"=""
"application/x-director,dcr,5"=""
"application/x-director,dir,5"=""
"application/x-director,dxr,5"=""
"image/cis-cod,cod,5"=""
"image/x-cmx,cmx,5"=""
"application/envoy,envy,5"=""
"application/x-msaccess,mdb,5"=""
```

```
"application/x-mscardfile,crd,5"=""
"application/x-msclip,clip,5"=""
"application/octet-stream,exe,5"=""
"application/x-msexcel,xla,5"=""
"application/x-msexcel,xlc,5"=""
"application/x-msexcel,xlm,5"=""
"application/x-msexcel,xls,5"=""
"application/x-msexcel,xlt,5"=""
"application/x-msexcel,xlw,5"=""
"application/x-msmediaview,m13,5"=""
"application/x-msmediaview,m14,5"=""
"application/x-msmoney,mny,5"=""
"application/x-mspowerpoint,ppt,5"=""
"application/x-msproject,mpp,5"=""
"application/x-mspublisher,pub,5"=""
"application/x-msterial,rm,5"=""
"application/x-msworks,wks,5"=""
"application/x-mswrite,wri,5"=""
"application/x-msmetafile,wmf,5"=""
"application/x-csh,csh,5"=""
"application/x-dvi,dvi,5"=""
"application/x-hdf,hdf,5"=""
"application/x-latex,latex,5"=""
"application/x-netcdf,nc,5"=""
"application/x-netcdf,cdf,5"=""
"application/x-sh,sh,5"=""
"application/x-tcl,tcl,5"=""
"application/x-tex,tex,5"=""
"application/x-texinfo,texinfo,5"=""
"application/x-texinfo,texi,5"=""
"application/x-troff,t,5"=""
"application/x-troff,tr,5"=""
"application/x-troff,roff,5"=""
"application/x-troff-man,man,5"=""
"application/x-troff-me,me,5"=""
"application/x-troff-ms,ms,5"=""
"application/x-wais-source,src,7"=""
"application/x-bcpio,bcpio,5"=""
"application/x-cpio,cpio,5"=""
"application/x-gtar,gtar,9"=""
"application/x-shar,shar,5"=""
"application/x-sv4cpio,sv4cpio,5"=""
"application/x-sv4crc,sv4crc,5"=""
"application/x-tar,tar,5"=""
"application/x-ustar,ustar,5"=""
"audio/basic,au,<"=""
"audio/basic,snd,<"=""
"audio/x-aiff,aif,<"=""
"audio/x-aiff,aiff,<"=""
"audio/x-aiff,aifc,<"=""
"audio/x-wav,wav,<"=""
"audio/x-pn-realaudio,ram,<"=""
"image/ief,ief,:"=""
"image/tiff,tiff,:"=""
"image/tiff,tif,:"=""
"image/x-cmu-raster,ras,:"=""
"image/x-portable-anymap,pnm,:"=""
"image/x-portable-bitmap,pbm,:"=""
"image/x-portable-graymap,pgm,:"=""
"image/x-portable-pixmap,ppm,:"=""
"image/x-rgb,rgb,:"=""
"image/x-xbitmap,xbm,:"=""
"image/x-xpixmap,xpm,:"=""
"image/x-xwindowdump,xwd,:"=""
"text/html,stm,h"=""
"text/plain,bas,0"=""
"text/plain,c,0"=""
"text/plain,h,0"=""
"text/richtext,rtx,0"=""
```

```
"text/tab-separated-values,tsv,0"=""
"text/x-setext,etx,0"=""
"application/x-perfmon,pmc,5"=""
"application/x-perfmon,pma,5"=""
"application/x-perfmon,pmr,5"=""
"application/x-perfmon,pml,5"=""
"application/x-perfmon,pmw,5"=""
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEASystems\TUXEDO\6.4\Environment]
"TUXDIR"="C:\TUXEDO"
"NLSPATH"="C:\TUXEDO\locale\LC"
"ULOGDIR"="C:\TUXEDO"
"ULOGPFX"="C:\ULOG"
"ULOGOUT"=dword:00000001
"TUXIPC_MSG_BYTES"=dword:00010000
"TUXIPC_MSG_HDRS"=dword:00001fc0
"TUXIPC_MSG_QUEUESES"=dword:00000320
"TUXIPC_MSG_QUEUE_BYTES"=dword:00010000
"TUXIPC_MSG_SEG_BYTES"=dword:00000040
"TUXIPC_MSG_SEGS"=dword:00007fff
"TUXIPC_PROC"=dword:00001770
"TUXIPC_SEM"=dword:00001770
"TUXIPC_SEM_IDS"=dword:00001770
"TUXIPC_SEM_UNDO"=dword:00001770
"TUXIPC_SHM_PROCS"=dword:000186a0
"TUXIPC_SHM_SEGS"=dword:0000c350
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEASystems\TUXEDO\6.4\Environment\Services]
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEASystems\TUXEDO\6.4\Environment\Services\3050]
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC]
"Type"=dword:00000020
"Start"=dword:00000002
"ErrorControl"=dword:00000000
"ImagePath"=hex(2):43,3a,5c,57,49,4e,4e,54,5c,53,79,73,74,65,6d,33,32,5c,69,6e,65,74,73,72,76,5c,69,6e,65,74,69,6e,66,6f,2e,65,78,65,00
"DisplayName"="World Wide Web Publishing Service"
"DependOnService"=hex(7):52,50,43,53,53,00,4e,54,4c,4d,53,53,50,00,00
"DependOnGroup"=hex(7):00
"ObjectName"="LocalSystem"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Parameters]
"MajorVersion"=dword:00000002
"MinorVersion"=dword:00000000
"AdminName"="Administrator"
"AdminEmail"="Admin@corp.com"
"MaxConnections"=dword:000186a0
"LogType"=dword:00000000
"LogFileDirectory"=hex(2):25,53,79,73,74,65,6d,52,6f,6f,74,25,5c,53,79,73,74,65,6d,6d,6d,33,32,5c,4c,6f,67,46,69,6c,65,73,00
"LogFileTruncateSize"=dword:01400000
"LogFilePeriod"=dword:00000001
```

```
"LogFileFormat"=dword:00000000
"LogSqlDataSource"="HTTPLOG"
"LogSqlTableName"="Internetlog"
"LogSqlUserName"="InternetAdmin"
"LogSqlPassword"="sqllog"
"Authorization"=dword:00000005
"AnonymousUserName"="IUSR_CL01"
"Default Load File"="Default.htm"
"Dir Browse Control"=dword:4000001e
"CheckForWAISDB"=dword:00000000
"CacheExtensions"=dword:00000001
"GlobalExpire"=dword:ffffff
"ServerSideIncludesEnabled"=dword:00000001
"ServerSideIncludesExtension"=".stm"
"DebugFlags"=dword:00000008
"ScriptTimeout"=dword:00000384
"ConnectionTimeout"=dword:00000384
"InstallPath"="C:\WINNT\System32\inetsrv"
"SecurePort"=dword:000001bb
"Filter
DLLs"="C:\WINNT\System32\inetsrv\lsspifilt.dll"
"AccessDeniedMessage"="エラー : アクセスは拒否され
ました。"
"NTAuthenticationProviders"="NTLM"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Script Map]
".idc"="C:\WINNT\System32\inetsrv\httpodbc.dll"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Virtual Roots]
"/,"="C:\inetpub\wwwroot,,1"
"/tpc,"="C:\client\tpapl\Release,,5"
"/tpc4,"="C:\client\tpapl\tpaplkeep,,5"
"/Scripts,"="C:\inetpub\scripts,,4"
"/iisadmin,"="C:\WINNT\System32\inetsrv\iisad
min,,1"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Performance]
"Library"="w3ctrs.DLL"
"Open"="OpenW3PerformanceData"
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"Last Counter"=dword:00000790
"Last Help"=dword:00000791
"First Counter"=dword:00000758
"First Help"=dword:00000759
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Security]
"Security"=hex:01,00,14,80,c0,00,00,cc,00,00,
00,14,00,00,00,34,00,00,00,02,\
00,20,00,01,00,00,00,02,80,18,00,ff,01,0f,00,01,
01,00,00,00,00,01,00,00,\
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,00,0
0,18,00,8d,01,02,00,01,01,00,\
00,00,00,00,01,00,00,00,00,5c,00,53,00,00,00,1
c,00,fd,01,02,00,01,02,00,00,\
00,00,00,05,20,00,00,00,23,02,00,00,65,00,72,0
0,00,00,1c,00,ff,01,0f,00,01,\
02,00,00,00,00,05,20,00,00,00,20,02,00,00,6
5,00,72,00,00,00,1c,00,ff,01,\
0f,00,01,02,00,00,00,00,05,20,00,00,00,25,0
2,00,00,65,00,72,00,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,05,12,00,0
0,00,25,02,00,00,01,01,00,00,\
```

```
00,00,00,05,12,00,00,00,01,01,00,00,00,00,00,0
5,12,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\W3SAMP]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Enum]
"0"="Root\LEGACY_W3SVC\0000"
"Count"=dword:00000001
"NextInstance"=dword:00000001
```

```
REGEDIT4
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Fujitsu\T
PC-C ISAPI Application]
"Term_Base"=dword:1
"NumWarehouses"=dword:2c88
"MaxUsers"=dword:1bd50
"MaxTerm of Client"=dword:128e
"CONTROL_Flag"=dword:00000000
```

```
#
# ubbconfig : TUXEDO configuration file
#
```

```
*RESOURCES
IPCKEY 80952
MASTER SITE1
#UID 1
#GID 1
PERM 0660
MAXACCESSERS 800
MAXSERVERS 200
MAXSERVICES 1000
MODEL SHM
LDBAL Y
```

```
*MACHINES
CL01 LMD=SITE1
APPDIR="c:\sv-aplora"
TUXCONFIG="c:\client\tuxconfig"
TUXDIR="c:\tuxedo"
ULOGPFX="c:\tuxlogtest"
```

```
*GROUPS
group1 LMD=SITE1 GRPNO=1
```

```
*SERVERS
DEFAULT: RESTART=Y MAXGEN=5
RQPERM=0660
tpccfml SRVGRP=group1 RQADDR=newq1
SRVID=1 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq2
SRVID=2 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq3
SRVID=3 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq4
SRVID=4 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq5
SRVID=5 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq6
SRVID=6 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
```

```
tpccfml SRVGRP=group1 RQADDR=newq7
SRVID=7 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq8
SRVID=8 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq9
SRVID=9 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq10
SRVID=10 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq11
SRVID=11 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq12
SRVID=12 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq13
SRVID=13 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq14
SRVID=14 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq15
SRVID=15 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=newq16
SRVID=16 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=newq17
SRVID=17 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=newq18
SRVID=18 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=newq19
SRVID=19 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=newq20
SRVID=20 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=newq21
SRVID=21 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=newq22
SRVID=22 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfml SRVGRP=group1 RQADDR=newq23
SRVID=23 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfml SRVGRP=group1 RQADDR=newq24
SRVID=24 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfml SRVGRP=group1 RQADDR=newq25
SRVID=25 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfml SRVGRP=group1 RQADDR=newq26
SRVID=26 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfml SRVGRP=group1 RQADDR=newq27
SRVID=27 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfml SRVGRP=group1 RQADDR=newq28
SRVID=28 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfml SRVGRP=group1 RQADDR=newq29
SRVID=29 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
```

```
*SERVICES
"NEWORDER" TRANTIME=0
  SRVGRP=group1
"PAYMENT" TRANTIME=0
  SRVGRP=group1
"ORDERSTATUS" TRANTIME=0
  SRVGRP=group1
"DELIVERY" TRANTIME=0
  SRVGRP=group1
"STOCKLVL" TRANTIME=0
  SRVGRP=group1
```

```
*ROUTING
```

```
===== client 2 =====
```

```
REGEDIT4
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Parameters]
"BandwidthLevel"=dword:ffffff
"ListenBackLog"=dword:00000019
"MemoryCacheSize"=dword:00000000
"ObjectCacheTTL"=dword:ffffff
"PoolThreadLimit"=dword:0000020d
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Parameters\Filter]
"FilterType"=dword:00000000
"NumGrantSites"=dword:00000000
"NumDenySites"=dword:00000000
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Parameters\MimeMap]
"text/html,html,h"=""
"image/gif,gif,g"=""
"image/jpeg,jpg,j"=""
"text/plain,txt,t"=""
"text/html,html,h"=""
"image/jpeg,jpeg,j"=""
"image/jpeg,jpe,j"=""
"image/bmp,bmp,b"=""
"application/octet-stream,*"=""
"application/pdf,pdf,p"=""
"application/octet-stream,bin"=""
"application/oda,oda,o"=""
"application/zip,zip,z"=""
"application/rtf,rtf,r"=""
"application/postscript,ps,p"=""
"application/postscript,ai,a"=""
"application/postscript,eps,e"=""
"application/mac-binhex40,hqx,h"=""
"application/msword,doc,d"=""
"application/msword,dot,d"=""
"application/winhlp,hlp,h"=""
"video/mpeg,mpeg,m"=""
"video/mpeg,mpg,m"=""
"video/mpeg,mpe,m"=""
"video/x-msvideo,avi,a"=""
"video/quicktime,qt,q"=""
"video/quicktime,mov,m"=""
"video/x-sgi-movie,movie,m"=""
"x-world/x-vrml,vrml,v"=""
"x-world/x-vrml,xaf,x"=""
"x-world/x-vrml,xof,x"=""
```

```
"x-world/x-vrml,flr,f"=""
"x-world/x-vrml,wrz,w"=""
"application/x-director,dcr,d"=""
"application/x-director,dir,d"=""
"application/x-director,dxr,d"=""
"image/cis-cod,cod,c"=""
"image/x-cmx,cmx,c"=""
"application/envoy,evy,e"=""
"application/x-msaccess,mdb,m"=""
"application/x-mscardfile,crd,c"=""
"application/x-msclip,clip,c"=""
"application/octet-stream,exe,e"=""
"application/x-msexcel,xla,x"=""
"application/x-msexcel,xlc,x"=""
"application/x-msexcel,xlm,x"=""
"application/x-msexcel,xls,x"=""
"application/x-msexcel,xlt,x"=""
"application/x-msexcel,xlw,x"=""
"application/x-msmediaview,m13,m"=""
"application/x-msmediaview,m14,m"=""
"application/x-msmoney,mny,m"=""
"application/x-mspowerpoint,ppt,p"=""
"application/x-msproject,mpp,m"=""
"application/x-mspublisher,pub,p"=""
"application/x-msterminal,rm,r"=""
"application/x-msworks,wks,w"=""
"application/x-mswrite,wri,w"=""
"application/x-msmetafile,wmf,w"=""
"application/x-csh,csh,c"=""
"application/x-dvi,dvi,d"=""
"application/x-hdf,hdf,h"=""
"application/x-latex,latex,l"=""
"application/x-netcdf,nc,n"=""
"application/x-netcdf,cdf,c"=""
"application/x-sh,sh,s"=""
"application/x-tcl,tcl,t"=""
"application/x-tex,tex,t"=""
"application/x-texinfo,texinfo,t"=""
"application/x-texinfo,txi,t"=""
"application/x-troff,t,t"=""
"application/x-troff,tr,t"=""
"application/x-troff,roff,r"=""
"application/x-troff-man,man,m"=""
"application/x-troff-me,me,m"=""
"application/x-troff-ms,ms,m"=""
"application/x-wais-source,src,s"=""
"application/x-bcpio,bcpio,b"=""
"application/x-cpio,cpio,c"=""
"application/x-gtar,gtar,g"=""
"application/x-shar,shar,s"=""
"application/x-sv4cpio,sv4cpio,s"=""
"application/x-sv4crc,sv4crc,s"=""
"application/x-tar,tar,t"=""
"application/x-ustar,ustar,u"=""
"audio/basic,au,a"=""
"audio/basic,snd,s"=""
"audio/x-aiff,aif,a"=""
"audio/x-aiff,aiff,a"=""
"audio/x-aiff,aifc,a"=""
"audio/x-wav,wav,w"=""
"audio/x-pn-realaudio,ram,r"=""
"image/ief,ief,i"=""
"image/tiff,tiff,t"=""
"image/tiff,tif,t"=""
"image/x-cmu-raster,ras,r"=""
"image/x-portable-anymap,pnm,p"=""
"image/x-portable-bitmap,pbm,p"=""
"image/x-portable-graymap,pgm,p"=""
"image/x-portable-pixmap,ppm,p"=""
```

```
"image/x-rgb,rgb,r"=""
"image/x-xbitmap,xbm,x"=""
"image/x-xpixmap,xpm,x"=""
"image/x-xwindowdump,xwd,x"=""
"text/html,stm,s"=""
"text/plain,bas,b"=""
"text/plain,c,c"=""
"text/plain,h,h"=""
"text/plain,h,0"=""
"text/richtext,rtx,r"=""
"text/tab-separated-values,tsv,t"=""
"text/x-setext,etx,e"=""
"application/x-perfmon,pmc,p"=""
"application/x-perfmon,pma,p"=""
"application/x-perfmon,pmr,p"=""
"application/x-perfmon,pml,p"=""
"application/x-perfmon,pmw,p"=""
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Performance]
"Library"="infoctrs.DLL"
"Open"="OpenINFOPerformanceData"
"Close"="CloseINFOPerformanceData"
"Collect"="CollectINFOPerformanceData"
"Last Counter"=dword:00000756
"Last Help"=dword:00000757
"First Counter"=dword:00000738
"First Help"=dword:00000739
```

```
REGEDIT4
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEASystems\TUXEDO\6.4\Environment]
"TUXDIR"="C:\TUXEDO"
"NLSPATH"="C:\TUXEDO\locale\LC"
"ULOGDIR"="C:\TUXEDO"
"ULOGPFX"="C:\ULOG"
"ULOGOUT"=dword:00000001
"TUXIPC_MSG_BYTES"=dword:00010000
"TUXIPC_MSG_HDRS"=dword:00001fc0
"TUXIPC_MSG_QUEUES"=dword:00000320
"TUXIPC_MSG_QUEUE_BYTES"=dword:00010000
"TUXIPC_MSG_SEG_BYTES"=dword:00000040
"TUXIPC_MSG_SEGS"=dword:00007fff
"TUXIPC_PROC"=dword:00001770
"TUXIPC_SEM"=dword:00001770
"TUXIPC_SEM_IDS"=dword:00001770
"TUXIPC_SEM_UNDO"=dword:00001770
"TUXIPC_SHM_PROCS"=dword:000186a0
"TUXIPC_SHM_SEGS"=dword:0000c350
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEASystems\TUXEDO\6.4\Environment\Services]
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEASystems\TUXEDO\6.4\Environment\Services\3050]
```

```
REGEDIT4
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC]
"Type"=dword:00000020
"Start"=dword:00000002
"ErrorControl"=dword:00000000
"ImagePath"=hex(2):43,3a,5c,57,49,4e,4e,54,5c,53,79,73,74,65,6d,33,32,5c,69,6e,\65,74,73,72,76,5c,69,6e,65,74,69,6e,66,6f,2e,65,78,65,00
```



```
"DisplayName"="World Wide Web Publishing
Service"
"DependOnService"=hex(7):52,50,43,53,53,00,4e,
54,4c,4d,53,53,50,00,00
"DependOnGroup"=hex(7):00
"ObjectName"="LocalSystem"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters]
"MajorVersion"=dword:00000002
"MinorVersion"=dword:00000000
"AdminName"="Administrator"
"AdminEmail"="Admin@corp.com"
"MaxConnections"=dword:000186a0
"LogType"=dword:00000000
"LogFileDirectory"=hex(2):25,53,79,73,74,65,6d,5
2,6f,6f,74,25,5c,53,79,73,74,\
65,6d,33,32,5c,4c,6f,67,46,69,6c,65,73,00
"LogFileTruncateSize"=dword:01400000
"LogFilePeriod"=dword:00000001
"LogFileFormat"=dword:00000000
"LogSqlDataSource"="HTTPLOG"
"LogSqlTableName"="Internetlog"
"LogSqlUserName"="InternetAdmin"
"LogSqlPassword"="sqllog"
"Authorization"=dword:00000005
"AnonymousUserName"="IUSR_CL23"
"Default Load File"="Default.htm"
"Dir Browse Control"=dword:4000001e
"CheckForWAISDB"=dword:00000000
"CacheExtensions"=dword:00000001
"GlobalExpire"=dword:ffffff
"ServerSideIncludesEnabled"=dword:00000001
"ServerSideIncludesExtension"=".stm"
"DebugFlags"=dword:00000008
"ScriptTimeout"=dword:00000384
"ConnectionTimeout"=dword:00000384
"InstallPath"="C:\WINNT\System32\inetsrv"
"SecurePort"=dword:000001bb
"Filter
DLLs"="C:\WINNT\System32\inetsrv\lsspifilt.dll"
"AccessDeniedMessage"="エラー : アクセスは拒否さ
れました。"
"NTAuthenticationProviders"="NTLM"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Script Map]
".idc"="C:\WINNT\System32\inetsrv\httpodbc.dll"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Virtual Roots]
"/,"="C:\inetPub\wwwroot,1"
"/tpc,"="C:\client\tpapl\Release,,5"
"/tpcr,"="C:\client\tpapl\Release_dll,,5"
"/Scripts,"="C:\inetPub\scripts,,4"
"/iisadmin,"="C:\WINNT\System32\inetsrv\iisad
min,,1"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Performance]
"Library"="w3ctrs.DLL"
"Open"="OpenW3PerformanceData"
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"Last Counter"=dword:00000790
"Last Help"=dword:00000791
"First Counter"=dword:00000758
"First Help"=dword:00000759
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Security]
"Security"=hex:01,00,14,80,c0,00,00,00,cc,00,00,
00,14,00,00,00,34,00,00,00,02,\
00,20,00,01,00,00,00,02,80,18,00,ff,01,0f,00,01,
01,00,00,00,00,00,01,00,00,\
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,00,0
0,18,00,8d,01,02,00,01,01,00,\
00,00,00,00,01,00,00,00,00,00,00,00,00,00,0,1
c,00,fd,01,02,00,01,02,00,00,\
00,00,00,05,20,00,00,00,23,02,00,00,00,00,00,0
0,00,00,1c,00,ff,01,0f,00,01,\
02,00,00,00,00,05,20,00,00,00,20,02,00,00,0
0,00,00,00,00,00,1c,00,ff,01,\
0f,00,01,02,00,00,00,00,00,05,20,00,00,00,25,0
2,00,00,00,00,00,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,05,12,00,0
0,00,25,02,00,00,01,01,00,00,\
00,00,00,05,12,00,00,00,01,01,00,00,00,00,00,0
5,12,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\W3SAMP]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Enum]
"0"="Root\LEGACY_W3SVC\00000"
"Count"=dword:00000001
"NextInstance"=dword:00000001
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Fujitsu]
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Fujitsu\T
PC-C ISAPI Application]
"Term_Base"=dword:0000128f
"NumWarehouses"=dword:00002c88
"MaxUsers"=dword:0001bd50
"MaxTerm of Client"=dword:0000128e
"CONTROL_Flag"=dword:00000000
```

```
#
# ubbconfig : TUXEDO configuration file
#
```

```
*RESOURCES
IPCKEY 80952
MASTER SITE1
#UID 1
#GID 1
PERM 0660
MAXACCESSERS 800
MAXSERVERS 200
MAXSERVICES 1000
MODEL SHM
LDBAL Y
```

```
*MACHINES
CL02 LMID=SITE1
APPDIR="c:\sv-apl\ora"
TUXCONFIG="c:\client\tuxconfig"
TUXDIR="c:\tuxedo"
ULOGPF_X="c:\tuxlogtest"
```

```
*GROUPS
group1 LMID=SITE1 GRPNO=1
```

```
*SERVERS
DEFAULT: RESTART=Y MAXGEN=5
ROPERM=0660
tpccfml SRVGRP=group1 RQADDR=newq1
SRVID=1 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq2
SRVID=2 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq3
SRVID=3 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq4
SRVID=4 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq5
SRVID=5 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq6
SRVID=6 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq7
SRVID=7 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq8
SRVID=8 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq9
SRVID=9 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq10
SRVID=10 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq11
SRVID=11 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq12
SRVID=12 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq13
SRVID=13 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq14
SRVID=14 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfmlp SRVGRP=group1 RQADDR=payq1
SRVID=15 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq2
SRVID=16 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq3
SRVID=17 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq4
SRVID=18 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq5
SRVID=19 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq6
SRVID=20 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq7
SRVID=21 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=ordq1
SRVID=22 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
```

```

tpccfml0 SRVGRP=group1 RQADDR=ordq2
SRVID=23 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfml1 SRVGRP=group1 RQADDR=delq1
SRVID=24 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfml2 SRVGRP=group1 RQADDR=delq2
SRVID=25 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfml3 SRVGRP=group1 RQADDR=delq3
SRVID=26 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfml4 SRVGRP=group1 RQADDR=stq1
SRVID=27 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfml5 SRVGRP=group1 RQADDR=stq2
SRVID=28 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfml6 SRVGRP=group1 RQADDR=stq3
SRVID=29 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"

```

*SERVICES

```

"NEWORDER" TRANTIME=0
SRVGRP=group1
"PAYMENT" TRANTIME=0
SRVGRP=group1
"ORDERSTATUS" TRANTIME=0
SRVGRP=group1
"DELIVERY" TRANTIME=0
SRVGRP=group1
"STOCKLVL" TRANTIME=0
SRVGRP=group1

```

*ROUTING

```

===== client 3 =====
REGEDIT4

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo]

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters]
"BandwidthLevel"=dword:ffffff
"ListenBackLog"=dword:00000019
"MemoryCacheSize"=dword:00000000
"ObjectCacheTTL"=dword:ffffff
"PoolThreadLimit"=dword:0000020d

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters\Filter]
"FilterType"=dword:00000000
"NumGrantSites"=dword:00000000
"NumDenySites"=dword:00000000

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters\MimeMap]
"text/html,htm,h"=""
"image/gif,gif,g"=""
"image/jpeg,jpg,,"=""
"text/plain,txt,0"=""
"text/html,html,h"=""
"image/jpeg,jpeg,,"=""
"image/jpeg,jpe,,"=""
"image/bmp,bmp,,"=""
"application/octet-stream,*,,5"=""
"application/pdf,pdf,,5"=""

```

```

"application/octet-stream,bin,,5"=""
"application/oda,oda,,5"=""
"application/zip,zip,,9"=""
"application/rtf,rtf,,5"=""
"application/postscript,ps,,5"=""
"application/postscript,ai,,5"=""
"application/postscript,eps,,5"=""
"application/mac-binhex40,hqx,,4"=""
"application/msword.doc,,5"=""
"application/msword.dot,,5"=""
"application/winhlp,hlp,,5"=""
"video/mpeg,mpeg,,"=""
"video/mpeg,mpg,,"=""
"video/mpeg,mpe,,"=""
"video/x-msvideo,avi,,"=""
"video/quicktime,qt,,"=""
"video/quicktime,mov,,"=""
"video/x-sgi-movie,movie,,"=""
"x-world/x-vrml,wrl,,5"=""
"x-world/x-vrml,xaf,,5"=""
"x-world/x-vrml,xof,,5"=""
"x-world/x-vrml,flr,,5"=""
"x-world/x-vrml,wrz,,5"=""
"application/x-director,dcr,,5"=""
"application/x-director,dir,,5"=""
"application/x-director,dxr,,5"=""
"image/cis-cod,cod,,5"=""
"image/x-cmx,cmx,,5"=""
"application/envoy,envy,,5"=""
"application/x-msaccess,mdb,,5"=""
"application/x-mscardfile,crd,,5"=""
"application/x-msclip,clip,,5"=""
"application/octet-stream,exe,,5"=""
"application/x-msexcel,xla,,5"=""
"application/x-msexcel,xlc,,5"=""
"application/x-msexcel,xlm,,5"=""
"application/x-msexcel,xls,,5"=""
"application/x-msexcel,xlt,,5"=""
"application/x-msexcel,xlw,,5"=""
"application/x-msmediaview,m13,,5"=""
"application/x-msmediaview,m14,,5"=""
"application/x-msmoney,mny,,5"=""
"application/x-mspowerpoint,ppt,,5"=""
"application/x-msproject,mpp,,5"=""
"application/x-mspublisher,pub,,5"=""
"application/x-msterminal,rm,,5"=""
"application/x-msworks,wks,,5"=""
"application/x-mswrite,wri,,5"=""
"application/x-msmetafile,wmf,,5"=""
"application/x-csh,csh,,5"=""
"application/x-dvi,dvi,,5"=""
"application/x-hdf,hdf,,5"=""
"application/x-latex,latex,,5"=""
"application/x-netcdf,nc,,5"=""
"application/x-netcdf,cdf,,5"=""
"application/x-sh,sh,,5"=""
"application/x-tcl,tcl,,5"=""
"application/x-tex,tex,,5"=""
"application/x-texinfo,texinfo,,5"=""
"application/x-texinfo,texi,,5"=""
"application/x-troff,t,5"=""
"application/x-troff,tr,5"=""
"application/x-troff,roff,5"=""
"application/x-troff-man,man,5"=""
"application/x-troff-me,me,5"=""
"application/x-troff-ms,ms,5"=""
"application/x-wais-source,src,,7"=""
"application/x-bcpio,bcpio,,5"=""
"application/x-cpio,cpio,,5"=""

```

```

"application/x-gtar,gtar,,9"=""
"application/x-shar,shar,,5"=""
"application/x-sv4cpio,sv4cpio,,5"=""
"application/x-sv4crc,sv4crc,,5"=""
"application/x-tar,tar,,5"=""
"application/x-ustar,ustar,,5"=""
"audio/basic,au,,"=""
"audio/basic,snd,,"=""
"audio/x-aiff,aiff,,"=""
"audio/x-aiff,aifc,,"=""
"audio/x-wav,wav,,"=""
"audio/x-pn-realaudio,ram,,"=""
"image/ief,ief,,"=""
"image/tiff,tiff,,"=""
"image/tiff,tif,,"=""
"image/x-cmu-raster,ras,,"=""
"image/x-portable-anymap,pnm,,"=""
"image/x-portable-bitmap,pbm,,"=""
"image/x-portable-graymap,pgm,,"=""
"image/x-portable-pixmap,ppm,,"=""
"image/x-rgb,rgb,,"=""
"image/x-xbitmap,xbm,,"=""
"image/x-xpixmap,xpm,,"=""
"image/x-xwindowdump,xwd,,"=""
"text/html,stm,h"=""
"text/plain,bas,0"=""
"text/plain,c,0"=""
"text/plain,h,0"=""
"text/richtext,rtx,0"=""
"text/tab-separated-values,tsv,0"=""
"text/x-setext,etx,0"=""
"application/x-perfmon,pmc,,5"=""
"application/x-perfmon,pma,,5"=""
"application/x-perfmon,pmr,,5"=""
"application/x-perfmon,pml,,5"=""
"application/x-perfmon,pmw,,5"=""

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Performance]
"Library"="infocfrs.DLL"
"Open"="OpenINFOPerformanceData"
"Close"="CloseINFOPerformanceData"
"Collect"="CollectINFOPerformanceData"
"Last Counter"=dword:00000756
"Last Help"=dword:00000757
"First Counter"=dword:00000738
"First Help"=dword:00000739

```

REGEDIT4

```

[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment]
"TUXDIR"="C:\TUXEDO"
"NLSPATH"="C:\TUXEDO\locale\LC"
"ULOGDIR"="C:\TUXEDO"
"ULOGPFX"="C:\ULOG"
"ULOGOUT"=dword:00000001
"TUXIPC_MSG_BYTES"=dword:00010000
"TUXIPC_MSG_HDRS"=dword:00001fc0
"TUXIPC_MSG_QUEUE_BYTES"=dword:00000320
"TUXIPC_MSG_QUEUE_BYTES"=dword:000100
00
"TUXIPC_MSG_SEG_BYTES"=dword:00000040
"TUXIPC_MSG_SEGS"=dword:00007fff
"TUXIPC_PROC"=dword:00001770
"TUXIPC_SEM"=dword:00001770
"TUXIPC_SEM_IDS"=dword:00001770
"TUXIPC_SEM_UNDO"=dword:00001770

```

```
"TUXIPC_SHM_PROCS"=dword:000186a0
"TUXIPC_SHM_SEGS"=dword:0000c350
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services]
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services\305
0]
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC]
"Type"=dword:00000020
"Start"=dword:00000002
"ErrorControl"=dword:00000000
"ImagePath"=hex(2):43,3a,5c,57,49,4e,4e,54,5c,5
3,79,73,74,65,6d,33,32,5c,69,6e,\
65,74,73,72,76,5c,69,6e,65,74,69,6e,66,6f,2e,65,
78,65,00
"DisplayName"="World Wide Web Publishing
Service"
"DependOnService"=hex(7):52,50,43,53,53,00,4e,
54,4c,4d,53,53,50,00,00
"DependOnGroup"=hex(7):00
"ObjectName"="LocalSystem"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters]
"MajorVersion"=dword:00000002
"MinorVersion"=dword:00000000
"AdminName"="Administrator"
"AdminEmail"="Admin@corp.com"
"MaxConnections"=dword:000186a0
"LogType"=dword:00000000
"LogFileDirectory"=hex(2):25,53,79,73,74,65,6d,5
2,6f,6f,74,25,5c,53,79,73,74,\
65,6d,33,32,5c,4c,6f,67,46,69,6c,65,73,00
"LogFileTruncateSize"=dword:01400000
"LogFilePeriod"=dword:00000001
"LogFileFormat"=dword:00000000
"LogSqlDataSource"="HTTPLOG"
"LogSqlTableName"="Internetlog"
"LogSqlUserName"="InternetAdmin"
"LogSqlPassword"="sqllog"
"Authorization"=dword:00000005
"AnonymousUserName"="IUSR_CL03"
"Default Load File"="Default.htm"
"Dir Browse Control"=dword:4000001e
"CheckForWAISDB"=dword:00000000
"CacheExtensions"=dword:00000001
"GlobalExpire"=dword:ffffff
"ServerSideIncludesEnabled"=dword:00000001
"ServerSideIncludesExtension"=".stm"
"DebugFlags"=dword:00000008
"ScriptTimeout"=dword:00000384
"ConnectionTimeOut"=dword:00000384
"InstallPath"="C:\WINNT\System32\inetnsvr"
"SecurePort"=dword:000001bb
"Filter
DLLs"="C:\WINNT\System32\inetnsvr\sspfilt.dll"
"AccessDeniedMessage"="エラー : アクセスは拒否さ
れました。"
"NTAuthenticationProviders"="NTLM"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Script Map]
```

```
".idc"="C:\WINNT\System32\inetnsvr\httpodbc.dll
"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Virtual Roots]
"/,"="C:\InetPub\wwwroot,1"
"/tpc,"="C:\client\tpapl\Release,5"
"/Scripts,"="C:\InetPub\scripts,4"
"/iisadmin,"="C:\WINNT\System32\inetnsvr\iisad
min,1"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Performance]
"Library"="w3ctrs.DLL"
"Open"="OpenW3PerformanceData"
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"Last Counter"=dword:00000790
"Last Help"=dword:00000791
"First Counter"=dword:00000758
"First Help"=dword:00000759
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Security]
"Security"=hex:01,00,14,80,c0,00,00,00,cc,00,00,
00,14,00,00,00,34,00,00,00,02,\
00,20,00,01,00,00,00,02,80,18,00,ff,01,0f,00,01,
01,00,00,00,00,00,01,00,00,\
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,00,0
0,18,00,8d,01,02,00,01,01,00,\
00,00,00,00,01,00,00,00,00,74,00,44,00,00,00,1
c,00,fd,01,02,00,01,02,00,00,\
00,00,00,05,20,00,00,00,23,02,00,00,65,00,74,0
0,00,00,1c,00,ff,01,0f,00,01,\
02,00,00,00,00,00,05,20,00,00,00,20,02,00,00,6
5,00,74,00,00,00,1c,00,ff,01,\
0f,00,01,02,00,00,00,00,00,05,20,00,00,00,25,0
2,00,00,65,00,74,00,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,05,12,00,0
0,00,25,02,00,00,01,01,00,00,\
00,00,00,05,12,00,00,00,01,01,00,00,00,00,00,0
5,12,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\W3SAMP]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Enum]
"0"="Root\LEGACY_W3SVC\00000"
"Count"=dword:00000001
"NextInstance"=dword:00000001
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Fujitsu\T
PC-C ISAPI Application]
"Term_Base"=dword:251d
"NumWarehouses"=dword:2c88
"MaxUsers"=dword:1bd50
"MaxTerm of Client"=dword:128e
"CONTROL_Flag"=dword:00000000
```

```
#
# ubbconfig : TUXEDO configuration file
#
```

```
*RESOURCES
IPCKEY 80952
MASTER SITE1
```

```
#UID 1
#GID 1
PERM 0660
MAXACCESSERS 800
MAXSERVERS 200
MAXSERVICES 1000
MODEL SHM
LDBAL Y
```

```
*MACHINES
CL03 LMID=SITE1
APPDIR="c:\sv-ap\ora"
TUXCONFIG="c:\client\tuxconfig"
TUXDIR="c:\tuxedo"
ULOGFPFX="c:\tuxlog\test"
```

```
*GROUPS
group1 LMID=SITE1 GRPNO=1
```

```
*SERVERS
DEFAULT: RESTART=Y MAXGEN=5
ROPERM=0660
tpccfml SRVGRP=group1 RQADDR=newq1
SRVID=1 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq2
SRVID=2 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq3
SRVID=3 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq4
SRVID=4 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq5
SRVID=5 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq6
SRVID=6 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq7
SRVID=7 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq8
SRVID=8 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq9
SRVID=9 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq10
SRVID=10 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq11
SRVID=11 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq12
SRVID=12 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq13
SRVID=13 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq14
SRVID=14 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=payq1
SRVID=15 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
```

```

tpccfm1p  SRVGRP=group1 RQADDR=payq2
SRVID=16 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfm1p  SRVGRP=group1 RQADDR=payq3
SRVID=17 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfm1p  SRVGRP=group1 RQADDR=payq4
SRVID=18 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfm1p  SRVGRP=group1 RQADDR=payq5
SRVID=19 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfm1p  SRVGRP=group1 RQADDR=payq6
SRVID=20 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfm1p  SRVGRP=group1 RQADDR=payq7
SRVID=21 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfm1o  SRVGRP=group1 RQADDR=ordq1
SRVID=22 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfm1o  SRVGRP=group1 RQADDR=ordq2
SRVID=23 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfm1d  SRVGRP=group1 RQADDR=delq1
SRVID=24 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfm1d  SRVGRP=group1 RQADDR=delq2
SRVID=25 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfm1d  SRVGRP=group1 RQADDR=delq3
SRVID=26 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfm1s  SRVGRP=group1 RQADDR=stoq1
SRVID=27 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfm1s  SRVGRP=group1 RQADDR=stoq2
SRVID=28 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfm1s  SRVGRP=group1 RQADDR=stoq3
SRVID=29 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"

*SERVICES
"NEWORDER"  TRANTIME=0
              SRVGRP=group1
"PAYMENT"   TRANTIME=0
              SRVGRP=group1
"ORDERSTATUS"  TRANTIME=0
              SRVGRP=group1
"DELIVERY"    TRANTIME=0
              SRVGRP=group1
"STOCKLVL"   TRANTIME=0
              SRVGRP=group1

*ROUTING

===== client 4 =====
REGEDIT4

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo]

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters]
"BandwidthLevel"=dword:ffffff
"ListenBackLog"=dword:00000019
"MemoryCacheSize"=dword:00000000

```

```

"ObjectCacheTTL"=dword:ffffff
"ThreadPoolLimit"=dword:0000020d

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters\Filter]
"FilterType"=dword:00000000
"NumGrantSites"=dword:00000000
"NumDenySites"=dword:00000000

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters\MimeMap]
"text/html,html,h"=""
"image/gif,gif,g"=""
"image/jpeg,jpg,."=""
"text/plain,txt,0"=""
"text/html,html,h"=""
"image/jpeg,jpeg,."=""
"image/jpeg,jpe,."=""
"image/bmp,bmp,."=""
"application/octet-stream,*"=""
"application/pdf,pdf,5"=""
"application/octet-stream,bin,5"=""
"application/oda,oda,5"=""
"application/zip,zip,9"=""
"application/rft,rft,5"=""
"application/postscript,ps,5"=""
"application/postscript,ai,5"=""
"application/postscript,eps,5"=""
"application/mac-binhex40,hqx,4"=""
"application/msword,doc,5"=""
"application/msword,dot,5"=""
"application/winhelp,help,5"=""
"video/mpeg,mpeg,."=""
"video/mpeg,mpg,."=""
"video/mpeg,mpe,."=""
"video/x-msvideo,avi,<"=""
"video/quicktime,qt,."=""
"video/quicktime,mov,."=""
"video/x-sgi-movie,movie,<"=""
"x-world/x-vrml,wrl,5"=""
"x-world/x-vrml,xaf,5"=""
"x-world/x-vrml,xof,5"=""
"x-world/x-vrml,flr,5"=""
"x-world/x-vrml,wrz,5"=""
"application/x-director,dcr,5"=""
"application/x-director,dir,5"=""
"application/x-director,dxr,5"=""
"image/cis-cod,cod,5"=""
"image/x-cmx,cmx,5"=""
"application/envoy,evy,5"=""
"application/x-msaccess,mdb,5"=""
"application/x-mscardfile,crd,5"=""
"application/x-msclip,clip,5"=""
"application/octet-stream,exe,5"=""
"application/x-msexcel,xla,5"=""
"application/x-msexcel,xlc,5"=""
"application/x-msexcel,xlm,5"=""
"application/x-msexcel,xls,5"=""
"application/x-msexcel,xlt,5"=""
"application/x-msexcel,xlw,5"=""
"application/x-msmediaview,m13,5"=""
"application/x-msmediaview,m14,5"=""
"application/x-msmoney,mny,5"=""
"application/x-mspowerpoint,ppt,5"=""
"application/x-msproject,mpp,5"=""
"application/x-mspublisher,pub,5"=""
"application/x-msterminal,term,5"=""
"application/x-msworks,wks,5"=""
"application/x-mswrite,wri,5"=""

```

```

"application/x-msmetafile,wmf,5"=""
"application/x-csh,csh,5"=""
"application/x-dvi,dvi,5"=""
"application/x-hdf,hdf,5"=""
"application/x-latex,latex,5"=""
"application/x-netcdf,nc,5"=""
"application/x-netcdf,cdf,5"=""
"application/x-sh,sh,5"=""
"application/x-tcl,tcl,5"=""
"application/x-tex,tex,5"=""
"application/x-texinfo,texinfo,5"=""
"application/x-texinfo,texi,5"=""
"application/x-troff,t,5"=""
"application/x-troff,t,5"=""
"application/x-troff,roff,5"=""
"application/x-troff-man,man,5"=""
"application/x-troff-me,me,5"=""
"application/x-troff-ms,ms,5"=""
"application/x-wais-source,src,5"=""
"application/x-texinfo,bcpio,5"=""
"application/x-cpio,cpio,5"=""
"application/x-gtar,gtar,9"=""
"application/x-shar,shar,5"=""
"application/x-sv4cpio,sv4cpio,5"=""
"application/x-sv4crc,sv4crc,5"=""
"application/x-tar,tar,5"=""
"application/x-ustar,ustar,5"=""
"audio/basic,au,<"=""
"audio/basic,snd,<"=""
"audio/x-aiff,aiff,<"=""
"audio/x-aiff,aiff,<"=""
"audio/x-aiff,aifc,<"=""
"audio/x-wav,wav,<"=""
"audio/x-pn-realaudio,ram,<"=""
"image/ief,ief,."=""
"image/tiff,tiff,."=""
"image/tiff,tif,."=""
"image/x-cmu-raster,ras,."=""
"image/x-portable-anymap,pnm,."=""
"image/x-portable-bitmap,pbm,."=""
"image/x-portable-graymap,pgm,."=""
"image/x-portable-pixmap,ppm,."=""
"image/x-rgb,rgb,."=""
"image/x-xbitmap,xbm,."=""
"image/x-xpixmap,xpm,."=""
"image/x-xwinddumpp,xwd,."=""
"text/html,stm,h"=""
"text/plain,bas,0"=""
"text/plain,c,0"=""
"text/plain,h,0"=""
"text/richtext,rtx,0"=""
"text/tab-separated-values,tsv,0"=""
"text/x-setext,etx,0"=""
"application/x-perfmon,pmc,5"=""
"application/x-perfmon,pma,5"=""
"application/x-perfmon,pmr,5"=""
"application/x-perfmon,pml,5"=""
"application/x-perfmon,pmw,5"=""

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Performance]
"Library"="infoctrs.DLL"
"Open"="OpenINFOPerformanceData"
"Close"="CloseINFOPerformanceData"
"Collect"="CollectINFOPerformanceData"
"Last Counter"=dword:00000756
"Last Help"=dword:00000757
"First Counter"=dword:00000738
"First Help"=dword:00000739

```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment]
"TUXDIR"="C:\TUXEDO"
"NLSPATH"="C:\TUXEDO\locale\LC"
"ULOGDIR"="C:\TUXEDO"
"ULOGPFX"="C:\ULOG"
"ULOGOUT"=dword:00000001
"TUXIPC_MSG_BYTES"=dword:00010000
"TUXIPC_MSG_HDRS"=dword:00001fc0
"TUXIPC_MSG_QUEUES"=dword:00000320
"TUXIPC_MSG_QUEUE_BYTES"=dword:000100
00
"TUXIPC_MSG_SEG_BYTES"=dword:00000040
"TUXIPC_MSG_SEGS"=dword:00007fff
"TUXIPC_PROC"=dword:00001770
"TUXIPC_SEM"=dword:00001770
"TUXIPC_SEM_IDS"=dword:00001770
"TUXIPC_SEM_UNDO"=dword:00001770
"TUXIPC_SHM_PROCS"=dword:000186a0
"TUXIPC_SHM_SEGS"=dword:0000c350
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services]
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services\305
0]
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC]
"Type"=dword:00000020
"Start"=dword:00000002
"ErrorControl"=dword:00000000
"ImagePath"=hex(2):43,3a,5c,57,49,4e,4e,54,5c,5
3,79,73,74,65,6d,33,32,5c,69,6e,\
65,74,73,72,76,5c,69,6e,65,74,69,6e,66,6f,2e,65,
78,65,00
"DisplayName"="World Wide Web Publishing
Service"
"DependOnService"=hex(7):52,50,43,53,53,00,4e,
54,4c,4d,53,53,50,00,00
"DependOnGroup"=hex(7):00
"ObjectName"="LocalSystem"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters]
"MajorVersion"=dword:00000002
"MinorVersion"=dword:00000000
"AdminName"="Administrator"
"AdminEmail"="Admin@corp.com"
"MaxConnections"=dword:000186a0
"LogType"=dword:00000000
"LogFileDirectory"=hex(2):25,53,79,73,74,65,6d,5
2,6f,6f,74,25,5c,53,79,73,74,\
65,6d,33,32,5c,4c,6f,67,46,69,6c,65,73,00
"LogFileTruncateSize"=dword:01400000
"LogFilePeriod"=dword:00000001
"LogFileFormat"=dword:00000000
"LogSqlDataSource"="HTTPLOG"
"LogSqlTableName"="Internetlog"
"LogSqlUserName"="InternetAdmin"
"LogSqlPassword"="sqllog"
"Authorization"=dword:00000005
"AnonymousUserName"="IUSR_CL04"
```

```
"Default Load File"="Default.htm"
"Dir Browse Control"=dword:4000001e
"CheckForWAISDB"=dword:00000000
"CacheExtensions"=dword:00000001
"GlobalExpire"=dword:ffffff
"ServerSideIncludesEnabled"=dword:00000001
"ServerSideIncludesExtension"=".stm"
"DebugFlags"=dword:00000008
"ScriptTimeout"=dword:00000384
"ConnectionTimeOut"=dword:00000384
"InstallPath"="C:\WINNT\System32\inetrv"
"SecurePort"=dword:000001bb
"Filter
DLLs"="C:\WINNT\System32\inetrv\sspifilt.dll"
"AccessDeniedMessage"="エラー : 予約は拒否さ
れました。"
"NTAuthenticationProviders"="NTLM"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Script Map]
".idc"="C:\WINNT\System32\inetrv\httpodbc.dll"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Virtual Roots]
"/,/"="C:\inetpub\wwwroot,1"
"/tpc,/"="C:\client\tpap\Release,5"
"/Scripts,/"="C:\inetpub\scripts,4"
"/iisadmin,/"="C:\WINNT\System32\inetrv\iisad
min,1"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Performance]
"Library"="w3ctrs.DLL"
"Open"="OpenW3PerformanceData"
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"Last Counter"=dword:00000790
"Last Help"=dword:00000791
"First Counter"=dword:00000758
"First Help"=dword:00000759
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Security]
"Security"=hex:01,00,14,80,c0,00,00,00,cc,00,00,
00,14,00,00,00,34,00,00,00,02,\
00,20,00,01,00,00,00,02,80,18,00,ff,01,0f,00,01,
01,00,00,00,00,00,01,00,00,\
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,00,0
0,18,00,8d,01,02,00,01,01,00,\
00,00,00,00,01,00,00,00,00,00,00,00,00,00,01
c,00,fd,01,02,00,01,02,00,00,\
00,00,00,05,20,00,00,00,23,02,00,00,00,00,00,0
0,00,00,1c,00,ff,01,0f,00,01,\
02,00,00,00,00,00,05,20,00,00,00,20,02,00,00,0
0,00,00,00,00,1c,00,ff,01,\
0f,00,01,02,00,00,00,00,00,05,20,00,00,00,25,0
2,00,00,00,00,00,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,05,12,00,0
0,00,25,02,00,00,01,01,00,00,\
00,00,00,05,12,00,00,00,01,01,00,00,00,00,00,0
5,12,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\W3SAMP]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Enum]
"0"="Root\LEGACY_W3SVC\0000"
```

```
"Count"=dword:00000001
"NextInstance"=dword:00000001
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Fujitsu\T
PC-C ISAPI Application]
"Term_Base"=dword:37ab
"NumWarehouses"=dword:2c88
"MaxUsers"=dword:1bd50
"MaxTerm of Client"=dword:128e
"CONTROL_Flag"=dword:00000000
```

```
#
# ubbconfig : TUXEDO configuration file
#
```

```
*RESOURCES
IPCKEY 80952
MASTER SITE1
#UID 1
#GID 1
PERM 0660
MAXACCESSERS 800
MAXSERVERS 200
MAXSERVICES 1000
MODEL SHM
LDBAL Y
```

```
*MACHINES
CL04 LMID=SITE1
APPDIR="c:\sv-ap\lora"
TUXCONFIG="c:\client\tuxconfig"
TUXDIR="c:\tuxedo"
ULOGPFX="c:\tuxlogtest"
```

```
*GROUPS
group1 LMID=SITE1 GRPNO=1
```

```
*SERVERS
DEFAULT: RESTART=Y MAXGEN=5
RQPERM=0660
tpccfml SRVGRP=group1 RQADDR=newq1
SRVID=1 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq2
SRVID=2 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq3
SRVID=3 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq4
SRVID=4 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq5
SRVID=5 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq6
SRVID=6 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq7
SRVID=7 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq8
SRVID=8 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq9
SRVID=9 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
```

```

tpccfml SRVGRP=group1 RQADDR=newq10
SRVID=10 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq11
SRVID=11 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq12
SRVID=12 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq13
SRVID=13 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq14
SRVID=14 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=payq1
SRVID=15 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=payq2
SRVID=16 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=payq3
SRVID=17 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=payq4
SRVID=18 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=payq5
SRVID=19 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=payq6
SRVID=20 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=payq7
SRVID=21 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=ordq1
SRVID=22 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfml SRVGRP=group1 RQADDR=ordq2
SRVID=23 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfml SRVGRP=group1 RQADDR=delq1
SRVID=24 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfml SRVGRP=group1 RQADDR=delq2
SRVID=25 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfml SRVGRP=group1 RQADDR=delq3
SRVID=26 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfml SRVGRP=group1 RQADDR=stoq1
SRVID=27 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfml SRVGRP=group1 RQADDR=stoq2
SRVID=28 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfml SRVGRP=group1 RQADDR=stoq3
SRVID=29 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"

*SERVICES
"NEWORDER" TRANTIME=0
SRVGRP=group1
"PAYMENT" TRANTIME=0
SRVGRP=group1
"ORDERSTATUS" TRANTIME=0
SRVGRP=group1

```

```

"DELIVERY" TRANTIME=0
SRVGRP=group1
"STOCKLVL" TRANTIME=0
SRVGRP=group1

*ROUTING

===== client 5 =====
REGEDIT4

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\inetInfo]

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\inetInfo\Parameters]
"BandwidthLevel"=dword:ffffff
"ListenBackLog"=dword:00000019
"MemoryCacheSize"=dword:00000000
"ObjectCacheTTL"=dword:ffffff
"PoolThreadLimit"=dword:0000020d

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\inetInfo\Parameters\Filter]
"FilterType"=dword:00000000
"NumGrantSites"=dword:00000000
"NumDenySites"=dword:00000000

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\inetInfo\Parameters\MimeMap]
"text/html,html,h"=""
"image/gif,gif,g"=""
"image/jpeg,jpg,,"=""
"text/plain,txt,0"=""
"text/html,html,h"=""
"image/jpeg,jpeg,,"=""
"image/jpeg,jpe,,"=""
"image/bmp,bmp,,"=""
"application/octet-stream,*,,5"=""
"application/pdf,pdf,,5"=""
"application/octet-stream,bin,,5"=""
"application/oda,oda,,5"=""
"application/zip,zip,,9"=""
"application/rft,rft,,5"=""
"application/postscript,ps,,5"=""
"application/postscript,ai,,5"=""
"application/postscript,eps,,5"=""
"application/mac-binhex40,hqx,,4"=""
"application/msword.doc,,5"=""
"application/msword.dot,,5"=""
"application/winhelp,hlp,,5"=""
"video/mpeg,mpeg,,"=""
"video/mpeg,mpg,,"=""
"video/mpeg,mpe,,"=""
"video/x-msvideo,avi,<,"=""
"video/quicktime,qt,,"=""
"video/quicktime,mov,,"=""
"video/x-sgi-movie,movie,<,"=""
"x-world/x-vrml,wrl,,5"=""
"x-world/x-vrml,xaf,,5"=""
"x-world/x-vrml,xof,,5"=""
"x-world/x-vrml,flr,,5"=""
"x-world/x-vrml,wrz,,5"=""
"application/x-director,dcr,,5"=""
"application/x-director,dir,,5"=""
"application/x-director,dxr,,5"=""
"image/cis-cod,cod,,5"=""
"image/x-cmx,cmx,,5"=""
"application/envoy,evy,,5"=""

```

```

"application/x-msaccess,mdb,,5"=""
"application/x-mscardfile,crd,,5"=""
"application/x-msclip,clip,,5"=""
"application/octet-stream,exe,,5"=""
"application/x-msexcel,xla,,5"=""
"application/x-msexcel,xlc,,5"=""
"application/x-msexcel,xlm,,5"=""
"application/x-msexcel,xls,,5"=""
"application/x-msexcel,xlt,,5"=""
"application/x-msexcel,xlw,,5"=""
"application/x-msmediaview,m13,,5"=""
"application/x-msmediaview,m14,,5"=""
"application/x-msmoney,mny,,5"=""
"application/x-mspowerpoint,ppt,,5"=""
"application/x-msproject,mpp,,5"=""
"application/x-mspublisher,pub,,5"=""
"application/x-msterminal,rm,,5"=""
"application/x-msworks,wks,,5"=""
"application/x-mswrite,wri,,5"=""
"application/x-msmetafile,wmf,,5"=""
"application/x-csh,csh,,5"=""
"application/x-dvi,dvi,,5"=""
"application/x-hdf,hdf,,5"=""
"application/x-latex,latex,,5"=""
"application/x-netcdf,nc,,5"=""
"application/x-netcdf,cdf,,5"=""
"application/x-sh,sh,,5"=""
"application/x-tcl,tcl,,5"=""
"application/x-tex,tex,,5"=""
"application/x-texinfo,texinfo,,5"=""
"application/x-texinfo,texi,,5"=""
"application/x-troff,t,,5"=""
"application/x-troff,tr,,5"=""
"application/x-troff,roff,,5"=""
"application/x-troff-man,man,,5"=""
"application/x-troff-me,me,,5"=""
"application/x-troff-ms,ms,,5"=""
"application/x-wais-source,src,,7"=""
"application/x-bcpio,bcpio,,5"=""
"application/x-cpio,cpio,,5"=""
"application/x-gtar,gtar,,9"=""
"application/x-shar,shar,,5"=""
"application/x-sv4cpio,sv4cpio,,5"=""
"application/x-sv4crc,sv4crc,,5"=""
"application/x-tar,tar,,5"=""
"application/x-ustar,ustar,,5"=""
"audio/basic,au,<,"=""
"audio/basic,snd,<,"=""
"audio/x-aiff,aiff,<,"=""
"audio/x-aiff,aiff,<,"=""
"audio/x-aiff,aifc,<,"=""
"audio/x-wav,wav,<,"=""
"audio/x-pn-realaudio,ram,<,"=""
"image/ief,ief,,"=""
"image/tiff,tiff,,"=""
"image/tiff,tif,,"=""
"image/x-cmu-raster,ras,,"=""
"image/x-portable-anymap,pnm,,"=""
"image/x-portable-bitmap,pbm,,"=""
"image/x-portable-graymap,pgm,,"=""
"image/x-portable-pixmap,ppm,,"=""
"image/x-rgb,rgb,,"=""
"image/x-xbitmap,xbm,,"=""
"image/x-xpixmap,xpm,,"=""
"image/x-xwindowdump,xwd,,"=""
"text/html,stm,,h"=""
"text/plain,bas,,0"=""
"text/plain,c,,0"=""
"text/plain,h,,0"=""

```

```
"text/richtext,rtx,,0"=""
"text/tab-separated-values,tsv,,0"=""
"text/x-setext,etx,,0"=""
"application/x-perfmon,pmc,,5"=""
"application/x-perfmon,pma,,5"=""
"application/x-perfmon,pmr,,5"=""
"application/x-perfmon,pml,,5"=""
"application/x-perfmon,pmw,,5"=""
REGEDIT4
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment]
"TUXDIR"="C:\TUXEDO"
"NLSPATH"="C:\TUXEDO\locale\VC"
"ULOGDIR"="C:\TUXEDO"
"ULOGPFX"="C:\ULOG"
"ULOGOUT"=dword:00000001
"TUXIPC_MSG_BYTES"=dword:00010000
"TUXIPC_MSG_HDRS"=dword:00001fc0
"TUXIPC_MSG_QUEUE"=dword:00000320
"TUXIPC_MSG_QUEUE_BYTES"=dword:000100
00
"TUXIPC_MSG_SEG_BYTES"=dword:00000040
"TUXIPC_MSG_SEGS"=dword:00007fff
"TUXIPC_PROC"=dword:00001770
"TUXIPC_SEM"=dword:00001770
"TUXIPC_SEM_IDS"=dword:00001770
"TUXIPC_SEM_UNDO"=dword:00001770
"TUXIPC_SHM_PROCS"=dword:000186a0
"TUXIPC_SHM_SEGS"=dword:0000c350
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services]
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services\305
0]
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC]
"Type"=dword:00000020
"Start"=dword:00000002
"ErrorControl"=dword:00000000
"ImagePath"=hex(2):43,3a,5c,57,49,4e,4e,54,5c,5
3,79,73,74,65,6d,33,32,5c,69,6e,\
65,74,73,72,76,5c,69,6e,65,74,69,6e,66,6f,2e,65,
78,65,00
"DisplayName"="World Wide Web Publishing
Service"
"DependOnService"=hex(7):52,50,43,53,53,00,4e,
54,4c,4d,53,53,50,00,00
"DependOnGroup"=hex(7):00
"ObjectName"="LocalSystem"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters]
"MajorVersion"=dword:00000002
"MinorVersion"=dword:00000000
"AdminName"="Administrator"
"AdminEmail"="Admin@corp.com"
"MaxConnections"=dword:000186a0
"LogType"=dword:00000000
"LogFileDirectory"=hex(2):25,53,79,73,74,65,6d,5
2,6f,6f,74,25,5c,53,79,73,74,\
65,6d,33,32,5c,4c,6f,67,46,69,6c,65,73,00
"LogFileTruncateSize"=dword:01400000
"LogFilePeriod"=dword:00000001
```

```
"LogFileFormat"=dword:00000000
"LogSqlDataSource"="HTTLPLOG"
"LogSqlTableName"="Internetlog"
"LogSqlUserName"="InternetAdmin"
"LogSqlPassword"="sqllog"
"Authorization"=dword:00000005
"AnonymousUserName"="IUSR_CL05"
"Default Load File"="Default.htm"
"Dir Browse Control"=dword:4000001e
"CheckForWAISDB"=dword:00000000
"CacheExtensions"=dword:00000001
"GlobalExpire"=dword:ffffff
"ServerSideIncludesEnabled"=dword:00000001
"ServerSideIncludesExtension"=".stm"
"DebugFlags"=dword:00000008
"ScriptTimeout"=dword:00000384
"ConnectionTimeout"=dword:00000384
"InstallPath"="C:\WINNT\System32\inetsrv"
"SecurePort"=dword:000001bb
"Filter
DLLs"="C:\WINNT\System32\inetsrv\sspi\filter.dll"
"AccessDeniedMessage"="エラー : アクセスは拒否さ
れました。"
"NTAuthenticationProviders"="NTLM"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Script Map]
".idc"="C:\WINNT\System32\inetsrv\lhttpodbc.dll"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Virtual Roots]
"/,"="C:\inetpub\wwwroot,1"
"/tpc,"="C:\client\tpap\Release,,5"
"/Scripts,"="C:\inetpub\scripts,,4"
"/iisadmin,"="C:\WINNT\System32\inetsrv\iisad
min,,1"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Performance]
"Library"="w3ctrs.DLL"
"Open"="OpenW3PerformanceData"
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"Last Counter"=dword:00000790
"Last Help"=dword:00000791
"First Counter"=dword:00000758
"First Help"=dword:00000759
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Security]
"Security"=hex:01,00,14,80,c0,00,00,00,cc,00,00,
00,14,00,00,00,34,00,00,02,\
00,20,00,01,00,00,02,80,18,00,ff,01,0f,00,01,
01,00,00,00,00,01,00,00,\
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,00,
0,18,00,8d,01,02,00,01,01,00,\
00,00,00,00,01,00,00,00,38,25,15,00,00,01,
c,00,fd,01,02,00,01,02,00,00,\
00,00,00,05,20,00,00,00,23,02,00,00,c8,00,14,0
0,00,00,1c,00,ff,01,0f,00,01,\
02,00,00,00,00,00,05,20,00,00,20,02,00,00,c
8,00,14,00,00,00,1c,00,ff,01,\
0f,00,01,02,00,00,00,00,05,20,00,00,00,25,0
2,00,00,c8,00,14,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,05,12,00,0
0,00,25,02,00,00,01,01,00,00,\
00,00,00,05,12,00,00,00,01,01,00,00,00,00,0
5,12,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\W3SAMP]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Enum]
"0"="Root\LEGACY_W3SVC\00000"
"Count"=dword:00000001
"NextInstance"=dword:00000001
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Fujitsu\T
PC-C ISAPI Application]
"Term_Base"=dword:4a39
"NumWarehouses"=dword:2c88
"MaxUsers"=dword:1bd50
"MaxTerm of Client"=dword:128e
"CONTROL_Flag"=dword:00000000
```

```
#
# ubbconfig : TUXEDO configuration file
#
```

```
*RESOURCES
IPCKEY 80952
MASTER SITE1
#UID 1
#GID 1
PERM 0660
MAXACCESSERS 800
MAXSERVERS 200
MAXSERVICES 1000
MODEL SHM
LDBAL Y
```

```
*MACHINES
CL05 LMID=SITE1
APPDIR="c:\sv-apl\ora"
TUXCONFIG="c:\client\tuxconfig"
TUXDIR="c:\tuxedo"
ULOGPFX="c:\tuxlog\test"
```

```
*GROUPS
group1 LMID=SITE1 GRPNO=1
```

```
*SERVERS
DEFAULT: RESTART=Y MAXGEN=5
ROPERM=0660
tpccfml SRVGRP=group1 RQADDR=newq1
SRVID=1 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq2
SRVID=2 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq3
SRVID=3 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq4
SRVID=4 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq5
SRVID=5 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq6
SRVID=6 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
```

```

tpccfml SRVGRP=group1 RQADDR=newq7
SRVID=7 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq8
SRVID=8 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq9
SRVID=9 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq10
SRVID=10 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq11
SRVID=11 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq12
SRVID=12 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq13
SRVID=13 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq14
SRVID=14 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfmlp SRVGRP=group1 RQADDR=payq1
SRVID=15 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq2
SRVID=16 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq3
SRVID=17 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq4
SRVID=18 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq5
SRVID=19 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq6
SRVID=20 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq7
SRVID=21 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml0 SRVGRP=group1 RQADDR=ordq1
SRVID=22 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfml0 SRVGRP=group1 RQADDR=ordq2
SRVID=23 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfml0 SRVGRP=group1 RQADDR=delq1
SRVID=24 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfml0 SRVGRP=group1 RQADDR=delq2
SRVID=25 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfml0 SRVGRP=group1 RQADDR=delq3
SRVID=26 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmls SRVGRP=group1 RQADDR=stog1
SRVID=27 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfmls SRVGRP=group1 RQADDR=stog2
SRVID=28 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfmls SRVGRP=group1 RQADDR=stog3
SRVID=29 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"

```

```

*SERVICES
"NEWORDER" TRANTIME=0
SRVGRP=group1
"PAYMENT" TRANTIME=0
SRVGRP=group1
"ORDERSTATUS" TRANTIME=0
SRVGRP=group1
"DELIVERY" TRANTIME=0
SRVGRP=group1
"STOCKLVL" TRANTIME=0
SRVGRP=group1

```

*ROUTING

```

===== client 6 =====
REGEDIT4

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\inetInfo\Parameters]
"BandwidthLevel"=dword:ffffff
"ListenBackLog"=dword:00000019
"MemoryCacheSize"=dword:00000000
"ObjectCacheTTL"=dword:ffffff
"PoolThreadLimit"=dword:0000020d

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\inetInfo\Parameters\Filter]
"FilterType"=dword:00000000
"NumGrantSites"=dword:00000000
"NumDenySites"=dword:00000000

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\inetInfo\Parameters\MimeMap]
"text/html,html,h"=""
"image/gif,gif,g"=""
"image/jpeg,jpg,,"=""
"text/plain,txt,0"=""
"text/html,html,,h"=""
"image/jpeg,jpeg,,"=""
"image/jpeg,jpe,,"=""
"image/bmp,bmp,,"=""
"application/octet-stream,*"=""
"application/pdf,pdf,5"=""
"application/octet-stream,bin,,5"=""
"application/oda,oda,,5"=""
"application/zip,zip,9"=""
"application/rft,rft,,5"=""
"application/postscript,ps,5"=""
"application/postscript,ai,5"=""
"application/postscript,eps,5"=""
"application/mac-binhex40,hqx,,4"=""
"application/msword,doc,,5"=""
"application/msword,dot,5"=""
"application/winhelp,hlp,,5"=""
"video/mpeg,mpeg,,"=""
"video/mpeg,mpg,,"=""
"video/mpeg,mpe,,"=""
"video/x-msvideo,avi,,<"=""
"video/quicktime,qt,,"=""
"video/quicktime,mov,,"=""
"video/x-sgi-movie,movie,,<"=""
"x-world/x-vrml,wrl,,5"=""
"x-world/x-vrml,xaf,,5"=""
"x-world/x-vrml,xof,,5"=""
"x-world/x-vrml,flr,,5"=""
"x-world/x-vrml,wrz,,5"=""
"application/x-director,dcr,,5"=""

```

```

"application/x-director,dir,,5"=""
"application/x-director,dxr,,5"=""
"image/cis-cod,cod,,5"=""
"image/x-cmx,cmx,,5"=""
"application/envoy,evy,,5"=""
"application/x-msaccess,mdb,5"=""
"application/x-mscardfile,crd,,5"=""
"application/x-msclip,clip,,5"=""
"application/octet-stream,exe,,5"=""
"application/x-msexcel,xla,,5"=""
"application/x-msexcel,xlc,,5"=""
"application/x-msexcel,xlm,,5"=""
"application/x-msexcel,xls,,5"=""
"application/x-msexcel,xlt,,5"=""
"application/x-msexcel,xlw,,5"=""
"application/x-msmediaview,m13,,5"=""
"application/x-msmediaview,m14,,5"=""
"application/x-msmoney,mny,,5"=""
"application/x-mspowerpoint,ppt,,5"=""
"application/x-msproject,mpp,,5"=""
"application/x-mspublisher,pub,,5"=""
"application/x-msterminal,frm,,5"=""
"application/x-msworks,wks,,5"=""
"application/x-mswrite,wri,,5"=""
"application/x-msmetafile,wmf,,5"=""
"application/x-csh,csh,,5"=""
"application/x-dvi,dvi,,5"=""
"application/x-hdf,hdf,,5"=""
"application/x-latex,latex,,5"=""
"application/x-netcdf,nc,,5"=""
"application/x-netcdf,cdf,,5"=""
"application/x-sh,sh,,5"=""
"application/x-tcl,tcl,,5"=""
"application/x-tex,tex,,5"=""
"application/x-texinfo,texinfo,,5"=""
"application/x-texinfo,texi,,5"=""
"application/x-troff,t,,5"=""
"application/x-troff,txt,,5"=""
"application/x-troff,roff,,5"=""
"application/x-troff-man,man,,5"=""
"application/x-troff-me,me,,5"=""
"application/x-troff-ms,ms,,5"=""
"application/x-wais-source,src,,7"=""
"application/x-bcpio,bcpio,,5"=""
"application/x-cpio,cpio,,5"=""
"application/x-gtar,gtar,,9"=""
"application/x-shar,shar,,5"=""
"application/x-sv4cpio,sv4cpio,,5"=""
"application/x-sv4crc,sv4crc,,5"=""
"application/x-tar,tar,,5"=""
"application/x-ustar,ustar,,5"=""
"audio/basic,au,,<"=""
"audio/basic,snd,,<"=""
"audio/x-aiff,aiff,,<"=""
"audio/x-aiff,aifc,,<"=""
"audio/x-wav,wav,,<"=""
"audio/x-pn-realaudio,ram,,<"=""
"image/ief,ief,,="=""
"image/tiff,tiff,,="=""
"image/tiff,tif,,="=""
"image/x-cmu-raster,ras,,="=""
"image/x-portable-anymap,pnm,,="=""
"image/x-portable-bitmap,pbm,,="=""
"image/x-portable-graymap,pgm,,="=""
"image/x-portable-pixmap,ppm,,="=""
"image/x-rgb,rgb,,="=""
"image/x-bitmap,xbm,,="=""
"image/x-pixmap,xpm,,="=""

```



```
"image/x-xwindowdump,xwd,,"=""
"text/html,stm,h"=""
"text/plain,bas,0"=""
"text/plain,c,0"=""
"text/plain,h,0"=""
"text/richtext,rtx,0"=""
"text/tab-separated-values,tsv,0"=""
"text/x-setext,etx,0"=""
"application/x-perfmon,pmc,5"=""
"application/x-perfmon,pma,5"=""
"application/x-perfmon,pmr,5"=""
"application/x-perfmon,pml,5"=""
"application/x-perfmon,pmw,5"=""
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment]
"TUXDIR"="C:\TUXEDO"
"NLSPATH"="C:\TUXEDO\locale\LC"
"ULOGDIR"="C:\TUXEDO"
"ULOGPFX"="C:\ULOG"
"ULOGOUT"=dword:00000001
"TUXIPC_MSG_BYTES"=dword:00010000
"TUXIPC_MSG_HDRS"=dword:00001fc0
"TUXIPC_MSG_QUEUE"=dword:00000320
"TUXIPC_MSG_QUEUE_BYTES"=dword:000100
00
"TUXIPC_MSG_SEG_BYTES"=dword:00000040
"TUXIPC_MSG_SEGS"=dword:00007fff
"TUXIPC_PROC"=dword:00001770
"TUXIPC_SEM"=dword:00001770
"TUXIPC_SEM_IDS"=dword:00001770
"TUXIPC_SEM_UNDO"=dword:00001770
"TUXIPC_SHM_PROCS"=dword:000186a0
"TUXIPC_SHM_SEGS"=dword:0000c350
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services]
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services\305
0]
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC]
"Type"=dword:00000020
"Start"=dword:00000002
"ErrorControl"=dword:00000000
"ImagePath"=hex(2):43,3a,5c,57,49,4e,4e,54,5c,5
3,79,73,74,65,6d,33,32,5c,69,6e,\
65,74,73,72,76,5c,69,6e,65,74,69,6e,66,6f,2e,65,
78,65,00
"DisplayName"="World Wide Web Publishing
Service"
"DependOnService"=hex(7):52,50,43,53,53,00,4e,
54,4c,4d,53,53,50,00,00
"DependOnGroup"=hex(7):00
"ObjectName"="LocalSystem"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters]
"MajorVersion"=dword:00000002
"MinorVersion"=dword:00000000
"AdminName"="Administrator"
"AdminEmail"="Admin@corp.com"
"MaxConnections"=dword:000186a0
```

```
"LogType"=dword:00000000
"LogFileDirectory"=hex(2):25,53,79,73,74,65,6d,5
2,6f,6f,74,25,5c,53,79,73,74,\
65,6d,33,32,5c,4c,6f,67,46,69,6c,65,73,00
"LogFileTruncateSize"=dword:01400000
"LogFilePeriod"=dword:00000001
"LogFileFormat"=dword:00000000
"LogSqlDataSource"="HTTPLOG"
"LogSqlTableName"="Internetlog"
"LogSqlUserName"="InternetAdmin"
"LogSqlPassword"="sqllog"
"Authorization"=dword:00000005
"AnonymousUserName"="IUSR_CL06"
"Default Load File"="Default.htm"
"Dir Browse Control"=dword:4000001e
"CheckForWAISDB"=dword:00000000
"CacheExtensions"=dword:00000001
"GlobalExpire"=dword:ffffff
"ServerSideIncludesEnabled"=dword:00000001
"ServerSideIncludesExtension"=".stm"
"DebugFlags"=dword:00000008
"ScriptTimeout"=dword:00000384
"ConnectionTimeOut"=dword:00000384
"InstallPath"="C:\WINNT\System32\inetrv"
"SecurePort"=dword:000001bb
"Filter
DLLs"="C:\WINNT\System32\inetrv\sspifilt.dll"
"AccessDeniedMessage"="エラー : 予けは拒否さ
れました。 "
"NTAuthenticationProviders"="NTLM"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Script Map]
".idc"="C:\WINNT\System32\inetrv\httpodbc.dll"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Virtual Roots]
"/"="C:\inetPub\wwwroot,1"
"/tpc"="C:\client\tpap\Release,5"
"/Scripts"="C:\inetPub\scripts,4"
"/iisadmin"="C:\WINNT\System32\inetrv\iisad
min,1"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Performance]
"Library"="w3ctrs.DLL"
"Open"="OpenW3PerformanceData"
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"Last Counter"=dword:00000790
"Last Help"=dword:00000791
"First Counter"=dword:00000758
"First Help"=dword:00000759
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Security]
"Security"=hex:01,00,14,80,c0,00,00,00,cc,00,00,
00,14,00,00,00,34,00,00,00,02,\
00,20,00,01,00,00,02,80,18,00,ff,01,0f,00,01,
01,00,00,00,00,01,00,00,\
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,00,
0,18,00,8d,01,02,00,01,01,00,\
00,00,00,00,01,00,00,00,00,00,00,00,00,00,1
c,00,fd,01,02,00,01,02,00,00,\
00,00,00,05,20,00,00,00,23,02,00,00,00,00,00,
0,00,00,1c,00,ff,01,0f,00,01,\
02,00,00,00,00,00,05,20,00,00,00,20,02,00,00,0
0,00,00,00,00,1c,00,ff,01,01,\
```

```
0f,00,01,02,00,00,00,00,05,20,00,00,00,25,0
2,00,00,00,00,00,00,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,00,05,12,00,0
0,00,25,02,00,00,01,01,00,00,\
00,00,00,05,12,00,00,00,01,01,00,00,00,00,00,0
5,12,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\W3SAMP]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Enum]
"0"="Root\LEGACY_W3SVC\00000"
"Count"=dword:00000001
"NextInstance"=dword:00000001
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Fujitsu\T
PC-C ISAPI Application]
"Term_Base"=dword:5cc7
"NumWarehouses"=dword:2c88
"MaxUsers"=dword:1bd50
"MaxTerm of Client"=dword:128e
"CONTROL_Flag"=dword:00000000
```

```
#
# ubbconfig : TUXEDO configuration file
#
```

```
*RESOURCES
IPCKEY 80952
MASTER SITE1
#UID 1
#GID 1
PERM 0660
MAXACCESSERS 800
MAXSERVERS 200
MAXSERVICES 1000
MODEL SHM
LDBAL Y
```

```
*MACHINES
CL06 LMID=SITE1
APPDIR="c:\sv-ap\lora"
TUXCONFIG="c:\client\tuxconfig"
TUXDIR="c:\tuxedo"
ULOGPFX="c:\tuxlog\test"
```

```
*GROUPS
group1 LMID=SITE1 GRPNO=1
```

```
*SERVERS
DEFAULT: RESTART=Y MAXGEN=5
ROPERM=0660
tpccfml SRVGRP=group1 RQADDR=newq1
SRVID=1 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq2
SRVID=2 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq3
SRVID=3 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq4
SRVID=4 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
```

```

tpccfml SRVGRP=group1 RQADDR=newq5
SRVID=5 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq6
SRVID=6 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq7
SRVID=7 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq8
SRVID=8 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq9
SRVID=9 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq10
SRVID=10 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq11
SRVID=11 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq12
SRVID=12 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq13
SRVID=13 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=payq1
SRVID=15 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=payq2
SRVID=16 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=payq3
SRVID=17 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=payq4
SRVID=18 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=payq5
SRVID=19 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=payq6
SRVID=20 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=payq7
SRVID=21 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=ordq1
SRVID=22 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfml SRVGRP=group1 RQADDR=ordq2
SRVID=23 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfml SRVGRP=group1 RQADDR=delq1
SRVID=24 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfml SRVGRP=group1 RQADDR=delq2
SRVID=25 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfml SRVGRP=group1 RQADDR=delq3
SRVID=26 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfml SRVGRP=group1 RQADDR=stoq1
SRVID=27 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"

```

```

tpccfmls SRVGRP=group1 RQADDR=stoq2
SRVID=28 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfmls SRVGRP=group1 RQADDR=stoq3
SRVID=29 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"

*SERVICES
"NEWORDER" TRANTIME=0
SRVGRP=group1
"PAYMENT" TRANTIME=0
SRVGRP=group1
"ORDERSTATUS" TRANTIME=0
SRVGRP=group1
"DELIVERY" TRANTIME=0
SRVGRP=group1
"STOCKLVL" TRANTIME=0
SRVGRP=group1

*ROUTING

===== client 7 =====
REGEDIT4

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters]
"BandwidthLevel"=dword:ffffff
"ListenBackLog"=dword:0000019
"MemoryCacheSize"=dword:00000000
"ObjectCacheTTL"=dword:ffffff
"PoolThreadLimit"=dword:0000020d

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters\Filter]
"FilterType"=dword:00000000
"NumGrantSites"=dword:00000000
"NumDenySites"=dword:00000000

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters\MimeMap]
"text/html,htm,h"=""
"image/gif,gif,g"=""
"image/jpeg,jpg,,"=""
"text/plain,txt,0"=""
"text/html,html,h"=""
"image/jpeg,jpeg,,"=""
"image/jpeg,jpe,,"=""
"image/bmp,bmp,,"=""
"application/octet-stream,*"=""
"application/pdf,pdf,,"=""
"application/octet-stream,bin,,"=""
"application/oda,oda,,"=""
"application/zip,zip,,"=""
"application/rtf,rtf,,"=""
"application/postscript,ps,,"=""
"application/postscript,ai,,"=""
"application/postscript,eps,,"=""
"application/mac-binhex40,hqx,,"=""
"application/msword,doc,,"=""
"application/msword,dot,,"=""
"application/winhelp,hlp,,"=""
"video/mpeg,mpeg,,"=""
"video/mpeg,mpg,,"=""
"video/mpeg,mpe,,"=""
"video/x-msvideo,avi,,"=""
"video/quicktime,qt,,"=""
"video/quicktime,mov,,"=""
"video/x-sgi-movie,movie,,"=""

```

```

"x-world/x-vrml,wrl,,"=""
"x-world/x-vrml,xaf,,"=""
"x-world/x-vrml,xof,,"=""
"x-world/x-vrml,flr,,"=""
"x-world/x-vrml,wrz,,"=""
"application/x-director,dcr,,"=""
"application/x-director,dir,,"=""
"application/x-director,dxr,,"=""
"image/cis-cod,cod,,"=""
"image/x-cmx,cmx,,"=""
"application/envoy,envy,,"=""
"application/x-msaccess,mdb,,"=""
"application/x-mscardfile,crd,,"=""
"application/x-msclip,clip,,"=""
"application/octet-stream,exe,,"=""
"application/x-msexcel,xla,,"=""
"application/x-msexcel,xlc,,"=""
"application/x-msexcel,xlm,,"=""
"application/x-msexcel,xls,,"=""
"application/x-msexcel,xlt,,"=""
"application/x-msexcel,xlw,,"=""
"application/x-msmediaview,m13,,"=""
"application/x-msmediaview,m14,,"=""
"application/x-msmoney,mny,,"=""
"application/x-mspowerpoint,ppt,,"=""
"application/x-msproject,mpp,,"=""
"application/x-mspublisher,pub,,"=""
"application/x-msterminal,trm,,"=""
"application/x-msworks,wks,,"=""
"application/x-mswrite,wri,,"=""
"application/x-msmetafile,wmf,,"=""
"application/x-csh,csh,,"=""
"application/x-dvi,dvi,,"=""
"application/x-hdf,hdf,,"=""
"application/x-latex,latex,,"=""
"application/x-netcdf,nc,,"=""
"application/x-netcdf,cdf,,"=""
"application/x-sh,sh,,"=""
"application/x-tcl,tcl,,"=""
"application/x-tex,tex,,"=""
"application/x-texinfo,texinfo,,"=""
"application/x-texinfo,texi,,"=""
"application/x-troff,t,,"=""
"application/x-troff,tr,,"=""
"application/x-troff,roff,,"=""
"application/x-troff-man,man,,"=""
"application/x-troff-me,me,,"=""
"application/x-troff-ms,ms,,"=""
"application/x-wais-source,src,,"=""
"application/x-bcpio,bcpio,,"=""
"application/x-cpio,cpio,,"=""
"application/x-gtar,gtar,,"=""
"application/x-shar,shar,,"=""
"application/x-sv4cpio,sv4cpio,,"=""
"application/x-sv4crc,sv4crc,,"=""
"application/x-tar,tar,,"=""
"application/x-ustar,ustar,,"=""
"audio/basic,au,,"=""
"audio/basic,snd,,"=""
"audio/x-aiff,aiff,,"=""
"audio/x-aiff,aiff,,"=""
"audio/x-aiff,aifc,,"=""
"audio/x-wav,wav,,"=""
"audio/x-pn-realaudio,ram,,"=""
"image/ief,ief,,"=""
"image/tiff,tiff,,"=""
"image/tiff,tif,,"=""
"image/x-cmu-raster,ras,,"=""
"image/x-portable-anymap,pnm,,"=""

```

```
"image/x-portable-bitmap,pbm,;"=""
"image/x-portable-graymap,pgm,;"=""
"image/x-portable-pixmap,ppm,;"=""
"image/x-rgb,rgb,;"=""
"image/x-xbitmap,xbm,;"=""
"image/x-xpixmap,xpm,;"=""
"image/x-xwindowdump,xwd,;"=""
"text/html,stm,h;"=""
"text/plain,bas,0;"=""
"text/plain,c,0;"=""
"text/plain,h,0;"=""
"text/richtext,rtx,0;"=""
"text/tab-separated-values,tsv,0;"=""
"text/x-setext,etx,0;"=""
"application/x-perfmon,pmc,5;"=""
"application/x-perfmon,pma,5;"=""
"application/x-perfmon,pmr,5;"=""
"application/x-perfmon,pml,5;"=""
"application/x-perfmon,pmw,5;"=""
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment]
"TUXDIR"="C:\TUXEDO"
"NLSPATH"="C:\TUXEDO\locale\VC"
"ULOGDIR"="C:\TUXEDO"
"ULOGPF"="C:\ULOG"
"ULOGOUT"="dword:00000001"
"TUXIPC_MSG_BYTES"="dword:00010000"
"TUXIPC_MSG_HDRS"="dword:00001fc0"
"TUXIPC_MSG_QUEUE"="dword:00000320"
"TUXIPC_MSG_QUEUE_BYTES"="dword:00010000"
"TUXIPC_MSG_SEG_BYTES"="dword:00000040"
"TUXIPC_MSG_SEGS"="dword:00007fff"
"TUXIPC_PROC"="dword:00001770"
"TUXIPC_SEM"="dword:00001770"
"TUXIPC_SEM_IDS"="dword:00001770"
"TUXIPC_SEM_UNDO"="dword:00001770"
"TUXIPC_SHM_PROCS"="dword:000186a0"
"TUXIPC_SHM_SEGS"="dword:0000c350"
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services]
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services\305
0]
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC]
"Type"="dword:00000020"
"Start"="dword:00000002"
"ErrorControl"="dword:00000000"
"ImagePath"="hex(2):43,3a,5c,57,49,4e,4e,54,5c,5
3,79,73,74,65,6d,33,32,5c,69,6e,\
65,74,73,72,76,5c,69,6e,65,74,69,6e,66,6f,2e,65,
78,65,00"
"DisplayName"="World Wide Web Publishing
Service"
"DependOnService"="hex(7):52,50,43,53,53,00,4e,
54,4c,4d,53,53,50,00,00"
"DependOnGroup"="hex(7):00"
"ObjectName"="LocalSystem"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters]
"MajorVersion"="dword:00000002"
"MinorVersion"="dword:00000000"
"AdminName"="Administrator"
"AdminEmail"="Admin@corp.com"
"MaxConnections"="dword:000186a0"
"LogType"="dword:00000000"
"LogFileDirectory"="hex(2):25,53,79,73,74,65,6d,5
2,6f,6f,74,25,5c,53,79,73,74,\
65,6d,33,32,5c,4c,6f,67,46,69,6c,65,73,00"
"LogFileTruncateSize"="dword:01400000"
"LogFilePeriod"="dword:00000001"
"LogFileFormat"="dword:00000000"
"LogSqlDataSource"="HTTPLOG"
"LogSqlTableName"="Internetlog"
"LogSqlUserName"="InternetAdmin"
"LogSqlPassword"="sqllog"
"Authorization"="dword:00000005"
"AnonymousUserName"="IUSR_CL07"
"Default Load File"="Default.htm"
"Dir Browse Control"="dword:4000001e"
"CheckForWAISDB"="dword:00000000"
"CacheExtensions"="dword:00000001"
"GlobalExpire"="dword:ffffff"
"ServerSideIncludesEnabled"="dword:00000001"
"ServerSideIncludesExtension"=".stm"
"DebugFlags"="dword:00000008"
"ScriptTimeout"="dword:00000384"
"ConnectionTimeout"="dword:00000384"
"InstallPath"="C:\WINNT\System32\inetshr"
"SecurePort"="dword:000001bb"
"Filter
DLLs"="C:\WINNT\System32\inetshr\sspifilt.dll"
"AccessDeniedMessage"="エラー : アクセスは拒否さ
れました。"
"NTAuthenticationProviders"="NTLM"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Script Map]
".idc"="C:\WINNT\System32\inetshr\httpodbc.dll"
"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Virtual Roots]
"/,;"="C:\inetPub\wwwroot,1"
"/tpc,;"="C:\client\tpap\Release,5"
"/Scripts,;"="C:\inetPub\scripts,4"
"/iisadmin,;"="C:\WINNT\System32\inetshr\iisad
min,1"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Performance]
"Library"="w3ctrs.DLL"
"Open"="OpenW3PerformanceData"
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"Last Counter"="dword:00000790"
"Last Help"="dword:00000791"
"First Counter"="dword:00000758"
"First Help"="dword:00000759"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Security]
"Security"="hex:01,00,14,80,c0,00,00,00,cc,00,00,
00,14,00,00,00,34,00,00,00,02,\
00,20,00,01,00,00,00,02,80,18,00,ff,01,0f,00,01,
01,00,00,00,00,00,01,00,00,\
```

```
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,0,0
0,18,00,8d,01,02,00,01,01,00,\
00,00,00,00,01,00,00,00,00,a8,e2,14,00,00,0,1
c,00,fd,01,02,00,01,02,00,00,\
00,00,00,05,20,00,00,00,23,02,00,00,00,00,0,0
0,00,00,1c,00,ff,01,0f,00,01,\
02,00,00,00,00,00,05,20,00,00,00,00,00,00,0,0
0,00,00,00,00,00,1c,00,ff,01,\
0f,00,01,02,00,00,00,00,05,20,00,00,00,0,25,0
2,00,00,00,00,00,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,05,12,00,0
0,00,25,02,00,00,01,01,00,00,\
00,00,00,05,12,00,00,00,01,01,00,00,00,00,0,0
5,12,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\W3SAMP]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Enum]
"0"="Root\LEGACY_W3SVC\00000"
"Count"="dword:00000001"
"NextInstance"="dword:00000001"
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Fujitsu\T
PC-C ISAPI Application]
"Term_Base"="dword:6f55"
"NumWarehouses"="dword:2c88"
"MaxUsers"="dword:1bd50"
"MaxTerm of Client"="dword:128e"
"CONTROL_Flag"="dword:00000000"
```

```
#
# ubbconfig : TUXEDO configuration file
#
```

```
*RESOURCES
IPCKEY 80952
MASTER SITE1
#UID 1
#GID 1
PERM 0660
MAXACCESSERS 800
MAXSERVERS 200
MAXSERVICES 1000
MODEL SHM
LDBAL Y
```

```
*MACHINES
CL07 LMID=SITE1
APPDIR="c:\sv-aplora"
TUXCONFIG="c:\client\tuxconfig"
TUXDIR="c:\tuxedo"
ULOGPF="c:\tuxlog\test"
```

```
*GROUPS
group1 LMID=SITE1 GRPNO=1
```

```
*SERVERS
DEFAULT: RESTART=Y MAXGEN=5
ROPERM=0660
tpccfml SRVGRP=group1 RQADDR=newq1
SRVID=1 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq2
SRVID=2 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
```

```

tpccfml SRVGRP=group1 RQADDR=newq3
SRVID=3 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq4
SRVID=4 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq5
SRVID=5 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq6
SRVID=6 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq7
SRVID=7 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq8
SRVID=8 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq9
SRVID=9 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq10
SRVID=10 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq11
SRVID=11 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq12
SRVID=12 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq13
SRVID=13 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq14
SRVID=14 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfmlp SRVGRP=group1 RQADDR=payq1
SRVID=15 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq2
SRVID=16 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq3
SRVID=17 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq4
SRVID=18 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq5
SRVID=19 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq6
SRVID=20 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq7
SRVID=21 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml0 SRVGRP=group1 RQADDR=ordq1
SRVID=22 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfml0 SRVGRP=group1 RQADDR=ordq2
SRVID=23 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfmld SRVGRP=group1 RQADDR=delq1
SRVID=24 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmld SRVGRP=group1 RQADDR=delq2
SRVID=25 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"

```

```

tpccfmld SRVGRP=group1 RQADDR=delq3
SRVID=26 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmld SRVGRP=group1 RQADDR=stoq1
SRVID=27 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfmld SRVGRP=group1 RQADDR=stoq2
SRVID=28 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfmld SRVGRP=group1 RQADDR=stoq3
SRVID=29 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"

```

```

*SERVICES
"NEWORDER" TRANTIME=0
SRVGRP=group1
"PAYMENT" TRANTIME=0
SRVGRP=group1
"ORDERSTATUS" TRANTIME=0
SRVGRP=group1
"DELIVERY" TRANTIME=0
SRVGRP=group1
"STOCKLVL" TRANTIME=0
SRVGRP=group1

```

*ROUTING

```

===== client 8 =====
REGEDIT4

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\inetInfo\Parameters]
"BandwidthLevel"=dword:ffffff
"ListenBackLog"=dword:00000019
"MemoryCacheSize"=dword:00000000
"ObjectCacheTTL"=dword:ffffff
"PoolThreadLimit"=dword:0000020d

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\inetInfo\Parameters\Filter]
"FilterType"=dword:00000000
"NumGrantSites"=dword:00000000
"NumDenySites"=dword:00000000

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\inetInfo\Parameters\MimeMap]
"text/html,html,h"=""
"image/gif,gif,g"=""
"image/jpeg,jpg,,"=""
"text/plain,txt,0"=""
"text/html,html,h"=""
"image/jpeg,jpeg,,"=""
"image/jpeg,jpe,,"=""
"image/bmp,bmp,,"=""
"application/octet-stream,*,,5"=""
"application/pdf,pdf,,5"=""
"application/octet-stream,bin,,5"=""
"application/oda,oda,,5"=""
"application/zip,zip,,9"=""
"application/rft,rft,,5"=""
"application/postscript,ps,,5"=""
"application/postscript,ai,,5"=""
"application/postscript,eps,,5"=""
"application/mac-binhex40,hqx,,4"=""
"application/msword,doc,,5"=""
"application/msword,dot,,5"=""
"application/winhelp,help,,5"=""
"video/mpeg,mpeg,,"=""

```

```

"video/mpeg,mpg,,"=""
"video/mpeg,mpe,,"=""
"video/x-msvideo,avi,<,"=""
"video/quicktime,qt,,"=""
"video/quicktime,mov,,"=""
"video/x-sgi-movie,movie,<,"=""
"x-world/x-vrml,wrl,,5"=""
"x-world/x-vrml,xaf,,5"=""
"x-world/x-vrml,xof,,5"=""
"x-world/x-vrml,flr,,5"=""
"x-world/x-vrml,wrz,,5"=""
"application/x-director,dcr,,5"=""
"application/x-director,dir,,5"=""
"application/x-director,dxr,,5"=""
"image/cis-cod,cod,,5"=""
"image/x-cmx,cmx,,5"=""
"application/envoy,envy,,5"=""
"application/x-msaccess,mdb,,5"=""
"application/x-mscardfile,crd,,5"=""
"application/x-msclip,clip,,5"=""
"application/octet-stream,exe,,5"=""
"application/x-msexcel,xla,,5"=""
"application/x-msexcel,xlc,,5"=""
"application/x-msexcel,xlm,,5"=""
"application/x-msexcel,xls,,5"=""
"application/x-msexcel,xlt,,5"=""
"application/x-msexcel,xlw,,5"=""
"application/x-msmediaview,m13,,5"=""
"application/x-msmediaview,m14,,5"=""
"application/x-msmoney,mny,,5"=""
"application/x-mspowerpoint,ppt,,5"=""
"application/x-msproject,mpp,,5"=""
"application/x-mspublisher,pub,,5"=""
"application/x-msterminal,frm,,5"=""
"application/x-msworks,wks,,5"=""
"application/x-mswrite,wri,,5"=""
"application/x-msmetafile,wmf,,5"=""
"application/x-csh,csh,,5"=""
"application/x-dvi,dvi,,5"=""
"application/x-hdf,hdf,,5"=""
"application/x-latex,latex,,5"=""
"application/x-netcdf,nc,,5"=""
"application/x-netcdf,cdf,,5"=""
"application/x-sh,sh,,5"=""
"application/x-tcl,tcl,,5"=""
"application/x-tex,tex,,5"=""
"application/x-texinfo,texinfo,,5"=""
"application/x-texinfo,texi,,5"=""
"application/x-troff,t,,5"=""
"application/x-troff,tf,,5"=""
"application/x-troff,roff,,5"=""
"application/x-troff-man,man,,5"=""
"application/x-troff-me,me,,5"=""
"application/x-troff-ms,ms,,5"=""
"application/x-wais-source,src,,7"=""
"application/x-bcpio,bcpio,,5"=""
"application/x-cpio,cpio,,5"=""
"application/x-gtar,gtar,,9"=""
"application/x-shar,shar,,5"=""
"application/x-sv4cpio,sv4cpio,,5"=""
"application/x-sv4crc,sv4crc,,5"=""
"application/x-tar,tar,,5"=""
"application/x-ustar,ustar,,5"=""
"audio/basic,au,<,"=""
"audio/basic,snd,<,"=""
"audio/x-aiff,aif,<,"=""
"audio/x-aiff,aiff,<,"=""
"audio/x-aiff,aifc,<,"=""
"audio/x-wav,wav,<,"=""

```

```
"audio/x-pn-realaudio,ram,,"="""
"image/ief,ief,,"="""
"image/tiff,tiff,,"="""
"image/tiff,tif,,"="""
"image/x-cmu-raster,ras,,"="""
"image/x-portable-anymap,pnm,,"="""
"image/x-portable-bitmap,pbm,,"="""
"image/x-portable-graymap,pgm,,"="""
"image/x-portable-pixmap,ppm,,"="""
"image/x-rgb,rgb,,"="""
"image/x-xbitmap,xbm,,"="""
"image/x-ppm,ppm,,"="""
"image/x-xpixmap,xpm,,"="""
"image/x-xwindowdump,xwd,,"="""
"text/html,stm,h,"="""
"text/plain,bas,0,"="""
"text/plain,c,0,"="""
"text/plain,h,0,"="""
"text/richtext,rtx,0,"="""
"text/tab-separated-values,tsv,0,"="""
"text/x-sectext,etx,0,"="""
"application/x-perfmon,pmc,5,"="""
"application/x-perfmon,pma,5,"="""
"application/x-perfmon,pmr,5,"="""
"application/x-perfmon,pml,5,"="""
"application/x-perfmon,pmw,5,"="""
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment]
"TUXDIR"="C:\TUXEDO"
"NLSPATH"="C:\TUXEDO\locale\C"
"ULOGDIR"="C:\TUXEDO"
"ULOGPFX"="C:\ULOG"
"ULOGOUT"=dword:00000001
"TUXIPC_MSG_BYTES"=dword:00010000
"TUXIPC_MSG_HDRS"=dword:00001fc0
"TUXIPC_MSG_QUEUES"=dword:00000320
"TUXIPC_MSG_QUEUE_BYTES"=dword:000100
00
"TUXIPC_MSG_SEG_BYTES"=dword:00000040
"TUXIPC_MSG_SEGS"=dword:00007fff
"TUXIPC_PROC"=dword:00001770
"TUXIPC_SEM"=dword:00001770
"TUXIPC_SEM_IDS"=dword:00001770
"TUXIPC_SEM_UNDO"=dword:00001770
"TUXIPC_SHM_PROCS"=dword:000186a0
"TUXIPC_SHM_SEGS"=dword:0000c350
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services]
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services\305
0]
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC]
"Type"=dword:00000020
"Start"=dword:00000002
"ErrorControl"=dword:00000000
"ImagePath"=hex(2):43,3a,5c,57,49,4e,54,5c,5
3,79,73,74,65,6d,33,32,5c,69,6e,\
65,74,73,72,76,5c,69,6e,65,74,69,6e,66,6f,2e,65,
78,65,00
"DisplayName"="World Wide Web Publishing
Service"
```

```
"DependOnService"=hex(7):52,50,43,53,53,00,4e,
54,4c,4d,53,53,50,00,00
"DependOnGroup"=hex(7):00
"ObjectName"="LocalSystem"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters]
"MajorVersion"=dword:00000002
"MinorVersion"=dword:00000000
"AdminName"="Administrator"
"AdminEmail"="Admin@corp.com"
"MaxConnections"=dword:000186a0
"LogType"=dword:00000000
"LogFileDirectory"=hex(2):25,53,79,73,74,65,6d,5
2,6f,6f,74,25,5c,53,79,73,74,\
65,6d,33,32,5c,4c,6f,67,46,69,6c,65,73,00
"LogFileTruncateSize"=dword:01400000
"LogFilePeriod"=dword:00000001
"LogFileFormat"=dword:00000000
"LogSqlDataSource"="HTTPLOG"
"LogSqlTableName"="Internetlog"
"LogSqlUserName"="InternetAdmin"
"LogSqlPassword"="sqllog"
"Authorization"=dword:00000005
"AnonymousUserName"="IUSR_CL08"
"Default Load File"="Default.htm"
"Dir Browse Control"=dword:4000001e
"CheckForWAISDB"=dword:00000000
"CacheExtensions"=dword:00000001
"GlobalExpire"=dword:ffffff
"ServerSideIncludesEnabled"=dword:00000001
"ServerSideIncludesExtension"=".stm"
"DebugFlags"=dword:00000008
"ScriptTimeout"=dword:00000384
"ConnectionTimeout"=dword:00000384
"InstallPath"="C:\WINNT\System32\inetsrv"
"SecurePort"=dword:000001bb
"Filter
DLLs"="C:\WINNT\System32\inetsrv\sspicfilt.dll"
"AccessDeniedMessage"="エラー : アクセスは拒否さ
れました。"
"NTAuthenticationProviders"="NTLM"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Script Map]
".idc"="C:\WINNT\System32\inetsrv\lhttpodbc.dll"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Virtual Roots]
"/,"="C:\inetpub\wwwroot,1"
"/tpc,"="C:\client\tpap\Release,5"
"/Scripts,"="C:\inetpub\scripts,4"
"/iisadmin,"="C:\WINNT\System32\inetsrv\iisad
min,1"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Performance]
"Library"="w3ctrs.DLL"
"Open"="OpenW3PerformanceData"
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"Last Counter"=dword:00000790
"Last Help"=dword:00000791
"First Counter"=dword:00000758
"First Help"=dword:00000759
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Security]
```

```
"Security"=hex:01,00,14,80,c0,00,00,00,cc,00,00,
00,14,00,00,00,34,00,00,00,02,\
00,20,00,01,00,00,02,80,18,00,ff,01,0f,00,01,
01,00,00,00,00,01,00,00,\
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,00,0
0,18,00,8d,01,02,00,01,01,00,\
00,00,00,00,01,00,00,00,00,00,00,00,00,00,1
c,00,fd,01,02,00,01,02,00,00,\
00,00,00,05,20,00,00,00,23,02,00,00,00,00,00,0
0,00,00,1c,00,ff,01,0f,00,01,\
02,00,00,00,00,00,05,20,00,00,00,00,20,02,00,00,0
0,00,00,00,00,01,c,00,ff,01,\
0f,00,01,02,00,00,00,00,05,20,00,00,00,25,0
2,00,00,00,00,00,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,05,12,00,0
0,00,25,02,00,00,01,01,00,00,\
00,00,00,05,12,00,00,00,01,01,00,00,00,00,00,0
5,12,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\W3SAMP]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Enum]
"0"="Root\LEGACY_W3SVC\0000"
"Count"=dword:00000001
"NextInstance"=dword:00000001
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Fujitsu\T
PC-C ISAPI Application]
"Term_Base"=dword:81e3
"NumWarehouses"=dword:2c88
"MaxUsers"=dword:1bd50
"MaxTerm of Client"=dword:128e
"CONTROL_Flag"=dword:00000000
```

```
#
# ubbconfig : TUXEDO configuration file
#
```

```
*RESOURCES
IPCKEY 80952
MASTER SITE1
#UID 1
#GID 1
PERM 0660
MAXACCESSERS 800
MAXSERVERS 200
MAXSERVICES 1000
MODEL SHM
LDBAL Y
```

```
*MACHINES
CL08 LMID=SITE1
APPDIR="c:\sv-apl\ora"
TUXCONFIG="c:\client\tuxconfig"
TUXDIR="c:\tuxedo"
ULOGPFX="c:\tuxlog\test"
```

```
*GROUPS
group1 LMID=SITE1 GRPNO=1
```

```
*SERVERS
DEFAULT: RESTART=Y MAXGEN=8
RQPERM=0660
```

```

tpccfml SRVGRP=group1 RQADDR=newq1
SRVID=1 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq2
SRVID=2 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq3
SRVID=3 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq4
SRVID=4 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq5
SRVID=5 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq6
SRVID=6 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq7
SRVID=7 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq8
SRVID=8 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq9
SRVID=9 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq10
SRVID=10 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq11
SRVID=11 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq12
SRVID=12 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq13
SRVID=13 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq14
SRVID=14 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfmlp SRVGRP=group1 RQADDR=payq1
SRVID=15 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq2
SRVID=16 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq3
SRVID=17 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq4
SRVID=18 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq5
SRVID=19 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq6
SRVID=20 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq7
SRVID=21 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlo SRVGRP=group1 RQADDR=ordq1
SRVID=22 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfmlo SRVGRP=group1 RQADDR=ordq2
SRVID=23 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"

```

```

tpccfmld SRVGRP=group1 RQADDR=delq1
SRVID=24 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmld SRVGRP=group1 RQADDR=delq2
SRVID=25 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmld SRVGRP=group1 RQADDR=delq3
SRVID=26 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmld SRVGRP=group1 RQADDR=stoq1
SRVID=27 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfmld SRVGRP=group1 RQADDR=stoq2
SRVID=28 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfmld SRVGRP=group1 RQADDR=stoq3
SRVID=29 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"

```

```

*SERVICES
"NEWORDER" TRANTIME=0
SRVGRP=group1
"PAYMENT" TRANTIME=0
SRVGRP=group1
"ORDERSTATUS" TRANTIME=0
SRVGRP=group1
"DELIVERY" TRANTIME=0
SRVGRP=group1
"STOCKLVL" TRANTIME=0
SRVGRP=group1

```

*ROUTING

```

===== client 9 =====
REGEDIT4

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\inetInfo\Parameters]
"BandwidthLevel"=dword:ffffff
"ListenBackLog"=dword:00000019
"MemoryCacheSize"=dword:00000000
"ObjectCacheTTL"=dword:ffffff
"PoolThreadLimit"=dword:0000020d

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\inetInfo\Parameters\Filter]
"FilterType"=dword:00000000
"NumGrantSites"=dword:00000000
"NumDenySites"=dword:00000000

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\inetInfo\Parameters\MimeMap]
"text/html,html,h"=""
"image/gif,gif,g"=""
"image/jpeg,jpg,,"=""
"text/plain,txt,0"=""
"text/html,html,,h"=""
"image/peg,jpeg,,"=""
"image/jpeg,jpe,,"=""
"image/bmp,bmp,,"=""
"application/octet-stream,*"=""
"application/pdf,pdf,5"=""
"application/octet-stream,bin,,5"=""
"application/oda,oda,,5"=""
"application/zip,zip,,9"=""
"application/rft,rft,,5"=""
"application/postscript,ps,5"=""
"application/postscript,ai,5"=""

```

```

"application/postscript,eps,,5"=""
"application/mac-binhex40,hqx,,4"=""
"application/msword,doc,,5"=""
"application/msword,dot,,5"=""
"application/winhlp,hlp,,5"=""
"video/mpeg,mpeg,,,"=""
"video/mpeg,mpg,,,"=""
"video/mpeg,mpe,,,"=""
"video/x-msvideo,avi,<"=""
"video/quicktime,qt,,,"=""
"video/quicktime,mov,,,"=""
"video/x-sgi-movie,movie,<"=""
"x-world/x-vrml,wrl,,5"=""
"x-world/x-vrml,xaf,,5"=""
"x-world/x-vrml,xof,,5"=""
"x-world/x-vrml,flr,,5"=""
"x-world/x-vrml,wrz,,5"=""
"application/x-director,dcr,,5"=""
"application/x-director,dir,,5"=""
"application/x-director,dxr,,5"=""
"image/cis-cod,cod,,5"=""
"image/x-cmx,cmx,,5"=""
"application/envoy,evy,,5"=""
"application/x-msaccess,mdb,,5"=""
"application/x-mscardfile,crd,,5"=""
"application/x-msclip,clip,,5"=""
"application/octet-stream,exe,,5"=""
"application/x-msexcel,xla,,5"=""
"application/x-msexcel,xlc,,5"=""
"application/x-msexcel,xlm,,5"=""
"application/x-msexcel,xls,,5"=""
"application/x-msexcel,xlt,,5"=""
"application/x-msexcel,xlw,,5"=""
"application/x-msmediaview,m13,,5"=""
"application/x-msmediaview,m14,,5"=""
"application/x-msmoney,mny,,5"=""
"application/x-mspowerpoint,ppt,,5"=""
"application/x-msproject,mpp,,5"=""
"application/x-mspublisher,pub,,5"=""
"application/x-msterminal,frm,,5"=""
"application/x-msworks,wks,,5"=""
"application/x-mswrite,wri,,5"=""
"application/x-msmetafile,wmf,,5"=""
"application/x-csh,csh,,5"=""
"application/x-dvi,dvi,,5"=""
"application/x-hdf,hdf,,5"=""
"application/x-latex,latex,,5"=""
"application/x-netcdf,nc,,5"=""
"application/x-netcdf,cdf,,5"=""
"application/x-sh,sh,,5"=""
"application/x-tcl,tcl,,5"=""
"application/x-tex,tex,,5"=""
"application/x-texinfo,texinfo,,5"=""
"application/x-texinfo,txi,,5"=""
"application/x-troff,t,,5"=""
"application/x-troff,tr,,5"=""
"application/x-troff,roff,,5"=""
"application/x-troff-man,man,,5"=""
"application/x-troff-me,me,,5"=""
"application/x-troff-ms,ms,,5"=""
"application/x-wais-source,src,,7"=""
"application/x-bcpio,bcpio,,5"=""
"application/x-cpio,cpio,,5"=""
"application/x-gtar,gtar,,9"=""
"application/x-shar,shar,,5"=""
"application/x-sv4cpio,sv4cpio,,5"=""
"application/x-sv4crc,sv4crc,,5"=""
"application/x-tar,tar,,5"=""
"application/x-ustar,ustar,,5"=""

```

```
"audio/basic,au,,"<=""
"audio/basic,snd,,"<=""
"audio/x-aiff,aiff,,"<=""
"audio/x-aiff,aiff,,"<=""
"audio/x-aiff,aiff,,"<=""
"audio/x-wav,wav,,"<=""
"audio/x-pn-realaudio,ram,,"<=""
"image/ief,ief,,"<=""
"image/tiff,tiff,,"<=""
"image/tiff,tiff,,"<=""
"image/x-cmu-raster,ras,,"<=""
"image/x-portable-anymap,pnm,,"<=""
"image/x-portable-bitmap,pbm,,"<=""
"image/x-portable-graymap,pgm,,"<=""
"image/x-portable-pixmap,ppm,,"<=""
"image/x-rgb,rgb,,"<=""
"image/x-xbitmap,xbm,,"<=""
"image/x-xpixmap,xpm,,"<=""
"image/x-xwindowdump,xwd,,"<=""
"text/html,stm,,"<=""
"text/plain,bas,,"<=""
"text/plain,c,,"<=""
"text/plain,h,,"<=""
"text/richtext,rtx,,"<=""
"text/tab-separated-values,tsv,,"<=""
"text/x-setext,etx,,"<=""
"application/x-perfmon,pmc,,"<=""
"application/x-perfmon,pma,,"<=""
"application/x-perfmon,pmr,,"<=""
"application/x-perfmon,pml,,"<=""
"application/x-perfmon,pmw,,"<=""
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment]
"TUXDIR"="C:\TUXEDO"
"NLSPATH"="C:\TUXEDO\locale\%C"
"ULOGDIR"="C:\TUXEDO"
"ULOGPFX"="C:\ULOG"
"ULOGOUT"=dword:00000001
"TUXIPC_MSG_BYTES"=dword:00010000
"TUXIPC_MSG_HDRS"=dword:00001fc0
"TUXIPC_MSG_QUEUE"=dword:00000320
"TUXIPC_MSG_QUEUE_BYTES"=dword:000100
00
"TUXIPC_MSG_SEG_BYTES"=dword:00000040
"TUXIPC_MSG_SEGS"=dword:00007fff
"TUXIPC_PROC"=dword:00001770
"TUXIPC_SEM"=dword:00001770
"TUXIPC_SEM_IDS"=dword:00001770
"TUXIPC_SEM_UNDO"=dword:00001770
"TUXIPC_SHM_PROCS"=dword:000186a0
"TUXIPC_SHM_SEGS"=dword:0000c350
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services]
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services\305
0]
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC]
"Type"=dword:00000020
"Start"=dword:00000002
"ErrorControl"=dword:00000000
```

```
"ImagePath"=hex(2):43,3a,5c,57,49,4e,4e,54,5c,5
3,79,73,74,65,6d,33,32,5c,69,6e,\
65,74,73,72,76,5c,69,6e,65,74,69,6e,66,6f,2e,65,
78,65,00
"DisplayName"="World Wide Web Publishing
Service"
"DependOnService"=hex(7):52,50,43,53,53,00,4e,
54,4c,4d,53,53,50,00,00
"DependOnGroup"=hex(7):00
"ObjectName"="LocalSystem"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters]
"MajorVersion"=dword:00000002
"MinorVersion"=dword:00000000
"AdminName"="Administrator"
"AdminEmail"="Admin@corp.com"
"MaxConnections"=dword:000186a0
"LogType"=dword:00000000
"LogFileDirectory"=hex(2):25,53,79,73,74,65,6d,5
2,6f,6f,74,25,5c,53,79,73,74,\
65,6d,33,32,5c,4c,6f,67,46,69,6c,65,73,00
"LogFileTruncateSize"=dword:01400000
"LogFilePeriod"=dword:00000001
"LogFileFormat"=dword:00000000
"LogSqlDataSource"="HTTPLOG"
"LogSqlTableName"="Internetlog"
"LogSqlUserName"="InternetAdmin"
"LogSqlPassword"="sqllog"
"Authorization"=dword:00000005
"AnonymousUserName"="IUSR_CL09"
"Default Load File"="Default.htm"
"Dir Browse Control"=dword:4000001e
"CheckForWAISSDB"=dword:00000000
"CacheExtensions"=dword:00000001
"GlobalExpire"=dword:ffffff
"ServerSideIncludesEnabled"=dword:00000001
"ServerSideIncludesExtension"=".stm"
"DebugFlags"=dword:00000008
"ScriptTimeout"=dword:00000384
"ConnectionTimeOut"=dword:00000384
"InstallPath"="C:\WINNT\System32\inetshr"
"SecurePort"=dword:000001bb
"Filter
DLLs"="C:\WINNT\System32\inetshr\sspi\flt.dll"
"AccessDeniedMessage"="エラー : 予約は拒否さ
れました。"
"NTAuthenticationProviders"="NTLM"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Script Map]
".idc"="C:\WINNT\System32\inetshr\httpodbc.dll"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Virtual Roots]
"/"="C:\InetPub\wwwroot,1"
"/tpc"="C:\client\tpap\Release,5"
"/Scripts"="C:\InetPub\Scripts,4"
"/iisadmin"="C:\WINNT\System32\inetshr\iisad
min,1"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Performance]
"Library"="w3ctrs.DLL"
"Open"="OpenW3PerformanceData"
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"Last Counter"=dword:00000790
```

```
"Last Help"=dword:00000791
"First Counter"=dword:00000758
"First Help"=dword:00000759
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Security]
"Security"=hex:01,00,14,80,c0,00,00,00,cc,00,00,
00,14,00,00,00,34,00,00,00,02,\
00,20,00,01,00,00,00,02,80,18,00,ff,01,0f,00,01,
01,00,00,00,00,01,00,00,\
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,00,0
0,18,00,8d,01,02,00,01,01,00,\
00,00,00,00,01,00,00,00,00,00,00,00,00,00,00,1
c,00,fd,01,02,00,01,02,00,00,\
00,00,00,05,20,00,00,00,23,02,00,00,00,00,00,0
0,00,00,1c,00,ff,01,0f,00,01,\
02,00,00,00,00,05,20,00,00,00,20,02,00,00,0
0,00,00,00,00,1c,00,ff,01,\
0f,00,01,02,00,00,00,00,05,20,00,00,00,25,0
2,00,00,00,00,00,00,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,05,12,00,0
0,00,25,02,00,00,01,01,00,00,\
00,00,00,05,12,00,00,01,01,00,00,00,00,0
5,12,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\W3SAMP]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Enum]
"0"="Root\LEGACY_W3SVC\00000"
"Count"=dword:00000001
"NextInstance"=dword:00000001
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Fujitsu\T
PC-C ISAPI Application]
"Term_Base"=dword:9471
"NumWarehouses"=dword:2c88
"MaxUsers"=dword:1bd50
"MaxTerm of Client"=dword:128e
"CONTROL_Flag"=dword:00000000
```

```
#
# ubbconfig : TUXEDO configuration file
#
```

```
*RESOURCES
IPCKEY 80952
MASTER SITE1
#UID 1
#GID 1
PERM 0660
MAXACCESSERS 800
MAXSERVERS 200
MAXSERVICES 1000
MODEL SHM
LDBAL Y
```

```
*MACHINES
CL09 LMID=SITE1
APPDIR="c:\sv-ap\lora"
TUXCONFIG="c:\client\tuxconfig"
"TXDIR"="c:\tuxedo"
ULOGPFX="c:\tuxlog\test"
```

```
*GROUPS
group1 LMID=SITE1 GRPNO=1
```

```
*SERVERS
DEFAULT: RESTART=Y MAXGEN=5
ROPERM=0660
tpccfml SRVGRP=group1 RQADDR=newq1
SRVID=1 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq2
SRVID=2 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq3
SRVID=3 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq4
SRVID=4 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq5
SRVID=5 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq6
SRVID=6 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq7
SRVID=7 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq8
SRVID=8 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq9
SRVID=9 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq10
SRVID=10 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq11
SRVID=11 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq12
SRVID=12 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq13
SRVID=13 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq14
SRVID=14 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfmlp SRVGRP=group1 RQADDR=payq1
SRVID=15 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq2
SRVID=16 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq3
SRVID=17 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq4
SRVID=18 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq5
SRVID=19 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq6
SRVID=20 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq7
SRVID=21 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
```

```
tpccfmlo SRVGRP=group1 RQADDR=ordq1
SRVID=22 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfmlo SRVGRP=group1 RQADDR=ordq2
SRVID=23 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfmld SRVGRP=group1 RQADDR=delq1
SRVID=24 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmld SRVGRP=group1 RQADDR=delq2
SRVID=25 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmld SRVGRP=group1 RQADDR=delq3
SRVID=26 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmld SRVGRP=group1 RQADDR=stoq1
SRVID=27 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfmld SRVGRP=group1 RQADDR=stoq2
SRVID=28 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfmld SRVGRP=group1 RQADDR=stoq3
SRVID=29 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
```

```
*SERVICES
"NEWORDER" TRANTIME=0
SRVGRP=group1
"PAYMENT" TRANTIME=0
SRVGRP=group1
"ORDERSTATUS" TRANTIME=0
SRVGRP=group1
"DELIVERY" TRANTIME=0
SRVGRP=group1
"STOCKLVL" TRANTIME=0
SRVGRP=group1
```

*ROUTING

```
===== client 10 =====
REGEDIT4
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters]
"BandwidthLevel"=dword:ffffff
"ListenBackLog"=dword:00000019
"MemoryCacheSize"=dword:00000000
"ObjectCacheTTL"=dword:ffffff
"PoolThreadLimit"=dword:0000020d
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters\Filter]
"FilterType"=dword:00000000
"NumGrantSites"=dword:00000000
"NumDenySites"=dword:00000000
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters\MimeMap]
"text/html,html,h"=""
"image/gif,gif,g"=""
"image/jpeg,jpg,,"=""
"text/plain,txt,0"=""
"text/html,html,h"=""
"image/jpeg,jpeg,,"=""
"image/jpeg,jpe,,"=""
"image/bmp,bmp,,"=""
"application/octet-stream,*"=""
"application/pdf,pdf,,"=""
```

```
"application/octet-stream,bin,,"=""
"application/oda,oda,,"=""
"application/zip,zip,9"=""
"application/rtf,rtf,5"=""
"application/postscript,ps,5"=""
"application/postscript,ai,5"=""
"application/postscript,eps,5"=""
"application/mac-binhex40,hqx,4"=""
"application/msword,doc,5"=""
"application/msword,dot,5"=""
"application/winhlb,hlp,5"=""
"video/mpeg,mpeg,,"=""
"video/mpeg,mpg,,"=""
"video/mpeg,mpe,,"=""
"video/x-msvideo,avi,<"=""
"video/quicktime,qt,,"=""
"video/quicktime,mov,,"=""
"video/x-sgi-movie,movie,<"=""
"x-world/x-vmrl,wrl,5"=""
"x-world/x-vmrl,xaf,5"=""
"x-world/x-vmrl,xof,5"=""
"x-world/x-vmrl,flr,5"=""
"x-world/x-vmrl,wrz,5"=""
"application/x-director,dcr,5"=""
"application/x-director,dir,5"=""
"application/x-director,dxr,5"=""
"image/cis-cod,cod,5"=""
"image/x-cmx,cmx,5"=""
"application/envoy,evy,5"=""
"application/x-msaccess,mdb,5"=""
"application/x-mscardfile,crd,5"=""
"application/x-msclip,clip,5"=""
"application/octet-stream,exe,5"=""
"application/x-msexcel,xla,5"=""
"application/x-msexcel,xlc,5"=""
"application/x-msexcel,xlm,5"=""
"application/x-msexcel,xls,5"=""
"application/x-msexcel,xlt,5"=""
"application/x-msexcel,xlw,5"=""
"application/x-msmediaview,m13,5"=""
"application/x-msmediaview,m14,5"=""
"application/x-msmoney,mny,5"=""
"application/x-mspowerpoint,ppt,5"=""
"application/x-msproject,mpp,5"=""
"application/x-mspublisher,pub,5"=""
"application/x-msterminal,trm,5"=""
"application/x-msworks,wks,5"=""
"application/x-mswrite,wri,5"=""
"application/x-msmetafile,wmf,5"=""
"application/x-csh,csh,5"=""
"application/x-dvi,dvi,5"=""
"application/x-hdf,hdf,5"=""
"application/x-latex,latex,5"=""
"application/x-netcdf,nc,5"=""
"application/x-netcdf,cdf,5"=""
"application/x-sh,sh,5"=""
"application/x-tcl,tcl,5"=""
"application/x-tex,tex,5"=""
"application/x-texinfo,texinfo,5"=""
"application/x-texinfo,texi,5"=""
"application/x-troff,t,5"=""
"application/x-troff,tr,5"=""
"application/x-troff,roff,5"=""
"application/x-troff-man,man,5"=""
"application/x-troff-me,me,5"=""
"application/x-troff-ms,ms,5"=""
"application/x-wais-source,src,7"=""
"application/x-bcpio,bcpio,5"=""
"application/x-cpio,cpio,5"=""
```



```
"application/x-gtar,gtar,,9"=""
"application/x-shar,shar,,5"=""
"application/x-sv4cpio,sv4cpio,,5"=""
"application/x-sv4crc,sv4crc,,5"=""
"application/x-tar,tar,,5"=""
"application/x-ustar,ustar,,5"=""
"audio/basic,au,,<"=""
"audio/basic,snd,,<"=""
"audio/x-aiff,aiff,,<"=""
"audio/x-aiff,aiff,,<"=""
"audio/x-aiff,aifc,,<"=""
"audio/x-wav,wav,,<"=""
"audio/x-pn-realaudio,ram,,<"=""
"image/ief,ief,,:"=""
"image/tiff,tiff,,:"=""
"image/tiff,tif,,:"=""
"image/x-cmu-raster,ras,,:"=""
"image/x-portable-anymap,pnm,,:"=""
"image/x-portable-bitmap,pbm,,:"=""
"image/x-portable-graymap,pgm,,:"=""
"image/x-portable-pixmap,ppm,,:"=""
"image/x-rgb,rgb,,:"=""
"image/x-xbitmap,xbm,,:"=""
"image/x-xpixmap,xpm,,:"=""
"image/x-xwindowdump,xwd,,:"=""
"text/html,stm,,h"=""
"text/plain,bas,,0"=""
"text/plain,c,,0"=""
"text/plain,h,,0"=""
"text/richtext,rtx,,0"=""
"text/tab-separated-values,tsv,,0"=""
"text/x-setext,etx,,0"=""
"application/x-perfmon,pmc,,5"=""
"application/x-perfmon,pma,,5"=""
"application/x-perfmon,pmr,,5"=""
"application/x-perfmon,pml,,5"=""
"application/x-perfmon,pmw,,5"=""
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment]
"TUXDIR"="C:\TUXEDO"
"NLSPATH"="C:\TUXEDO\locale\LC"
"ULOGDIR"="C:\TUXEDO"
"ULOGPF"="C:\ULOG"
"ULOGOUT"="dword:00000001"
"TUXIPC_MSG_BYTES"="dword:00010000"
"TUXIPC_MSG_HDRS"="dword:00001fc0"
"TUXIPC_MSG_QUEUE"="dword:00000320"
"TUXIPC_MSG_QUEUE_BYTES"="dword:000100
00"
"TUXIPC_MSG_SEG_BYTES"="dword:00000040"
"TUXIPC_MSG_SEGS"="dword:00007fff"
"TUXIPC_PROC"="dword:00001770"
"TUXIPC_SEM"="dword:00001770"
"TUXIPC_SEM_IDS"="dword:00001770"
"TUXIPC_SEM_UNDO"="dword:00001770"
"TUXIPC_SHM_PROCS"="dword:000186a0"
"TUXIPC_SHM_SEGS"="dword:0000c350"
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services]
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services\305
0]
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC]
"Type"=dword:00000020
"Start"=dword:00000002
"ErrorControl"=dword:00000000
"ImagePath"=hex(2):43,3a,5c,57,49,4e,4e,54,5c,5
3,79,73,74,65,6d,33,32,5c,69,6e,\
65,74,73,72,76,5c,69,6e,65,74,69,6e,66,6f,2e,65,
78,65,00
"DisplayName"="World Wide Web Publishing
Service"
"DependOnService"=hex(7):52,50,43,53,53,00,4e,
54,4c,4d,53,53,50,00,00
"DependOnGroup"=hex(7):00
"ObjectName"="LocalSystem"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters]
"MajorVersion"=dword:00000002
"MinorVersion"=dword:00000000
"AdminName"="Administrator"
"AdminEmail"="Admin@corp.com"
"MaxConnections"=dword:000186a0
"LogType"=dword:00000000
"LogFileDirectory"=hex(2):25,53,79,73,74,65,6d,5
2,6f,6f,74,25,5c,53,79,73,74,\
65,6d,33,32,5c,4c,6f,67,46,69,6c,65,73,00
"LogFileTruncateSize"=dword:01400000
"LogFilePeriod"=dword:00000001
"LogFileFormat"=dword:00000000
"LogSqlDataSource"="HTTPLOG"
"LogSqlTableName"="Internetlog"
"LogSqlUserName"="InternetAdmin"
"LogSqlPassword"="sqllog"
"Authorization"=dword:00000005
"AnonymousUserName"="IUSR_CL10"
"Default Load File"="Default.htm"
"Dir Browse Control"=dword:4000001e
"CheckForWAI5DB"=dword:00000000
"CacheExtensions"=dword:00000001
"GlobalExpire"=dword:ffffff
"ServerSideIncludesEnabled"=dword:00000001
"ServerSideIncludesExtension"=".stm"
"DebugFlags"=dword:00000008
"ScriptTimeout"=dword:00000384
"ConnectionTimeOut"=dword:00000384
"InstallPath"="C:\WINNT\System32\inetshr"
"SecurePort"=dword:000001bb
"Filter
DLLs"="C:\WINNT\System32\inetshr\sspifilt.dll"
"AccessDeniedMessage"="エラー : アクセスは拒否さ
れました。"
"NTAuthenticationProviders"="NTLM"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Script Map]
".idc"="C:\WINNT\System32\inetshr\httpodbc.dll"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Virtual Roots]
"/,"="C:\InetPub\wwwroot,,1"
"/tpc,"="C:\client\tpap\Release,,5"
"/Scripts,"="C:\InetPub\scripts,,4"
"/iisadmin,"="C:\WINNT\System32\inetshr\iisad
min,,1"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Performance]
"Library"="w3ctrs.DLL"
"Open"="OpenW3PerformanceData"
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"Last Counter"=dword:00000790
"Last Help"=dword:00000791
"First Counter"=dword:00000758
"First Help"=dword:00000759
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Security]
"Security"=hex:01,00,14,80,c0,00,00,cc,00,00,
00,14,00,00,00,34,00,00,00,02,\
00,20,00,01,00,00,00,02,80,18,00,ff,01,0f,00,01,
01,00,00,00,00,01,00,00,\
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,00,0
0,18,00,8d,01,02,00,01,01,00,\
00,00,00,00,01,00,00,00,00,00,00,00,00,00,1
c,00,fd,01,02,00,01,02,00,00,\
00,00,00,05,20,00,00,00,23,02,00,00,00,00,00,0
0,00,00,1c,00,ff,01,0f,00,01,\
02,00,00,00,00,05,20,00,00,00,20,02,00,00,0
0,00,00,00,00,1c,00,ff,01,\
0f,00,01,02,00,00,00,00,05,20,00,00,00,25,0
2,00,00,00,00,00,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,05,12,00,0
0,00,25,02,00,00,01,01,00,00,\
00,00,00,05,12,00,00,00,01,01,00,00,00,00,0
5,12,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\W3SAMP]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Enum]
"0"="Root\LEGACY_W3SVC\00000"
"Count"=dword:00000001
"NextInstance"=dword:00000001
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Fujitsu\T
PC-C ISAPI Application]
"Term_Base"=dword:a6ff
"NumWarehouses"=dword:2c88
"MaxUsers"=dword:1bd50
"MaxTerm of Client"=dword:128e
"CONTROL_Flag"=dword:00000000
```

```
#
# ubbconfig : TUXEDO configuration file
#
```

```
*RESOURCES
IPCKEY 80952
MASTER SITE1
#UID 1
#GID 1
PERM 0660
MAXACCESSERS 800
MAXSERVERS 200
MAXSERVICES 1000
MODEL SHM
LDBAL Y
```

```
*MACHINES
CL10 LMID=SITE1
```

```

APPDIR="c:\sv-aplora"
TUXCONFIG="c:\client\tuxconfig"
TUXDIR="c:\tuxedo"
ULOGPFX="c:\tuxlog\test"

```

*GROUPS

```
group1 LMID=SITE1 GRPNO=1
```

*SERVERS

```

DEFAULT: RESTART=Y MAXGEN=5
RQPERM=0660
tpccfml SRVGRP=group1 RQADDR=newq1
SRVID=1 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq2
SRVID=2 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq3
SRVID=3 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq4
SRVID=4 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq5
SRVID=5 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq6
SRVID=6 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq7
SRVID=7 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq8
SRVID=8 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq9
SRVID=9 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq10
SRVID=10 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq11
SRVID=11 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq12
SRVID=12 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq13
SRVID=13 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq14
SRVID=14 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfmlp SRVGRP=group1 RQADDR=payq1
SRVID=15 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq2
SRVID=16 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq3
SRVID=17 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq4
SRVID=18 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq5
SRVID=19 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"

```

```

tpccfmlp SRVGRP=group1 RQADDR=payq6
SRVID=20 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq7
SRVID=21 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=ordq1
SRVID=22 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfml SRVGRP=group1 RQADDR=ordq2
SRVID=23 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfml SRVGRP=group1 RQADDR=delq1
SRVID=24 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfml SRVGRP=group1 RQADDR=delq2
SRVID=25 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfml SRVGRP=group1 RQADDR=delq3
SRVID=26 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmls SRVGRP=group1 RQADDR=stoq1
SRVID=27 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfmls SRVGRP=group1 RQADDR=stoq2
SRVID=28 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfmls SRVGRP=group1 RQADDR=stoq3
SRVID=29 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"

```

*SERVICES

```

"NEWORDER" TRANTIME=0
SRVGRP=group1
"PAYMENT" TRANTIME=0
SRVGRP=group1
"ORDERSTATUS" TRANTIME=0
SRVGRP=group1
"DELIVERY" TRANTIME=0
SRVGRP=group1
"STOCKLVL" TRANTIME=0
SRVGRP=group1

```

*ROUTING

```

===== client 11 =====
REGEDIT4

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters]
"BandwidthLevel"=dword:ffffff
"ListenBackLog"=dword:00000019
"MemoryCacheSize"=dword:00000000
"ObjectCacheTTL"=dword:ffffff
"PoolThreadLimit"=dword:0000020d

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters\Filter]
"FilterType"=dword:00000000
"NumGrantSites"=dword:00000000
"NumDenySites"=dword:00000000

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters\MimeMap]
"text/html,htm,,h"=""
"image/gif,gif,g"=""
"image/jpeg,jpg,,"=""
"text/plain,txt,0"=""

```

```

"text/html,html,,h"=""
"image/jpeg,jpeg,,"=""
"image/jpeg,jpe,,"=""
"image/bmp,bmp,,"=""
"application/octet-stream,*,,5"=""
"application/pdf,pdf,,5"=""
"application/octet-stream,bin,,5"=""
"application/oda,oda,,5"=""
"application/zip,zip,,9"=""
"application/rtf,rtf,,5"=""
"application/postscript,ps,,5"=""
"application/postscript,ai,,5"=""
"application/postscript,eps,,5"=""
"application/mac-binhex40,hqx,,4"=""
"application/msword,doc,,5"=""
"application/msword,dot,,5"=""
"application/winhlp,hlp,,5"=""
"video/mpeg,mpeg,,"=""
"video/mpeg,mpg,,"=""
"video/mpeg,mpe,,"=""
"video/x-msvideo,avi,<,,5"=""
"video/quicktime,qt,,"=""
"video/quicktime,mov,,"=""
"video/x-sgi-movie,movie,<,,5"=""
"x-world/x-vrml,wrl,,5"=""
"x-world/x-vrml,xaf,,5"=""
"x-world/x-vrml,xof,,5"=""
"x-world/x-vrml,flr,,5"=""
"x-world/x-vrml,wrz,,5"=""
"application/x-director,dcr,,5"=""
"application/x-director,dir,,5"=""
"application/x-director,dxr,,5"=""
"image/cis-cod,cod,,5"=""
"image/x-cmx,cmx,,5"=""
"application/envoy,evy,,5"=""
"application/x-msaccess,mdb,,5"=""
"application/x-mscardfile,crd,,5"=""
"application/x-msclip,clip,,5"=""
"application/octet-stream,exe,,5"=""
"application/x-msexcel,xla,,5"=""
"application/x-msexcel,xlc,,5"=""
"application/x-msexcel,xlm,,5"=""
"application/x-msexcel,xls,,5"=""
"application/x-msexcel,xlt,,5"=""
"application/x-msexcel,xlw,,5"=""
"application/x-msmediaview,m13,,5"=""
"application/x-msmediaview,m14,,5"=""
"application/x-msmoney,mny,,5"=""
"application/x-mspowerpoint,ppt,,5"=""
"application/x-msproject,mpp,,5"=""
"application/x-mspublisher,pub,,5"=""
"application/x-msterminal,frm,,5"=""
"application/x-msworks,wks,,5"=""
"application/x-mswrite,wri,,5"=""
"application/x-msmetafile,wmf,,5"=""
"application/x-csh,csh,,5"=""
"application/x-dvi,dvi,,5"=""
"application/x-hdf,hdf,,5"=""
"application/x-latex,latex,,5"=""
"application/x-netcdf,nc,,5"=""
"application/x-netcdf,cdf,,5"=""
"application/x-sh,sh,,5"=""
"application/x-tcl,tcl,,5"=""
"application/x-tex,tex,,5"=""
"application/x-texinfo,texinfo,,5"=""
"application/x-texinfo,texi,,5"=""
"application/x-troff,t,,5"=""
"application/x-troff,tf,,5"=""
"application/x-troff,roff,,5"=""

```

```

"application/x-troff-man,man,,5"=""
"application/x-troff-me,me,,5"=""
"application/x-troff-ms,ms,,5"=""
"application/x-wais-source,src,,7"=""
"application/x-bcpio,bcpio,,5"=""
"application/x-cpio,cpio,,5"=""
"application/x-gtar,gtar,,9"=""
"application/x-shar,shar,,5"=""
"application/x-sv4cpio,sv4cpio,,5"=""
"application/x-sv4crc,sv4crc,,5"=""
"application/x-tar,tar,,5"=""
"application/x-ustar,ustar,,5"=""
"audio/basic,au,,<"=""
"audio/basic,snd,,<"=""
"audio/x-aiff,aif,,<"=""
"audio/x-aiff,aiff,,<"=""
"audio/x-aiff,aifc,,<"=""
"audio/x-wav,wav,,<"=""
"audio/x-pn-realaudio,ram,,<"=""
"image/ief,ief,,:"=""
"image/tiff,tiff,,:"=""
"image/tiff,tif,,:"=""
"image/x-cmu-raster,ras,,:"=""
"image/x-portable-anymap,pnm,,:"=""
"image/x-portable-bitmap,pbm,,:"=""
"image/x-portable-graymap,pgm,,:"=""
"image/x-portable-pixmap,ppm,,:"=""
"image/x-rgb,rgb,,:"=""
"image/x-xbitmap,xbm,,:"=""
"image/x-ppixmap,xpm,,:"=""
"image/x-xwindowdump,xwd,,:"=""
"text/html,stm,,h"=""
"text/plain,bas,,0"=""
"text/plain,c,,0"=""
"text/plain,h,,0"=""
"text/richtext,rtx,,0"=""
"text/tab-separated-values,tsv,,0"=""
"text/x-setext,etx,,0"=""
"application/x-perfmon,pmc,,5"=""
"application/x-perfmon,pma,,5"=""
"application/x-perfmon,pmr,,5"=""
"application/x-perfmon,pml,,5"=""
"application/x-perfmon,pmw,,5"=""

```

REGEDIT4

```

[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment]
"TUXDIR"="C:\TUXEDO"
"NLSPATH"="C:\TUXEDO\locale\VC"
"ULOGDIR"="C:\TUXEDO"
"ULOGPF"="C:\ULOG"
"ULOGOUT"=dword:00000001
"TUXIPC_MSG_BYTES"=dword:00010000
"TUXIPC_MSG_HDRS"=dword:00001fc0
"TUXIPC_MSG_QUEUEUS"=dword:00000320
"TUXIPC_MSG_QUEUE_BYTES"=dword:000100
00
"TUXIPC_MSG_SEG_BYTES"=dword:00000040
"TUXIPC_MSG_SEGS"=dword:00007fff
"TUXIPC_PROC"=dword:00001770
"TUXIPC_SEM"=dword:00001770
"TUXIPC_SEM_IDS"=dword:00001770
"TUXIPC_SEM_UNDO"=dword:00001770
"TUXIPC_SHM_PROCS"=dword:000186a0
"TUXIPC_SHM_SEGS"=dword:0000c350

```

```

[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services]

```

```

[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services\305
0]

```

REGEDIT4

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC]
"Type"=dword:00000020
"Start"=dword:00000002
"ErrorControl"=dword:00000000
"ImagePath"=hex(2):43,3a,5c,57,49,4e,4e,54,5c,5
3,79,73,74,65,6d,33,32,5c,69,6e,\
65,74,73,72,76,5c,69,6e,65,74,69,6e,66,6f,2e,65,
78,65,00
"DisplayName"="World Wide Web Publishing
Service"
"DependOnService"=hex(7):52,50,43,53,53,00,4e,
54,4c,4d,53,53,00,00,00
"DependOnGroup"=hex(7):00
"ObjectName"="LocalSystem"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters]
"MajorVersion"=dword:00000002
"MinorVersion"=dword:00000000
"AdminName"="Administrator"
"AdminEmail"="Admin@corp.com"
"MaxConnections"=dword:000186a0
"LogType"=dword:00000000
"LogFileDirectory"=hex(2):25,53,79,73,74,65,6d,5
2,6f,6f,74,25,5c,53,79,73,74,\
65,6d,33,32,5c,4c,6f,67,46,69,6c,65,73,00
"LogFileTruncateSize"=dword:01400000
"LogFilePeriod"=dword:00000001
"LogFileFormat"=dword:00000000
"LogSqlDataSource"="HTTPLOG"
"LogSqlTableName"="Internetlog"
"LogSqlUserName"="InternetAdmin"
"LogSqlPassword"="sqllog"
"Authorization"=dword:00000005
"AnonymousUserName"="IUSR_CL11"
"Default Load File"="Default.htm"
"Dir Browse Control"=dword:4000001e
"CheckForWAISDB"=dword:00000000
"CacheExtensions"=dword:00000001
"GlobalExpire"=dword:ffffff
"ServerSideIncludesEnabled"=dword:00000001
"ServerSideIncludesExtension"=".stm"
"DebugFlags"=dword:00000008
"ScriptTimeout"=dword:00000384
"ConnectionTimeout"=dword:00000384
"InstallPath"="C:\WINNT\System32\inetstrv"
"SecurePort"=dword:000001bb
"Filter
DLLs"="C:\WINNT\System32\inetstrv\sspifilt.dll"
"AccessDeniedMessage"="エラー : アクセスは拒否さ
れました。"
"NTAuthenticationProviders"="NTLM"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Script Map]
".idc"="C:\WINNT\System32\inetstrv\httpodbc.dll"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Virtual Roots]
"/,"="C:\inetpub\wwwroot,1"

```

```

"/tpc,"="C:\client\tpap\Release,,5"
"/Scripts,"="C:\inetpub\Scripts,,4"
"/iisadmin,"="C:\WINNT\System32\inetstrv\iisad
min,,1"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Performance]
"Library"="w3ctrs.DLL"
"Open"="OpenW3PerformanceData"
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"Last Counter"=dword:00000790
"Last Help"=dword:00000791
"First Counter"=dword:00000758
"First Help"=dword:00000759

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Security]
"Security"=hex:01,00,14,80,c0,00,00,00,cc,00,00,
00,14,00,00,00,34,00,00,02,\
00,20,00,01,00,00,02,80,18,00,ff,01,0f,00,01,
01,00,00,00,00,01,00,00,\
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,00,
0,18,00,8d,01,02,00,01,01,00,\
00,00,00,00,01,00,00,00,6d,00,63,00,00,1
c,00,fd,01,02,00,01,02,00,00,\
00,00,00,05,20,00,00,00,23,02,00,00,69,00,61,0
0,00,00,1c,00,ff,01,0f,00,01,\
02,00,00,00,00,05,20,00,00,00,20,02,00,00,6
9,00,61,00,00,00,1c,00,ff,01,\
0f,00,01,02,00,00,00,00,05,20,00,00,00,25,0
2,00,00,69,00,61,00,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,05,12,00,0
0,25,02,00,00,01,01,00,00,\
00,00,00,05,12,00,00,00,01,01,00,00,00,00,0
5,12,00,00,00

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\W3SAMP]

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Enum]
"0"="Root\LEGACY_W3SVC\0000"
"Count"=dword:00000001
"NextInstance"=dword:00000001

```

REGEDIT4

```

[HKEY_LOCAL_MACHINE\SOFTWARE\Fujitsu\T
PC-C ISAPI Application]
"Term_Base"=dword:b98d
"NumWarehouses"=dword:2c88
"MaxUsers"=dword:1bd50
"MaxTerm of Client"=dword:128e
"CONTROL_Flag"=dword:00000000

```

```

#
# ubbconfig : TUXEDO configuration file
#

```

```

*RESOURCES
IPCKEY 80952
MASTER SITE1
#UID 1
#GID 1
PERM 0660
MAXACCESSERS 800
MAXSERVERS 200
MAXSERVICES 1000

```

```

MODEL          SHM
LDBAL          Y

*MACHINES
CL11 LMID=SITE1
  APPDIR="c:\sv-aplora"
  TUXCONFIG="c:\client\tuxconfig"
  TUXDIR="c:\tuxedo"
  ULOGPFX="c:\tuxlog\test"

*GROUPS
group1  LMID=SITE1 GRPNO=1

*SERVERS
DEFAULT: RESTART=Y MAXGEN=5
RPERM=0660
tpccfml  SRVGRP=group1 RQADDR=newq1
SRVID=1 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml  SRVGRP=group1 RQADDR=newq2
SRVID=2 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml  SRVGRP=group1 RQADDR=newq3
SRVID=3 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml  SRVGRP=group1 RQADDR=newq4
SRVID=4 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml  SRVGRP=group1 RQADDR=newq5
SRVID=5 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml  SRVGRP=group1 RQADDR=newq6
SRVID=6 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml  SRVGRP=group1 RQADDR=newq7
SRVID=7 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml  SRVGRP=group1 RQADDR=newq8
SRVID=8 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml  SRVGRP=group1 RQADDR=newq9
SRVID=9 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml  SRVGRP=group1 RQADDR=newq10
SRVID=10 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml  SRVGRP=group1 RQADDR=newq11
SRVID=11 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml  SRVGRP=group1 RQADDR=newq12
SRVID=12 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml  SRVGRP=group1 RQADDR=newq13
SRVID=13 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml  SRVGRP=group1 RQADDR=newq14
SRVID=14 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfmlp  SRVGRP=group1 RQADDR=payq1
SRVID=15 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp  SRVGRP=group1 RQADDR=payq2
SRVID=16 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp  SRVGRP=group1 RQADDR=payq3
SRVID=17 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"

```

```

tpccfmlp  SRVGRP=group1 RQADDR=payq4
SRVID=18 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp  SRVGRP=group1 RQADDR=payq5
SRVID=19 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp  SRVGRP=group1 RQADDR=payq6
SRVID=20 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp  SRVGRP=group1 RQADDR=payq7
SRVID=21 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlo  SRVGRP=group1 RQADDR=ordq1
SRVID=22 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfmlo  SRVGRP=group1 RQADDR=ordq2
SRVID=23 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfmld  SRVGRP=group1 RQADDR=delq1
SRVID=24 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmld  SRVGRP=group1 RQADDR=delq2
SRVID=25 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmld  SRVGRP=group1 RQADDR=delq3
SRVID=26 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmls  SRVGRP=group1 RQADDR=stoq1
SRVID=27 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfmls  SRVGRP=group1 RQADDR=stoq2
SRVID=28 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfmls  SRVGRP=group1 RQADDR=stoq3
SRVID=29 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"

*SERVICES
"NEWORDER"  TRANTIME=0
              SRVGRP=group1
"PAYMENT"   TRANTIME=0
              SRVGRP=group1
"ORDERSTATUS" TRANTIME=0
              SRVGRP=group1
"DELIVERY"  TRANTIME=0
              SRVGRP=group1
"STOCKLVL"  TRANTIME=0
              SRVGRP=group1

*ROUTING

===== client 12 =====
REGEDIT4

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters]
"BandwidthLevel"=dword:ffffff
"ListenBackLog"=dword:00000019
"MemoryCacheSize"=dword:00000000
"ObjectCacheTTL"=dword:ffffff
"PoolThreadLimit"=dword:0000020d

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters\Filter]
"FilterType"=dword:00000000
"NumGrantSites"=dword:00000000
"NumDenySites"=dword:00000000

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters\MimeMap]
"text/html,html,h"=""
"image/gif,gif,g"=""
"image/jpeg,jpg,,"=""
"text/plain,txt,0"=""
"text/html,html,h"=""
"image/jpeg,jpeg,,"=""
"image/jpeg,jpe,,"=""
"image/bmp,bmp,,"=""
"application/octet-stream,*,,5"=""
"application/pdf,pdf,,5"=""
"application/octet-stream,bin,,5"=""
"application/oda,oda,,5"=""
"application/zip,zip,,9"=""
"application/rft,rft,,5"=""
"application/postscript,ps,,5"=""
"application/postscript,ai,,5"=""
"application/postscript,eps,,5"=""
"application/mac-binhex40,hqx,,4"=""
"application/msword,doc,,5"=""
"application/msword,dot,,5"=""
"application/winhlp,hlp,,5"=""
"video/mpeg,mpeg,,"=""
"video/mpeg,mpeg,,"=""
"video/mpeg,mpe,,"=""
"video/x-msvideo,avi,<,"=""
"video/quicktime,qt,,"=""
"video/quicktime,mov,,"=""
"video/x-sgi-movie,movie,<,"=""
"x-world/x-vrml,wrl,,5"=""
"x-world/x-vrml,xaf,,5"=""
"x-world/x-vrml,xof,,5"=""
"x-world/x-vrml,flr,,5"=""
"x-world/x-vrml,wrz,,5"=""
"application/x-director,dcr,,5"=""
"application/x-director,dir,,5"=""
"application/x-director,dxr,,5"=""
"image/cis-cod,cod,,5"=""
"image/x-cmx,cmx,,5"=""
"application/envoy,evy,,5"=""
"application/x-msaccess,mdb,,5"=""
"application/x-mscardfile,crd,,5"=""
"application/x-msclip,clip,,5"=""
"application/octet-stream,exe,,5"=""
"application/x-msexcel,xla,,5"=""
"application/x-msexcel,xlc,,5"=""
"application/x-msexcel,xlm,,5"=""
"application/x-msexcel,xls,,5"=""
"application/x-msexcel,xlt,,5"=""
"application/x-msexcel,xlw,,5"=""
"application/x-msmediaview,m13,,5"=""
"application/x-msmediaview,m14,,5"=""
"application/x-msmoney,mny,,5"=""
"application/x-mspowerpoint,ppt,,5"=""
"application/x-msproject,mpp,,5"=""
"application/x-mspublisher,pub,,5"=""
"application/x-msterminal,rm,,5"=""
"application/x-msworks,wks,,5"=""
"application/x-mswrite,wri,,5"=""
"application/x-msmetafile,wmf,,5"=""
"application/x-csh,csh,,5"=""
"application/x-dvi,dvi,,5"=""
"application/x-hdf,hdf,,5"=""
"application/x-latex,latex,,5"=""
"application/x-netcdf,nc,,5"=""
"application/x-netcdf,cdf,,5"=""
"application/x-sh,sh,,5"=""
"application/x-tcl,tcl,,5"=""

```

```
"application/x-tex,tex,5"=""
"application/x-texinfo,texinfo,5"=""
"application/x-texinfo,texi,5"=""
"application/x-troff,t,5"=""
"application/x-troff,tr,5"=""
"application/x-troff,roff,5"=""
"application/x-troff-man,man,5"=""
"application/x-troff-me,me,5"=""
"application/x-troff-ms,ms,5"=""
"application/x-wais-source,src,7"=""
"application/x-bcpio,bcpio,5"=""
"application/x-cpio,cpio,5"=""
"application/x-gtar,gtar,9"=""
"application/x-shar,shar,5"=""
"application/x-sv4cpio,sv4cpio,5"=""
"application/x-sv4crc,sv4crc,5"=""
"application/x-tar,tar,5"=""
"application/x-ustar,ustar,5"=""
"audio/basic,au,<"=""
"audio/basic,snd,<"=""
"audio/x-aiff,aif,<"=""
"audio/x-aiff,aiff,<"=""
"audio/x-aiff,aifc,<"=""
"audio/x-wav,wav,<"=""
"audio/x-pn-realaudio,ram,<"=""
"image/ief,ief,:"=""
"image/tiff,tiff,:"=""
"image/tiff,tif,:"=""
"image/x-cmu-raster,ras,:"=""
"image/x-portable-anymap,pnm,:"=""
"image/x-portable-bitmap,pbm,:"=""
"image/x-portable-graymap,pgm,:"=""
"image/x-portable-pixmap,ppm,:"=""
"image/x-rgb,rgb,:"=""
"image/x-xbitmap,xbm,:"=""
"image/x-xpixmap,xpm,:"=""
"image/x-xwindowdump,xwd,:"=""
"text/html,stm,h"=""
"text/plain,bas,0"=""
"text/plain,c,0"=""
"text/plain,h,0"=""
"text/richtext,rtx,0"=""
"text/tab-separated-values,tsv,0"=""
"text/x-setext,etx,0"=""
"application/x-perfmon,pmc,5"=""
"application/x-perfmon,pma,5"=""
"application/x-perfmon,pmr,5"=""
"application/x-perfmon,pml,5"=""
"application/x-perfmon,pmw,5"=""
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment]
"TUXDIR"="C:\TUXEDO"
"NLSPATH"="C:\TUXEDO\locale\LC"
"ULOGDIR"="C:\TUXEDO"
"ULOGPF"="C:\ULOG"
"ULOGOUT"=dword:00000001
"TUXIPC_MSG_BYTES"=dword:00010000
"TUXIPC_MSG_HDRS"=dword:00001fc0
"TUXIPC_MSG_QUEUE"=dword:00000320
"TUXIPC_MSG_QUEUE_BYTES"=dword:000100
00
"TUXIPC_MSG_SEG_BYTES"=dword:00000040
"TUXIPC_MSG_SEGS"=dword:00007fff
"TUXIPC_PROC"=dword:00001770
"TUXIPC_SEM"=dword:00001770
"TUXIPC_SEM_IDS"=dword:00001770
```

```
"TUXIPC_SEM_UNDO"=dword:00001770
"TUXIPC_SHM_PROCS"=dword:000186a0
"TUXIPC_SHM_SEGS"=dword:0000c350
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services]
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services\305
0]
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC]
"Type"=dword:00000020
"Start"=dword:00000002
"ErrorControl"=dword:00000000
"ImagePath"=hex(2):43,3a,5c,57,49,4e,4e,54,5c,5
3,79,73,74,65,6d,33,32,5c,69,6e,\
65,74,73,72,76,5c,69,6e,65,74,69,6e,66,6f,2e,65,
78,65,00
"DisplayName"="World Wide Web Publishing
Service"
"DependOnService"=hex(7):52,50,43,53,53,00,4e,
54,4c,4d,53,53,50,00,00
"DependOnGroup"=hex(7):00
"ObjectName"="LocalSystem"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters]
"MajorVersion"=dword:00000002
"MinorVersion"=dword:00000000
"AdminName"="Administrator"
"AdminEmail"="Admin@corp.com"
"MaxConnections"=dword:000186a0
"LogType"=dword:00000000
"LogFileDirectory"=hex(2):25,53,79,73,74,65,6d,5
2,6f,6f,74,25,5c,53,79,73,74,\
65,6d,33,32,5c,4c,6f,67,46,69,6c,65,73,00
"LogFileTruncateSize"=dword:01400000
"LogFilePeriod"=dword:00000001
"LogFileFormat"=dword:00000000
"LogSqlDataSource"="HTTPLOG"
"LogSqlTableName"="Internetlog"
"LogSqlUserName"="InternetAdmin"
"LogSqlPassword"="sqllog"
"Authorization"=dword:00000005
"AnonymousUserName"="IUSR_CL11"
"Default Load File"="Default.htm"
"Dir Browse Control"=dword:4000001e
"CheckForWAISDB"=dword:00000000
"CacheExtensions"=dword:00000001
"GlobalExpire"=dword:ffffff
"ServerSideIncludesEnabled"=dword:00000001
"ServerSideIncludesExtension"=".stm"
"DebugFlags"=dword:00000008
"ScriptTimeout"=dword:00000384
"ConnectionTimeout"=dword:00000384
"InstallPath"="C:\WINNT\System32\inet\svr"
"SecurePort"=dword:000001bb
"Filter
DLLs"="C:\WINNT\System32\inet\svr\lss\pifilt.dll"
"AccessDeniedMessage"="エラー : 予約は拒否さ
れました。"
"NTAuthenticationProviders"="NTLM"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Script Map]
```

```
".idc"="C:\WINNT\System32\inet\svr\http\podb.dll"
"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Virtual Roots]
"/"="C:\inet\pub\wwwroot,1"
"/tpc"="C:\inet\pub\tpap\Release,5"
"/Scripts"="C:\inet\pub\scripts,4"
"/iisadmin"="C:\WINNT\System32\inet\svr\iisad
min,1"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Performance]
"Library"="w3ctrs.DLL"
"Open"="OpenW3PerformanceData"
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"Last Counter"=dword:00000790
"Last Help"=dword:00000791
"First Counter"=dword:00000758
"First Help"=dword:00000759
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Security]
"Security"=hex:01,00,14,80,c0,00,00,00,cc,00,00,
00,14,00,00,00,34,00,00,00,02,\
00,20,00,01,00,00,00,02,80,18,00,ff,01,0f,00,01,
01,00,00,00,00,00,01,00,00,\
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,00,0
0,18,00,8d,01,02,00,01,01,00,\
00,00,00,00,01,00,00,00,00,48,e4,14,00,00,00,1
c,00,fd,01,02,00,01,02,00,00,\
00,00,00,05,20,00,00,00,23,02,00,00,65,00,72,0
0,00,00,1c,00,ff,01,0f,00,01,\
02,00,00,00,00,05,20,00,00,00,20,02,00,00,6
5,00,72,00,00,00,1c,00,ff,01,\
0f,00,01,02,00,00,00,00,05,20,00,00,25,0
2,00,00,65,00,72,00,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,05,12,00,0
0,00,25,02,00,00,01,01,00,00,\
00,00,00,05,12,00,00,00,01,01,00,00,00,00,0
5,12,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\W3SAMP]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Enum]
"0"="Root\LEGACY_W3SVC\00000"
"Count"=dword:00000001
"NextInstance"=dword:00000001
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Fujitsu\T
PC-C ISAPI Application]
"Term_Base"=dword:cc1b
"NumWarehouses"=dword:2c88
"MaxUsers"=dword:1bd50
"MaxTerm of Client"=dword:128e
"CONTROL_Flag"=dword:00000000
```

```
#
# ubbconfig : TUXEDO configuration file
#
```

```
*RESOURCES
IPCKEY 80952
MASTER SITE1
```

```
#UID 1
#GID 1
PERM 0660
MAXACCESSERS 800
MAXSERVERS 200
MAXSERVICES 1000
MODEL SHM
LDBAL Y

*MACHINES
CL12 LMID=SITE1
  APPDIR="c:\sv-aplora"
  TUXCONFIG="c:\client\tuxconfig"
  TUXDIR="c:\tuxedo"
  ULOGPFX="c:\tuxlog\tst"

*GROUPS
group1 LMID=SITE1 GRPNO=1

*SERVERS
DEFAULT: RESTART=Y MAXGEN=5
ROPERM=0660
tpccfml SRVGRP=group1 RQADDR=newq1
SRVID=1 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq2
SRVID=2 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq3
SRVID=3 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq4
SRVID=4 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq5
SRVID=5 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq6
SRVID=6 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq7
SRVID=7 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq8
SRVID=8 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq9
SRVID=9 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq10
SRVID=10 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq11
SRVID=11 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq12
SRVID=12 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq13
SRVID=13 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq14
SRVID=14 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfmlp SRVGRP=group1 RQADDR=payq1
SRVID=15 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
```

```
tpccfmlp SRVGRP=group1 RQADDR=payq2
SRVID=16 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq3
SRVID=17 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq4
SRVID=18 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq5
SRVID=19 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq6
SRVID=20 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq7
SRVID=21 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlo SRVGRP=group1 RQADDR=ordq1
SRVID=22 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfmlo SRVGRP=group1 RQADDR=ordq2
SRVID=23 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfmld SRVGRP=group1 RQADDR=delq1
SRVID=24 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmld SRVGRP=group1 RQADDR=delq2
SRVID=25 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmld SRVGRP=group1 RQADDR=delq3
SRVID=26 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmlls SRVGRP=group1 RQADDR=stoq1
SRVID=27 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfmlls SRVGRP=group1 RQADDR=stoq2
SRVID=28 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfmlls SRVGRP=group1 RQADDR=stoq3
SRVID=29 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"

*SERVICES
"NEWORDER" TRANTIME=0
SRVGRP=group1
"PAYMENT" TRANTIME=0
SRVGRP=group1
"ORDERSTATUS" TRANTIME=0
SRVGRP=group1
"DELIVERY" TRANTIME=0
SRVGRP=group1
"STOCKLVL" TRANTIME=0
SRVGRP=group1

*ROUTING

===== client 13 =====
REGEDIT4

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters]
"BandwidthLevel"=dword:ffffff
"ListenBackLog"=dword:00000019
"MemoryCacheSize"=dword:00000000
"ObjectCacheTTL"=dword:ffffff
"PoolThreadLimit"=dword:0000020d
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters\Filter]
"FilterType"=dword:00000000
"NumGrantSites"=dword:00000000
"NumDenySites"=dword:00000000
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters\MimeMap]
"text/html,html,,h"=""
"image/gif,gif,,g"=""
"image/jpeg,jpg,,j"=""
"text/plain,txt,0"=""
"text/html,html,,h"=""
"image/jpeg,jpeg,,j"=""
"image/jpeg,jpe,,j"=""
"image/bmp,bmp,,b"=""
"application/octet-stream,*,,5"=""
"application/pdf,pdf,,5"=""
"application/octet-stream,bin,,5"=""
"application/oda,oda,,5"=""
"application/zip,zip,,9"=""
"application/rft,rft,,5"=""
"application/postscript,ps,5"=""
"application/postscript,ai,,5"=""
"application/postscript,eps,,5"=""
"application/mac-binhex40,hqx,,4"=""
"application/msword,doc,,5"=""
"application/msword,dot,,5"=""
"application/winhlip,hlp,,5"=""
"video/mpeg,mpeg,,m"=""
"video/mpeg,mpg,,m"=""
"video/mpeg,mpe,,m"=""
"video/x-msvideo,avi,,<"=""
"video/quicktime,qt,,q"=""
"video/quicktime,mov,,m"=""
"video/x-sgi-movie,movie,,<"=""
"x-world/x-vrml,wrl,,5"=""
"x-world/x-vrml,xaf,,5"=""
"x-world/x-vrml,xof,,5"=""
"x-world/x-vrml,flr,,5"=""
"x-world/x-vrml,wrz,,5"=""
"application/x-director,dcr,,5"=""
"application/x-director,dir,,5"=""
"application/x-director,dxr,,5"=""
"image/cis-cod,cod,,5"=""
"image/x-cmx,cmx,,5"=""
"application/envoy,envy,,5"=""
"application/x-msaccess,mdb,,5"=""
"application/x-mscardfile,crd,,5"=""
"application/x-msclip,clip,,5"=""
"application/octet-stream,exe,,5"=""
"application/x-msexcel,xla,,5"=""
"application/x-msexcel,xlc,,5"=""
"application/x-msexcel,xlm,,5"=""
"application/x-msexcel,xls,,5"=""
"application/x-msexcel,xlt,,5"=""
"application/x-msexcel,xlw,,5"=""
"application/x-msmediaview,m13,,5"=""
"application/x-msmediaview,m14,,5"=""
"application/x-msmoney,mny,,5"=""
"application/x-mspowerpoint,ppt,,5"=""
"application/x-msproject,mpp,,5"=""
"application/x-mspublisher,pub,,5"=""
"application/x-msterminal,trm,,5"=""
"application/x-msworks,wks,,5"=""
"application/x-mswrite,wri,,5"=""
"application/x-msmetafile,wmf,,5"=""
"application/x-csh,csh,,5"=""
"application/x-dvi,dvi,,5"=""
```

```
"application/x-hdf,hdf,,5"=""
"application/x-latex,latex,,5"=""
"application/x-netcdf,nc,,5"=""
"application/x-netcdf,cdf,,5"=""
"application/x-sh,sh,,5"=""
"application/x-tcl,tcl,,5"=""
"application/x-tex,tex,,5"=""
"application/x-texinfo,texinfo,,5"=""
"application/x-texinfo,texi,,5"=""
"application/x-troff,t,,5"=""
"application/x-troff,tr,,5"=""
"application/x-troff,roff,,5"=""
"application/x-troff-man,man,,5"=""
"application/x-troff-me,me,,5"=""
"application/x-troff-ms,ms,,5"=""
"application/x-wais-source,src,,7"=""
"application/x-bcpio,bcpio,,5"=""
"application/x-cpio,cpio,,5"=""
"application/x-gtar,gtar,,9"=""
"application/x-shar,shar,,5"=""
"application/x-sv4cpio,sv4cpio,,5"=""
"application/x-sv4crc,sv4crc,,5"=""
"application/x-tar,tar,,5"=""
"application/x-ustar,ustar,,5"=""
"audio/basic,au,,<"=""
"audio/basic,snd,,<"=""
"audio/x-aiff,aiff,,<"=""
"audio/x-aiff,aiff,,<"=""
"audio/x-aiff,aiff,,<"=""
"audio/x-wav,wav,,<"=""
"audio/x-pn-realaudio,ram,,<"=""
"image/ief,ief,,:"=""
"image/tiff,tiff,,:"=""
"image/tiff,tif,,:"=""
"image/x-cmu-raster,ras,,:"=""
"image/x-portable-anymap,pnm,,:"=""
"image/x-portable-bitmap,pbm,,:"=""
"image/x-portable-graymap,pgm,,:"=""
"image/x-portable-pixmap,ppm,,:"=""
"image/x-rgb,rgb,,:"=""
"image/x-xbitmap,xbm,,:"=""
"image/x-xpixmap,xpm,,:"=""
"image/x-xwindowdump,xwd,,:"=""
"text/html,stm,,h"=""
"text/plain,bas,,0"=""
"text/plain,c,,0"=""
"text/plain,h,,0"=""
"text/richtext,rtx,,0"=""
"text/tab-separated-values,tsv,,0"=""
"text/x-setext,etx,,0"=""
"application/x-perfmon,pmc,,5"=""
"application/x-perfmon,pma,,5"=""
"application/x-perfmon,pmr,,5"=""
"application/x-perfmon,pml,,5"=""
"application/x-perfmon,pmw,,5"=""
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment]
"TUXDIR"="C:\TUXEDO"
"NLSPATH"="C:\TUXEDO\locale\VC"
"ULOGDIR"="C:\TUXEDO"
"ULOGPF"="C:\ULOG"
"ULOGOUT"=dword:00000001
"TUXIPC_MSG_BYTES"=dword:00010000
"TUXIPC_MSG_HDRS"=dword:00001fc0
"TUXIPC_MSG_QUEUES"=dword:00000320
```

```
"TUXIPC_MSG_QUEUE_BYTES"=dword:000100
00
"TUXIPC_MSG_SEG_BYTES"=dword:00000040
"TUXIPC_MSG_SEGS"=dword:00007fff
"TUXIPC_PROC"=dword:00001770
"TUXIPC_SEM"=dword:00001770
"TUXIPC_SEM_IDS"=dword:00001770
"TUXIPC_SEM_UNDO"=dword:00001770
"TUXIPC_SHM_PROCS"=dword:000186a0
"TUXIPC_SHM_SEGS"=dword:0000c350
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services]
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services\305
0]
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC]
"Type"=dword:00000020
"Start"=dword:00000002
"ErrorControl"=dword:00000000
"ImagePath"=hex(2):43,3a,5c,57,49,4e,4e,54,5c,5
3,79,73,74,65,6d,33,32,5c,69,6e,\
65,74,73,72,76,5c,69,6e,65,74,69,6e,66,6f,2e,65,
78,65,00
"DisplayName"="World Wide Web Publishing
Service"
"DependOnService"=hex(7):52,50,43,53,53,00,4e,
54,4c,4d,53,53,50,00,00
"DependOnGroup"=hex(7):00
"ObjectName"="LocalSystem"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters]
"MajorVersion"=dword:00000002
"MinorVersion"=dword:00000000
"AdminName"="Administrator"
"AdminEmail"="Admin@corp.com"
"MaxConnections"=dword:000186a0
"LogType"=dword:00000000
"LogFileDirectory"=hex(2):25,53,79,73,74,65,6d,5
2,6f,6f,74,25,5c,53,79,73,74,\
65,6d,33,32,5c,4c,6f,67,46,69,6c,65,73,00
"LogFileTruncateSize"=dword:01400000
"LogFilePeriod"=dword:00000001
"LogFileFormat"=dword:00000000
"LogSqlDataSource"="HTTPLOG"
"LogSqlTableName"="Internetlog"
"LogSqlUserName"="InternetAdmin"
"LogSqlPassword"="sqllog"
"Authorization"=dword:00000005
"AnonymousUserName"="IUSR_CL13"
"Default Load File"="Default.htm"
"Dir Browse Control"=dword:4000001e
"CheckForWAISDB"=dword:00000000
"CacheExtensions"=dword:00000001
"GlobalExpire"=dword:ffffff
"ServerSideIncludesEnabled"=dword:00000001
"ServerSideIncludesExtension"=".stm"
"DebugFlags"=dword:00000008
"ScriptTimeout"=dword:00000384
"ConnectionTimeout"=dword:00000384
"InstallPath"="C:\WINNT\System32\inetshr"
"SecurePort"=dword:000001b1
```

```
"Filter
DLLs"="C:\WINNT\System32\inetshr\lsspifill.dll"
"AccessDeniedMessage"="エラー : アクセスは拒否さ
れました。 "
"NTAuthenticationProviders"="NTLM"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Script Map]
".idc"="C:\WINNT\System32\inetshr\httpodbc.dll"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Virtual Roots]
"/"="C:\inetpub\wwwroot,,1"
"/tpc"="C:\client\tpap\Release,,5"
"/tpcr"="C:\client\tpap\Release_dll,,5"
"/Scripts"="C:\inetpub\scripts,,4"
"/iisadmin"="C:\WINNT\System32\inetshr\iisad
min,,1"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Performance]
"Library"="w3ctrs.DLL"
"Open"="OpenW3PerformanceData"
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"Last Counter"=dword:00000790
"Last Help"=dword:00000791
"First Counter"=dword:00000758
"First Help"=dword:00000759
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Security]
"Security"=hex:01,00,14,80,c0,00,00,00,cc,00,00,
00,14,00,00,00,34,00,00,00,02,\
00,20,00,01,00,00,00,02,80,18,00,ff,01,0f,00,01,
01,00,00,00,00,00,01,00,00,\
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,00,0
0,18,00,8d,01,02,00,01,01,00,\
00,00,00,00,01,00,00,00,00,00,00,00,00,00,01,
c,00,fd,01,02,00,01,02,00,00,\
00,00,00,05,20,00,00,23,02,00,00,00,00,00,00,0
0,00,01,c,00,ff,01,0f,00,01,\
02,00,00,00,00,05,20,00,00,00,20,02,00,00,0
0,00,00,00,00,01,c,00,ff,01,\
0f,00,01,02,00,00,00,00,05,20,00,00,00,25,0
2,00,00,00,00,00,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,00,05,12,00,0
0,00,25,02,00,00,01,01,00,00,\
00,00,00,05,12,00,00,00,01,01,00,00,00,00,00,0
5,12,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\W3SAMP]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Enum]
"0"="Root\LEGACY_W3SVC\0000"
"Count"=dword:00000001
"NextInstance"=dword:00000001
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Fujitsu\T
PC-C ISAPI Application]
"Term_Base"=dword:dea9
"NumWarehouses"=dword:2c88
"MaxUsers"=dword:1bd50
"MaxTerm of Client"=dword:128e
```

```
"CONTROL_Flag"=dword:00000000
```

```
#
# ubbconfig : TUXEDO configuration file
#
```

```
*RESOURCES
```

```
IPCKEY      80952
MASTER     SITE1
#UID       1
#GID       1
PERM       0660
MAXACCESSERS 800
MAXSERVERS 200
MAXSERVICES 1000
MODEL      SHM
LDBAL      Y
```

```
*MACHINES
```

```
CL13 LMID=SITE1
  APPDIR="c:\sv-aplora"
  TUXCONFIG="c:\client\tuxconfig"
  TUXDIR="c:\tuxedo"
  ULOGPFX="c:\tuxlog\test"
```

```
*GROUPS
```

```
group1 LMID=SITE1 GRPNO=1
```

```
*SERVERS
```

```
DEFAULT: RESTART=Y MAXGEN=5
RQPERM=0660
tpccfml SRVGRP=group1 RQADDR=newq1
SRVID=1 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq2
SRVID=2 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq3
SRVID=3 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq4
SRVID=4 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq5
SRVID=5 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq6
SRVID=6 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq7
SRVID=7 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq8
SRVID=8 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq9
SRVID=9 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq10
SRVID=10 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq11
SRVID=11 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq12
SRVID=12 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
```

```
tpccfml SRVGRP=group1 RQADDR=newq13
SRVID=13 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq14
SRVID=14 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=payq1
SRVID=15 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=payq2
SRVID=16 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=payq3
SRVID=17 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=payq4
SRVID=18 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=payq5
SRVID=19 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=payq6
SRVID=20 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=payq7
SRVID=21 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=ordq1
SRVID=22 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfml SRVGRP=group1 RQADDR=ordq2
SRVID=23 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfml SRVGRP=group1 RQADDR=delq1
SRVID=24 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfml SRVGRP=group1 RQADDR=delq2
SRVID=25 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfml SRVGRP=group1 RQADDR=delq3
SRVID=26 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfml SRVGRP=group1 RQADDR=stoq1
SRVID=27 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfml SRVGRP=group1 RQADDR=stoq2
SRVID=28 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfml SRVGRP=group1 RQADDR=stoq3
SRVID=29 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
```

```
*SERVICES
```

```
"NEWORDER" TRANTIME=0
SRVGRP=group1
"PAYMENT" TRANTIME=0
SRVGRP=group1
"ORDERSTATUS" TRANTIME=0
SRVGRP=group1
"DELIVERY" TRANTIME=0
SRVGRP=group1
"STOCKLVL" TRANTIME=0
SRVGRP=group1
```

```
*ROUTING
```

```
===== client 14 =====
REGEDIT4
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\inetInfo\Parameters]
"BandwidthLevel"=dword:ffffff
"ListenBackLog"=dword:00000019
"MemoryCacheSize"=dword:00000000
"ObjectCacheTTL"=dword:ffffff
"PoolThreadLimit"=dword:0000020d
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\inetInfo\Parameters\Filter]
"FilterType"=dword:00000000
"NumGrantSites"=dword:00000000
"NumDenySites"=dword:00000000
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\inetInfo\Parameters\MimeMap]
"text/html,html,h"=""
"image/gif,gif,g"=""
"image/jpeg,jpg,,"=""
"text/plain,txt,0"=""
"text/html,html,h"=""
"image/jpeg,jpeg,,"=""
"image/jpeg,jpe,,"=""
"image/bmp,bmp,,"=""
"application/octet-stream,*,,5"=""
"application/pdf,pdf,,5"=""
"application/octet-stream,bin,,5"=""
"application/oda,oda,,5"=""
"application/zip,zip,,9"=""
"application/rtf,rtf,,5"=""
"application/postscript,ps,5"=""
"application/postscript,ai,,5"=""
"application/postscript,eps,5"=""
"application/mac-binhex40,hqx,4"=""
"application/msword,doc,5"=""
"application/msword,dot,5"=""
"application/winhttp,hlp,5"=""
"video/mpeg,mpeg,,"=""
"video/mpeg,mpg,,"=""
"video/mpeg,mpe,,"=""
"video/x-msvideo,avi,<"=""
"video/quicktime,qt,,"=""
"video/quicktime,mov,,"=""
"video/x-sgi-movie,movie,<"=""
"x-world/x-vrml,wr1,5"=""
"x-world/x-vrml,xaf,5"=""
"x-world/x-vrml,xof,5"=""
"x-world/x-vrml,flr,5"=""
"x-world/x-vrml,wrz,5"=""
"application/x-director,dcr,5"=""
"application/x-director,dir,5"=""
"application/x-director,dxr,5"=""
"image/cis-cod,cod,5"=""
"image/x-cmx,cmx,5"=""
"application/envoy,evy,5"=""
"application/x-msaccess,mdb,5"=""
"application/x-mscardfile,crd,5"=""
"application/x-msclip,clip,5"=""
"application/octet-stream,exe,5"=""
"application/x-msexcel,xla,5"=""
"application/x-msexcel,xlc,5"=""
"application/x-msexcel,xlm,5"=""
"application/x-msexcel,xls,5"=""
"application/x-msexcel,xlt,5"=""
"application/x-msexcel,xlw,5"=""
"application/x-msmediaview,m13,5"=""
"application/x-msmediaview,m14,5"=""
"application/x-msmoney,mny,5"=""
```



```

"application/x-mspowerpoint.ppt,,5"=""
"application/x-msproject.mpp,,5"=""
"application/x-mspublisher.pub,,5"=""
"application/x-msterminal.trm,,5"=""
"application/x-msworks.wks,,5"=""
"application/x-mswrite.wri,,5"=""
"application/x-msmetafile.wmf,,5"=""
"application/x-csh.csh,,5"=""
"application/x-dvi.dvi,,5"=""
"application/x-hdf.hdf,,5"=""
"application/x-latex.latex,,5"=""
"application/x-netcdf.nc,,5"=""
"application/x-netcdf.cdf,,5"=""
"application/x-sh.sh,,5"=""
"application/x-tcl.tcl,,5"=""
"application/x-tex.tex,,5"=""
"application/x-texinfo.texinfo,,5"=""
"application/x-texinfo.texi,,5"=""
"application/x-troff.t,,5"=""
"application/x-troff.tr,,5"=""
"application/x-troff.roff,,5"=""
"application/x-troff-man.man,,5"=""
"application/x-troff-me.me,,5"=""
"application/x-troff-ms.ms,,5"=""
"application/x-wais-source.src,,7"=""
"application/x-bcpio.bcpio,,5"=""
"application/x-cpio.cpio,,5"=""
"application/x-gtar.gtar,,9"=""
"application/x-shar.shar,,5"=""
"application/x-sv4cpio.sv4cpio,,5"=""
"application/x-sv4crc.sv4crc,,5"=""
"application/x-tar.tar,,5"=""
"application/x-ustar.ustar,,5"=""
"audio/basic.au,,<"=""
"audio/basic.snd,,<"=""
"audio/x-aiff.aiff,,<"=""
"audio/x-aiff.aiff,,<"=""
"audio/x-aiff.aifc,,<"=""
"audio/x-wav.wav,,<"=""
"audio/x-pn-realaudio.ram,,<"=""
"image/ief.ief,,:"=""
"image/tiff.tiff,,:"=""
"image/tiff.tif,,:"=""
"image/x-cmu-raster.ras,,:"=""
"image/x-portable-anymap.pnm,,:"=""
"image/x-portable-bitmap.pbm,,:"=""
"image/x-portable-graymap.pgm,,:"=""
"image/x-portable-pixmap.ppm,,:"=""
"image/x-rgb.rgb,,:"=""
"image/x-xbitmap.xbm,,:"=""
"image/x-xpixmap.xpm,,:"=""
"image/x-xwindowdump.xwd,,:"=""
"text/html.stm,,h"=""
"text/plain.bas,,0"=""
"text/plain.c,,0"=""
"text/plain.h,,0"=""
"text/richtext.rtx,,0"=""
"text/tab-separated-values.tsv,,0"=""
"text/x-setext.etx,,0"=""
"application/x-perfmon.pmc,,5"=""
"application/x-perfmon.pma,,5"=""
"application/x-perfmon.pmr,,5"=""
"application/x-perfmon.pml,,5"=""
"application/x-perfmon.pmw,,5"=""

```

REGEDIT4

[HKEY_LOCAL_MACHINE\SOFTWARE\BEA Systems\TUXEDO\6.4\Environment]

```

"TUXDIR"="C:\TUXEDO"
"NLSPATH"="C:\TUXEDO\locale\C"
"ULOGDIR"="C:\TUXEDO"
"ULOGPFX"="C:\ULOG"
"ULOGOUT"=dword:00000001
"TUXIPC_MSG_BYTES"=dword:00010000
"TUXIPC_MSG_HDRS"=dword:00001fc0
"TUXIPC_MSG_QUEUEES"=dword:00000320
"TUXIPC_MSG_QUEUE_BYTES"=dword:00010000
"TUXIPC_MSG_SEG_BYTES"=dword:00000040
"TUXIPC_MSG_SEGS"=dword:00007fff
"TUXIPC_PROC"=dword:00001770
"TUXIPC_SEM"=dword:00001770
"TUXIPC_SEM_IDS"=dword:00001770
"TUXIPC_SEM_UNDO"=dword:00001770
"TUXIPC_SHM_PROCS"=dword:00000800
"TUXIPC_SHM_SEGS"=dword:00000100

```

[HKEY_LOCAL_MACHINE\SOFTWARE\BEA Systems\TUXEDO\6.4\Environment\Services]

[HKEY_LOCAL_MACHINE\SOFTWARE\BEA Systems\TUXEDO\6.4\Environment\Services\305 0]

REGEDIT4

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC]
"Type"=dword:00000020
"Start"=dword:00000002
"ErrorControl"=dword:00000000
"ImagePath"=hex(2):43,3a,5c,57,49,4e,4e,54,5c,53,79,73,74,65,6d,33,32,5c,69,6e,\
65,74,73,72,76,5c,69,6e,65,74,69,6e,66,6f,2e,65,78,65,00
"DisplayName"="World Wide Web Publishing Service"
"DependOnService"=hex(7):52,50,43,53,53,00,4e,54,4c,4d,53,53,50,00,00
"DependOnGroup"=hex(7):00
"ObjectName"="LocalSystem"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Parameters]
"MajorVersion"=dword:00000002
"MinorVersion"=dword:00000000
"AdminName"="Administrator"
"AdminEmail"="Admin@corp.com"
"MaxConnections"=dword:000186a0
"LogType"=dword:00000000
"LogFileDirectory"=hex(2):25,53,79,73,74,65,6d,65,2,6f,6f,74,25,5c,53,79,73,74,\
65,6d,33,32,5c,4c,6f,67,46,69,6c,65,73,00
"LogFileTruncateSize"=dword:01400000
"LogFilePeriod"=dword:00000001
"LogFileFormat"=dword:00000000
"LogSqlDataSource"="HTTPLOG"
"LogSqlTableName"="Internetlog"
"LogSqlUserName"="InternetAdmin"
"LogSqlPassword"="sqllog"
"Authorization"=dword:00000005
"AnonymousUserName"="IUSR_CL24"
"Default Load File"="Default.htm"
"Dir Browse Control"=dword:4000001e
"CheckForWAISDB"=dword:00000000
"CacheExtensions"=dword:00000001
"GlobalExpire"=dword:ffffff

```

```

"ServerSideIncludesEnabled"=dword:00000001
"ServerSideIncludesExtension"=".stm"
"DebugFlags"=dword:00000008
"ScriptTimeout"=dword:00000384
"ConnectionTimeout"=dword:00000384
"InstallPath"="C:\WINNT\System32\inetnsrv"
"SecurePort"=dword:000001bb
"Filter
DLLs"="C:\WINNT\System32\inetnsrv\lsspifilt.dll"
"AccessDeniedMessage"="エラー: 予知は拒否されました。"
"NTAuthenticationProviders"="NTLM"

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Parameters\Script Map]
".idc"="C:\WINNT\System32\inetnsrv\httpodbc.dll"

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Parameters\Virtual Roots]
"/."="C:\inetPub\wwwroot,,1"
"/tpc."="C:\client\tpap\Release,,5"
"/Scripts."="C:\inetPub\Scripts,,4"
"/iisadmin."="C:\WINNT\System32\inetnsrv\iisadmin,,1"

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Performance]
"Library"="w3ctrs.DLL"
"Open"="OpenW3PerformanceData"
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"Last Counter"=dword:00000790
"Last Help"=dword:00000791
"First Counter"=dword:00000758
"First Help"=dword:00000759

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Security]
"Security"=hex:01,00,14,80,c0,00,00,00,cc,00,00,00,14,00,00,00,34,00,00,00,02,\
00,20,00,01,00,00,02,80,18,00,ff,01,0f,00,01,01,00,00,00,00,01,00,00,\
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,00,00,0,18,00,8d,01,02,00,01,01,00,\
00,00,00,00,01,00,00,00,00,00,00,00,00,00,00,1c,00,fd,01,02,00,01,02,00,\
00,00,00,05,20,00,00,00,23,02,00,00,00,00,00,00,0,00,00,1c,00,ff,01,0f,00,01,\
02,00,00,00,00,00,05,20,00,00,00,20,02,00,00,0,00,00,00,00,00,1c,00,ff,01,\
0f,00,01,02,00,00,00,00,05,20,00,00,00,25,0,00,00,00,00,00,00,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,05,12,00,0,00,00,25,02,00,00,01,01,00,00,\
00,00,00,05,12,00,00,00,01,01,00,00,00,00,00,0,5,12,00,00,00

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\W3SAMP]

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Enum]
"0"="Root\LEGACY_W3SVC\0000"
"Count"=dword:00000001
"NextInstance"=dword:00000001

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Fujitsu\T
PC-C ISAPI Application]
"Term_Base"=dword:f137
"NumWarehouses"=dword:2c88
"MaxUsers"=dword:1bd50
"MaxTerm of Client"=dword:128e
"CONTROL_Flag"=dword:00000000

#
#   ubbconfig : TUXEDO configuration file
#

*RESOURCES
IPCKEY      80952
MASTER     SITE1
#UID        1
#GID        1
PERM        0660
MAXACCESSERS 800
MAXSERVERS 200
MAXSERVICES 1000
MODEL       SHM
LDBAL       Y

*MACHINES
CL14 LMID=SITE1
  APPDIR="c:\sv-aplora"
  TUXCONFIG="c:\client\tuxconfig"
  TUXDIR="c:\tuxedo"
  ULOGPFX="c:\tuxlog\test"

*GROUPS
group1     LMID=SITE1 GRPNO=1

*SERVERS
DEFAULT:  RESTART=Y MAXGEN=5
RPERM=0660
tpccfml  SRVGRP=group1 RQADDR=newq1
SRVID=1  REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml  SRVGRP=group1 RQADDR=newq2
SRVID=2  REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml  SRVGRP=group1 RQADDR=newq3
SRVID=3  REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml  SRVGRP=group1 RQADDR=newq4
SRVID=4  REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml  SRVGRP=group1 RQADDR=newq5
SRVID=5  REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml  SRVGRP=group1 RQADDR=newq6
SRVID=6  REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml  SRVGRP=group1 RQADDR=newq7
SRVID=7  REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml  SRVGRP=group1 RQADDR=newq8
SRVID=8  REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml  SRVGRP=group1 RQADDR=newq9
SRVID=9  REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml  SRVGRP=group1 RQADDR=newq10
SRVID=10 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
```

```
tpccfml  SRVGRP=group1 RQADDR=newq11
SRVID=11 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml  SRVGRP=group1 RQADDR=newq12
SRVID=12 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml  SRVGRP=group1 RQADDR=newq13
SRVID=13 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml  SRVGRP=group1 RQADDR=newq14
SRVID=14 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfmlp SRVGRP=group1 RQADDR=payq1
SRVID=15 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq2
SRVID=16 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq3
SRVID=17 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq4
SRVID=18 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq5
SRVID=19 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq6
SRVID=20 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq7
SRVID=21 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=ordq1
SRVID=22 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfmlp SRVGRP=group1 RQADDR=ordq2
SRVID=23 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfmlp SRVGRP=group1 RQADDR=delq1
SRVID=24 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmlp SRVGRP=group1 RQADDR=delq2
SRVID=25 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmlp SRVGRP=group1 RQADDR=delq3
SRVID=26 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmls SRVGRP=group1 RQADDR=stoq1
SRVID=27 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfmls SRVGRP=group1 RQADDR=stoq2
SRVID=28 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfmls SRVGRP=group1 RQADDR=stoq3
SRVID=29 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"

*SERVICES
"NEWORDER" TRANTIME=0
  SRVGRP=group1
"PAYMENT" TRANTIME=0
  SRVGRP=group1
"ORDERSTATUS" TRANTIME=0
  SRVGRP=group1
"DELIVERY" TRANTIME=0
  SRVGRP=group1
"STOCKLVL" TRANTIME=0
  SRVGRP=group1
```

*ROUTING

===== client 15 =====
REGEDIT4

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters]
"BandwidthLevel"=dword:ffffff
"ListenBackLog"=dword:00000019
"MemoryCacheSize"=dword:00000000
"ObjectCacheTTL"=dword:ffffff
"PoolThreadLimit"=dword:0000020d
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters\Filter]
"FilterType"=dword:00000000
"NumGrantSites"=dword:00000000
"NumDenySites"=dword:00000000
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters\MimeMap]
"text/html,html,h"=""
"image/gif,gif,g"=""
"image/jpeg,jpg,,j"=""
"text/plain,txt,0"=""
"text/html,html,h"=""
"image/jpeg,jpeg,,j"=""
"image/jpeg,jpe,,j"=""
"image/bmp,bmp,,b"=""
"application/octet-stream,*,,5"=""
"application/pdf,pdf,,5"=""
"application/octet-stream,bin,,5"=""
"application/oda,oda,,5"=""
"application/zip,zip,,9"=""
"application/rtf,rtf,,5"=""
"application/postscript,ps,,5"=""
"application/postscript,ai,,5"=""
"application/postscript,eps,,5"=""
"application/mac-binhex40,hqx,,4"=""
"application/msword,doc,,5"=""
"application/msword,dot,,5"=""
"application/winhlp,hlp,,5"=""
"video/mpeg,mpeg,,m"=""
"video/mpeg,mpg,,m"=""
"video/mpeg,mpe,,m"=""
"video/x-msvideo,avi,<,,5"=""
"video/quicktime,qt,,5"=""
"video/quicktime,mov,,5"=""
"video/x-sgi-movie,movie,<,,5"=""
"x-world/x-vmrl,wrl,,5"=""
"x-world/x-vmrl,xaf,,5"=""
"x-world/x-vmrl,xof,,5"=""
"x-world/x-vmrl,flr,,5"=""
"x-world/x-vmrl,wrz,,5"=""
"application/x-director,dcr,,5"=""
"application/x-director,dir,,5"=""
"application/x-director,dxr,,5"=""
"image/cis-cod,cod,,5"=""
"image/x-cmx,cmx,,5"=""
"application/envoy,evy,,5"=""
"application/x-msaccess,mdb,,5"=""
"application/x-mscardfile,crd,,5"=""
"application/x-msclip,clip,,5"=""
"application/octet-stream,exe,,5"=""
"application/x-msexcel,xla,,5"=""
"application/x-msexcel,xlc,,5"=""
"application/x-msexcel,xlm,,5"=""
```

```

"application/x-msexcel,xls,,5"=""
"application/x-msexcel,xlt,,5"=""
"application/x-msexcel,xlw,,5"=""
"application/x-msmediaview,m13,,5"=""
"application/x-msmediaview,m14,,5"=""
"application/x-msmoney,mny,,5"=""
"application/x-mspowerpoint,ppt,,5"=""
"application/x-msproject,mpp,,5"=""
"application/x-mspublisher,pub,,5"=""
"application/x-msterminal,tm,,5"=""
"application/x-msworks,wks,,5"=""
"application/x-mswrite,wri,,5"=""
"application/x-msmetafile,wmf,,5"=""
"application/x-csh,csh,,5"=""
"application/x-dvi,dvi,,5"=""
"application/x-hdf,hdf,,5"=""
"application/x-latex,latex,,5"=""
"application/x-netcdf,nc,,5"=""
"application/x-netcdf,cdf,,5"=""
"application/x-sh,sh,,5"=""
"application/x-tcl,tcl,,5"=""
"application/x-tex,tex,,5"=""
"application/x-texinfo,texinfo,,5"=""
"application/x-texinfo,texi,,5"=""
"application/x-troff,t,,5"=""
"application/x-troff,tr,,5"=""
"application/x-troff,roff,,5"=""
"application/x-troff-man,man,,5"=""
"application/x-troff-me,me,,5"=""
"application/x-troff-ms,ms,,5"=""
"application/x-wais-source,src,,7"=""
"application/x-bcpio,bcpio,,5"=""
"application/x-cpio,cpio,,5"=""
"application/x-gtar,gtar,,9"=""
"application/x-shar,shar,,5"=""
"application/x-sv4cpio,sv4cpio,,5"=""
"application/x-sv4crc,sv4crc,,5"=""
"application/x-tar,tar,,5"=""
"application/x-ustar,ustar,,5"=""
"audio/basic,au,,<"=""
"audio/basic,snd,,<"=""
"audio/x-aiff,aif,,<"=""
"audio/x-aiff,aiff,,<"=""
"audio/x-aiff,aifc,,<"=""
"audio/x-wav,wav,,<"=""
"audio/x-pn-realaudio,ram,,<"=""
"image/ief,ief,,:"=""
"image/tiff,tiff,,:"=""
"image/tiff,tif,,:"=""
"image/x-cmu-raster,ras,,:"=""
"image/x-portable-anymap,pnm,,:"=""
"image/x-portable-bitmap,pbm,,:"=""
"image/x-portable-graymap,pgm,,:"=""
"image/x-portable-pixmap,ppm,,:"=""
"image/x-rgb,rgb,,:"=""
"image/x-xbitmap,xbm,,:"=""
"image/x-xpixmap,xpm,,:"=""
"image/x-xwindowdump,xwd,,:"=""
"text/html,stm,,h"=""
"text/plain,bas,,0"=""
"text/plain,c,,0"=""
"text/plain,h,,0"=""
"text/richtext,rtx,,0"=""
"text/tab-separated-values,tsv,,0"=""
"text/x-setext,etx,,0"=""
"application/x-perfmon,pmc,,5"=""
"application/x-perfmon,pma,,5"=""
"application/x-perfmon,pmr,,5"=""
"application/x-perfmon,pml,,5"=""

```

```
"application/x-perfmon,pmw,,5"=""
```

REGEDIT4

```

[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment]
"TUXDIR"="C:\TUXEDO"
"NLSPATH"="C:\TUXEDO\locale\C"
"ULOGDIR"="C:\TUXEDO"
"ULOGPFX"="C:\ULOG"
"ULOGOUT"=dword:00000001
"TUXIPC_MSG_BYTES"=dword:00010000
"TUXIPC_MSG_HDRS"=dword:00001fc0
"TUXIPC_MSG_QUEUES"=dword:00000320
"TUXIPC_MSG_QUEUE_BYTES"=dword:000100
00
"TUXIPC_MSG_SEG_BYTES"=dword:00000040
"TUXIPC_MSG_SEGS"=dword:00007fff
"TUXIPC_PROC"=dword:00001770
"TUXIPC_SEM"=dword:00001770
"TUXIPC_SEM_IDS"=dword:00001770
"TUXIPC_SEM_UNDO"=dword:00001770
"TUXIPC_SHM_PROCS"=dword:000186a0
"TUXIPC_SHM_SEGS"=dword:0000c350

```

```

[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services]

```

```

[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services\305
0]

```

REGEDIT4

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC]
"Type"=dword:00000020
"Start"=dword:00000002
"ErrorControl"=dword:00000000
"ImagePath"=hex(2):43,3a,5c,57,49,4e,4e,54,5c,5
3,79,73,74,65,6d,33,32,5c,69,6e,\
65,74,73,72,76,5c,69,6e,65,74,69,6e,66,6f,2e,65,
78,65,00
"DisplayName"="World Wide Web Publishing
Service"
"DependOnService"=hex(7):52,50,43,53,53,00,4e,
54,4c,4d,53,53,50,00,00
"DependOnGroup"=hex(7):00
"ObjectName"="LocalSystem"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters]
"MajorVersion"=dword:00000002
"MinorVersion"=dword:00000000
"AdminName"="Administrator"
"AdminEmail"="Admin@corp.com"
"MaxConnections"=dword:000186a0
"LogType"=dword:00000000
"LogFileDirectory"=hex(2):25,53,79,73,74,65,6d,5
2,6f,6f,74,25,5c,53,79,73,74,\
65,6d,33,32,5c,4c,6f,67,46,69,6c,65,73,00
"LogFileTruncateSize"=dword:01400000
"LogFilePeriod"=dword:00000001
"LogFileFormat"=dword:00000000
"LogSqlDataSource"="HTTPLOG"
"LogSqlTableName"="Internetlog"
"LogSqlUserName"="InternetAdmin"
"LogSqlPassword"="sqllog"
"Authorization"=dword:00000005

```

```

"AnonymousUserName"="IUSR_CL15"
"Default Load File"="Default.htm"
"Dir Browse Control"=dword:4000001e
"CheckForWAISDB"=dword:00000000
"CacheExtensions"=dword:00000001
"GlobalExpire"=dword:ffffff
"ServerSideIncludesEnabled"=dword:00000001
"ServerSideIncludesExtension"=".stm"
"DebugFlags"=dword:00000008
"ScriptTimeout"=dword:00000384
"ConnectionTimeout"=dword:00000384
"InstallPath"="C:\WINNT\System32\inetnsrv"
"SecurePort"=dword:000001bb
"Filter
DLLs"="C:\WINNT\System32\inetnsrv\lsspifilt.dll"
"AccessDeniedMessage"="エラー : アクセスは拒否さ
れました。"
"NTAuthenticationProviders"="NTLM"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Script Map]
".idc"="C:\WINNT\System32\inetnsrv\httpodbc.dll"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Virtual Roots]
"/,"="C:\inetPub\wwwroot,,1"
"/tpc,"="C:\client\tpap\Release,,5"
"/Scripts,"="C:\inetPub\scripts,,4"
"/iisadmin,"="C:\WINNT\System32\inetnsrv\iisad
min,,1"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Performance]
"Library"="w3ctrs.DLL"
"Open"="OpenW3PerformanceData"
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"Last Counter"=dword:00000790
"Last Help"=dword:00000791
"First Counter"=dword:00000758
"First Help"=dword:00000759

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Security]
"Security"=hex:01,00,14,80,c0,00,00,00,cc,00,00,
00,14,00,00,00,34,00,00,00,02,\
00,20,00,01,00,00,00,02,80,18,00,ff,01,0f,00,01,
01,00,00,00,00,01,00,00,\
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,00,0
0,18,00,8d,01,02,00,01,01,00,\
00,00,00,00,01,00,00,00,00,00,00,00,00,00,00,1
c,00,fd,01,02,00,01,02,00,0,\
00,00,00,05,20,00,00,00,23,02,00,00,00,00,00,0
0,00,00,1c,00,ff,01,0f,00,01,\
02,00,00,00,00,05,20,00,00,00,20,02,00,00,0
0,00,00,00,00,1c,00,ff,01,\
0f,00,01,02,00,00,00,00,05,20,00,00,00,25,0
2,00,00,00,00,00,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,05,12,00,0
0,00,25,02,00,00,01,01,00,00,\
00,00,00,05,12,00,00,00,01,01,00,00,00,00,0
5,12,00,00,00

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\W3SAMP]

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Enum]

```

```
"0"="Root\LEGACY_W3SVC\0000"
"Count"=dword:00000001
"NextInstance"=dword:00000001
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Fujitsu\T
PC-C ISAPI Application]
"Term_Base"=dword:103c5
"NumWarehouses"=dword:2c88
"MaxUsers"=dword:1bd50
"MaxTerm of Client"=dword:128e
"CONTROL_Flag"=dword:00000000
```

```
#
# ubbconfig : TUXEDO configuration file
#
```

```
*RESOURCES
IPCKEY 80952
MASTER SITE1
#UID 1
#GID 1
PERM 0660
MAXACCESSERS 800
MAXSERVERS 200
MAXSERVICES 1000
MODEL SHM
LDBAL Y
```

```
*MACHINES
CL15 LMID=SITE1
APPPDIR="c:\sv-aplora"
TUXCONFIG="c:\client\tuxconfig"
TUXDIR="c:\tuxedo"
ULOGPFEX="c:\tuxlog\test"
```

```
*GROUPS
group1 LMID=SITE1 GRPNO=1
```

```
*SERVERS
DEFAULT: RESTART=Y MAXGEN=5
RPERM=0660
tpccfml SRVGRP=group1 RQADDR=newq1
SRVID=1 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq2
SRVID=2 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq3
SRVID=3 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq4
SRVID=4 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq5
SRVID=5 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq6
SRVID=6 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq7
SRVID=7 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq8
SRVID=8 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
```

```
tpccfml SRVGRP=group1 RQADDR=newq9
SRVID=9 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq10
SRVID=10 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq11
SRVID=11 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq12
SRVID=12 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq13
SRVID=13 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq14
SRVID=14 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfmlp SRVGRP=group1 RQADDR=payq1
SRVID=15 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq2
SRVID=16 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq3
SRVID=17 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq4
SRVID=18 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq5
SRVID=19 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq6
SRVID=20 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq7
SRVID=21 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=ordq1
SRVID=22 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfmlp SRVGRP=group1 RQADDR=ordq2
SRVID=23 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfmlp SRVGRP=group1 RQADDR=delq1
SRVID=24 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmlp SRVGRP=group1 RQADDR=delq2
SRVID=25 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmlp SRVGRP=group1 RQADDR=delq3
SRVID=26 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmls SRVGRP=group1 RQADDR=stoq1
SRVID=27 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfmls SRVGRP=group1 RQADDR=stoq2
SRVID=28 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfmls SRVGRP=group1 RQADDR=stoq3
SRVID=29 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"

*SERVICES
"NEWORDER" TRANTIME=0
SRVGRP=group1
"PAYMENT" TRANTIME=0
SRVGRP=group1
```

```
"ORDERSTATUS" TRANTIME=0
SRVGRP=group1
"DELIVERY" TRANTIME=0
SRVGRP=group1
"STOCKLVL" TRANTIME=0
SRVGRP=group1
```

*ROUTING

```
===== client 16 =====
REGEDIT4
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters]
"BandwidthLevel"=dword:ffffff
"ListenBackLog"=dword:00000019
"MemoryCacheSize"=dword:00000000
"ObjectCacheTTL"=dword:ffffff
"PoolThreadLimit"=dword:0000020d
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters\Filter]
"FilterType"=dword:00000000
"NumGrantSites"=dword:00000000
"NumDenySites"=dword:00000000
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters\MimeMap]
"text/html,html,h"=""
"image/gif,gif,g"=""
"image/jpeg,jpg,,"=""
"text/plain,txt,0"=""
"text/html,html,h"=""
"image/jpeg,jpeg,,"=""
"image/jpeg,jpe,,"=""
"image/bmp,bmp,,"=""
"application/octet-stream,*,,5"=""
"application/pdf,pdf,,5"=""
"application/octet-stream,bin,,5"=""
"application/oda,oda,,5"=""
"application/zip,zip,,9"=""
"application/rft,rft,,5"=""
"application/postscript,ps,,5"=""
"application/postscript,ai,,5"=""
"application/postscript,eps,,5"=""
"application/mac-binhex40,hqx,,4"=""
"application/msword,doc,,5"=""
"application/msword,dot,,5"=""
"application/winhlp,hlp,,5"=""
"video/mpeg,mpeg,,"=""
"video/mpeg,mpg,,"=""
"video/mpeg,mpe,,"=""
"video/x-msvideo,avi,<,"=""
"video/quicktime,qt,,"=""
"video/quicktime,mov,,"=""
"video/x-sgi-movie,movie,<,"=""
"x-world/x-vrml,wrl,,5"=""
"x-world/x-vrml,xaf,,5"=""
"x-world/x-vrml,xof,,5"=""
"x-world/x-vrml,flr,,5"=""
"x-world/x-vrml,wrz,,5"=""
"application/x-director,dcr,,5"=""
"application/x-director,dir,,5"=""
"application/x-director,dxr,,5"=""
"image/cis-cod,cod,,5"=""
"image/x-cmx,cmx,,5"=""
"application/envoy,evy,,5"=""
"application/x-msaccess,mdb,,5"=""
```

```

"application/x-mscardfile,crd,,5"=""
"application/x-msclip,clip,,5"=""
"application/octet-stream,exe,,5"=""
"application/x-msexcel,xla,,5"=""
"application/x-msexcel,xlc,,5"=""
"application/x-msexcel,xlm,,5"=""
"application/x-msexcel,xls,,5"=""
"application/x-msexcel,xlt,,5"=""
"application/x-msexcel,xlw,,5"=""
"application/x-msmediaview,m13,,5"=""
"application/x-msmediaview,m14,,5"=""
"application/x-msmoney,mny,,5"=""
"application/x-mspowerpoint,ppt,,5"=""
"application/x-msproject,mpp,,5"=""
"application/x-mspublisher,pub,,5"=""
"application/x-msterminal,frm,,5"=""
"application/x-msworks,wks,,5"=""
"application/x-mswrite,wri,,5"=""
"application/x-msmetafile,wmf,,5"=""
"application/x-csh,csh,,5"=""
"application/x-dvi,dvi,,5"=""
"application/x-hdf,hdf,,5"=""
"application/x-latex,latex,,5"=""
"application/x-netcdf,nc,,5"=""
"application/x-netcdf,cdf,,5"=""
"application/x-sh,sh,,5"=""
"application/x-tcl,tcl,,5"=""
"application/x-tex,tex,,5"=""
"application/x-texinfo,texinfo,,5"=""
"application/x-texinfo,texi,,5"=""
"application/x-troff,t,,5"=""
"application/x-troff,tr,,5"=""
"application/x-troff,roff,,5"=""
"application/x-troff-man,man,,5"=""
"application/x-troff-me,me,,5"=""
"application/x-troff-ms,ms,,5"=""
"application/x-wais-source,src,,7"=""
"application/x-bcpio,bcpio,,5"=""
"application/x-cpio,cpio,,5"=""
"application/x-gtar,gtar,,9"=""
"application/x-shar,shar,,5"=""
"application/x-sv4cpio,sv4cpio,,5"=""
"application/x-sv4crc,sv4crc,,5"=""
"application/x-tar,tar,,5"=""
"application/x-ustar,ustar,,5"=""
"audio/basic,au,,<"=""
"audio/basic,snd,,<"=""
"audio/x-aiff,aif,,<"=""
"audio/x-aiff,aiff,,<"=""
"audio/x-aiff,aifc,,<"=""
"audio/x-wav,wav,,<"=""
"audio/x-pn-realaudio,ram,,<"=""
"image/ief,ief,,:"=""
"image/tiff,tiff,,:"=""
"image/tiff,tif,,:"=""
"image/x-cmu-raster,ras,,:"=""
"image/x-portable-anymap,pnm,,:"=""
"image/x-portable-bitmap,pbm,,:"=""
"image/x-portable-graymap,pgm,,:"=""
"image/x-portable-pixmap,ppm,,:"=""
"image/x-rgb,rgb,,:"=""
"image/x-xbitmap,xbm,,:"=""
"image/x-xpixmap,xpm,,:"=""
"image/x-xwindowdump,xwd,,:"=""
"text/html,stm,,h"=""
"text/plain,bas,,0"=""
"text/plain,c,,0"=""
"text/plain,h,,0"=""
"text/richtext,rtx,,0"=""

```

```

"text/tab-separated-values,tsv,,0"=""
"text/x-setext,etx,,0"=""
"application/x-perfmon,pmc,,5"=""
"application/x-perfmon,pma,,5"=""
"application/x-perfmon,pmr,,5"=""
"application/x-perfmon,pml,,5"=""
"application/x-perfmon,pmw,,5"=""

```

REGEDIT4

```

[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment]
"TUXDIR"="C:\TUXEDO"
"NLSPATH"="C:\TUXEDO\locale\C"
"ULOGDIR"="C:\TUXEDO"
"ULOGPFX"="C:\ULOG"
"ULOGOUT"=dword:00000001
"TUXIPC_MSG_BYTES"=dword:00010000
"TUXIPC_MSG_HDRS"=dword:00001fc0
"TUXIPC_MSG_QUEUEUS"=dword:00000320
"TUXIPC_MSG_QUEUE_BYTES"=dword:000100
00
"TUXIPC_MSG_SEG_BYTES"=dword:00000040
"TUXIPC_MSG_SEGS"=dword:00007fff
"TUXIPC_PROC"=dword:00001770
"TUXIPC_SEM"=dword:00001770
"TUXIPC_SEM_IDS"=dword:00001770
"TUXIPC_SEM_UNDO"=dword:00001770
"TUXIPC_SHM_PROCS"=dword:000186a0
"TUXIPC_SHM_SEGS"=dword:0000c350

```

```

[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services]

```

```

[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services\305
0]

```

REGEDIT4

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC]
"Type"=dword:00000020
"Start"=dword:00000002
"ErrorControl"=dword:00000000
"ImagePath"=hex(2):43,3a,5c,57,49,4e,4e,54,5c,5
3,79,73,74,65,6d,33,32,5c,69,6e,\
65,74,73,72,76,5c,69,6e,65,74,69,6e,66,6f,2e,65,
78,65,00
"DisplayName"="World Wide Web Publishing
Service"
"DependOnService"=hex(7):52,50,43,53,00,00,4e,
54,4c,4d,53,53,00,00,00
"DependOnGroup"=hex(7):00
"ObjectName"="LocalSystem"

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters]
"MajorVersion"=dword:00000002
"MinorVersion"=dword:00000000
"AdminName"="Administrator"
"AdminEmail"="Admin@corp.com"
"MaxConnections"=dword:000186a0
"LogType"=dword:00000000
"LogFileDirectory"=hex(2):25,53,79,73,74,65,6d,5
2,6f,6f,74,25,5c,53,79,73,74,\
65,6d,33,32,5c,4c,6f,67,46,69,6c,65,73,00
"LogFileTruncateSize"=dword:01400000
"LogFilePeriod"=dword:00000001

```

```

"LogFileFormat"=dword:00000000
"LogSqlDataSource"="HTTPLOG"
"LogSqlTableName"="Internetlog"
"LogSqlUserName"="InternetAdmin"
"LogSqlPassword"="sqllog"
"Authorization"=dword:00000005
"AnonymousUserName"="IUSR_CL16"
"Default Load File"="Default.htm"
"Dir Browse Control"=dword:4000001e
"CheckForWAISDB"=dword:00000000
"CacheExtensions"=dword:00000001
"GlobalExpire"=dword:ffffff
"ServerSideIncludesEnabled"=dword:00000001
"ServerSideIncludesExtension"=".stm"
"DebugFlags"=dword:00000008
"ScriptTimeout"=dword:00000384
"ConnectionTimeout"=dword:00000384
"InstallPath"="C:\WINNT\System32\inetnsrv"
"SecurePort"=dword:000001bb
"Filter
DLLs"="C:\WINNT\System32\inetnsrv\lsspifilt.dll"
"AccessDeniedMessage"="エラー : アクセスは拒否さ
れました。"
"NTAuthenticationProviders"="NTLM"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Script Map]
".idc"="C:\WINNT\System32\inetnsrv\httpodbc.dll"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Virtual Roots]
"/,"="C:\inetpub\wwwroot,1"
"/tpc,"="C:\client\tpap\Release,,5"
"/Scripts,"="C:\inetpub\scripts,4"
"/iisadmin,"="C:\WINNT\System32\inetnsrv\iisad
min,,1"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Performance]
"Library"="w3ctrs.DLL"
"Open"="OpenW3PerformanceData"
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"Last Counter"=dword:00000790
"Last Help"=dword:00000791
"First Counter"=dword:00000758
"First Help"=dword:00000759

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Security]
"Security"=hex:01,00,14,80,c0,00,00,00,cc,00,00,
00,14,00,00,00,34,00,00,00,02,\
00,20,00,01,00,00,00,02,80,18,00,ff,01,0f,00,01,
01,00,00,00,00,00,01,00,00,\
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,00,0
0,18,00,8d,01,02,00,01,01,00,\
00,00,00,00,01,00,00,00,80,72,14,00,00,00,1
c,00,fd,01,02,00,01,02,00,00,\
00,00,00,05,20,00,00,00,23,02,00,00,00,00,00,0
0,00,00,1c,00,ff,01,0f,00,01,\
02,00,00,00,00,00,05,20,00,00,00,20,02,00,00,0
0,00,00,00,00,00,1c,00,ff,01,\
0f,00,01,02,00,00,00,00,05,20,00,00,00,25,0
2,00,00,00,00,00,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,00,05,12,00,0
0,00,25,02,00,00,01,01,00,00,\
00,00,00,05,12,00,00,00,01,01,00,00,00,00,00,0
5,12,00,00,00

```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\W3SAMP]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Enum]
```

```
"0"="Root\LEGACY_W3SVC\0000"
"Count"=dword:00000001
"NextInstance"=dword:00000001
```

```
REGEDIT4
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Fujitsu\TCP-C ISAPI Application]
```

```
"Term_Base"=dword:11653
"NumWarehouses"=dword:2c88
"MaxUsers"=dword:1bd50
"MaxTerm of Client"=dword:128e
"CONTROL_Flag"=dword:00000000
```

```
#
# ubbconfig : TUXEDO configuration file
#
```

```
*RESOURCES
```

```
IPCKEY      80952
MASTER     SITE1
#UID       1
#GID       1
PERM       0660
MAXACCESSERS 800
MAXSERVERS 200
MAXSERVICES 1000
MODEL      SHM
LDBAL      Y
```

```
*MACHINES
```

```
CL16 LMID=SITE1
  APPDIR="c:\sv-aplora"
  TUXCONFIG="c:\client\tuxconfig"
  TUXDIR="c:\tuxedo"
  ULOGPFX="c:\tuxlog\test"
```

```
*GROUPS
```

```
group1 LMID=SITE1 GRPNO=1
```

```
*SERVERS
```

```
DEFAULT: RESTART=Y MAXGEN=5
RQPERM=0660
tpccfml SRVGRP=group1 RQADDR=newq1
SRVID=1 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq2
SRVID=2 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq3
SRVID=3 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq4
SRVID=4 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq5
SRVID=5 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq6
SRVID=6 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
```

```
tpccfml SRVGRP=group1 RQADDR=newq7
SRVID=7 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq8
SRVID=8 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq9
SRVID=9 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq10
SRVID=10 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq11
SRVID=11 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq12
SRVID=12 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq13
SRVID=13 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq14
SRVID=14 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfmlp SRVGRP=group1 RQADDR=payq1
SRVID=15 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq2
SRVID=16 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq3
SRVID=17 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq4
SRVID=18 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq5
SRVID=19 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq6
SRVID=20 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq7
SRVID=21 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlo SRVGRP=group1 RQADDR=ordq1
SRVID=22 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfmlo SRVGRP=group1 RQADDR=ordq2
SRVID=23 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfmld SRVGRP=group1 RQADDR=delq1
SRVID=24 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmld SRVGRP=group1 RQADDR=delq2
SRVID=25 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmld SRVGRP=group1 RQADDR=delq3
SRVID=26 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmlls SRVGRP=group1 RQADDR=stoq1
SRVID=27 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfmlls SRVGRP=group1 RQADDR=stoq2
SRVID=28 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfmlls SRVGRP=group1 RQADDR=stoq3
SRVID=29 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
```

```
*SERVICES
```

```
"NEWORDER" TRANTIME=0
SRVGRP=group1
"PAYMENT" TRANTIME=0
SRVGRP=group1
"ORDERSTATUS" TRANTIME=0
SRVGRP=group1
"DELIVERY" TRANTIME=0
SRVGRP=group1
"STOCKLVL" TRANTIME=0
SRVGRP=group1
```

```
*ROUTING
```

```
==== client 17 ====
REGEDIT4
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\inetInfo\Parameters]
```

```
"BandwidthLevel"=dword:ffffff
"ListenBackLog"=dword:00000019
"MemoryCacheSize"=dword:00000000
"ObjectCacheTTL"=dword:ffffff
"PoolThreadLimit"=dword:0000020d
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\inetInfo\Parameters\Filter]
```

```
"FilterType"=dword:00000000
"NumGrantSites"=dword:00000000
"NumDenySites"=dword:00000000
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\inetInfo\Parameters\MimeMap]
```

```
"text/html,html,h"=""
"image/gif,gif,g"=""
"image/jpeg,jpg,,"=""
"text/plain,txt,0"=""
"text/html,html,h"=""
"image/jpeg,jpeg,,"=""
"image/jpeg,jpe,,"=""
"image/bmp,bmp,,"=""
"application/octet-stream,*,,5"=""
"application/pdf,pdf,,5"=""
"application/octet-stream,bin,,5"=""
"application/oda,oda,,5"=""
"application/zip,zip,,9"=""
"application/rtf,rtf,,5"=""
"application/postscript,ps,,5"=""
"application/postscript,ai,,5"=""
"application/postscript,eps,,5"=""
"application/mac-binhex40,hqx,,4"=""
"application/msword,doc,,5"=""
"application/msword,dot,,5"=""
"application/winhlp,hlp,,5"=""
"video/mpeg,mpeg,,"=""
"video/mpeg,mpg,,"=""
"video/mpeg,mpe,,"=""
"video/x-msvideo,avi,<=""=""
"video/quicktime,qt,,"=""
"video/quicktime,mov,,"=""
"video/x-sgi-movie,movie,<=""=""
"x-world/x-vrml,wrl,,5"=""
"x-world/x-vrml,xaf,,5"=""
"x-world/x-vrml,xof,,5"=""
"x-world/x-vrml,flr,,5"=""
"x-world/x-vrml,wrz,,5"=""
"application/x-director,dcr,,5"=""
```

```

"application/x-director,dir,,5"=""
"application/x-director,dxr,,5"=""
"image/cis-cod,cod,,5"=""
"image/x-cmx,cmx,,5"=""
"application/envoy,env,,5"=""
"application/x-msaccess,mdb,,5"=""
"application/x-mscardfile,crd,,5"=""
"application/x-msclip,clip,,5"=""
"application/octet-stream,exe,,5"=""
"application/x-msexcel,xla,,5"=""
"application/x-msexcel,xlc,,5"=""
"application/x-msexcel,xlm,,5"=""
"application/x-msexcel,xls,,5"=""
"application/x-msexcel,xlt,,5"=""
"application/x-msexcel,xlw,,5"=""
"application/x-msmediaview,m13,,5"=""
"application/x-msmediaview,m14,,5"=""
"application/x-msmoney,mny,,5"=""
"application/x-mspowerpoint,ppt,,5"=""
"application/x-msproject,mpp,,5"=""
"application/x-mspublisher,pub,,5"=""
"application/x-sterminal,term,,5"=""
"application/x-msworks,wks,,5"=""
"application/x-mswrite,wri,,5"=""
"application/x-msmetafile,wmf,,5"=""
"application/x-csh,csh,,5"=""
"application/x-dvi,dvi,,5"=""
"application/x-hdf,hdf,,5"=""
"application/x-latex,latex,,5"=""
"application/x-netcdf,nc,,5"=""
"application/x-netcdf,cdf,,5"=""
"application/x-sh,sh,,5"=""
"application/x-tcl,tcl,,5"=""
"application/x-tex,tex,,5"=""
"application/x-texinfo,texinfo,,5"=""
"application/x-texinfo,texi,,5"=""
"application/x-troff,t,,5"=""
"application/x-troff,tr,,5"=""
"application/x-troff,roff,,5"=""
"application/x-troff-man,man,,5"=""
"application/x-troff-me,me,,5"=""
"application/x-troff-ms,ms,,5"=""
"application/x-wais-source,src,,7"=""
"application/x-bcpio,bcpio,,5"=""
"application/x-cpio,cpio,,5"=""
"application/x-gtar,gtar,,9"=""
"application/x-shar,shar,,5"=""
"application/x-sv4cpio,sv4cpio,,5"=""
"application/x-sv4crc,sv4crc,,5"=""
"application/x-tar,tar,,5"=""
"application/x-ustar,ustar,,5"=""
"audio/basic,au,,<"=""
"audio/basic,snd,,<"=""
"audio/x-aiff,aiff,,<"=""
"audio/x-aiff,aifc,,<"=""
"audio/x-wav,wav,,<"=""
"audio/x-pn-realaudio,ram,,<"=""
"image/ief,ief,,:"=""
"image/tiff,tiff,,:"=""
"image/tif,tif,,:"=""
"image/x-cmu-raster,ras,,:"=""
"image/x-portable-anymap,pnm,,:"=""
"image/x-portable-bitmap,pbm,,:"=""
"image/x-portable-graymap,pgm,,:"=""
"image/x-portable-pixmap,ppm,,:"=""
"image/x-rgb,rgb,,:"=""
"image/x-xbitmap,xbm,,:"=""
"image/x-xpixmap,xpm,,:"=""

```

```

"image/x-xwindowdump,xwd,,:"=""
"text/html,stm,h"=""
"text/plain,bas,,0"=""
"text/plain,c,,0"=""
"text/plain,h,,0"=""
"text/richtext,rtx,,0"=""
"text/tab-separated-values,tsv,,0"=""
"text/x-setext,etx,,0"=""
"application/x-perfmon,pmc,,5"=""
"application/x-perfmon,pma,,5"=""
"application/x-perfmon,pmr,,5"=""
"application/x-perfmon,pml,,5"=""
"application/x-perfmon,pmw,,5"=""

```

REGEDIT4

```

[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment]
"TUXDIR"="C:\TUXEDO"
"NLSPATH"="C:\TUXEDO\locale\C"
"ULOGDIR"="C:\TUXEDO"
"ULOGPF"="C:\ULOG"
"ULOGOUT"=dword:00000001
"TUXIPC_MSG_BYTES"=dword:00010000
"TUXIPC_MSG_HDRS"=dword:00001fc0
"TUXIPC_MSG_QUEUE"=dword:00000320
"TUXIPC_MSG_QUEUE_BYTES"=dword:000100
00
"TUXIPC_MSG_SEG_BYTES"=dword:00000040
"TUXIPC_MSG_SEGS"=dword:00007fff
"TUXIPC_PROC"=dword:00001770
"TUXIPC_SEM"=dword:00001770
"TUXIPC_SEM_IDS"=dword:00001770
"TUXIPC_SEM_UNDO"=dword:00001770
"TUXIPC_SHM_PROCS"=dword:000186a0
"TUXIPC_SHM_SEGS"=dword:0000c350

```

```

[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services]

```

```

[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services\305
0]

```

REGEDIT4

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC]
"Type"=dword:00000020
"Start"=dword:00000002
"ErrorControl"=dword:00000000
"ImagePath"=hex(2):43,3a,5c,57,49,4e,4e,54,5c,5
3,79,73,74,65,6d,33,32,5c,69,6e,\
65,74,73,72,76,5c,69,6e,65,74,69,6e,66,6f,2e,65,
78,65,00
"DisplayName"="World Wide Web Publishing
Service"
"DependOnService"=hex(7):52,50,43,53,53,00,4e,
54,4c,4d,53,53,50,00,00
"DependOnGroup"=hex(7):00
"ObjectName"="LocalSystem"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters]
"MajorVersion"=dword:00000002
"MinorVersion"=dword:00000000
"AdminName"="Administrator"
"AdminEmail"="Admin@corp.com"
"MaxConnections"=dword:000186a0

```

```

"LogType"=dword:00000000
"LogFileDirectory"=hex(2):25,53,79,73,74,65,6d,5
2,6f,6f,74,25,5c,53,79,73,74,\
65,6d,33,32,5c,4c,6f,67,46,69,6c,65,73,00
"LogFileTruncateSize"=dword:01400000
"LogFilePeriod"=dword:00000001
"LogFileFormat"=dword:00000000
"LogSqlDataSource"="HTTPLOG"
"LogSqlTableName"="Internetlog"
"LogSqlUserName"="InternetAdmin"
"LogSqlPassword"="sqllog"
"Authorization"=dword:00000005
"AnonymousUserName"="IUSR_CL17"
"Default Load File"="Default.htm"
"Dir Browse Control"=dword:4000001e
"CheckForWAISDB"=dword:00000000
"CacheExtensions"=dword:00000001
"GlobalExpire"=dword:ffffff
"ServerSideIncludesEnabled"=dword:00000001
"ServerSideIncludesExtension"=".stm"
"DebugFlags"=dword:00000008
"ScriptTimeout"=dword:00000384
"ConnectionTimeout"=dword:00000384
"InstallPath"="C:\WINNT\System32\inetnsrv"
"SecurePort"=dword:000001bb
"Filter
DLLs"="C:\WINNT\System32\inetnsrv\sspifilt.dll"
"AccessDeniedMessage"="エラー: 予知は拒否さ
れました。"
"NTAuthenticationProviders"="NTLM"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Script Map]
".idc"="C:\WINNT\System32\inetnsrv\httpodbc.dll"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Virtual Roots]
"/"="C:\inetPub\wwwroot,1"
"/tpc"="C:\client\tpap\Release,,5"
"/Scripts"="C:\inetPub\Scripts,,4"
"/iisadmin"="C:\WINNT\System32\inetnsrv\iisad
min,,1"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Performance]
"Library"="w3ctrs.DLL"
"Open"="OpenW3PerformanceData"
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"Last Counter"=dword:00000790
"Last Help"=dword:00000791
"First Counter"=dword:00000758
"First Help"=dword:00000759

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Security]
"Security"=hex:01,00,14,80,c0,00,00,cc,00,00,
00,14,00,00,00,34,00,00,02,\
00,20,00,01,00,00,02,80,18,00,ff,01,0f,00,01,
01,00,00,00,00,01,00,00,\
00,00,00,02,00,00,02,00,8c,00,05,00,00,00,00,0
0,18,00,8d,01,02,00,01,01,00,\
00,00,00,00,01,00,00,00,50,db,14,00,00,00,1
c,00,fd,01,02,00,01,02,00,00,\
00,00,00,05,20,00,00,00,23,02,00,00,00,00,00,0
0,00,00,1c,00,ff,01,0f,00,01,\
02,00,00,00,00,05,20,00,00,00,20,02,00,00,0
0,00,00,00,00,00,1c,00,ff,01,\

```

```
0f,00,01,02,00,00,00,00,05,20,00,00,00,25,0
2,00,00,00,00,00,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,05,12,00,0
0,00,25,02,00,00,01,01,00,00,\
00,00,00,05,12,00,00,00,01,01,00,00,00,00,0
5,12,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\W3SAMP]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Enum]
```

```
"0"="Root\LEGACY_W3SVC\0000"
"Count"=dword:00000001
"NextInstance"=dword:00000001
```

```
REGEDIT4
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Fujitsu\T
PC-C ISAPI Application]
```

```
"Term_Base"=dword:128e1
"NumWarehouses"=dword:2c88
"MaxUsers"=dword:1bd50
"MaxTerm of Client"=dword:128e
"CONTROL_Flag"=dword:00000000
```

```
#
# ubbconfig : TUXEDO configuration file
#
```

```
*RESOURCES
```

```
IPCKEY 80952
MASTER SITE1
#UID 1
#GID 1
PERM 0660
MAXACCESSERS 800
MAXSERVERS 200
MAXSERVICES 1000
MODEL SHM
LDBAL Y
```

```
*MACHINES
```

```
CL17 LMID=SITE1
APPPDIR="c:\sv-aplora"
TUXCONFIG="c:\client\tuxconfig"
TUXDIR="c:\tuxedo"
ULOGPFX="c:\tuxlog\test"
```

```
*GROUPS
```

```
group1 LMID=SITE1 GRPNO=1
```

```
*SERVERS
```

```
DEFAULT: RESTART=Y MAXGEN=5
RQPERM=0660
tpccfml SRVGRP=group1 RQADDR=newq1
SRVID=1 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq2
SRVID=2 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq3
SRVID=3 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq4
SRVID=4 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
```

```
tpccfml SRVGRP=group1 RQADDR=newq5
SRVID=5 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq6
SRVID=6 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq7
SRVID=7 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq8
SRVID=8 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq9
SRVID=9 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq10
SRVID=10 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq11
SRVID=11 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq12
SRVID=12 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq13
SRVID=13 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq14
SRVID=14 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfmlp SRVGRP=group1 RQADDR=payq1
SRVID=15 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq2
SRVID=16 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq3
SRVID=17 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq4
SRVID=18 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq5
SRVID=19 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq6
SRVID=20 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq7
SRVID=21 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlo SRVGRP=group1 RQADDR=ordq1
SRVID=22 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfmlo SRVGRP=group1 RQADDR=ordq2
SRVID=23 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfmld SRVGRP=group1 RQADDR=delq1
SRVID=24 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmld SRVGRP=group1 RQADDR=delq2
SRVID=25 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmld SRVGRP=group1 RQADDR=delq3
SRVID=26 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmlls SRVGRP=group1 RQADDR=stoq1
SRVID=27 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
```

```
tpccfmlls SRVGRP=group1 RQADDR=stoq2
SRVID=28 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfmlls SRVGRP=group1 RQADDR=stoq3
SRVID=29 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
```

```
*SERVICES
```

```
"NEWORDER" TRANTIME=0
SRVGRP=group1
"PAYMENT" TRANTIME=0
SRVGRP=group1
"ORDERSTATUS" TRANTIME=0
SRVGRP=group1
"DELIVERY" TRANTIME=0
SRVGRP=group1
"STOCKLVL" TRANTIME=0
SRVGRP=group1
```

```
*ROUTING
```

```
===== client 18 =====
REGEDIT4
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters]
```

```
"BandwidthLevel"=dword:ffffff
"ListenBackLog"=dword:00000019
"MemoryCacheSize"=dword:00000000
"ObjectCacheTTL"=dword:ffffff
"PoolThreadLimit"=dword:0000020d
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters\Filter]
```

```
"FilterType"=dword:00000000
"NumGrantSites"=dword:00000000
"NumDenySites"=dword:00000000
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters\MimeMap]
```

```
"text/html,htm,h"=""
"image/gif,gif,g"=""
"image/jpeg,jpg,,"=""
"text/plain,txt,0"=""
"text/html,html,h"=""
"image/jpeg,jpeg,,"=""
"image/jpeg,jpe,,"=""
"image/bmp,bmp,,"=""
"application/octet-stream,*,,5"=""
"application/pdf,pdf,,5"=""
"application/octet-stream,bin,,5"=""
"application/oda,oda,,5"=""
"application/zip,zip,,9"=""
"application/rft,rft,,5"=""
"application/postscript,ps,,5"=""
"application/postscript,ai,,5"=""
"application/postscript,eps,,5"=""
"application/mac-binhex40,hqx,,4"=""
"application/msword,doc,,5"=""
"application/msword,dot,,5"=""
"application/winhlp,hlp,,5"=""
"video/mpeg,mpeg,,"=""
"video/mpeg,mpg,,"=""
"video/mpeg,mpe,,"=""
"video/x-msvideo,avi,<,,="=""
"video/quicktime,qt,,"=""
"video/quicktime,mov,,"=""
"video/x-sgi-movie,movie,<,,="=""
```



```

"x-world/x-vrml,wrl,5"=""
"x-world/x-vrml,xaf,5"=""
"x-world/x-vrml,xof,5"=""
"x-world/x-vrml,flr,5"=""
"x-world/x-vrml,wrz,5"=""
"application/x-director,dcr,5"=""
"application/x-director,dir,5"=""
"application/x-director,dxr,5"=""
"image/cis-cod,cod,5"=""
"image/x-cmx,cmx,5"=""
"application/envoy,evy,5"=""
"application/x-msaccess,mdb,5"=""
"application/x-mscardfile,crd,5"=""
"application/x-msclip,cpl,5"=""
"application/octet-stream,exe,5"=""
"application/x-msexcel,xla,5"=""
"application/x-msexcel,xlc,5"=""
"application/x-msexcel,xlm,5"=""
"application/x-msexcel,xls,5"=""
"application/x-msexcel,xlt,5"=""
"application/x-msexcel,xlw,5"=""
"application/x-msmediaview,m13,5"=""
"application/x-msmediaview,m14,5"=""
"application/x-msmoney,mny,5"=""
"application/x-mspowerpoint,ppt,5"=""
"application/x-msproject,mpp,5"=""
"application/x-mspublisher,pub,5"=""
"application/x-msterminal,tm,5"=""
"application/x-msworks,wks,5"=""
"application/x-mswrite,wri,5"=""
"application/x-msmetafile,wmf,5"=""
"application/x-csh,csh,5"=""
"application/x-dvi,dvi,5"=""
"application/x-hdf,hdf,5"=""
"application/x-latex,latex,5"=""
"application/x-netcdf,nc,5"=""
"application/x-netcdf,cdf,5"=""
"application/x-sh,sh,5"=""
"application/x-tcl,tcl,5"=""
"application/x-tex,tex,5"=""
"application/x-texinfo,texinfo,5"=""
"application/x-texinfo,texi,5"=""
"application/x-troff,t,5"=""
"application/x-troff,tr,5"=""
"application/x-troff,roff,5"=""
"application/x-troff-man,man,5"=""
"application/x-troff-me,me,5"=""
"application/x-troff-ms,ms,5"=""
"application/x-wais-source,src,7"=""
"application/x-bcpio,bcpio,5"=""
"application/x-cpio,cpio,5"=""
"application/x-gtar,gtar,9"=""
"application/x-shar,shar,5"=""
"application/x-sv4cpio,sv4cpio,5"=""
"application/x-sv4crc,sv4crc,5"=""
"application/x-tar,tar,5"=""
"application/x-ustar,ustar,5"=""
"audio/basic,au,<"=""
"audio/basic,snd,<"=""
"audio/x-aiff,aif,<"=""
"audio/x-aiff,aiff,<"=""
"audio/x-aiff,aifc,<"=""
"audio/x-wav,wav,<"=""
"audio/x-pn-realaudio,ram,<"=""
"image/ief,ief,5"=""
"image/tiff,tiff,5"=""
"image/tiff,tif,5"=""
"image/x-cmu-raster,ras,5"=""
"image/x-portable-anymap,pnm,5"=""

```

```

"image/x-portable-bitmap,pbm,5"=""
"image/x-portable-graymap,pgm,5"=""
"image/x-portable-pixmap,ppm,5"=""
"image/x-rgb,rgb,5"=""
"image/x-xbitmap,xbm,5"=""
"image/x-xpixmap,xpm,5"=""
"image/x-xwindowdump,xwd,5"=""
"text/html,stm,h,5"=""
"text/plain,bas,0"=""
"text/plain,c,0"=""
"text/plain,h,0"=""
"text/richtext,rtx,0"=""
"text/tab-separated-values,tsv,0"=""
"text/x-setext,etx,0"=""
"application/x-perfmon,pmc,5"=""
"application/x-perfmon,pma,5"=""
"application/x-perfmon,pmr,5"=""
"application/x-perfmon,pml,5"=""
"application/x-perfmon,pmw,5"=""

```

REGEDIT4

```

[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment]
"TUXDIR"="C:\TUXEDO"
"NLSPATH"="C:\TUXEDO\locale\C"
"ULOGDIR"="C:\TUXEDO"
"ULOGPF"="C:\ULOG"
"ULOGOUT"=dword:00000001
"TUXIPC_MSG_BYTES"=dword:00010000
"TUXIPC_MSG_HDRS"=dword:00001fc0
"TUXIPC_MSG_QUEUES"=dword:00000320
"TUXIPC_MSG_QUEUE_BYTES"=dword:000100
00
"TUXIPC_MSG_SEG_BYTES"=dword:00000040
"TUXIPC_MSG_SEGS"=dword:00007fff
"TUXIPC_PROC"=dword:00001770
"TUXIPC_SEM"=dword:00001770
"TUXIPC_SEM_IDS"=dword:00001770
"TUXIPC_SEM_UNDO"=dword:00001770
"TUXIPC_SHM_PROCS"=dword:000186a0
"TUXIPC_SHM_SEGS"=dword:0000c350

```

```

[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services]

```

```

[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services\305
0]

```

REGEDIT4

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC]
"Type"=dword:00000020
"Start"=dword:00000002
"ErrorControl"=dword:00000000
"ImagePath"=hex(2):43,3a,5c,57,49,4e,4e,54,5c,5
3,79,73,74,65,6d,33,32,5c,69,6e,\
65,74,73,72,76,5c,69,6e,65,74,69,6e,66,6f,2e,65,
78,65,00
"DisplayName"="World Wide Web Publishing
Service"
"DependOnService"=hex(7):52,50,43,53,53,00,4e,
54,4c,4d,53,53,50,00,00
"DependOnGroup"=hex(7):00
"ObjectName"="LocalSystem"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters]
"MajorVersion"=dword:00000002
"MinorVersion"=dword:00000000
"AdminName"="Administrator"
"AdminEmail"="Admin@corp.com"
"MaxConnections"=dword:000186a0
"LogType"=dword:00000000
"LogFileDirectory"=hex(2):25,53,79,73,74,65,6d,5
2,6f,6f,74,25,5c,53,79,73,74,\
65,6d,33,32,5c,4c,6f,67,46,69,6c,65,73,00
"LogFileTruncateSize"=dword:01400000
"LogFilePeriod"=dword:00000001
"LogFileFormat"=dword:00000000
"LogSqlDataSource"="HTTPLOG"
"LogSqlTableName"="Internetlog"
"LogSqlUserName"="InternetAdmin"
"LogSqlPassword"="sqllog"
"Authorization"=dword:00000005
"AnonymousUserName"="IUSR_CL18"
"Default Load File"="Default.htm"
"Dir Browse Control"=dword:4000001e
"CheckForWAISDB"=dword:00000000
"CacheExtensions"=dword:00000001
"GlobalExpire"=dword:ffffff
"ServerSideIncludesEnabled"=dword:00000001
"ServerSideIncludesExtension"=".stm"
"DebugFlags"=dword:00000008
"ScriptTimeout"=dword:00000384
"ConnectionTimeout"=dword:00000384
"InstallPath"="C:\WINNT\System32\inetnsrv"
"SecurePort"=dword:000001bb
"Filter
DLs"="C:\WINNT\System32\inetnsrv\lsspifilt.dll"
"AccessDeniedMessage"="エラー：アクセス拒否さ
れました。"
"NTAuthenticationProviders"="NTLM"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Script Map]
".idc"="C:\WINNT\System32\inetnsrv\httpodbc.dll
"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Virtual Roots]
"/,"="C:\inetPub\wwwroot,1"
"/tpc,"="C:\client\tpap\Release,5"
"/Scripts,"="C:\inetPub\Scripts,4"
"/iisadmin,"="C:\WINNT\System32\inetnsrv\iisad
min,1"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Performance]
"Library"="w3ctrs.DLL"
"Open"="OpenW3PerformanceData"
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"Last Counter"=dword:00000790
"Last Help"=dword:00000791
"First Counter"=dword:00000758
"First Help"=dword:00000759

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Security]
"Security"=hex:01,00,14,80,c0,00,00,cc,00,00,
00,14,00,00,00,34,00,00,02,\
00,20,00,01,00,00,02,80,18,00,ff,01,0f,00,01,
01,00,00,00,00,01,00,00,\

```

```

00,00,20,02,00,00,02,00,8c,00,05,00,00,00,00,0
0,18,00,8d,01,02,00,01,01,00,\
00,00,00,00,01,00,00,00,00,e0,0d,15,00,00,00,1
c,00,fd,01,02,00,01,02,00,00,\
00,00,00,05,20,00,00,00,23,02,00,00,00,00,0
0,00,00,1c,00,ff,01,0f,00,01,\
02,00,00,00,00,00,05,20,00,00,00,20,02,00,00,0
0,00,00,00,00,00,1c,00,ff,01,\
0f,00,01,02,00,00,00,00,05,20,00,00,00,25,0
2,00,00,00,00,00,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,00,05,12,00,0
0,00,25,02,00,00,01,01,00,00,\
00,00,00,05,12,00,00,00,01,01,00,00,00,00,0
5,12,00,00,00

```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\W3SAMP]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Enum]
"0"="Root\LEGACY_W3SVC\0000"
"Count"=dword:00000001
"NextInstance"=dword:00000001
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Fujitsu\T
PC-C ISAPI Application]
"Term_Base"=dword:13b6f
"NumWarehouses"=dword:2c88
"MaxUsers"=dword:1bd50
"MaxTerm of Client"=dword:128e
"CONTROL_Flag"=dword:00000000
```

```
#
# ubbconfig : TUXEDO configuration file
#
```

*RESOURCES

```

IPCKEY      80952
MASTER     SITE1
#UID       1
#GID       1
PERM       0660
MAXACCESSERS 800
MAXSERVERS 200
MAXSERVICES 1000
MODEL      SHM
LDBAL      Y

```

*MACHINES

```

CL18 LMID=SITE1
  APPDIR="c:\sv-aplora"
  TUXCONFIG="c:\client\tuxconfig"
  TUXDIR="c:\tuxedo"
  ULOGPFX="c:\tuxlog\test"

```

*GROUPS

```
group1 LMID=SITE1 GRPNO=1
```

*SERVERS

```

DEFAULT: RESTART=Y MAXGEN=5
RPERM=0660
tpccfml SRVGRP=group1 RQADDR=newq1
SRVID=1 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq2
SRVID=2 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"

```

```

tpccfml SRVGRP=group1 RQADDR=newq3
SRVID=3 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq4
SRVID=4 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq5
SRVID=5 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq6
SRVID=6 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq7
SRVID=7 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq8
SRVID=8 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq9
SRVID=9 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq10
SRVID=10 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq11
SRVID=11 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq12
SRVID=12 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq13
SRVID=13 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq14
SRVID=14 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfmlp SRVGRP=group1 RQADDR=payq1
SRVID=15 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq2
SRVID=16 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq3
SRVID=17 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq4
SRVID=18 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq5
SRVID=19 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq6
SRVID=20 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq7
SRVID=21 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlo SRVGRP=group1 RQADDR=ordq1
SRVID=22 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfmlo SRVGRP=group1 RQADDR=ordq2
SRVID=23 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfmld SRVGRP=group1 RQADDR=delq1
SRVID=24 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmld SRVGRP=group1 RQADDR=delq2
SRVID=25 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"

```

```

tpccfmld SRVGRP=group1 RQADDR=delq3
SRVID=26 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmld SRVGRP=group1 RQADDR=stoq1
SRVID=27 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfmld SRVGRP=group1 RQADDR=stoq2
SRVID=28 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfmld SRVGRP=group1 RQADDR=stoq3
SRVID=29 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"

```

*SERVICES

```

"NEWORDER" TRANTIME=0
  SRVGRP=group1
"PAYMENT" TRANTIME=0
  SRVGRP=group1
"ORDERSTATUS" TRANTIME=0
  SRVGRP=group1
"DELIVERY" TRANTIME=0
  SRVGRP=group1
"STOCKLVL" TRANTIME=0
  SRVGRP=group1

```

*ROUTING

```

===== client 19 =====
REGEDIT4

```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters]
"BandwidthLevel"=dword:ffffff
"ListenBackLog"=dword:00000019
"MemoryCacheSize"=dword:00000000
"ObjectCacheTTL"=dword:ffffff
"PoolThreadLimit"=dword:0000020d

```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters\Filter]
"FilterType"=dword:00000000
"NumGrantSites"=dword:00000000
"NumDenySites"=dword:00000000

```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters\MimeMap]
"text/html,html,h"=""
"image/gif,gif,g"=""
"image/jpeg,jpg,,"=""
"text/plain,txt,0"=""
"text/html,html,h"=""
"image/jpeg,jpeg,,"=""
"image/jpeg,jpe,,"=""
"image/bmp,bmp,,"=""
"application/octet-stream,*,,5"=""
"application/pdf,pdf,,5"=""
"application/octet-stream,bin,,5"=""
"application/oda,oda,,5"=""
"application/zip,zip,,9"=""
"application/rft,rft,,5"=""
"application/postscript,ps,,5"=""
"application/postscript,ai,,5"=""
"application/postscript,eps,,5"=""
"application/mac-binhex40,hqx,,4"=""
"application/msword,doc,,5"=""
"application/msword,dot,,5"=""
"application/winhlp,hlp,,5"=""
"video/mpeg,mpeg,,"=""

```

```

"video/mpeg.mpg,,"="
"video/mpeg.mpe,,"="
"video/x-msvideo.avi,,"="
"video/quicktime.qt,,"="
"video/quicktime.mov,,"="
"video/x-sgi-movie.movie,,"="
"x-world/x-vmrl.wrl,5"="
"x-world/x-vmrl.xaf,5"="
"x-world/x-vmrl.xof,5"="
"x-world/x-vmrl.flr,5"="
"x-world/x-vmrl.wrz,5"="
"application/x-director.dcr,5"="
"application/x-director.dir,5"="
"application/x-director.dxr,5"="
"image/cis-cod.cod,5"="
"image/x-cmx.cmx,5"="
"application/envoy.evy,5"="
"application/x-msaccess.mdb,5"="
"application/x-mscardfile.crd,5"="
"application/x-msclip.clp,5"="
"application/octet-stream.exe,5"="
"application/x-msexcel.xla,5"="
"application/x-msexcel.xls,5"="
"application/x-msexcel.xlm,5"="
"application/x-msexcel.xls,5"="
"application/x-msexcel.xlt,5"="
"application/x-msexcel.xlw,5"="
"application/x-msmediaview.m13,5"="
"application/x-msmediaview.m14,5"="
"application/x-msmoney.mny,5"="
"application/x-mspowerpoint.ppt,5"="
"application/x-msproject.mpp,5"="
"application/x-mspublisher.pub,5"="
"application/x-msterminal.trm,5"="
"application/x-msworks.wks,5"="
"application/x-mswrite.wrl,5"="
"application/x-msmetafile.wmf,5"="
"application/x-csh.csh,5"="
"application/x-dvi.dvi,5"="
"application/x-hdf.hdf,5"="
"application/x-latex.latex,5"="
"application/x-netcdf.nc,5"="
"application/x-netcdf.cdf,5"="
"application/x-sh.sh,5"="
"application/x-tcl.tcl,5"="
"application/x-tex.tex,5"="
"application/x-texinfo.texinfo,5"="
"application/x-texinfo.texi,5"="
"application/x-troff.tr,5"="
"application/x-troff.tr,5"="
"application/x-troff.roff,5"="
"application/x-troff-man.man,5"="
"application/x-troff-me.me,5"="
"application/x-troff-ms.ms,5"="
"application/x-wais-source.src,7"="
"application/x-bcpio.bcpio,5"="
"application/x-cpio.cpio,5"="
"application/x-gtar.gtar,9"="
"application/x-shar.shar,5"="
"application/x-sv4cpio.sv4cpio,5"="
"application/x-sv4crc.sv4crc,5"="
"application/x-tar.tar,5"="
"application/x-ustar.ustar,5"="
"audio/basic.au,,"="
"audio/basic.snd,,"="
"audio/x-aiff.aif,,"="
"audio/x-aiff.aiff,,"="
"audio/x-aiff.aifc,,"="
"audio/x-wav.wav,,"="

```

```

"audio/x-pn-realaudio.ram,,"="
"image/ief.ief,,"="
"image/tiff.tiff,,"="
"image/tiff.tif,,"="
"image/x-cmu-raster.ras,,"="
"image/x-portable-anymap.pnm,,"="
"image/x-portable-bitmap.pbm,,"="
"image/x-portable-graymap.pgm,,"="
"image/x-portable-pixmap.ppm,,"="
"image/x-rgb.rgb,,"="
"image/x-xbitmap.xbm,,"="
"image/x-xpixmap.xpm,,"="
"image/x-xwindowdump.xwd,,"="
"text/html.stm,h"="
"text/plain.bas,0"="
"text/plain.c,0"="
"text/plain.h,0"="
"text/richtext.rtx,0"="
"text/tab-separated-values.tsv,0"="
"text/x-setext.etx,0"="
"application/x-perfmon.pmc,5"="
"application/x-perfmon.pma,5"="
"application/x-perfmon.pmr,5"="
"application/x-perfmon.pml,5"="
"application/x-perfmon.pmw,5"="

```

REGEDIT4

```

[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment]
"TUXDIR"="C:\TUXEDO"
"NLSPATH"="C:\TUXEDO\locale\C"
"ULOGDIR"="C:\TUXEDO"
"ULOGPF"="C:\ULOG"
"ULOGOUT"=dword:00000001
"TUXIPC_MSG_BYTES"=dword:00010000
"TUXIPC_MSG_HDRS"=dword:00001fc0
"TUXIPC_MSG_QUEUE_SIZE"=dword:00000320
"TUXIPC_MSG_QUEUE_BYTES"=dword:000100
00
"TUXIPC_MSG_SEG_BYTES"=dword:00000040
"TUXIPC_MSG_SEGS"=dword:00007fff
"TUXIPC_PROC"=dword:00001770
"TUXIPC_SEM"=dword:00001770
"TUXIPC_SEM_IDS"=dword:00001770
"TUXIPC_SEM_UNDO"=dword:00001770
"TUXIPC_SHM_PROCS"=dword:000186a0
"TUXIPC_SHM_SEGS"=dword:0000c350

```

```

[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services]

```

```

[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services\305
0]

```

REGEDIT4

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC]
"Type"=dword:00000020
"Start"=dword:00000002
"ErrorControl"=dword:00000000
"ImagePath"=hex(2):43,3a,5c,5f,49,4e,4e,54,5c,5
3,79,73,74,65,6d,33,32,5c,69,6e,\
65,74,73,72,76,5c,69,6e,65,74,69,6e,66,6f,2e,65,
78,65,00
"DisplayName"="World Wide Web Publishing
Service"

```

```

"DependOnService"=hex(7):52,50,43,53,53,00,4e,
54,4c,4d,53,53,50,00,00
"DependOnGroup"=hex(7):00
"ObjectName"="LocalSystem"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters]
"MajorVersion"=dword:00000002
"MinorVersion"=dword:00000000
"AdminName"="Administrator"
"AdminEmail"="Admin@corp.com"
"MaxConnections"=dword:000186a0
"LogType"=dword:00000000
"LogFileDirectory"=hex(2):25,53,79,73,74,65,6d,5
2,6f,6f,74,25,5c,53,79,73,74,\
65,6d,33,32,5c,4c,6f,67,46,69,6c,65,73,00
"LogFileTruncateSize"=dword:01400000
"LogFilePeriod"=dword:00000001
"LogFileFormat"=dword:00000000
"LogSqlDataSource"="HTTPLOG"
"LogSqlTableName"="Internetlog"
"LogSqlUserName"="InternetAdmin"
"LogSqlPassword"="sqllog"
"Authorization"=dword:00000005
"AnonymousUserName"="IUSR_CL19"
"Default Load File"="Default.htm"
"Dir Browse Control"=dword:4000001e
"CheckForWAISDB"=dword:00000000
"CacheExtensions"=dword:00000001
"GlobalExpire"=dword:ffffff
"ServerSideIncludesEnabled"=dword:00000001
"ServerSideIncludesExtension"=".stm"
"DebugFlags"=dword:00000008
"ScriptTimeout"=dword:00000384
"ConnectionTimeout"=dword:00000384
"InstallPath"="C:\WINNT\System32\inetrsrv"
"SecurePort"=dword:000001bb
"Filter
DLLs"="C:\WINNT\System32\inetrsrv\lsspifilt.dll"
"AccessDeniedMessage"="エラー : アクセスは拒否さ
れました。"
"NTAuthenticationProviders"="NTLM"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Script Map]
".idc"="C:\WINNT\System32\inetrsrv\httpodbc.dll"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Virtual Roots]
"/,"="C:\inetPub\wwwroot,1"
"/tpc,"="C:\client\tpap\Release,5"
"/Scripts,"="C:\inetPub\Scripts,4"
"/iisadmin,"="C:\WINNT\System32\inetrsrv\iisad
min,1"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Performance]
"Library"="w3ctrs.DLL"
"Open"="OpenW3PerformanceData"
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"Last Counter"=dword:00000790
"Last Help"=dword:00000791
"First Counter"=dword:00000758
"First Help"=dword:00000759

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Security]

```

```
"Security"=hex:01,00,14,80,c0,00,00,00,cc,00,00,
00,14,00,00,00,34,00,00,00,02,\
  00,20,00,01,00,00,00,02,80,18,00,ff,01,0f,00,01,
01,00,00,00,00,00,01,00,00,\
  00,00,20,02,00,00,02,00,8c,00,05,00,00,00,0,0
0,18,00,8d,01,02,00,01,01,00,\
  00,00,00,00,01,00,00,00,00,00,00,00,00,00,1
c,00,fd,01,02,00,01,02,00,00,\
  00,00,00,05,20,00,00,00,23,02,00,00,00,00,0,0
0,00,0,1c,00,ff,01,0f,00,01,\
  02,00,00,00,00,00,05,20,00,00,00,00,20,02,00,0,0
0,00,00,00,00,0,1c,00,ff,01,\
  0f,00,01,02,00,00,00,00,05,20,00,00,00,25,0
2,00,00,00,00,00,00,00,18,\
  00,fd,01,02,00,01,01,00,00,00,00,05,12,00,0
0,00,25,02,00,00,01,01,00,00,\
  00,00,00,05,12,00,00,00,01,01,00,00,00,00,0
5,12,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\W3SAMP]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Enum]
"0"="Root\LEGACY_W3SVC\0000"
"Count"=dword:00000001
"NextInstance"=dword:00000001
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Fujitsu\T
PC-C ISAPI Application]
"Term_Base"=dword:14dfd
"NumWarehouses"=dword:2c88
"MaxUsers"=dword:1bd50
"MaxTerm of Client"=dword:128e
"CONTROL_Flag"=dword:00000000
```

```
#
# ubbconfig : TUXEDO configuration file
#
```

```
*RESOURCES
IPCKEY      80952
MASTER     SITE1
#UID        1
#GID        1
PERM        0660
MAXACCESSERS  800
MAXSERVERS  200
MAXSERVICES 1000
MODEL       SHM
LDBAL       Y
```

```
*MACHINES
CL19 LMID=SITE1
  APPDIR="c:\sv-aplora"
  TUXCONFIG="c:\client\tuxconfig"
  TUXDIR="c:\tuxedo"
  ULOGPFX="c:\tuxlog\test"
```

```
*GROUPS
group1 LMID=SITE1 GRPNO=1
```

```
*SERVERS
DEFAULT: RESTART=Y MAXGEN=5
RQPERM=0660
```

```
tpccfml SRVGRP=group1 RQADDR=newq1
SRVID=1 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq2
SRVID=2 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq3
SRVID=3 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq4
SRVID=4 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq5
SRVID=5 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq6
SRVID=6 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq7
SRVID=7 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq8
SRVID=8 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq9
SRVID=9 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq10
SRVID=10 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq11
SRVID=11 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq12
SRVID=12 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq13
SRVID=13 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq14
SRVID=14 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfmlp SRVGRP=group1 RQADDR=payq1
SRVID=15 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq2
SRVID=16 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq3
SRVID=17 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq4
SRVID=18 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq5
SRVID=19 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq6
SRVID=20 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq7
SRVID=21 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml0 SRVGRP=group1 RQADDR=ordq1
SRVID=22 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfml0 SRVGRP=group1 RQADDR=ordq2
SRVID=23 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
```

```
tpccfml SRVGRP=group1 RQADDR=delq1
SRVID=24 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfml SRVGRP=group1 RQADDR=delq2
SRVID=25 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfml SRVGRP=group1 RQADDR=delq3
SRVID=26 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmls SRVGRP=group1 RQADDR=stoq1
SRVID=27 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfmls SRVGRP=group1 RQADDR=stoq2
SRVID=28 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfmls SRVGRP=group1 RQADDR=stoq3
SRVID=29 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
```

```
*SERVICES
"NEWORDER" TRANTIME=0
  SRVGRP=group1
"PAYMENT" TRANTIME=0
  SRVGRP=group1
"ORDERSTATUS" TRANTIME=0
  SRVGRP=group1
"DELIVERY" TRANTIME=0
  SRVGRP=group1
"STOCKLVL" TRANTIME=0
  SRVGRP=group1
```

*ROUTING

```
==== client 20 =====
REGEDIT4
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters]
"BandwidthLevel"=dword:ffffff
"ListenBackLog"=dword:00000019
"MemoryCacheSize"=dword:00000000
"ObjectCacheTTL"=dword:ffffff
"PoolThreadLimit"=dword:0000020d
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters\Filter]
"FilterType"=dword:00000000
"NumGrantSites"=dword:00000000
"NumDenySites"=dword:00000000
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters\MimeMap]
"text/html,html,h"=""
"image/gif,gif,g"=""
"image/jpeg,jpg,,"=""
"text/plain,txt,0"=""
"text/html,html,h"=""
"image/jpeg,jpeg,,"=""
"image/jpeg,jpe,,"=""
"image/bmp,bmp,,"=""
"application/octet-stream,*,,5"=""
"application/pdf,pdf,,5"=""
"application/octet-stream,bin,,5"=""
"application/oda,oda,,5"=""
"application/zip,zip,,9"=""
"application/rtf,rtf,,5"=""
"application/postscript,ps,,5"=""
"application/postscript,ai,,5"=""
```

```

"application/postscript,eps,5"=""
"application/mac-binhex40,hqx,4"=""
"application/msword,doc,5"=""
"application/msword,dot,5"=""
"application/winhlp,hlp,5"=""
"video/mpeg,mpeg,5"=""
"video/mpeg,mpg,5"=""
"video/mpeg,mpe,5"=""
"video/x-msvideo,avi,5"=""
"video/quicktime,qt,5"=""
"video/quicktime,mov,5"=""
"video/x-sgi-movie,movie,5"=""
"x-world/x-vmrl,wrl,5"=""
"x-world/x-vmrl,xaf,5"=""
"x-world/x-vmrl,xof,5"=""
"x-world/x-vmrl,flr,5"=""
"x-world/x-vmrl,wrz,5"=""
"application/x-director,dcr,5"=""
"application/x-director,dir,5"=""
"application/x-director,dxr,5"=""
"image/cis-cod,cod,5"=""
"image/x-cmx,cmx,5"=""
"application/envoy,evy,5"=""
"application/x-msaccess,mdb,5"=""
"application/x-mscardfile,crd,5"=""
"application/x-mscclip,clp,5"=""
"application/octet-stream,exe,5"=""
"application/x-msexcel,xla,5"=""
"application/x-msexcel,xlc,5"=""
"application/x-msexcel,xlm,5"=""
"application/x-msexcel,xls,5"=""
"application/x-msexcel,xlt,5"=""
"application/x-msexcel,xlw,5"=""
"application/x-msmediaview,m13,5"=""
"application/x-msmediaview,m14,5"=""
"application/x-msmoney,mny,5"=""
"application/x-mspowerpoint,ppt,5"=""
"application/x-msproject,mpp,5"=""
"application/x-mspublisher,pub,5"=""
"application/x-msterminal,tm,5"=""
"application/x-msworks,wks,5"=""
"application/x-mswrite,wri,5"=""
"application/x-msmetafile,wmf,5"=""
"application/x-csh,csh,5"=""
"application/x-dvi,dvi,5"=""
"application/x-hdf,hdf,5"=""
"application/x-latex,latex,5"=""
"application/x-netcdf,nc,5"=""
"application/x-netcdf,cdf,5"=""
"application/x-sh,sh,5"=""
"application/x-tcl,tcl,5"=""
"application/x-tex,tex,5"=""
"application/x-texinfo,texinfo,5"=""
"application/x-texinfo,texi,5"=""
"application/x-troff,t,5"=""
"application/x-troff,tr,5"=""
"application/x-troff,roff,5"=""
"application/x-troff-man,man,5"=""
"application/x-troff-me,me,5"=""
"application/x-troff-ms,ms,5"=""
"application/x-wais-source,src,7"=""
"application/x-bcpio,bcpio,5"=""
"application/x-cpio,cpio,5"=""
"application/x-gtar,gtar,9"=""
"application/x-shar,shar,5"=""
"application/x-sv4cpio,sv4cpio,5"=""
"application/x-sv4crc,sv4crc,5"=""
"application/x-tar,tar,5"=""
"application/x-ustar,ustar,5"=""

```

```

"audio/basic,au,5"=""
"audio/basic,snd,5"=""
"audio/x-aiff,aif,5"=""
"audio/x-aiff,aiff,5"=""
"audio/x-aiff,aifc,5"=""
"audio/x-wav,wav,5"=""
"audio/x-pn-realaudio,ram,5"=""
"image/ief,ief,5"=""
"image/tiff,tiff,5"=""
"image/tiff,tif,5"=""
"image/x-cmu-raster,ras,5"=""
"image/x-portable-anymap,pnm,5"=""
"image/x-portable-bitmap,pbm,5"=""
"image/x-portable-graymap,pgm,5"=""
"image/x-portable-pixmap,ppm,5"=""
"image/x-rgb,rgb,5"=""
"image/x-xbitmap,xbm,5"=""
"image/x-pximap,xpm,5"=""
"image/x-xwindowdump,xwd,5"=""
"text/html,htm,5"=""
"text/html,html,5"=""
"text/plain,bas,0"=""
"text/plain,c,0"=""
"text/plain,h,0"=""
"text/richtext,rtx,0"=""
"text/tab-separated-values,tsv,0"=""
"text/x-setext,etx,0"=""
"application/x-perfmon,pmc,5"=""
"application/x-perfmon,pma,5"=""
"application/x-perfmon,pmr,5"=""
"application/x-perfmon,pml,5"=""
"application/x-perfmon,pmw,5"=""

```

REGEDIT4

```

[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment]
"TXUDIR"="C:\TUXEDO"
"NLSPATH"="C:\TUXEDO\locale\C"
"ULOGDIR"="C:\TUXEDO"
"ULOGPFX"="C:\ULOG"
"ULOGOUT"=dword:00000001
"TUXIPC_MSG_BYTES"=dword:00010000
"TUXIPC_MSG_HDRS"=dword:00001fc0
"TUXIPC_MSG_QUEUE"=dword:00000320
"TUXIPC_MSG_QUEUE_BYTES"=dword:000100
00
"TUXIPC_MSG_SEG_BYTES"=dword:00000040
"TUXIPC_MSG_SEGS"=dword:00007fff
"TUXIPC_PROC"=dword:00001770
"TUXIPC_SEM"=dword:00001770
"TUXIPC_SEM_IDS"=dword:00001770
"TUXIPC_SEM_UNDO"=dword:00001770
"TUXIPC_SHM_PROCS"=dword:000186a0
"TUXIPC_SHM_SEGS"=dword:0000c350

```

```

[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services]

```

```

[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services\305
0]

```

REGEDIT4

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC]
"Type"=dword:00000020
"Start"=dword:00000002
"ErrorControl"=dword:00000000

```

```

"ImagePath"=hex(2):43,3a,5c,57,49,4e,4e,54,5c,5
3,79,73,74,65,6d,33,32,5c,69,6e,
65,74,73,72,76,5c,69,6e,65,74,69,6e,66,6f,2e,65,
78,65,00
"DisplayName"="World Wide Web Publishing
Service"
"DependOnService"=hex(7):52,50,43,53,53,00,4e,
54,4c,4d,53,53,50,00,00
"DependOnGroup"=hex(7):00
"ObjectName"="LocalSystem"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters]
"MajorVersion"=dword:00000002
"MinorVersion"=dword:00000000
"AdminName"="Administrator"
"AdminEmail"="Admin@corp.com"
"MaxConnections"=dword:000186a0
"LogType"=dword:00000000
"LogFileDirectory"=hex(2):25,53,79,73,74,65,6d,5
2,6f,6f,74,25,5c,53,79,73,74,
65,6d,33,32,5c,4c,6f,67,46,69,6c,65,73,00
"LogFileTruncateSize"=dword:01400000
"LogFilePrefix"=dword:00000001
"LogFileFormat"=dword:00000000
"LogSqlDataSource"="HTTPLOG"
"LogSqlTableName"="Internetlog"
"LogSqlUserName"="InternetAdmin"
"LogSqlPassword"="sqllog"
"Authorization"=dword:00000005
"AnonymousUserName"="IUSR_CL25"
"Default Load File"="Default.htm"
"Dir Browse Control"=dword:4000001e
"CheckForWAISDB"=dword:00000000
"CacheExtensions"=dword:00000001
"GlobalExpire"=dword:ffffff
"ServerSideIncludesEnabled"=dword:00000001
"ServerSideIncludesExtension"=".stm"
"DebugFlags"=dword:00000008
"ScriptTimeout"=dword:00000384
"ConnectionTimeout"=dword:00000384
"InstallPath"="C:\WINNT\System32\inetnsrv"
"SecurePort"=dword:000001bb
"Filter
DLs"="C:\WINNT\System32\inetnsrv\lsspifilt.dll"
"AccessDeniedMessage"="エラー: 予期は拒否さ
れました。"
"NTAuthenticationProviders"="NTLM"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Script Map]
".idc"="C:\WINNT\System32\inetnsrv\httpodbc.dll"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Virtual Roots]
"/."="C:\inetpub\wwwroot,1"
"/tpc"="C:\client\tpap\Release,5"
"/Scripts"="C:\inetpub\scripts,4"
"/iisadmin"="C:\WINNT\System32\inetnsrv\iisad
min,1"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Performance]
"Library"="w3ctrs.DLL"
"Open"="OpenW3PerformanceData"
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"Last Counter"=dword:00000790

```

```
"Last Help"=dword:00000791
"First Counter"=dword:00000758
"First Help"=dword:00000759
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Security]
"Security"=hex:01,00,14,80,c0,00,00,cc,00,00,00,14,00,00,34,00,00,02,\
00,20,00,01,00,00,02,80,18,00,ff,01,0f,00,01,\
01,00,00,00,00,01,00,00,\
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,0,\
0,18,00,8d,01,02,00,01,01,00,\
00,00,00,00,01,00,00,00,00,00,ff,ff,00,00,1c,0,\
0,fd,01,02,00,01,02,00,00,\
00,00,00,05,20,00,00,00,23,02,00,00,d8,00,14,0,\
0,00,00,1c,00,ff,01,0f,00,01,\
02,00,00,00,00,05,20,00,00,00,20,02,00,00,d,\
8,00,14,00,00,1c,00,ff,01,\
0f,00,01,02,00,00,00,00,05,20,00,00,25,0,\
2,00,00,d8,00,14,00,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,05,12,00,0,\
0,00,25,02,00,00,01,01,00,00,\
00,00,00,05,12,00,00,01,01,00,00,00,00,0,\
5,12,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\W3SAMP]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Enum]
"0"="Root\LEGACY_W3SVC\0000"
"Count"=dword:00000001
"NextInstance"=dword:00000001
```

```
REGEDIT4
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Fujitsu\T
PC-C ISAPI Application]
"Term_Base"=dword:1608b
"NumWarehouses"=dword:2c88
"MaxUsers"=dword:1bd50
"MaxTerm of Client"=dword:128e
"CONTROL_Flag"=dword:00000000
```

```
#
# ubbconfig : TUXEDO configuration file
#
```

```
*RESOURCES
IPCKEY 80952
MASTER SITE1
#UID 1
#GID 1
PERM 0660
MAXACCESSERS 800
MAXSERVERS 200
MAXSERVICES 1000
MODEL SHM
LDBAL Y
```

```
*MACHINES
CL20 LMID=SITE1
APPDIR="c:\sv-aplora"
TUXCONFIG="c:\client\tuxconfig"
TUXDIR="c:\tuxedo"
ULOGPFX="c:\tuxlog\test"
```

```
*GROUPS
group1 LMID=SITE1 GRPNO=1
```

```
*SERVERS
DEFAULT: RESTART=Y MAXGEN=5
RQPERM=0660
tpccfml SRVGRP=group1 RQADDR=newq1
SRVID=1 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq2
SRVID=2 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq3
SRVID=3 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq4
SRVID=4 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq5
SRVID=5 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq6
SRVID=6 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq7
SRVID=7 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq8
SRVID=8 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq9
SRVID=9 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq10
SRVID=10 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq11
SRVID=11 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq12
SRVID=12 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq13
SRVID=13 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq14
SRVID=14 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfmlp SRVGRP=group1 RQADDR=payq1
SRVID=15 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq2
SRVID=16 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq3
SRVID=17 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq4
SRVID=18 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq5
SRVID=19 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq6
SRVID=20 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq7
SRVID=21 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
```

```
tpccfml SRVGRP=group1 RQADDR=ordq1
SRVID=22 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfml SRVGRP=group1 RQADDR=ordq2
SRVID=23 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfml SRVGRP=group1 RQADDR=delq1
SRVID=24 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfml SRVGRP=group1 RQADDR=delq2
SRVID=25 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfml SRVGRP=group1 RQADDR=delq3
SRVID=26 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmls SRVGRP=group1 RQADDR=stoq1
SRVID=27 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfmls SRVGRP=group1 RQADDR=stoq2
SRVID=28 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfmls SRVGRP=group1 RQADDR=stoq3
SRVID=29 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
```

```
*SERVICES
"NEWORDER" TRANTIME=0
SRVGRP=group1
"PAYMENT" TRANTIME=0
SRVGRP=group1
"ORDERSTATUS" TRANTIME=0
SRVGRP=group1
"DELIVERY" TRANTIME=0
SRVGRP=group1
"STOCKLVL" TRANTIME=0
SRVGRP=group1
```

```
*ROUTING
```

```
===== client 21 =====
REGEDIT4
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Parameters]
"BandwidthLevel"=dword:ffffff
"ListenBackLog"=dword:00000019
"MemoryCacheSize"=dword:00000000
"ObjectCacheTTL"=dword:ffffff
"PoolThreadLimit"=dword:0000020d
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Parameters\Filter]
"FilterType"=dword:00000000
"NumGrantSites"=dword:00000000
"NumDenySites"=dword:00000000
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Parameters\MimeMap]
"text/html,html,,h"=""
"image/gif,gif,,g"=""
"image/jpeg,jpg,,j"=""
"text/plain,txt,,0"=""
"text/html,html,,h"=""
"image/jpeg,jpeg,,j"=""
"image/jpeg,jpe,,j"=""
"image/bmp,bmp,,b"=""
"application/octet-stream,*,,5"=""
"application/pdf,pdf,,5"=""
```

```

"application/octet-stream,bin,,5"=""
"application/oda,oda,,5"=""
"application/zip,zip,,9"=""
"application/rf,rf,,5"=""
"application/postscript,ps,,5"=""
"application/postscript,ai,,5"=""
"application/postscript,eps,,5"=""
"application/mac-binhex40,hqx,,4"=""
"application/msword,doc,,5"=""
"application/msword,dot,,5"=""
"application/winhelp,hlp,,5"=""
"video/mpeg,mpeg,,,"=""
"video/mpeg,mpg,,,"=""
"video/mpeg,mpe,,,"=""
"video/x-msvideo,avi,,<"=""
"video/quicktime,qt,,,"=""
"video/quicktime,mov,,,"=""
"video/x-sgi-movie,movie,,<"=""
"x-world/x-vrml,wrl,,5"=""
"x-world/x-vrml,xaf,,5"=""
"x-world/x-vrml,xof,,5"=""
"x-world/x-vrml,flr,,5"=""
"x-world/x-vrml,wrz,,5"=""
"application/x-director,dcr,,5"=""
"application/x-director,dir,,5"=""
"application/x-director,dxr,,5"=""
"image/cis-cod,cod,,5"=""
"image/x-cmx,cmx,,5"=""
"application/envoy,evy,,5"=""
"application/x-msaccess,mdb,,5"=""
"application/x-mscardfile,crd,,5"=""
"application/x-msclip,clip,,5"=""
"application/octet-stream,exe,,5"=""
"application/x-msexcel,xla,,5"=""
"application/x-msexcel,xlc,,5"=""
"application/x-msexcel,xlm,,5"=""
"application/x-msexcel,xls,,5"=""
"application/x-msexcel,xlt,,5"=""
"application/x-msexcel,xlw,,5"=""
"application/x-msmediaview,m13,,5"=""
"application/x-msmediaview,m14,,5"=""
"application/x-msmoney,mny,,5"=""
"application/x-mspowerpoint,ppt,,5"=""
"application/x-msproject,mpp,,5"=""
"application/x-mspublisher,pub,,5"=""
"application/x-msterminal,tm,,5"=""
"application/x-msworks,wks,,5"=""
"application/x-mswrite,wri,,5"=""
"application/x-msmetafile,wmf,,5"=""
"application/x-csh,csh,,5"=""
"application/x-dvi,dvi,,5"=""
"application/x-hdf,hdf,,5"=""
"application/x-latex,latex,,5"=""
"application/x-netcdf,nc,,5"=""
"application/x-netcdf,cdf,,5"=""
"application/x-sh,sh,,5"=""
"application/x-tcl,tcl,,5"=""
"application/x-tex,tex,,5"=""
"application/x-texinfo,texinfo,,5"=""
"application/x-texinfo,texi,,5"=""
"application/x-troff,t,,5"=""
"application/x-troff,tr,,5"=""
"application/x-troff,roff,,5"=""
"application/x-troff-man,man,,5"=""
"application/x-troff-me,me,,5"=""
"application/x-troff-ms,ms,,5"=""
"application/x-wais-source,src,,7"=""
"application/x-bcpio,bcpio,,5"=""
"application/x-cpio,cpio,,5"=""

```

```

"application/x-gtar,gtar,,9"=""
"application/x-shar,shar,,5"=""
"application/x-sv4cpio,sv4cpio,,5"=""
"application/x-sv4crc,sv4crc,,5"=""
"application/x-tar,tar,,5"=""
"application/x-ustar,ustar,,5"=""
"audio/basic,au,,<"=""
"audio/basic,snd,,<"=""
"audio/x-aiff,aif,,<"=""
"audio/x-aiff,aiff,,<"=""
"audio/x-aiff,aifc,,<"=""
"audio/x-wav,wav,,<"=""
"audio/x-pn-realaudio,ram,,<"=""
"image/ief,ief,,,"=""
"image/tiff,tiff,,,"=""
"image/tiff,tif,,,"=""
"image/x-cmu-raster,ras,,,"=""
"image/x-portable-anymap,pnm,,,"=""
"image/x-portable-bitmap,pbm,,,"=""
"image/x-portable-graymap,pgm,,,"=""
"image/x-portable-pixmap,ppm,,,"=""
"image/x-rgb,rgb,,,"=""
"image/x-xbitmap,xbm,,,"=""
"image/x-xpixmap,xpm,,,"=""
"image/x-xwindowdump,xwd,,,"=""
"text/html,stm,,h"=""
"text/plain,bas,,0"=""
"text/plain,c,,0"=""
"text/plain,h,,0"=""
"text/richtext,rtx,,0"=""
"text/tab-separated-values,tsv,,0"=""
"text/x-setext,etx,,0"=""
"application/x-perfmon,pmc,,5"=""
"application/x-perfmon,pma,,5"=""
"application/x-perfmon,pmr,,5"=""
"application/x-perfmon,pml,,5"=""
"application/x-perfmon,pmw,,5"=""

```

REGEDIT4

```

[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment]
"TUXDIR"="C:\TUXEDO"
"NLSPATH"="C:\TUXEDO\locale\C"
"ULOGDIR"="C:\TUXEDO"
"ULOGPFX"="C:\ULOG"
"ULOGOUT"=dword:00000001
"TUXIPC_MSG_BYTES"=dword:00010000
"TUXIPC_MSG_HDRS"=dword:00001fc0
"TUXIPC_MSG_QUEUES"=dword:00000320
"TUXIPC_MSG_QUEUE_BYTES"=dword:000100
00
"TUXIPC_MSG_SEG_BYTES"=dword:00000040
"TUXIPC_MSG_SEGS"=dword:00007fff
"TUXIPC_PROC"=dword:00001770
"TUXIPC_SEM"=dword:00001770
"TUXIPC_SEM_IDS"=dword:00001770
"TUXIPC_SEM_UNDO"=dword:00001770
"TUXIPC_SHM_PROCS"=dword:000186a0
"TUXIPC_SHM_SEGS"=dword:0000c350

```

```

[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services]

```

```

[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services\305
0]

```

REGEDIT4

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC]
"Type"=dword:00000020
"Start"=dword:00000002
"ErrorControl"=dword:00000000
"ImagePath"=hex(2):43,3a,5c,57,49,4e,4e,54,5c,5
3,79,73,74,65,6d,33,32,5c,69,6e,\
65,74,73,72,76,5c,69,6e,65,74,69,6e,66,6f,2e,65,
78,65,00
"DisplayName"="World Wide Web Publishing
Service"
"DependOnService"=hex(7):52,50,43,53,53,00,4e,
54,4c,4d,53,53,50,00,00
"DependOnGroup"=hex(7):00
"ObjectName"="LocalSystem"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters]
"MajorVersion"=dword:00000002
"MinorVersion"=dword:00000000
"AdminName"="Administrator"
"AdminEmail"="Admin@corp.com"
"MaxConnections"=dword:000186a0
"LogType"=dword:00000000
"LogFileDirectory"=hex(2):25,53,79,73,74,65,6d,5
2,6f,6f,74,25,5c,53,79,73,74,\
65,6d,33,32,5c,4c,6f,67,46,69,6c,65,73,00
"LogFileTruncateSize"=dword:01400000
"LogFilePeriod"=dword:00000001
"LogFileFormat"=dword:00000000
"LogSqlDataSource"="HTTPLOG"
"LogSqlTableName"="Internetlog"
"LogSqlUserName"="InternetAdmin"
"LogSqlPassword"="sqllog"
"Authorization"=dword:00000005
"AnonymousUserName"="IUSR_CL21"
"Default Load File"="Default.htm"
"Dir Browse Control"=dword:4000001e
"CheckForWAISDB"=dword:00000000
"CacheExtensions"=dword:00000001
"GlobalExpire"=dword:ffffff
"ServerSideIncludesEnabled"=dword:00000001
"ServerSideIncludesExtension"=".stm"
"DebugFlags"=dword:00000008
"ScriptTimeout"=dword:00000384
"ConnectionTimeout"=dword:00000384
"InstallPath"="C:\WINNT\System32\inetnsrv"
"SecurePort"=dword:000001bb
"Filter
DLLs"="C:\WINNT\System32\inetnsrv\sspifill.dll"
"AccessDeniedMessage"="エラー : アクセスは拒否さ
れました。"
"NTAuthenticationProviders"="NTLM"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Script Map]
".idc"="C:\WINNT\System32\inetnsrv\httpodbc.dll"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Virtual Roots]
"/,"="C:\InetPub\wwwroot,,1"
"/tpc,"="C:\client\tpap\Release,,5"
"/Scripts,"="C:\InetPub\Scripts,,4"
"/iisadmin,"="C:\WINNT\System32\inetnsrv\iisad
min,,1"

```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Performance]
"Library"="w3ctrs.DLL"
"Open"="OpenW3PerformanceData"
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"Last Counter"=dword:00000790
"Last Help"=dword:00000791
"First Counter"=dword:00000758
"First Help"=dword:00000759
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Security]
"Security"=hex:01,00,14,80,c0,00,00,cc,00,00,
00,14,00,00,00,34,00,00,00,02,\
00,20,00,01,00,00,00,02,80,18,00,ff,01,0f,00,01,
01,00,00,00,00,01,00,00,\
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,0,0
0,18,00,8d,01,02,00,01,01,00,\
00,00,00,00,01,00,00,00,00,74,00,44,00,00,0,1
c,00,fd,01,02,00,01,02,00,00,\
00,00,00,05,20,00,00,00,23,02,00,00,65,00,74,0
0,00,00,1c,00,ff,01,0f,00,01,\
02,00,00,00,00,00,05,20,00,00,00,20,02,00,00,6
5,00,74,00,00,00,1c,00,ff,01,\
0f,00,01,02,00,00,00,00,05,20,00,00,00,25,0
2,00,00,65,00,74,00,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,05,12,00,0
0,00,25,02,00,00,01,01,00,00,\
00,00,00,05,12,00,00,00,01,01,00,00,00,00,0
5,12,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\W3SAMP]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Enum]
"0"="Root\LEGACY_W3SVC\0000"
"Count"=dword:00000001
"NextInstance"=dword:00000001
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Fujitsu\T
PC-C ISAPI Application]
"Term_Base"=dword:17319
"NumWarehouses"=dword:2c88
"MaxUsers"=dword:1bd50
"MaxTerm of Client"=dword:128e
"CONTROL_Flag"=dword:00000000
```

```
#
# ubbconfig : TUXEDO configuration file
#
```

```
*RESOURCES
IPCKEY 80952
MASTER SITE1
#UID 1
#GID 1
PERM 0660
MAXACCESSERS 800
MAXSERVERS 200
MAXSERVICES 1000
MODEL SHM
LDBAL Y
```

```
*MACHINES
CL21 LMID=SITE1
```

```
APPDIR="c:\sv-ap\hora"
TUXCONFIG="c:\client\tuxconfig"
TUXDIR="c:\tuxedo"
ULOGPFX="c:\tuxlogtest"
```

```
*GROUPS
group1 LMID=SITE1 GRPNO=1
```

```
*SERVERS
DEFAULT: RESTART=Y MAXGEN=5
RQPERM=0660
tpccfml SRVGRP=group1 RQADDR=newq1
SRVID=1 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq2
SRVID=2 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq3
SRVID=3 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq4
SRVID=4 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq5
SRVID=5 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq6
SRVID=6 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq7
SRVID=7 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq8
SRVID=8 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq9
SRVID=9 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq10
SRVID=10 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq11
SRVID=11 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq12
SRVID=12 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq13
SRVID=13 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq14
SRVID=14 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfmlp SRVGRP=group1 RQADDR=payq1
SRVID=15 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq2
SRVID=16 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq3
SRVID=17 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq4
SRVID=18 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq5
SRVID=19 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
```

```
tpccfmlp SRVGRP=group1 RQADDR=payq6
SRVID=20 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq7
SRVID=21 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=ordq1
SRVID=22 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfml SRVGRP=group1 RQADDR=ordq2
SRVID=23 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfml SRVGRP=group1 RQADDR=delq1
SRVID=24 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfml SRVGRP=group1 RQADDR=delq2
SRVID=25 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfml SRVGRP=group1 RQADDR=delq3
SRVID=26 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfml SRVGRP=group1 RQADDR=stoq1
SRVID=27 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfml SRVGRP=group1 RQADDR=stoq2
SRVID=28 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfml SRVGRP=group1 RQADDR=stoq3
SRVID=29 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
```

```
*SERVICES
"NEWORDER" TRANTIME=0
SRVGRP=group1
"PAYMENT" TRANTIME=0
SRVGRP=group1
"ORDERSTATUS" TRANTIME=0
SRVGRP=group1
"DELIVERY" TRANTIME=0
SRVGRP=group1
"STOCKLVL" TRANTIME=0
SRVGRP=group1
```

*ROUTING

```
===== client 22 =====
REGEDIT4
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters]
"BandwidthLevel"=dword:ffffff
"ListenBackLog"=dword:00000019
"MemoryCacheSize"=dword:00000000
"ObjectCacheTTL"=dword:ffffff
"PoolThreadLimit"=dword:0000020d
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters\Filter]
"FilterType"=dword:00000000
"NumGrantSites"=dword:00000000
"NumDenySites"=dword:00000000
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\InetInfo\Parameters\MimeMap]
"text/html,html,h"=""
"image/gif,gif,g"=""
"image/jpeg,jpg,,"=""
"text/plain,txt,0"=""
```



```
"text/html,html,h"=""
"image/jpeg,jpeg,,"=""
"image/jpeg,jpe,,"=""
"image/bmp,bmp,,"=""
"application/octet-stream,*,,5"=""
"application/pdf,pdf,,5"=""
"application/octet-stream,bin,,5"=""
"application/oda,oda,,5"=""
"application/zip,zip,,9"=""
"application/rtf,rtf,,5"=""
"application/postscript,ps,,5"=""
"application/postscript,ai,,5"=""
"application/postscript,eps,,5"=""
"application/mac-binhex40,hqx,,4"=""
"application/msword,doc,,5"=""
"application/msword,dot,,5"=""
"application/winhlp,hlp,,5"=""
"video/mpeg,mpeg,,"=""
"video/mpeg,mpg,,"=""
"video/mpeg,mpe,,"=""
"video/x-msvideo,avi,<,"=""
"video/quicktime,qt,,"=""
"video/quicktime,mov,,"=""
"video/x-sgi-movie,movie,<,"=""
"x-world/x-vrml,vrml,,5"=""
"x-world/x-vrml,xaf,,5"=""
"x-world/x-vrml,xof,,5"=""
"x-world/x-vrml,flr,,5"=""
"x-world/x-vrml,wrz,,5"=""
"application/x-director,dcr,,5"=""
"application/x-director,dir,,5"=""
"application/x-director,dxr,,5"=""
"image/cis-cod,cod,,5"=""
"image/x-cmx,cmx,,5"=""
"application/envoy,evy,,5"=""
"application/x-msaccess,mdb,,5"=""
"application/x-mscardfile,crd,,5"=""
"application/x-msclip,clip,,5"=""
"application/octet-stream,exe,,5"=""
"application/x-msexcel,xla,,5"=""
"application/x-msexcel,xlc,,5"=""
"application/x-msexcel,xlm,,5"=""
"application/x-msexcel,xls,,5"=""
"application/x-msexcel,xlt,,5"=""
"application/x-msexcel,xlw,,5"=""
"application/x-msmediaview,m13,,5"=""
"application/x-msmediaview,m14,,5"=""
"application/x-msmoney,mny,,5"=""
"application/x-mspowerpoint,ppt,,5"=""
"application/x-msproject,mpp,,5"=""
"application/x-mspublisher,pub,,5"=""
"application/x-msterial,frm,,5"=""
"application/x-msworks,wks,,5"=""
"application/x-mswrite,wri,,5"=""
"application/x-msmetafile,wmf,,5"=""
"application/x-csh,csh,,5"=""
"application/x-dvi,dvi,,5"=""
"application/x-hdf,hdf,,5"=""
"application/x-latex,latex,,5"=""
"application/x-netcdf,nc,,5"=""
"application/x-netcdf,cdf,,5"=""
"application/x-sh,sh,,5"=""
"application/x-tcl,tcl,,5"=""
"application/x-tex,tex,,5"=""
"application/x-texinfo,texinfo,,5"=""
"application/x-texinfo,texi,,5"=""
"application/x-troff,t,,5"=""
"application/x-troff,tr,,5"=""
"application/x-troff,roff,,5"=""
```

```
"application/x-troff-man,man,,5"=""
"application/x-troff-me,me,,5"=""
"application/x-troff-ms,ms,,5"=""
"application/x-wais-source,src,,7"=""
"application/x-bcpio,bcpio,,5"=""
"application/x-cpio,cpio,,5"=""
"application/x-gtar,gtar,,9"=""
"application/x-shar,shar,,5"=""
"application/x-sv4cpio,sv4cpio,,5"=""
"application/x-sv4crc,sv4crc,,5"=""
"application/x-tar,tar,,5"=""
"application/x-ustar,ustar,,5"=""
"audio/basic,au,<,"=""
"audio/basic,snd,<,"=""
"audio/x-aiff,aif,<,"=""
"audio/x-aiff,aiff,<,"=""
"audio/x-aiff,aifc,<,"=""
"audio/x-wav,wav,<,"=""
"audio/x-pn-realaudio,ram,<,"=""
"image/ief,ief,,"=""
"image/tiff,tiff,,"=""
"image/tiff,tif,,"=""
"image/x-cmu-raster,ras,,"=""
"image/x-portable-anymap,pnm,,"=""
"image/x-portable-bitmap,pbm,,"=""
"image/x-portable-graymap,pgm,,"=""
"image/x-portable-pixmap,ppm,,"=""
"image/x-rgb,rgb,,"=""
"image/x-xbitmap,xbm,,"=""
"image/x-xpixmap,xpm,,"=""
"image/x-xwindowdump,xwd,,"=""
"text/html,stm,h"=""
"text/plain,bas,0"=""
"text/plain,c,0"=""
"text/plain,h,0"=""
"text/richtext,rtx,0"=""
"text/tab-separated-values,tsv,0"=""
"text/x-setext,etx,0"=""
"application/x-perfmon,pmc,,5"=""
"application/x-perfmon,pma,,5"=""
"application/x-perfmon,pmr,,5"=""
"application/x-perfmon,pml,,5"=""
"application/x-perfmon,pmw,,5"=""
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment]
"TUXDIR"="C:\TUXEDO"
"NLSPATH"="C:\TUXEDO\locale\C"
"ULOGDIR"="C:\TUXEDO"
"ULOGPF"="C:\ULOG"
"ULOGOUT"=dword:00000001
"TUXIPC_MSG_BYTES"=dword:00010000
"TUXIPC_MSG_HDRS"=dword:00001fc0
"TUXIPC_MSG_QUEUE"=dword:00000320
"TUXIPC_MSG_QUEUE_BYTES"=dword:000100
00
"TUXIPC_MSG_SEG_BYTES"=dword:00000040
"TUXIPC_MSG_SEGS"=dword:00007fff
"TUXIPC_PROC"=dword:00001770
"TUXIPC_SEM"=dword:00001770
"TUXIPC_SEM_IDS"=dword:00001770
"TUXIPC_SEM_UNDO"=dword:00001770
"TUXIPC_SHM_PROCS"=dword:000186a0
"TUXIPC_SHM_SEGS"=dword:0000c350
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services]
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA
Systems\TUXEDO\6.4\Environment\Services\305
0]
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC]
"Type"=dword:00000020
"Start"=dword:00000002
"ErrorControl"=dword:00000000
"ImagePath"=hex(2):43,3a,5c,57,49,4e,4e,54,5c,5
3,79,73,74,65,6d,33,32,5c,69,6e,
65,74,73,72,76,5c,69,6e,65,74,69,6e,66,6f,2e,65,
78,65,00
"DisplayName"="World Wide Web Publishing
Service"
"DependOnService"=hex(7):52,50,43,53,53,00,4e,
54,4c,4d,53,53,50,00,00
"DependOnGroup"=hex(7):00
"ObjectName"="LocalSystem"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters]
"MajorVersion"=dword:00000002
"MinorVersion"=dword:00000000
"AdminName"="Administrator"
"AdminEmail"="Admin@corp.com"
"MaxConnections"=dword:000186a0
"LogType"=dword:00000000
"LogFileDirectory"=hex(2):25,53,79,73,74,65,6d,5
2,6f,6f,74,25,5c,53,79,73,74,
65,6d,33,32,5c,4c,6f,67,46,69,6c,65,73,00
"LogFileTruncateSize"=dword:01400000
"LogFilePeriod"=dword:00000001
"LogFileFormat"=dword:00000000
"LogDataSource"="HTTPLOG"
"LogSqlTableName"="Internetlog"
"LogSqlUserName"="InternetAdmin"
"LogSqlPassword"="sqllog"
"Authorization"=dword:00000005
"AnonymousUserName"="IUSR_CL22"
"Default Load File"="Default.htm"
"Dir Browse Control"=dword:4000001e
"CheckForWAISDB"=dword:00000000
"CacheExtensions"=dword:00000001
"GlobalExpire"=dword:ffffff
"ServerSideIncludesEnabled"=dword:00000001
"ServerSideIncludesExtension"=".stm"
"DebugFlags"=dword:00000008
"ScriptTimeout"=dword:00000384
"ConnectionTimeout"=dword:00000384
"InstallPath"="C:\WINNT\System32\inetnsrv"
"SecurePort"=dword:000001bb
"Filter
DLLs"="C:\WINNT\System32\inetnsrv\lsspifilt.dll"
"AccessDeniedMessage"="エラー : アクセスは拒否さ
れました。"
"NTAuthenticationProviders"="NTLM"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Script Map]
".idc"="C:\WINNT\System32\inetnsrv\httpodbc.dll"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Virtual Roots]
"/,"="C:\inetPub\wwwroot,1"
```

```
"/tpc,"="C:\client\tpapl\Release,,5"
"/Scripts,"="C:\inetPub\scripts,,4"
"/iisadmin,"="C:\WINNT\System32\inetrv\iisad
min,,1"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Performance]
"Library"="w3ctrs.DLL"
"Open"="OpenW3PerformanceData"
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"Last Counter"=dword:00000790
"Last Help"=dword:00000791
"First Counter"=dword:00000758
"First Help"=dword:00000759
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Security]
"Security"=hex:01,00,14,80,c0,00,00,cc,00,00,
00,14,00,00,00,34,00,00,02,\
00,20,00,01,00,00,02,80,18,00,ff,01,0f,00,01,
01,00,00,00,00,01,00,00,\
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,0
0,18,00,8d,01,02,00,01,01,00,\
00,00,00,00,01,00,00,00,74,00,69,00,00,0,1
c,00,fd,01,02,00,01,02,00,00,\
00,00,00,05,20,00,00,00,23,02,00,00,76,00,65,0
0,00,00,1c,00,ff,01,0f,00,01,\
02,00,00,00,00,00,05,20,00,00,00,20,02,00,00,7
6,00,65,00,00,00,1c,00,ff,01,\
0f,00,01,02,00,00,00,00,05,20,00,00,00,25,0
2,00,00,76,00,65,00,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,05,12,00,0
0,00,25,02,00,00,01,01,00,00,\
00,00,00,05,12,00,00,00,01,01,00,00,00,00,0
5,12,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\W3SAMP]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Enum]
"0"="Root\LEGACY_W3SVC\0000"
"Count"=dword:00000001
"NextInstance"=dword:00000001
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Fujitsu\T
PC-C ISAPI Application]
"Term_Base"=dword:185a7
"NumWarehouses"=dword:2c88
"MaxUsers"=dword:1bd50
"MaxTerm of Client"=dword:128e
"CONTROL_Flag"=dword:00000000
```

```
#
# ubbconfig : TUXEDO configuration file
#
```

```
*RESOURCES
IPCKEY 80952
MASTER SITE1
#UID 1
#GID 1
PERM 0660
MAXACCESSERS 800
MAXSERVERS 200
MAXSERVICES 1000
```

```
MODEL SHM
LDBAL Y

*MACHINES
CL22 LMID=SITE1
APPDIR="c:\sv-ap\lora"
TUXCONFIG="c:\client\tuxconfig"
TUXDIR="c:\tuxedo"
ULOGPFX="c:\tuxlogtest"
```

```
*GROUPS
group1 LMID=SITE1 GRPNO=1
```

```
*SERVERS
DEFAULT: RESTART=Y MAXGEN=5
RQPERM=0660
tpccfml SRVGRP=group1 RQADDR=newq1
SRVID=1 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq2
SRVID=2 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq3
SRVID=3 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq4
SRVID=4 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq5
SRVID=5 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq6
SRVID=6 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq7
SRVID=7 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq8
SRVID=8 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq9
SRVID=9 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq10
SRVID=10 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq11
SRVID=11 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq12
SRVID=12 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq13
SRVID=13 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq14
SRVID=14 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=payq1
SRVID=15 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=payq2
SRVID=16 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=payq3
SRVID=17 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
```

```
tpccfml SRVGRP=group1 RQADDR=payq4
SRVID=18 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=payq5
SRVID=19 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=payq6
SRVID=20 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=payq7
SRVID=21 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=ordq1
SRVID=22 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfml SRVGRP=group1 RQADDR=ordq2
SRVID=23 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfml SRVGRP=group1 RQADDR=delq1
SRVID=24 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfml SRVGRP=group1 RQADDR=delq2
SRVID=25 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfml SRVGRP=group1 RQADDR=delq3
SRVID=26 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfml SRVGRP=group1 RQADDR=stoq1
SRVID=27 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfml SRVGRP=group1 RQADDR=stoq2
SRVID=28 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfml SRVGRP=group1 RQADDR=stoq3
SRVID=29 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
```

```
*SERVICES
"NEWORDER" TRANTIME=0
SRVGRP=group1
"PAYMENT" TRANTIME=0
SRVGRP=group1
"ORDERSTATUS" TRANTIME=0
SRVGRP=group1
"DELIVERY" TRANTIME=0
SRVGRP=group1
"STOCKLVL" TRANTIME=0
SRVGRP=group1
```

*ROUTING

```
==== client 23 =====
REGEDIT4
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\inetInfo\Parameters]
"BandwidthLevel"=dword:ffffff
"ListenBackLog"=dword:00000019
"MemoryCacheSize"=dword:00000000
"ObjectCacheTTL"=dword:ffffff
"PoolThreadLimit"=dword:0000020d
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\inetInfo\Parameters\Filter]
"FilterType"=dword:00000000
"NumGrantSites"=dword:00000000
"NumDenySites"=dword:00000000
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Netlnfop\Parameters\MimeMap]
"text/html,html,h"=""
"image/gif,gif,g"=""
"image/jpeg,jpg,:"=""
"text/plain,txt,0"=""
"text/html,html,h"=""
"image/jpeg,jpeg,:"=""
"image/jpeg,jpe,:"=""
"image/bmp,bmp,:"=""
"application/octet-stream,*,,5"=""
"application/pdf,pdf,5"=""
"application/octet-stream.bin,5"=""
"application/oda,oda,5"=""
"application/zip,zip,9"=""
"application/rft,rft,5"=""
"application/postscript,ps,5"=""
"application/postscript,ai,5"=""
"application/postscript,eps,5"=""
"application/mac-binhex40,hqx,4"=""
"application/msword,doc,5"=""
"application/msword,dot,5"=""
"application/winhlp,hlp,5"=""
"video/mpeg,mpeg,:"=""
"video/mpeg,mpg,:"=""
"video/mpeg,mpe,:"=""
"video/x-msvideo,avi,<"=""
"video/quicktime,qt,:"=""
"video/quicktime,mov,:"=""
"video/x-sgi-movie,movie,<"=""
"x-world/x-vrml,wrl,5"=""
"x-world/x-vrml,xaf,5"=""
"x-world/x-vrml,xof,5"=""
"x-world/x-vrml,flr,5"=""
"x-world/x-vrml,wrz,5"=""
"application/x-director,dcr,5"=""
"application/x-director,dir,5"=""
"application/x-director,dxr,5"=""
"image/cis-cod,cod,5"=""
"image/x-cmx,cmx,5"=""
"application/envoy,evy,5"=""
"application/x-msaccess,mdb,5"=""
"application/x-mscardfile,crd,5"=""
"application/x-msclip,clp,5"=""
"application/octet-stream,exe,5"=""
"application/x-msexcel,xla,5"=""
"application/x-msexcel,xlc,5"=""
"application/x-msexcel,xlm,5"=""
"application/x-msexcel,xls,5"=""
"application/x-msexcel,xlt,5"=""
"application/x-msexcel,xlw,5"=""
"application/x-msmediaview,m13,5"=""
"application/x-msmediaview,m14,5"=""
"application/x-msmoney,mny,5"=""
"application/x-mspowerpoint,ppt,5"=""
"application/x-msproject,mpp,5"=""
"application/x-mspublisher,pub,5"=""
"application/x-msterminal,tm,5"=""
"application/x-msworks,wks,5"=""
"application/x-mswrite,wri,5"=""
"application/x-msmetafile,wmf,5"=""
"application/x-csh,csh,5"=""
"application/x-dvi,dvi,5"=""
"application/x-hdf,hdf,5"=""
"application/x-latex,latex,5"=""
"application/x-netcdf,nc,5"=""
"application/x-netcdf,cdf,5"=""
"application/x-sh,sh,5"=""
"application/x-tcl,tcl,5"=""
```

```
"application/x-tex,tex,5"=""
"application/x-texinfo,texinfo,5"=""
"application/x-texinfo,txi,5"=""
"application/x-troff,t,5"=""
"application/x-troff,tf,5"=""
"application/x-troff-man,man,5"=""
"application/x-troff-me,me,5"=""
"application/x-troff-ms,ms,5"=""
"application/x-wais-source,src,7"=""
"application/x-bcpio,bcpio,5"=""
"application/x-cpio,cpio,5"=""
"application/x-gtar,gtar,9"=""
"application/x-shar,shar,5"=""
"application/x-sv4cpio,sv4cpio,5"=""
"application/x-sv4crc,sv4crc,5"=""
"application/x-tar,tar,5"=""
"application/x-ustar,ustar,5"=""
"audio/basic,au,<"=""
"audio/basic,snd,<"=""
"audio/x-aiff,aif,<"=""
"audio/x-aiff,aiff,<"=""
"audio/x-aiff,aifc,<"=""
"audio/x-wav,wav,<"=""
"audio/x-pn-realaudio,ram,<"=""
"image/ief,ief,:"=""
"image/tiff,tiff,:"=""
"image/tiff,tif,:"=""
"image/x-cmu-raster,ras,:"=""
"image/x-portable-anymap,pnm,:"=""
"image/x-portable-bitmap,pbm,:"=""
"image/x-portable-graymap,pgm,:"=""
"image/x-portable-pixmap,ppm,:"=""
"image/x-rgb,rgb,:"=""
"image/x-xbitmap,xbm,:"=""
"image/x-xpixmap,xpm,:"=""
"image/x-xwindowdump,xwd,:"=""
"text/html,stm,h"=""
"text/plain,bas,0"=""
"text/plain,c,0"=""
"text/plain,h,0"=""
"text/richtext,rtx,0"=""
"text/tab-separated-values,tsv,0"=""
"text/x-setext,etx,0"=""
"application/x-perfmon,pmc,5"=""
"application/x-perfmon,pma,5"=""
"application/x-perfmon,pmr,5"=""
"application/x-perfmon,pml,5"=""
"application/x-perfmon,pmw,5"=""
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA Systems\TUXEDO\6.4\Environment]
"TUXDIR"="C:\TUXEDO"
"NLSPATH"="C:\TUXEDO\locale\C"
"ULOGDIR"="C:\TUXEDO"
"ULOGPFX"="C:\ULOG"
"ULOGOUT"=dword:00000001
"TUXIPC_MSG_BYTES"=dword:00010000
"TUXIPC_MSG_HDRS"=dword:00001fc0
"TUXIPC_MSG_QUEUE"=dword:00000320
"TUXIPC_MSG_QUEUE_BYTES"=dword:00010000
"TUXIPC_MSG_SEG_BYTES"=dword:00000040
"TUXIPC_MSG_SEGS"=dword:00007fff
"TUXIPC_PROC"=dword:00001770
"TUXIPC_SEM"=dword:00001770
"TUXIPC_SEM_IDS"=dword:00001770
```

```
"TUXIPC_SEM_UNDO"=dword:00001770
"TUXIPC_SHM_PROCS"=dword:000186a0
"TUXIPC_SHM_SEGS"=dword:0000c350
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA Systems\TUXEDO\6.4\Environment\Services]
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEA Systems\TUXEDO\6.4\Environment\Services\3050]
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC]
"Type"=dword:00000020
"Start"=dword:00000002
"ErrorControl"=dword:00000000
"ImagePath"=hex(2):43,3a,5c,57,49,4e,4e,54,5c,53,79,73,74,65,6d,33,32,5c,69,6e,65,74,73,72,76,5c,69,6e,65,74,69,6e,66,6f,2e,65,78,65,00
"DisplayName"="World Wide Web Publishing Service"
"DependOnService"=hex(7):52,50,43,53,53,00,4e,54,4c,4d,53,53,50,00,00
"DependOnGroup"=hex(7):00
"ObjectName"="LocalSystem"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Parameters]
"MajorVersion"=dword:00000002
"MinorVersion"=dword:00000000
"AdminName"="Administrator"
"AdminEmail"="Admin@corp.com"
"MaxConnections"=dword:000186a0
"LogType"=dword:00000000
"LogFileDirectory"=hex(2):25,53,79,73,74,65,6d,52,6f,6f,74,25,5c,53,79,73,74,65,6d,6d,6d,33,32,5c,4c,6f,67,46,69,6c,65,73,00
"LogFileTruncateSize"=dword:01400000
"LogFilePeriod"=dword:00000001
"LogFileFormat"=dword:00000000
"LogSqlDataSource"="HTTPLOG"
"LogSqlTableName"="Internetlog"
"LogSqlUserName"="InternetAdmin"
"LogSqlPassword"="sqllog"
"Authorization"=dword:00000005
"AnonymousUserName"="IUSR_CL02"
"Default Load File"="Default.htm"
"Dir Browse Control"=dword:4000001e
"CheckForWAISDB"=dword:00000000
"CacheExtensions"=dword:00000001
"GlobalExpire"=dword:ffffff
"ServerSideIncludesEnabled"=dword:00000001
"ServerSideIncludesExtension"=".stm"
"DebugFlags"=dword:00000008
"ScriptTimeout"=dword:00000384
"ConnectionTimeout"=dword:00000384
"InstallPath"="C:\WINNT\System32\inetnsv"
"SecurePort"=dword:000001bb
"Filter
DLs"="C:\WINNT\System32\inetnsv\lsspifilt.dll"
"AccessDeniedMessage"="エラー: 予知は拒否されました。"
"NTAuthenticationProviders"="NTLM"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Parameters\Script Map]
```

```

".idc"="C:\WINNT\System32\inet\httpodbc.dll"
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Parameters\Virtual Roots]
"/,"="C:\inetpub\wwwroot,1"
"/tpc,"="C:\client\tpap\Release,5"
"/Scripts,"="C:\inetpub\scripts,4"
"/iisadmin,"="C:\WINNT\System32\inet\iisadmin,1"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Performance]
"Library"="w3ctrs.DLL"
"Open"="OpenW3PerformanceData"
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"Last Counter"=dword:00000790
"Last Help"=dword:00000791
"First Counter"=dword:00000758
"First Help"=dword:00000759

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Security]
"Security"=hex:01,00,14,80,c0,00,00,00,cc,00,00,00,14,00,00,00,34,00,00,02,\
00,20,00,01,00,00,02,80,18,00,ff,01,0f,00,01,01,00,00,00,00,01,00,00,\
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,00,00,01,80,8d,01,02,00,01,01,00,\
00,00,00,00,01,00,00,00,68,20,15,00,00,00,1c,00,fd,01,02,00,01,02,00,00,\
00,00,00,05,20,00,00,00,23,02,00,00,00,00,00,00,00,00,1c,00,ff,01,\
02,00,00,00,00,05,20,00,00,00,20,02,00,00,00,00,00,00,00,00,1c,00,ff,01,\
0f,00,01,02,00,00,00,00,05,20,00,00,00,25,02,00,00,00,00,00,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,05,12,00,00,00,25,02,00,00,01,01,00,00,\
00,00,05,12,00,00,00,01,01,00,00,00,00,00,05,12,00,00,00

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\W3SAMP]

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Enum]
"0"="Root\LEGACY_W3SVC\0000"
"Count"=dword:00000001
"NextInstance"=dword:00000001

```

REGEDIT4

```

[HKEY_LOCAL_MACHINE\SOFTWARE\Fujitsu\TPC-C ISAPI Application]
"Term_Base"=dword:19835
"NumWarehouses"=dword:2c88
"MaxUsers"=dword:1bd50
"MaxTerm of Client"=dword:128e
"CONTROL_Flag"=dword:00000000

```

```

#
# ubbconfig : TUXEDO configuration file
#

```

```

*RESOURCES
IPCKEY 80952
MASTER SITE1

```

```

#UID 1
#GID 1
PERM 0660
MAXACCESSERS 800
MAXSERVERS 200
MAXSERVICES 1000
MODEL SHM
LDBAL Y

```

```

*MACHINES
CL23 LMD=SITE1
APPDIR="c:\sv-aplora"
TUXCONFIG="c:\client\tuxconfig"
TUXDIR="c:\tuxedo"
ULOGPFX="c:\tuxlogtest"

```

```

*GROUPS
group1 LMD=SITE1 GRPNO=1

```

```

*SERVERS
DEFAULT: RESTART=Y MAXGEN=5
RQPERM=0660
tpccfml SRVGRP=group1 RQADDR=newq1
SRVID=1 REPLYQ=N CLOPT="-s"
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq2
SRVID=2 REPLYQ=N CLOPT="-s"
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq3
SRVID=3 REPLYQ=N CLOPT="-s"
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq4
SRVID=4 REPLYQ=N CLOPT="-s"
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq5
SRVID=5 REPLYQ=N CLOPT="-s"
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq6
SRVID=6 REPLYQ=N CLOPT="-s"
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq7
SRVID=7 REPLYQ=N CLOPT="-s"
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq8
SRVID=8 REPLYQ=N CLOPT="-s"
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq9
SRVID=9 REPLYQ=N CLOPT="-s"
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq10
SRVID=10 REPLYQ=N CLOPT="-s"
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq11
SRVID=11 REPLYQ=N CLOPT="-s"
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq12
SRVID=12 REPLYQ=N CLOPT="-s"
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq13
SRVID=13 REPLYQ=N CLOPT="-s"
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq14
SRVID=14 REPLYQ=N CLOPT="-s"
NEWORDER:NEWORDER"
tpccfmlp SRVGRP=group1 RQADDR=payq1
SRVID=15 REPLYQ=N CLOPT="-s"
PAYMENT:PAYMENT"

```

```

tpccfmlp SRVGRP=group1 RQADDR=payq2
SRVID=16 REPLYQ=N CLOPT="-s"
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq3
SRVID=17 REPLYQ=N CLOPT="-s"
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq4
SRVID=18 REPLYQ=N CLOPT="-s"
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq5
SRVID=19 REPLYQ=N CLOPT="-s"
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq6
SRVID=20 REPLYQ=N CLOPT="-s"
PAYMENT:PAYMENT"
tpccfmlp SRVGRP=group1 RQADDR=payq7
SRVID=21 REPLYQ=N CLOPT="-s"
PAYMENT:PAYMENT"
tpccfmlo SRVGRP=group1 RQADDR=ordq1
SRVID=22 REPLYQ=N CLOPT="-s"
ORDERSTATUS:ORDERSTATUS"
tpccfmlo SRVGRP=group1 RQADDR=ordq2
SRVID=23 REPLYQ=N CLOPT="-s"
ORDERSTATUS:ORDERSTATUS"
tpccfmld SRVGRP=group1 RQADDR=delq1
SRVID=24 REPLYQ=N CLOPT="-s"
DELIVERY:DELIVERY"
tpccfmld SRVGRP=group1 RQADDR=delq2
SRVID=25 REPLYQ=N CLOPT="-s"
DELIVERY:DELIVERY"
tpccfmld SRVGRP=group1 RQADDR=delq3
SRVID=26 REPLYQ=N CLOPT="-s"
DELIVERY:DELIVERY"
tpccfmfs SRVGRP=group1 RQADDR=stoq1
SRVID=27 REPLYQ=N CLOPT="-s"
STOCKLVL:STOCKLVL"
tpccfmfs SRVGRP=group1 RQADDR=stoq2
SRVID=28 REPLYQ=N CLOPT="-s"
STOCKLVL:STOCKLVL"
tpccfmfs SRVGRP=group1 RQADDR=stoq3
SRVID=29 REPLYQ=N CLOPT="-s"
STOCKLVL:STOCKLVL"

```

```

*SERVICES
"NEWORDER" TRANTIME=0
SRVGRP=group1
"PAYMENT" TRANTIME=0
SRVGRP=group1
"ORDERSTATUS" TRANTIME=0
SRVGRP=group1
"DELIVERY" TRANTIME=0
SRVGRP=group1
"STOCKLVL" TRANTIME=0
SRVGRP=group1

```

*ROUTING

```

===== client 24 =====
REGEDIT4

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Parameters]
"BandwidthLevel"=dword:ffffff
"ListenBackLog"=dword:00000019
"MemoryCacheSize"=dword:00000000
"ObjectCacheTTL"=dword:ffffff
"PoolThreadLimit"=dword:0000020d

```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Parameters\Filter]
"FilterType"=dword:00000000
"NumGrantSites"=dword:00000000
"NumDenySites"=dword:00000000
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Parameters\MimeMap]
"text/html,html,h"=""
"image/gif,gif,g"=""
"image/jpeg,jpg,,"=""
"text/plain,txt,0"=""
"text/html,html,h"=""
"image/jpeg,jpeg,,"=""
"image/jpeg,jpe,,"=""
"image/bmp,bmp,,"=""
"application/octet-stream,*"=""
"application/pdf,pdf,,"=""
"application/octet-stream,bin,,"=""
"application/oda,oda,,"=""
"application/zip,zip,9"=""
"application/rif,rif,,"=""
"application/postscript,ps,5"=""
"application/postscript,ai,5"=""
"application/postscript,eps,5"=""
"application/mac-binhex40,hqx,4"=""
"application/msword,doc,5"=""
"application/msword,dot,5"=""
"application/winhlp,hlp,5"=""
"video/mpeg,mpeg,,"=""
"video/mpeg,mpg,,"=""
"video/mpeg,mpe,,"=""
"video/x-msvideo,avi,<"=""
"video/quicktime,qt,,"=""
"video/quicktime,mov,,"=""
"video/x-sgi-movie,movie,,"=""
"x-world/x-vmrl,wrl,5"=""
"x-world/x-vmrl,xaf,5"=""
"x-world/x-vmrl,xof,5"=""
"x-world/x-vmrl,flr,5"=""
"x-world/x-vmrl,wrz,5"=""
"application/x-director,dcr,5"=""
"application/x-director,dir,5"=""
"application/x-director,dxr,5"=""
"image/cis-cod,cod,5"=""
"image/x-cmx,cmx,5"=""
"application/envoy,evy,5"=""
"application/x-msaccess,mdb,5"=""
"application/x-mscardfile,crd,5"=""
"application/x-msclip,clip,5"=""
"application/octet-stream,exe,5"=""
"application/x-msexcel,xla,5"=""
"application/x-msexcel,xlc,5"=""
"application/x-msexcel,xlm,5"=""
"application/x-msexcel,xls,5"=""
"application/x-msexcel,xlt,5"=""
"application/x-msexcel,xlw,5"=""
"application/x-msmediaview,m13,5"=""
"application/x-msmediaview,m14,5"=""
"application/x-msmoney,mny,5"=""
"application/x-mspowerpoint,ppt,5"=""
"application/x-msproject,mpp,5"=""
"application/x-mspublisher,pub,5"=""
"application/x-msterminal,tm,5"=""
"application/x-msworks,wks,5"=""
"application/x-mswrite,wri,5"=""
"application/x-msmetafile,wmf,5"=""
"application/x-csh,csh,5"=""
"application/x-dvi,dvi,5"=""
```

```
"application/x-hdf,hdf,5"=""
"application/x-latex,latex,5"=""
"application/x-netcdf,nc,5"=""
"application/x-netcdf,cdf,5"=""
"application/x-sh,sh,5"=""
"application/x-tcl,tcl,5"=""
"application/x-tex,tex,5"=""
"application/x-texinfo,texinfo,5"=""
"application/x-texinfo,texi,5"=""
"application/x-troff,t,5"=""
"application/x-troff,tr,5"=""
"application/x-troff,roff,5"=""
"application/x-troff-man,man,5"=""
"application/x-troff-me,me,5"=""
"application/x-troff-ms,ms,5"=""
"application/x-wais-source,src,7"=""
"application/x-bcpio,bcpio,5"=""
"application/x-cpio,cpio,5"=""
"application/x-gtar,gtar,9"=""
"application/x-shar,shar,5"=""
"application/x-sv4cpio,sv4cpio,5"=""
"application/x-sv4crc,sv4crc,5"=""
"application/x-tar,tar,5"=""
"application/x-ustar,ustar,5"=""
"audio/basic,au,<"=""
"audio/basic,snd,<"=""
"audio/x-aiff,aiff,<"=""
"audio/x-aiff,aiff,<"=""
"audio/x-aiff,aifc,<"=""
"audio/x-wav,wav,<"=""
"audio/x-pn-realaudio,ram,<"=""
"image/ief,ief,,"=""
"image/tiff,tiff,,"=""
"image/tiff,tif,,"=""
"image/x-cmu-raster,ras,,"=""
"image/x-portable-anymap,pnm,,"=""
"image/x-portable-bitmap,pbm,,"=""
"image/x-portable-graymap,pgm,,"=""
"image/x-portable-pixmap,ppm,,"=""
"image/x-rgb,rgb,,"=""
"image/x-xbitmap,xbm,,"=""
"image/x-xpixmap,xpm,,"=""
"image/x-xwindowdump,xwd,,"=""
"text/html,stm,h"=""
"text/plain,bas,0"=""
"text/plain,c,0"=""
"text/plain,h,0"=""
"text/richtext,rtx,0"=""
"text/tab-separated-values,tsv,0"=""
"text/x-setext,etx,0"=""
"application/x-perfmon,pmc,5"=""
"application/x-perfmon,pma,5"=""
"application/x-perfmon,pmr,5"=""
"application/x-perfmon,pml,5"=""
"application/x-perfmon,pmw,5"=""
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEASystems\TUXEDO\6.4\Environment]
"TUXDIR"="C:\TUXEDO"
"NLSPATH"="C:\TUXEDO\locale\C"
"ULOGDIR"="C:\TUXEDO"
"ULOGPF"="C:\ULOG"
"ULOGOUT"=dword:00000001
"TUXIPC_MSG_BYTES"=dword:00010000
"TUXIPC_MSG_HDRS"=dword:00001fc0
"TUXIPC_MSG_QUEUE"=dword:00000320
```

```
"TUXIPC_MSG_QUEUE_BYTES"=dword:00010000
"TUXIPC_MSG_SEG_BYTES"=dword:00000040
"TUXIPC_MSG_SEGS"=dword:000007ff
"TUXIPC_PROC"=dword:00001770
"TUXIPC_SEM"=dword:00001770
"TUXIPC_SEM_IDS"=dword:00001770
"TUXIPC_SEM_UNDO"=dword:00001770
"TUXIPC_SHM_PROCS"=dword:000186a0
"TUXIPC_SHM_SEGS"=dword:0000c350
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEASystems\TUXEDO\6.4\Environment\Services]
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\BEASystems\TUXEDO\6.4\Environment\Services\3050]
```

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC]
"Type"=dword:00000020
"Start"=dword:00000002
"ErrorControl"=dword:00000000
"ImagePath"=hex(2):43,3a,5c,57,49,4e,4e,54,5c,53,79,73,74,65,6d,33,32,5c,69,6e,65,74,73,72,76,5c,69,6e,65,74,69,6e,66,6f,2e,65,78,65,00
"DisplayName"="World Wide Web Publishing Service"
"DependOnService"=hex(7):52,50,43,53,53,00,4e,54,4c,4d,53,53,50,00,00
"DependOnGroup"=hex(7):00
"ObjectName"="LocalSystem"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Parameters]
"MajorVersion"=dword:00000002
"MinorVersion"=dword:00000000
"AdminName"="Administrator"
"AdminEmail"="Admin@corp.com"
"MaxConnections"=dword:000186a0
"LogType"=dword:00000000
"LogFileDirectory"=hex(2):25,53,79,73,74,65,6d,52,6f,6f,74,25,5c,53,79,73,74,65,6d,33,32,5c,4c,6f,67,46,69,6c,65,73,00
"LogFileTruncateSize"=dword:01400000
"LogFilePeriod"=dword:00000001
"LogFileFormat"=dword:00000000
"LogSqlDataSource"="HTTLPLOG"
"LogSqlTableName"="Internetlog"
"LogSqlUserName"="InternetAdmin"
"LogSqlPassword"="sqllog"
"Authorization"=dword:00000005
"AnonymousUserName"="IUSR_CL25"
"Default Load File"="Default.htm"
"Dir Browse Control"=dword:4000001e
"CheckForWAISDB"=dword:00000000
"CacheExtensions"=dword:00000001
"GlobalExpire"=dword:ffffff
"ServerSideIncludesEnabled"=dword:00000001
"ServerSideIncludesExtension"=".stm"
"DebugFlags"=dword:00000008
"ScriptTimeout"=dword:00000384
"ConnectionTimeout"=dword:00000384
"InstallPath"="C:\WINNT\System32\inetnsrv"
"SecurePort"=dword:000001bb
```

```

"Filter
DLLs"="C:\WINNT\System32\inet\sspifilt.dll"
"AccessDeniedMessage"="エラー：アクセスは拒否され
ました。"
"NTAuthenticationProviders"="NTLM"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Script Map]
".idc"="C:\WINNT\System32\inet\httpodbc.dll"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Parameters\Virtual Roots]
"/,"="C:\inetPub\wwwroot,1"
"/tpc,"="C:\client\tpap\Release,,5"
"/tpce,"="C:\client\tpap\Release_dll,,5"
"/Scripts,"="C:\inetPub\scripts,,4"
"/isadmin,"="C:\WINNT\System32\inet\isad
min,,1"

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Performance]
"Library"="w3ctrs.DLL"
"Open"="OpenW3PerformanceData"
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"Last Counter"=dword:00000790
"Last Help"=dword:00000791
"First Counter"=dword:00000758
"First Help"=dword:00000759

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Security]
"Security"=hex:01,00,14,80,c0,00,00,00,cc,00,00,
00,14,00,00,00,34,00,00,02,\
00,20,00,01,00,00,02,80,18,00,ff,01,0f,00,01,
01,00,00,00,00,01,00,00,\
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,0,
0,18,00,8d,01,02,00,01,01,00,\
00,00,00,01,00,00,00,00,00,00,00,00,00,00,1
c,00,fd,01,02,00,01,02,00,00,\
00,00,00,05,20,00,00,23,02,00,00,00,00,0,
0,00,00,1c,00,ff,01,0f,00,01,\
02,00,00,00,00,05,20,00,00,00,20,02,00,0,
0,00,00,00,00,00,1c,00,ff,01,\
0f,00,01,02,00,00,00,00,05,20,00,00,00,25,0
2,00,00,00,00,00,00,00,18,\
00,fd,01,02,00,01,01,00,00,00,00,00,05,12,0,0
0,00,25,02,00,00,01,01,00,00,\
00,00,00,05,12,00,00,00,01,01,00,00,00,0,0
5,12,00,00,00

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\W3SAMP]

```

```

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentCont
rolSet\Services\W3SVC\Enum]
"0"="Root\LEGACY_W3SVC\0000"
"Count"=dword:00000001
"NextInstance"=dword:00000001

```

REGEDIT4

```

[HKEY_LOCAL_MACHINE\SOFTWARE\Fujitsu\T
PC-C ISAPI Application]
"Term_Base"=dword:1aac3
"NumWarehouses"=dword:2c88
"MaxUsers"=dword:1bd50
"MaxTerm of Client"=dword:128e

```

```
"CONTROL_Flag"=dword:00000000
```

```
#
# ubbconfig : TUXEDO configuration file
#
```

```

*RESOURCES
IPCKEY 80952
MASTER SITE1
#UID 1
#GID 1
PERM 0660
MAXACCESSERS 800
MAXSERVERS 200
MAXSERVICES 1000
MODEL SHM
LDBAL Y

```

```

*MACHINES
CL24 LMID=SITE1
APDIR="c:\sv-apl\ora"
TUXCONFIG="c:\client\tuxconfig"
TUXDIR="c:\tuxedo"
ULOGPFX="c:\tuxlog\test"

```

```

*GROUPS
group1 LMID=SITE1 GRPNO=1

```

```

*SERVERS
DEFAULT: RESTART=Y MAXGEN=5
RQPERM=0660
tpccfml SRVGRP=group1 RQADDR=newq1
SRVID=1 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq2
SRVID=2 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq3
SRVID=3 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq4
SRVID=4 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq5
SRVID=5 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq6
SRVID=6 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq7
SRVID=7 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq8
SRVID=8 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq9
SRVID=9 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq10
SRVID=10 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq11
SRVID=11 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq12
SRVID=12 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"

```

```

tpccfml SRVGRP=group1 RQADDR=newq13
SRVID=13 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=newq14
SRVID=14 REPLYQ=N CLOPT="-s
NEWORDER:NEWORDER"
tpccfml SRVGRP=group1 RQADDR=payq1
SRVID=15 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=payq2
SRVID=16 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=payq3
SRVID=17 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=payq4
SRVID=18 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=payq5
SRVID=19 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=payq6
SRVID=20 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfml SRVGRP=group1 RQADDR=payq7
SRVID=21 REPLYQ=N CLOPT="-s
PAYMENT:PAYMENT"
tpccfmlo SRVGRP=group1 RQADDR=ordq1
SRVID=22 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfmlo SRVGRP=group1 RQADDR=ordq2
SRVID=23 REPLYQ=N CLOPT="-s
ORDERSTATUS:ORDERSTATUS"
tpccfmld SRVGRP=group1 RQADDR=delq1
SRVID=24 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmld SRVGRP=group1 RQADDR=delq2
SRVID=25 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmld SRVGRP=group1 RQADDR=delq3
SRVID=26 REPLYQ=N CLOPT="-s
DELIVERY:DELIVERY"
tpccfmfs SRVGRP=group1 RQADDR=stoq1
SRVID=27 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfmfs SRVGRP=group1 RQADDR=stoq2
SRVID=28 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"
tpccfmfs SRVGRP=group1 RQADDR=stoq3
SRVID=29 REPLYQ=N CLOPT="-s
STOCKLVL:STOCKLVL"

```

```

*SERVICES
"NEWORDER" TRANTIME=0
SRVGRP=group1
"PAYMENT" TRANTIME=0
SRVGRP=group1
"ORDERSTATUS" TRANTIME=0
SRVGRP=group1
"DELIVERY" TRANTIME=0
SRVGRP=group1
"STOCKLVL" TRANTIME=0
SRVGRP=group1

```

```
*ROUTING
```

Appendix E: Database Creation Code

===== driver.sh =====

#!/sh

```
STEP=1
START=0
END=0
CONTINUE=1
PROGRAM="driver.sh"
STOPFILE="stop"
TRACEFILE="log/trace.log"
JUNKFILE="junk"
```

usage()

```
{
  echo ""
  echo "Usage: $PROGRAM [<startstepno> [<stopstepno>] [<startstepno>] [-step <stepno>]"
  echo "  [<startstepno> <stopstepno>] - allows user to run a specified"
  echo "  range of steps."
  echo "  [<startstepno>] - runs from"
  echo "  step number <start> till"
  echo "  the end of the script."
  echo "  [-step <stepno>] - runs only"
  echo "  step number <stepno> and"
  echo "  then stops."
  echo ""
  echo "  STEP  FUNCTION"
  echo "-----"
```

```
  echo "  1  Create Database."
  echo "  2  Create Rollback Segments"
```

for Build."

```
  echo "  3  Shutdown Database."
  echo "  4  Startup Database for Build"
  echo "  5  Create User TPCC."
  echo "  6  Create Tablespaces."
  echo "  7  Assign Temporary
```

Tablespace to user TPCC."

```
  echo "  8  Create Data Dictionary
```

Views."

```
  echo "  9  Create Warehouse."
  echo " 10  Create District."
  echo " 11  Create Customer."
  echo " 12  Create History."
  echo " 13  Create Order."
  echo " 14  Create Neworder."
  echo " 15  Create Orderline."
  echo " 16  Create Stock."
  echo " 17  Create Item."
  echo " 18  Load Warehouse."
  echo " 19  Load District."
  echo " 20  Load Item."
  echo " 21  Load History"
  echo " 22  Load Neworder"
  echo " 23  Load Order/Orderline"
  echo " 24  Load Customer"
```

```
  echo " 25  Load Stock"
  echo " 26  Create Warehouse Index."
  echo " 27  Create District Index."
  echo " 28  Create Item Index."
  echo " 29  Create Customer1 Index."
  echo " 30  Create Customer2 Index."
  echo " 31  Create Stock Index."
  echo " 32  Create Orders1 Index."
  echo " 33  Create Orders2 Index."
  echo " 34  Create Neworder Index."
  echo " 35  Create Orderline Index."
  echo " 36  Analyze
```

Tables/Clusters/Indexes."

```
  echo " 37  Create Statistics Tables."
  echo " 38  Load Stored Procedures."
  echo " 39  Create Rollback Segments
```

for Runs."

```
  echo " 40  Generate Space Report."
  echo " 41  Misc."
  echo " 42  Offline Rollback segments
```

for Build."

```
  echo "-----"
```

exit 1;

}

torun()

```
{
  mv -f *.log* log > junk 2>&1
  rm -f $JUNKFILE*
```

if test -f \$STOPFILE

then

echo "An error has occurred, please look into the \$TRACEFILE file."

echo "OR you need to remove the 'stop' file found in the current directory."

echo "The 'stop' file need to be removed after an error occurs and before resuming."

exit 1;

fi

if test \$STEP -ge \$START

then

if test \$STEP -le \$END

then

STEP=`expr \$STEP + 1`

return 0

else

if test \$CONTINUE -eq 0

then

STEP=`expr \$STEP + 1`

return 0

fi

fi

STEP=`expr \$STEP + 1`

return 1

}

case \$# in

0) usage;

;;

1) case \$1 in

-h) usage

;;

[0-9]*) START=\$1

CONTINUE=0

;;

```
*) usage
  ;;
esac
;;
2) case $1 in
  -step) shift
        case $1 in
          [0-9]*) ;;
          *) usage
                ;;
        esac
        START=$1
        END=$1
        CONTINUE=1
        ;;
  [0-9]*) START=$1
        shift
        case $1 in
          [0-9]*) ;;
          *) usage
                ;;
        esac
        END=$1
        CONTINUE=1
        ;;
*) usage
  ;;
esac
```

esac

*) usage

;;

esac

if torun

then

echo "Creating the Database ..."

echo "Creating the Database ..." `date` "\n" >>

\$TRACEFILE

./step2_tpcc.sh

if test \$? -ne 0

then

echo "Creating the Database failed." `date` "\n"

>> \$TRACEFILE

echo "Look at log/step2createdb.log for more

details." `date` "\n" >> \$TRACEFILE

echo "Stopped" >> \$STOPFILE

echo "Creating the Database failed ..."

else

echo "Creating the Database done." `date` "\n"

>> \$TRACEFILE

echo "Creating the Database done ..."

fi

fi

if torun

then

echo "Creating the Rollback Segments ..."

echo "Creating the Rollback Segments ..." `date`

"\n" >> \$TRACEFILE

./step3creatorollback.sh

if test \$? -ne 0

then

echo "Creating the Rollback Segments failed."

`date` "\n" >> \$TRACEFILE

echo "Look at log/step3creatorollback.log for

more details." `date` "\n" >> \$TRACEFILE

echo "Stopped" >> \$STOPFILE

echo "Creating the Rollback Segments

failed ..."

```

else
  echo "Creating the Rollback Segments done."
`date` "\n" >> $TRACEFILE
  echo "Creating the Rollback Segments done ..."
fi
fi

if torun
then
  echo "Shutting down the Database ..."
  echo "Shutting down the Database ..." `date` "\n"
>> $TRACEFILE
  ./stepshut.sh
  if test $? -ne 0
  then
    echo "Shutting down the Database failed."
`date` "\n" >> $TRACEFILE
    echo "Look at log/stepshut.log for more details."
`date` "\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Shutting down the Database failed ..."
  else
    echo "Shutting down the Database done."
`date` "\n" >> $TRACEFILE
    echo "Shutting down the Database done ..."
  fi
fi

if torun
then
  echo "Start up Database for Build ..."
  echo "Start up Database for Build ..." `date` "\n"
>> $TRACEFILE
  ./stepstartb.sh
  if test $? -ne 0
  then
    echo "Start up Database for Build failed." `date`
"\n" >> $TRACEFILE
    echo "Look at log/stepstartb.log for more
details." `date` "\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Start up Database for Build failed ..."
  else
    echo "Start up Database for Build done." `date`
"\n" >> $TRACEFILE
    echo "Start up Database for Build done ..."
  fi
fi

if torun
then
  echo "Create User TPCC ..."
  echo "Create User TPCC ..." `date` "\n" >>
$TRACEFILE
  ./stepcreateuser.sh
  if test $? -ne 0
  then
    echo "Create User TPCC failed." `date` "\n" >>
$TRACEFILE
    echo "Look at log/stepcreateuser.log for more
details." `date` "\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create User TPCC failed ..."
  else
    echo "Create User TPCC done." `date` "\n" >>
$TRACEFILE
    echo "Create User TPCC done ..."
  fi
fi

```

```

if torun
then
  echo "Create Tablespaces ..."
  echo "Create Tablespaces ..." `date` "\n" >>
$TRACEFILE
  ./step5_tpcc.sh
  if test $? -ne 0
  then
    echo "Create Tablespaces failed." `date` "\n" >>
$TRACEFILE
    echo "Look at log/step5createts.log for more
details." `date` "\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create Tablespaces failed ..."
  else
    echo "Create Tablespaces done." `date` "\n" >>
$TRACEFILE
    echo "Create Tablespaces done ..."
  fi
fi

if torun
then
  echo "Assign Temporary Tablespace to user
TPCC ..."
  echo "Assign Temporary Tablespace to user
TPCC ..." `date` "\n" >> $TRACEFILE
  ./stepusertemp.sh
  if test $? -ne 0
  then
    echo "Assign Temporary Tablespace to user
TPCC failed." `date` "\n" >> $TRACEFILE
    echo "Look at log/stepusertemp.log for more
details." `date` "\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Assign Temporary Tablespace to user
TPCC failed ..."
  else
    echo "Assign Temporary Tablespace to user
TPCC done." `date` "\n" >> $TRACEFILE
    echo "Assign Temporary Tablespace to user
TPCC done ..."
  fi
fi

if torun
then
  echo "Create Data Dictionary views ..."
  echo "Create Data Dictionary views ..." `date`
"\n" >> $TRACEFILE
  ./step6_tpcc.sh
  if test $? -ne 0
  then
    echo "Create Data Dictionary views failed."
`date` "\n" >> $TRACEFILE
    echo "Look at log/step6createddviews.log for
more details." `date` "\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create Data Dictionary views failed ..."
  else
    echo "Create Data Dictionary views done."
`date` "\n" >> $TRACEFILE
    echo "Create Data Dictionary views done ..."
  fi
fi

if torun
then

```

```

  echo "Create Warehouse ..."
  echo "Create Warehouse ..." `date` "\n" >>
$TRACEFILE
  ./step9_tpcc.sh
  if test $? -ne 0
  then
    echo "Create Warehouse failed." `date` "\n" >>
$TRACEFILE
    echo "Look at log/step9createware.log for more
details." `date` "\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create Warehouse failed ..."
  else
    echo "Create Warehouse done." `date` "\n" >>
$TRACEFILE
    echo "Create Warehouse done ..."
  fi
fi

if torun
then
  echo "Create District ..."
  echo "Create District ..." `date` "\n" >>
$TRACEFILE
  ./step10_tpcc.sh
  if test $? -ne 0
  then
    echo "Create District failed." `date` "\n" >>
$TRACEFILE
    echo "Look at log/step10createdist.log for more
details." `date` "\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create District failed ..."
  else
    echo "Create District done." `date` "\n" >>
$TRACEFILE
    echo "Create District done ..."
  fi
fi

if torun
then
  echo "Create Customer ..."
  echo "Create Customer ..." `date` "\n" >>
$TRACEFILE
  ./step11_tpcc.sh
  if test $? -ne 0
  then
    echo "Create Customer failed.\n" `date` "\n" >>
$TRACEFILE
    echo "Look at log/step11createcust.log for more
details." `date` "\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create Customer failed ..."
  else
    echo "Create Customer done." `date` "\n" >>
$TRACEFILE
    echo "Create Customer done ..."
  fi
fi

if torun
then
  echo "Create History ..."
  echo "Create History ..." `date` "\n" >>
$TRACEFILE
  ./step12_tpcc.sh
  if test $? -ne 0
  then

```



```

    echo "Create History failed." `date` "\n" >>
$TRACEFILE
    echo "Look at log/step12createhist.log for more
details." `date` "\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create History failed ..."
else
    echo "Create History done." `date` "\n" >>
$TRACEFILE
    echo "Create History done ..."
fi
fi

if torun
then
    echo "Create Order ..."
    echo "Create Order ..." `date` "\n" >>
$TRACEFILE
    ./step13_tpcc.sh
    if test $? -ne 0
    then
        echo "Create Order failed." `date` "\n" >>
$TRACEFILE
        echo "Look at log/step13createordr.log for more
details." `date` "\n" >> $TRACEFILE
        echo "Stopped" >> $STOPFILE
        echo "Create Order failed ..."
    else
        echo "Create Order done." `date` "\n" >>
$TRACEFILE
        echo "Create Order done ..."
    fi
fi

if torun
then
    echo "Create Neworder ..."
    echo "Create Neworder ..." `date` "\n" >>
$TRACEFILE
    ./step14_tpcc.sh
    if test $? -ne 0
    then
        echo "Create Neworder failed." `date` "\n" >>
$TRACEFILE
        echo "Look at log/step14createnord.log for
more details." `date` "\n" >> $TRACEFILE
        echo "Stopped" >> $STOPFILE
        echo "Create Neworder failed ..."
    else
        echo "Create Neworder done." `date` "\n" >>
$TRACEFILE
        echo "Create Neworder done ..."
    fi
fi

if torun
then
    echo "Create Orderline ..."
    echo "Create Orderline ..." `date` "\n" >>
$TRACEFILE
    ./step15_tpcc.sh
    if test $? -ne 0
    then
        echo "Create Orderline failed." `date` "\n" >>
$TRACEFILE
        echo "Look at log/step15createordl.log for more
details." `date` "\n" >> $TRACEFILE
        echo "Stopped" >> $STOPFILE
        echo "Create Orderline failed ..."

```

```

    else
        echo "Create Orderline done." `date` "\n" >>
$TRACEFILE
        echo "Create Orderline done ..."
    fi
fi

if torun
then
    echo "Create Stock ..."
    echo "Create Stock ..." `date` "\n" >>
$TRACEFILE
    ./step16_tpcc.sh
    if test $? -ne 0
    then
        echo "Create Stock failed." `date` "\n" >>
$TRACEFILE
        echo "Look at log/step16createstok.log for more
details." `date` "\n" >> $TRACEFILE
        echo "Stopped" >> $STOPFILE
        echo "Create Stock failed ..."
    else
        echo "Create Stock done." `date` "\n" >>
$TRACEFILE
        echo "Create Stock done ..."
    fi
fi

if torun
then
    echo "Create Item ..."
    echo "Create Item ..." `date` "\n" >>
$TRACEFILE
    ./step17_tpcc.sh
    if test $? -ne 0
    then
        echo "Create Item failed." `date` "\n" >>
$TRACEFILE
        echo "Look at log/step17createitem.log for
more details." `date` "\n" >> $TRACEFILE
        echo "Stopped" >> $STOPFILE
        echo "Create Item failed ..."
    else
        echo "Create Item done." `date` "\n" >>
$TRACEFILE
        echo "Create Item done ..."
    fi
fi

if torun
then
    echo "Load Warehouse ..."
    echo "Load Warehouse ..." `date` "\n" >>
$TRACEFILE
    ./step20_tpcc.sh
    if test $? -ne 0
    then
        echo "Load Warehouse failed." `date` "\n" >>
$TRACEFILE
        echo "Look at log/step20loadware.log for more
details." `date` "\n" >> $TRACEFILE
        echo "Stopped" >> $STOPFILE
        echo "Load Warehouse failed ..."
    else
        echo "Load Warehouse done." `date` "\n" >>
$TRACEFILE
        echo "Load Warehouse done ..."
    fi
fi

```

```

if torun
then
    echo "Load District ..."
    echo "Load District ..." `date` "\n" >>
$TRACEFILE
    ./step21_tpcc.sh
    if test $? -ne 0
    then
        echo "Load District failed." `date` "\n" >>
$TRACEFILE
        echo "Look at log/step21loaddist.log for more
details." `date` "\n" >> $TRACEFILE
        echo "Stopped" >> $STOPFILE
        echo "Load District failed ..."
    else
        echo "Load District done." `date` "\n" >>
$TRACEFILE
        echo "Load District done ..."
    fi
fi

if torun
then
    echo "Load Item ..."
    echo "Load Item ..." `date` "\n" >> $TRACEFILE
    ./step22_tpcc.sh
    if test $? -ne 0
    then
        echo "Load Item failed." `date` "\n" >>
$TRACEFILE
        echo "Look at log/step22loaditem.log for more
details." `date` "\n" >> $TRACEFILE
        echo "Stopped" >> $STOPFILE
        echo "Load Item failed ..."
    else
        echo "Load Item done." `date` "\n" >>
$TRACEFILE
        echo "Load Item done ..."
    fi
fi

if torun
then
    echo "Load History ..."
    echo "Load History ..." `date` "\n" >>
$TRACEFILE
    ./step23_tpcc.sh
    if test $? -ne 0
    then
        echo "Load History failed." `date` "\n" >>
$TRACEFILE
        echo "Look at log/step23loadhist*.log for more
details." `date` "\n" >> $TRACEFILE
        echo "Stopped" >> $STOPFILE
        echo "Load History failed ..."
    else
        echo "Load History done." `date` "\n" >>
$TRACEFILE
        echo "Load History done ..."
    fi
fi

if torun
then
    echo "Load Neworder ..."
    echo "Load Neworder ..." `date` "\n" >>
$TRACEFILE
    ./step24_tpcc.sh

```

```

if test $? -ne 0
then
  echo "Load Neworder failed." `date` "\n" >>
$TRACEFILE
  echo "Look at log/step24loadnord*.log for more
details." `date` "\n" >> $TRACEFILE
  echo "Stopped" >> $STOPFILE
  echo "Load Neworder failed ..."
else
  echo "Load Neworder done." `date` "\n" >>
$TRACEFILE
  echo "Load Neworder done ..."
fi
fi

if torun
then
  echo "Load Order/Orderline ..."
  echo "Load Order/Orderline ..." `date` "\n" >>
$TRACEFILE
  ./step25_tpcc.sh
  if test $? -ne 0
  then
    echo "Load Order/Orderline failed." `date` "\n"
>> $TRACEFILE
    echo "Look at log/step25loadordrordl*.log for
more details." `date` "\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Load Order/Orderline failed ..."
  else
    echo "Load Order/Orderline done." `date` "\n"
>> $TRACEFILE
    echo "Load Order/Orderline done ..."
  fi
fi

if torun
then
  echo "Load Customer ..."
  echo "Load Customer ..." `date` "\n" >>
$TRACEFILE
  ./step26_tpcc.sh
  if test $? -ne 0
  then
    echo "Load Customer failed." `date` "\n" >>
$TRACEFILE
    echo "Look at log/step26loadcust*.log for more
details." `date` "\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Load Customer failed ..."
  else
    echo "Load Customer done." `date` "\n" >>
$TRACEFILE
    echo "Load Customer done ..."
  fi
fi

if torun
then
  echo "Load Stock ..."
  echo "Load Stock ..." `date` "\n" >>
$TRACEFILE
  ./step27_tpcc.sh
  if test $? -ne 0
  then
    echo "Load Stock failed." `date` "\n" >>
$TRACEFILE
    echo "Look at log/step27loadstok*.log for more
details." `date` "\n" >> $TRACEFILE

```

```

    echo "Stopped" >> $STOPFILE
    echo "Load Stock failed ..."
  else
    echo "Load Stock done." `date` "\n" >>
$TRACEFILE
    echo "Load Stock done ..."
  fi
fi

if torun
then
  echo "Create Warehouse Index ..."
  echo "Create Warehouse Index ..." `date` "\n" >>
$TRACEFILE
  ./step29_tpcc.sh
  if test $? -ne 0
  then
    echo "Create Warehouse Index failed." `date`
"\n" >> $TRACEFILE
    echo "Look at log/step29createiware.log for
more details." `date` "\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create Warehouse Index failed ..."
  else
    echo "Create Warehouse Index done." `date`
"\n" >> $TRACEFILE
    echo "Create Warehouse Index done ..."
  fi
fi

if torun
then
  echo "Create District Index ..."
  echo "Create District Index ..." `date` "\n" >>
$TRACEFILE
  ./step30_tpcc.sh
  if test $? -ne 0
  then
    echo "Create District Index failed." `date` "\n"
>> $TRACEFILE
    echo "Look at log/step30createidist.log for more
details." `date` "\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create District Index failed ..."
  else
    echo "Create District Index done." `date` "\n" >>
$TRACEFILE
    echo "Create District Index done ..."
  fi
fi

if torun
then
  echo "Create Item Index ..."
  echo "Create Item Index ..." `date` "\n" >>
$TRACEFILE
  ./step31_tpcc.sh
  if test $? -ne 0
  then
    echo "Create Item Index failed." `date` "\n" >>
$TRACEFILE
    echo "Look at log/step31createiitem.log for
more details." `date` "\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create Item Index failed ..."
  else
    echo "Create Item Index done." `date` "\n" >>
$TRACEFILE
    echo "Create Item Index done ..."

```

```

  fi
fi

if torun
then
  echo "Create Customer1 Index ..."
  echo "Create Customer1 Index ..." `date` "\n" >>
$TRACEFILE
  ./step32_tpcc.sh
  if test $? -ne 0
  then
    echo "Create Customer1 Index failed." `date`
"\n" >> $TRACEFILE
    echo "Look at log/step32createicust1.log for
more details." `date` "\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create Customer1 Index failed ..."
  else
    echo "Create Customer1 Index done." `date`
"\n" >> $TRACEFILE
    echo "Create Customer1 Index done ..."
  fi
fi

if torun
then
  echo "Create Customer2 Index ..."
  echo "Create Customer2 Index ..." `date` "\n" >>
$TRACEFILE
  ./step33_tpcc.sh
  if test $? -ne 0
  then
    echo "Create Customer2 Index failed." `date`
"\n" >> $TRACEFILE
    echo "Look at log/step33createicust2.log for
more details." `date` "\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create Customer2 Index failed ..."
  else
    echo "Create Customer2 Index done." `date`
"\n" >> $TRACEFILE
    echo "Create Customer2 Index done ..."
  fi
fi

if torun
then
  echo "Create Stock Index ..."
  echo "Create Stock Index ..." `date` "\n" >>
$TRACEFILE
  ./step34_tpcc.sh
  if test $? -ne 0
  then
    echo "Create Stock Index failed." `date` "\n" >>
$TRACEFILE
    echo "Look at log/step34createistok.log for
more details." `date` "\n" >> $TRACEFILE
    echo "Stopped" >> $STOPFILE
    echo "Create Stock Index failed ..."
  else
    echo "Create Stock Index done." `date` "\n" >>
$TRACEFILE
    echo "Create Stock Index done ..."
  fi
fi

if torun
then
  echo "Create Order1 Index ..."

```

```

echo "Create Order1 Index ..." `date` "\n" >>
$TRACEFILE
./step35_tpcc.sh
if test $? -ne 0
then
echo "Create Order1 Index failed." `date` "\n"
>> $TRACEFILE
echo "Look at log/step35createiordr1.log for
more details." `date` "\n" >> $TRACEFILE
echo "Stopped" >> $STOPFILE
echo "Create Order1 Index failed ..."
else
echo "Create Order1 Index done." `date` "\n"
>> $TRACEFILE
echo "Create Order1 Index done ..."
fi
fi

if torun
then
echo "Create Order2 Index ..."
echo "Create Order2 Index ..." `date` "\n" >>
$TRACEFILE
./step36_tpcc.sh
if test $? -ne 0
then
echo "Create Order2 Index failed." `date` "\n"
>> $TRACEFILE
echo "Look at log/step36createiordr2.log for
more details." `date` "\n" >> $TRACEFILE
echo "Stopped" >> $STOPFILE
echo "Create Order2 Index failed ..."
else
echo "Create Order2 Index done." `date` "\n"
>> $TRACEFILE
echo "Create Order2 Index done ..."
fi
fi

if torun
then
echo "Create Neworder Index ..."
echo "Create Neworder Index ..." `date` "\n" >>
$TRACEFILE
./step37_tpcc.sh
if test $? -ne 0
then
echo "Create Neworder Index failed." `date` "\n"
>> $TRACEFILE
echo "Look at log/step37createinord.log for
more details." `date` "\n" >> $TRACEFILE
echo "Stopped" >> $STOPFILE
echo "Create Neworder Index failed ..."
else
echo "Create Neworder Index done." `date` "\n"
>> $TRACEFILE
echo "Create Neworder Index done ..."
fi
fi

if torun
then
echo "Create Orderline Index ..."
echo "Create Orderline Index ..." `date` "\n" >>
$TRACEFILE
./step38_tpcc.sh
if test $? -ne 0
then

```

```

echo "Create Orderline Index failed." `date` "\n"
>> $TRACEFILE
echo "Look at log/step38createiordl.log for
more details." `date` "\n" >> $TRACEFILE
echo "Stopped" >> $STOPFILE
echo "Create Orderline Index failed ..."
else
echo "Create Orderline Index done." `date` "\n"
>> $TRACEFILE
echo "Create Orderline Index done ..."
fi
fi

if torun
then
echo "Analyze Tables/Clusters/Indexes ..."
echo "Analyze Tables/Clusters/Indexes ..." `date`
"\n" >> $TRACEFILE
./step39_tpcc.sh
if test $? -ne 0
then
echo "Analyze Tables/Clusters/Indexes failed."
`date` "\n" >> $TRACEFILE
echo "Look at log/step39analyze.log for more
details." `date` "\n" >> $TRACEFILE
echo "Stopped" >> $STOPFILE
echo "Analyze Tables/Clusters/Indexes
failed ..."
else
echo "Analyze Tables/Clusters/Indexes done."
`date` "\n" >> $TRACEFILE
echo "Analyze Tables/Clusters/Indexes
done ..."
fi
fi

if torun
then
echo "Create Statistics Tables ..."
echo "Create Statistics Tables ..." `date` "\n" >>
$TRACEFILE
./step40_tpcc.sh
if test $? -ne 0
then
echo "Create Statistics Tables failed." `date`
"\n" >> $TRACEFILE
echo "Look at log/step40createsstats.log for
more details." `date` "\n" >> $TRACEFILE
echo "Stopped" >> $STOPFILE
echo "Create Statistics Tables failed ..."
else
echo "Create Statistics Tables done." `date` "\n"
>> $TRACEFILE
echo "Create Statistics Tables done ..."
fi
fi

if torun
then
echo "Load Stored Procedures ..."
echo "Load Stored Procedures ..." `date` "\n" >>
$TRACEFILE
./step41_tpcc.sh
if test $? -ne 0
then
echo "Load Stored Procedures failed." `date`
"\n" >> $TRACEFILE
echo "Look at log/step41createstoredprocs.log
for more details." `date` "\n" >> $TRACEFILE

```

```

echo "Stopped" >> $STOPFILE
echo "Load Stored Procedures failed ..."
else
echo "Load Stored Procedures done." `date`
"\n" >> $TRACEFILE
echo "Load Stored Procedures done ..."
fi
fi

if torun
then
echo "Create Rollback Segments for Runs ..."
echo "Create Rollback Segments for Runs ..."
`date` "\n" >> $TRACEFILE
./step18_tpcc.sh
if test $? -ne 0
then
echo "Create Rollback Segments for Runs
failed." `date` "\n" >> $TRACEFILE
echo "Look at log/step18createrollsegs.log for
more details." `date` "\n" >> $TRACEFILE
echo "Stopped" >> $STOPFILE
echo "Create Rollback Segments for Runs
failed ..."
else
echo "Create Rollback Segments for Runs
done." `date` "\n" >> $TRACEFILE
echo "Create Rollback Segments for Runs
done ..."
fi
fi

if torun
then
echo "Generate Space Reports ..."
echo "Generate Space Reports ..." `date` "\n" >>
$TRACEFILE
./step42_tpcc.sh
if test $? -ne 0
then
echo "Generate Space Reports failed." `date`
"\n" >> $TRACEFILE
echo "Look at log/step42createspacestats.log
for more details." `date` "\n" >> $TRACEFILE
echo "Stopped" >> $STOPFILE
echo "Generate Space Reports failed ..."
else
echo "Generate Space Reports done." `date`
"\n" >> $TRACEFILE
echo "Generate Space Reports done ..."
fi
fi

if torun
then
echo "Misc ..."
echo "Misc ..." `date` "\n" >> $TRACEFILE
./step43_tpcc.sh
if test $? -ne 0
then
echo "Misc failed." `date` "\n" >> $TRACEFILE
echo "Look at log/step43createmisc.log for
more details." `date` "\n" >> $TRACEFILE
echo "Stopped" >> $STOPFILE
echo "Misc failed ..."
else
echo "Misc done." `date` "\n" >> $TRACEFILE
echo "Misc done ..."
fi
fi

```

```

fi
if torun
then
echo "Offline Rollback Segments for Build ..."
echo "Offline Rollback Segments for Build ..."
`date` "\n" >> $TRACEFILE
./step47_tpcc.sh
if test $? -ne 0
then
echo "Offline Rollback Segments for Build
failed." `date` "\n" >> $TRACEFILE
echo "Look at log/step47offlinerollsegs.log for
more details." `date` "\n" >> $TRACEFILE
echo "Stopped" >> $STOPFILE
echo "Offline Rollback Segments for Build
failed ..."
else
echo "Offline Rollback Segments for Build
done." `date` "\n" >> $TRACEFILE
echo "Offline Rollback Segments for Build
done ..."
fi
fi

torun

===== stepshut.sh =====

#!sh

$SEQPLUS internal/internal @stepshut > junk
2>&1

if test $? -ne 0
then
exit 1;
else
exit 0;
fi

===== stepshut.sql =====

spool stepshut.log;

set echo on;

alter system switch logfile;
alter system switch logfile;

shutdown immediate;

set echo off;
spool off;

exit sql.sqlcode;

===== stepstartb.sh =====

#!sh

$SEQPLUS internal/internal @stepstartb > junk
2>&1

```

```

if test $? -eq 0
then
exit 1;
else
exit 0;
fi

===== stepstartb.sql =====

spool stepstartb.log;

set echo on;

startup pfile=p_build2.ora open;

set echo off;
spool off;

exit sql.sqlcode;

===== p_build2.ora =====

compatible = 8.1.5.0.0
db_name = tpcc
control_files = /oracle/dbs/control_001
sort_area_size = 104857
#parallel_max_servers = 100
#recovery_parallelism = 40
db_files = 1024
db_writer_processes = 4
db_block_buffers = 800000
#db_block_buffers = 1000000
#db_block_buffers = 2048000
#db_block_buffers = 3072000
#db_block_buffers = 2000
dml_locks = 100
#log_buffer = 1048576
log_buffer = 10485
max_rollback_segments = 40
#processes = 1196
processes = 200
sessions = 200
transactions = 200
shared_pool_size = 150000
cursor_space_for_time = TRUE
rollback_segments =
(t1,t2,t3,t4,t5,t6,t7,t8,t9,t10,t11,t12,t13,t14,t15,t16,t
17,t18,t19,t20,t21,t22,t23,t24,t25,t26,t27,t28,t29,t3
0)
db_block_size = 2048

===== p_create.ora =====

compatible = 8.1.5.0.0
db_name = tpcc
control_files = /oracle/dbs/control_001
#db_files = 1146
db_files = 1000
db_block_buffers = 2000
dml_locks = 500
#log_buffer = 1048576
log_buffer = 10485

```

```

#processes = 1196
processes = 100
db_block_size = 2048
max_rollback_segments = 900
db_writer_processes = 10

===== addfile.sh =====

sqlplus tpcc/tpcc <<!
spool step5addfile_$1.log
set echo on
alter tablespace $1 add datafile '$2' size $3
reuse;
set echo off
spool off
exit sql.sqlcode;
!

===== addtempfile.sh =====

#!sh

$SEQPLUS tpcc/tpcc <<!
spool step5addfile_$1.log
set echo on
alter tablespace $1 add tempfile '$2' size $3
reuse;
set echo off
spool off
exit sql.sqlcode;
!

===== addtemptps.sh =====

#!sh

$SEQPLUS tpcc/tpcc <<!
spool step5createts_$1.log
set echo on
drop tablespace $1 including contents;
create temporary tablespace $1 tempfile '$2'
size $3 reuse extent management local uniform
size $4;
set echo off
spool off
exit sql.sqlcode;
!

===== addts.sh =====

sqlplus tpcc/tpcc <<!
spool step5createts_$1.log
set echo on
drop tablespace $1 including contents;
create tablespace $1 datafile '$2' size $3 reuse
extent management local uniform size $4
nologging;
set echo off
spool off

```

```

exit sql.sqlcode;
!

===== step10_tpcc.sh =====

$SQLPLUS tpcc/tpcc @crt_tab_dist > junk 2>&1

if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi

===== step10createdist.sql =====

spool step10createdist.log;
set echo on;
drop table dist;

set timing on;

create table dist (
  d_id      number(2,0),
  d_w_id    number(5,0),
  d_ytd     number,
  d_tax     number,
  d_next_o_id number,
  d_name    varchar2(10),
  d_street_1 varchar2(20),
  d_street_2 varchar2(20),
  d_city    varchar2(20),
  d_state   char(2),
  d_zip     char(9)
)
  initrans 4
  pctfree 99
  pctused 0
  storage ( freelists 22 freelist groups 43 )
  tablespace dist_0;

spool off;
set echo off;
exit sql.sqlcode;

===== step11_tpcc.sh =====

sqlplus tpcc/tpcc @step11createcust >> setup.log
#sqlplus tpcc/tpcc @STEP11 >> setup.log

if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi

===== step11createcust.sql =====

spool step11createcust.log;

```

```

set echo on;
drop table cust;
drop cluster custcluster including tables;

set timing on;

create cluster custcluster (
  c_id      number(5,0)
  ,c_d_id   number(2,0)
  ,c_w_id   number(5,0)
)
  single table
  hashkeys 405000000
  hash is (c_id * 135000 + c_w_id * 10 + c_d_id)
  size 850
  initrans 3
  pctfree 0
  storage ( buffer_pool recycle freelists 22 freelist
  groups 43 )
  tablespace cust_0;

create table cust (
  c_id      number(5,0),
  c_d_id    number(2,0),
  c_w_id    number(5,0),
  c_discount number,
  c_credit  char(2),
  c_last    varchar2(16),
  c_first   varchar2(16),
  c_credit_lim number,
  c_balance number,
  c_ytd_payment number,
  c_payment_cnt number,
  c_delivery_cnt number,
  c_street_1 varchar2(20),
  c_street_2 varchar2(20),
  c_city    varchar2(20),
  c_state   char(2),
  c_zip     char(9),
  c_phone   char(16),
  c_since   date,
  c_middle  char(2),
  c_data    varchar2(500)
)
  cluster custcluster (c_id
  ,c_d_id
  ,c_w_id
  );

spool off;
set echo off;
exit sql.sqlcode;

===== step12_tpcc.sh =====

$SQLPLUS tpcc/tpcc @step12createhist > junk
2>&1

if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi

```

```

===== step12createhist.sql =====

spool step12createhist.log;
set echo on;
drop table hist;

set timing on;

create table hist (
  h_c_id     number,
  h_c_d_id   number,
  h_c_w_id   number,
  h_d_id     number,
  h_w_id     number,
  h_date     date,
  h_amount   number,
  h_data     varchar2(24)
)
  initrans 4
  pctfree 5
  storage ( freelists 22 freelist groups 43 )
  tablespace hist_0;

spool off;
set echo off;
exit sql.sqlcode;

===== step13_tpcc.sh =====

sqlplus tpcc/tpcc @step13createordr > junk 2>&1

if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi

===== step13createordr.sql =====

spool step13createordr.log;
set echo on;
drop table ordr;

set timing on;

create table ordr (
  o_id      number,
  o_w_id    number,
  o_d_id    number,
  o_c_id    number,
  o_carrier_id number,
  o_ol_cnt  number,
  o_all_local number,
  o_entry_d date
)
  initrans 4
  pctfree 5
  storage ( freelists 22 freelist groups 43 )
  tablespace ordr_0;

spool off;
set echo off;
exit sql.sqlcode;

```

```

===== step14_tpcc.sh =====

$SQLPLUS tpcc/tpcc @step14createnord > junk
2>&1

if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi

===== step14createnord.sql =====

spool step14createnord.log;
set echo on;
drop table nord;

set timing on;

create table nord (
  no_w_id number,
  no_d_id number,
  no_o_id number,
  constraint inord primary key (no_w_id,
no_d_id, no_o_id)
)
organization index
initrans 4
pctfree 5
storage ( freelists 22 freelist groups 43 )
tablespace nord_0;
spool off;
set echo off;
exit sql.sqlcode;

===== step15_tpcc.sh =====

sqlplus tpcc/tpcc @step15createordl >> setup.log

if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi

===== step15createordl.sql =====

spool step15createordl.log;
set echo on;
drop table ordl;

set timing on;

create table ordl (
  ol_w_id number,
  ol_d_id number,
  ol_o_id number,

```

```

  ol_number number,
  ol_i_id number,
  ol_delivery_d date,
  ol_amount number,
  ol_supply_w_id number,
  ol_quantity number,
  ol_dist_info char(24),
  constraint iordl primary key (ol_w_id,
ol_d_id, ol_o_id, ol_number)
)
organization index
initrans 4
pctfree 5
storage ( freelists 22 freelist groups 43 )
tablespace ordl_0;
spool off;
set echo off;
exit sql.sqlcode;

```

```

===== step16_tpcc.sh =====

sqlplus tpcc/tpcc @step16createstok > step16.log
#sqlplus tpcc/tpcc @STEP16 > step16.log

if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi

```

```

===== step16createstok.sql =====

spool step16createstok.log;
set echo on;
drop table stok;
drop cluster stokcluster including tables;

set timing on;

```

```

create cluster stokcluster (
  s_i_id number(6,0)
, s_w_id number(5,0)
)
single table
hashkeys 1350000000
hash is (s_i_id * 13500 + s_w_id)
size 350
initrans 3
pctfree 0
storage ( buffer_pool keep freelists 22 freelist
groups 43 )
tablespace stock_0;

```

```

create table stok (
  s_i_id number(6,0),
  s_w_id number(5,0),
  s_quantity number,
  s_ytd number,
  s_order_cnt number,
  s_remote_cnt number,
  s_data varchar2(50),
  s_dist_01 char(24),
  s_dist_02 char(24),

```

```

  s_dist_03 char(24),
  s_dist_04 char(24),
  s_dist_05 char(24),
  s_dist_06 char(24),
  s_dist_07 char(24),
  s_dist_08 char(24),
  s_dist_09 char(24),
  s_dist_10 char(24)
)
cluster stokcluster (s_i_id
, s_w_id
);
spool off;
set echo off;
exit sql.sqlcode;

```

```

===== step17_tpcc.sh =====

$SQLPLUS tpcc/tpcc @step17createitem > junk
2>&1

if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi

```

```

===== step17createitem.sql =====

spool step17createitem.log;
set echo on;
drop table item;
drop cluster itemcluster including tables;

```

```

set timing on;

create cluster itemcluster (
  i_id number(6,0)
)
single table
hashkeys 100000
hash is (i_id + 1)
size 120
initrans 3
pctfree 0
storage ( buffer_pool keep freelists 22 freelist
groups 43 )
tablespace item_0;

```

```

create table item (
  i_id number(6,0),
  i_name varchar2(24),
  i_price number,
  i_data varchar2(50),
  i_im_id number
)
cluster itemcluster (i_id
);
spool off;
set echo off;
exit sql.sqlcode;

```

```
===== step18_tpcc.sh =====
```

```
$SEQPLUS tpcc/tpcc @step18_ika > junk 2>&1
```

```
if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi
```

```
===== step18creatorollsegs.sql
=====
```

```
spool step18creatorollsegs.log
set echo on
```

```
alter rollback segment t_0_1 offline;
alter rollback segment t_0_2 offline;
alter rollback segment t_0_3 offline;
alter rollback segment t_0_4 offline;
alter rollback segment t_0_5 offline;
alter rollback segment t_0_6 offline;
alter rollback segment t_0_7 offline;
alter rollback segment t_0_8 offline;
alter rollback segment t_0_9 offline;
alter rollback segment t_0_10 offline;
alter rollback segment t_0_11 offline;
alter rollback segment t_0_12 offline;
alter rollback segment t_0_13 offline;
alter rollback segment t_0_14 offline;
alter rollback segment t_0_15 offline;
alter rollback segment t_0_16 offline;
alter rollback segment t_0_17 offline;
alter rollback segment t_0_18 offline;
alter rollback segment t_0_19 offline;
alter rollback segment t_0_20 offline;
alter rollback segment t_0_21 offline;
alter rollback segment t_0_22 offline;
alter rollback segment t_0_23 offline;
alter rollback segment t_0_24 offline;
```

```
alter rollback segment t_0_25 offline;
alter rollback segment t_0_26 offline;
alter rollback segment t_0_27 offline;
alter rollback segment t_0_28 offline;
alter rollback segment t_0_29 offline;
alter rollback segment t_0_30 offline;
alter rollback segment t_0_31 offline;
alter rollback segment t_0_32 offline;
alter rollback segment t_0_33 offline;
alter rollback segment t_0_34 offline;
alter rollback segment t_0_35 offline;
alter rollback segment t_0_36 offline;
alter rollback segment t_0_37 offline;
alter rollback segment t_0_38 offline;
alter rollback segment t_0_39 offline;
alter rollback segment t_0_40 offline;
alter rollback segment t_0_41 offline;
alter rollback segment t_0_42 offline;
alter rollback segment t_0_43 offline;
alter rollback segment t_0_44 offline;
alter rollback segment t_0_45 offline;
alter rollback segment t_0_46 offline;
alter rollback segment t_0_47 offline;
alter rollback segment t_0_48 offline;
alter rollback segment t_0_49 offline;
alter rollback segment t_0_50 offline;
alter rollback segment t_0_51 offline;
alter rollback segment t_0_52 offline;
alter rollback segment t_0_53 offline;
alter rollback segment t_0_54 offline;
alter rollback segment t_0_55 offline;
alter rollback segment t_0_56 offline;
alter rollback segment t_0_57 offline;
alter rollback segment t_0_58 offline;
```

```
alter rollback segment t_0_59 offline;
alter rollback segment t_0_60 offline;
alter rollback segment t_0_61 offline;
alter rollback segment t_0_62 offline;
alter rollback segment t_0_63 offline;
alter rollback segment t_0_64 offline;
alter rollback segment t_0_65 offline;
alter rollback segment t_0_66 offline;
alter rollback segment t_0_67 offline;
alter rollback segment t_0_68 offline;
alter rollback segment t_0_69 offline;
alter rollback segment t_0_70 offline;
alter rollback segment t_0_71 offline;
alter rollback segment t_0_72 offline;
alter rollback segment t_0_73 offline;
alter rollback segment t_0_74 offline;
alter rollback segment t_0_75 offline;
alter rollback segment t_0_76 offline;
alter rollback segment t_0_77 offline;
alter rollback segment t_0_78 offline;
alter rollback segment t_0_79 offline;
alter rollback segment t_0_80 offline;
alter rollback segment t_0_81 offline;
alter rollback segment t_0_82 offline;
alter rollback segment t_0_83 offline;
alter rollback segment t_0_84 offline;
alter rollback segment t_0_85 offline;
alter rollback segment t_0_86 offline;
alter rollback segment t_0_87 offline;
alter rollback segment t_0_88 offline;
alter rollback segment t_0_89 offline;
alter rollback segment t_0_90 offline;
alter rollback segment t_0_91 offline;
alter rollback segment t_0_92 offline;
alter rollback segment t_0_93 offline;
```



```

create rollback segment t_0_146 tablespace
roll_0;
create rollback segment t_0_147 tablespace
roll_0;
create rollback segment t_0_148 tablespace
roll_0;
create rollback segment t_0_149 tablespace
roll_0;
create rollback segment t_0_150 tablespace
roll_0;
create rollback segment t_0_151 tablespace
roll_0;
create rollback segment t_0_152 tablespace
roll_0;
create rollback segment t_0_153 tablespace
roll_0;
create rollback segment t_0_154 tablespace
roll_0;
create rollback segment t_0_155 tablespace
roll_0;
create rollback segment t_0_156 tablespace
roll_0;
create rollback segment t_0_157 tablespace
roll_0;
create rollback segment t_0_158 tablespace
roll_0;
create rollback segment t_0_159 tablespace
roll_0;
create rollback segment t_0_160 tablespace
roll_0;
create rollback segment t_0_161 tablespace
roll_0;
create rollback segment t_0_162 tablespace
roll_0;
create rollback segment t_0_163 tablespace
roll_0;
create rollback segment t_0_164 tablespace
roll_0;
create rollback segment t_0_165 tablespace
roll_0;
create rollback segment t_0_166 tablespace
roll_0;
create rollback segment t_0_167 tablespace
roll_0;
create rollback segment t_0_168 tablespace
roll_0;
create rollback segment t_0_169 tablespace
roll_0;
create rollback segment t_0_170 tablespace
roll_0;
create rollback segment t_0_171 tablespace
roll_0;
create rollback segment t_0_172 tablespace
roll_0;
create rollback segment t_0_173 tablespace
roll_0;
create rollback segment t_0_174 tablespace
roll_0;
create rollback segment t_0_175 tablespace
roll_0;
create rollback segment t_0_176 tablespace
roll_0;
create rollback segment t_0_177 tablespace
roll_0;
create rollback segment t_0_178 tablespace
roll_0;
create rollback segment t_0_179 tablespace
roll_0;

```

```

create rollback segment t_0_180 tablespace
roll_0;
create rollback segment t_0_181 tablespace
roll_0;
create rollback segment t_0_182 tablespace
roll_0;
create rollback segment t_0_183 tablespace
roll_0;
create rollback segment t_0_184 tablespace
roll_0;
create rollback segment t_0_185 tablespace
roll_0;
create rollback segment t_0_186 tablespace
roll_0;
create rollback segment t_0_187 tablespace
roll_0;
create rollback segment t_0_188 tablespace
roll_0;
create rollback segment t_0_189 tablespace
roll_0;
create rollback segment t_0_190 tablespace
roll_0;
create rollback segment t_0_191 tablespace
roll_0;
create rollback segment t_0_192 tablespace
roll_0;
create rollback segment t_0_193 tablespace
roll_0;
create rollback segment t_0_194 tablespace
roll_0;
create rollback segment t_0_195 tablespace
roll_0;
create rollback segment t_0_196 tablespace
roll_0;
create rollback segment t_0_197 tablespace
roll_0;
create rollback segment t_0_198 tablespace
roll_0;
create rollback segment t_0_199 tablespace
roll_0;
create rollback segment t_0_200 tablespace
roll_0;

```

```

spool off
set echo off
exit sql.sqlcode

```

```

===== step20_tpcc.sh =====

```

```

$TPCCLOAD -M 13500 -w > step20loadware.log
2>&1

```

```

if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi

```

```

===== step21_tpcc.sh =====

```

```

$TPCCLOAD -M 13500 -d > step21loadlist.log
2>&1

```

```

if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi

```

```

===== step22_tpcc.sh =====

```

```

$TPCCLOAD -M 13500 -i > step22loaditem.log
2>&1

```

```

if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi

```

```

===== step23_tpcc.sh =====

```

```

l=1
SW=1
EW=1350
INC=1350
while [ $l -le 10 ]
do

```

```

$TPCCLOAD -M 13500 -h -b $SW -e $EW >
step23loadhist${l}.log0 2>&1 &
l=`expr $l + 1`
SW=`expr $SW + $INC`
EW=`expr $EW + $INC`
done
wait

```

```

===== step24_tpcc.sh =====

```

```

l=1
SW=1
EW=1350
INC=1350
while [ $l -le 10 ]
do
$TPCCLOAD -M 13500 -n -b $SW -e $EW
>step24loadnord${l}.log0 2>&1 &
l=`expr $l + 1`
SW=`expr $SW + $INC`
EW=`expr $EW + $INC`
done
wait

```

```

===== step25_tpcc.sh =====

```

```

l=1
SW=1

```

```

EW=270
INC=270
while [ $1 -le 50 ]
do

$TPCCLOAD -M 13500 -o
/oracle/dbs/dummy0.dat -b $SW -e $EW
>step25loadordrord1.log${!} 2>&1 &
I='expr $1 + 1'
SW='expr $SW + $INC'
EW='expr $EW + $INC'
done
wait

===== step26_tpcc.sh =====

I=1
J=1
SW=1
EW=450
INC=450
while [ $1 -le 30 ]
do

$TPCCLOAD -M 13500 -c -b $SW -e $EW >
step26loadcust${!}.log0 2>&1 &
I='expr $1 + 1'
SW='expr $SW + $INC'
EW='expr $EW + $INC'
done
wait

===== step27_tpcc.sh =====

I=1
SI=1
EI=2500
INC=2500
while [ $1 -le 40 ]
do
$TPCCLOAD -M 13500 -S -j $SI -k $EI
>step27loadstok${!}.log0 2>&1 &
I='expr $1 + 1'
SI='expr $SI + $INC'
EI='expr $EI + $INC'
done
wait

===== step29_tpcc.sh =====

$SQLPLUS tpcc/tpcc @step29createiware >
29.log 2>&1

if test $? -ne 0
then
exit 1;
else
exit 0;
fi

```

```

fi

===== step29createiware.sql =====

spool step29createiware.log;
set echo on;
drop index iware;

set timing on;

create unique index iware on ware (w_id)
initrans 3
parallel 16
pctfree 1
storage ( freelists 22 freelist groups 43 )
tablespace iware_0;
spool off;
set echo off;
exit sql.sqlcode;

===== step2_tpcc.sh =====

$SQLPLUS internal/internal @step2createdb >
junk 2>&1

if test $? -ne 0
then
exit 1;
else
exit 0;
fi

===== step2createdb.sh =====

$SQLPLUS internal/internal @step2createdb >
junk 2>&1

if test $? -ne 0
then
exit 1;
else
exit 0;
fi

===== step2createdb.sql =====

spool step2createdb.log

set echo on

startup pfile=p_create.ora nomount
create database tpcc
controlfile reuse
maxdatafiles 2048
maxinstances 1
datafile '/ora_dev/SYS' size 3000M reuse
logfile '/ora_dev/LOG_1' size 1000M reuse,
'/ora_dev/LOG_2' size 1000M reuse;
spool off

```

```

set echo off
exit sql.sqlcode

===== step30_tpcc.sh =====

$SQLPLUS tpcc/tpcc @step30createidist > junk
2>&1

if test $? -ne 0
then
exit 1;
else
exit 0;
fi

===== step30createidist.sql =====

spool step30createidist.log;
set echo on;
drop index idist;

set timing on;

create unique index idist on dist (d_w_id, d_id)
initrans 3
parallel 16
pctfree 5
storage ( freelists 22 freelist groups 43 )
tablespace idist_0;
spool off;
set echo off;
exit sql.sqlcode;

===== step31_tpcc.sh =====

$SQLPLUS tpcc/tpcc @step31createiitem > junk
2>&1

if test $? -ne 0
then
exit 1;
else
exit 0;
fi

===== step31createiitem.sql =====

spool step31createiitem.log;
set echo on;
drop index iitem;

set timing on;

create unique index iitem on item (i_id)
initrans 4
parallel 16
pctfree 5
storage ( freelists 22 freelist groups 43 )
tablespace item_0;

```

```
spool off;
set echo off;
exit sql.sqlcode;
```

```
===== step32_tpcc.sh =====
```

```
sqlplus tpcc/tpcc @step32createicust1 >
step32.log 2>&1
```

```
if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi
```

```
===== step32createicust1.sql =====
```

```
spool step32createicust1.log;
set echo on;
drop index icust1;
```

```
set timing on;
```

```
create unique index icust1 on cust (c_w_id, c_d_id,
c_id)
```

```
  initrans 3
  parallel 16
  pctfree 1
  storage ( freelists 22 freelist groups 43 )
  tablespace icust1_0;
spool off;
set echo off;
exit sql.sqlcode;
```

```
===== step33_tpcc.sh =====
```

```
sqlplus tpcc/tpcc @step33createicust2 >
step33.log 2>&1
```

```
if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi
```

```
===== step33createicust2.sql =====
```

```
spool step33createicust2.log;
set echo on;
drop index icust2;
```

```
set timing on;
```

```
create unique index icust2 on cust (c_last, c_w_id,
c_d_id, c_first, c_id)
```

```
  initrans 3
  parallel 16
```

```
pctfree 1
storage ( freelists 22 freelist groups 43 )
tablespace icust2_0;
```

```
spool off;
set echo off;
exit sql.sqlcode;
```

```
===== step34_tpcc.sh =====
```

```
sqlplus tpcc/tpcc @step34createistok > step34.log
2>&1
```

```
if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi
```

```
===== step34createistok.sql =====
```

```
spool step34createistok.log;
set echo on;
drop index istok;
```

```
set timing on;
```

```
create unique index istok on stok (s_i_id, s_w_id)
```

```
  initrans 3
  parallel 16
  pctfree 1
  storage ( freelists 22 freelist groups 43 )
  tablespace istok_0;
spool off;
set echo off;
exit sql.sqlcode;
```

```
===== step35_tpcc.sh =====
```

```
sqlplus tpcc/tpcc @step35createiordr1 >
step35.log 2>&1
```

```
if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi
```

```
===== step35createiordr1.sql =====
```

```
spool step35createiordr1.log;
set echo on;
drop index iordr1;
```

```
set timing on;
```

```
create unique index iordr1 on ordr (o_w_id,
o_d_id, o_id)
```

```
initrans 3
parallel 16
pctfree 1
storage ( freelists 22 freelist groups 43 )
tablespace iordr1_0;
```

```
spool off;
set echo off;
exit sql.sqlcode;
```

```
===== step36_tpcc.sh =====
```

```
sqlplus tpcc/tpcc @step36createiordr2 >
step36.log 2>&1
```

```
if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi
```

```
===== step36createiordr2.sql =====
```

```
spool step36createiordr2.log;
set echo on;
drop index iordr2;
```

```
set timing on;
```

```
create unique index iordr2 on ordr (o_w_id,
o_d_id, o_c_id, o_id)
```

```
  initrans 4
  parallel 16
  pctfree 25
  storage ( freelists 22 freelist groups 43 )
  tablespace iordr2_0;
spool off;
set echo off;
exit sql.sqlcode;
```

```
===== step37_tpcc.sh =====
```

```
$$SQLPLUS tpcc/tpcc @step37createinord > junk
2>&1
```

```
if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi
```

```
===== step37createinord.sql =====
```

```
exit sql.sqlcode;
```

```
===== step38_tpcc.sh =====
```

```
sqlplus tpcc/tpcc @step38createiordl > step38.log
2>&1
```

```
if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi
```

```
===== step38createiordl.sql =====
```

```
exit sql.sqlcode;
```

```
===== step39_tpcc.sh =====
```

```
$SEQLPLUS tpcc/tpcc @step39analyze >
step39.log 2>&1
```

```
if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi
```

```
===== step39analyze.sql =====
```

```
spool step39analyze.log;
set echo on;
analyze cluster stokcluster estimate statistics;
analyze index istok estimate statistics;
analyze table ware compute statistics;
analyze cluster itemcluster estimate statistics;
analyze index iordl estimate statistics;
analyze index iordr1 estimate statistics;
analyze index iordr2 estimate statistics;
analyze index inord estimate statistics;
analyze table dist compute statistics;
analyze table ord estimate statistics;
rem analyze table nord estimate statistics;
rem analyze table ordl estimate statistics;
analyze table hist estimate statistics;
analyze index iitem estimate statistics;
analyze index iware compute statistics;
analyze index icust1 estimate statistics;
analyze index icust2 estimate statistics;
analyze index idist compute statistics;
analyze cluster custcluster estimate statistics;
```

```
set echo off;
spool off;
```

```
exit sql.sqlcode;
```

```
===== step3creatorollback.sh =====
```

```
$SEQLPLUS internal/internal
@step3creatorollback > junk 2>&1
```

```
if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi
```

```
===== step3creatorollback.sql
=====
```

```
spool step3creatorollback.log
set echo on
```

```
alter rollback segment t1 offline;
drop public rollback segment t1;
create public rollback segment t1
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t1 online;
alter rollback segment t2 offline;
drop public rollback segment t2;
create public rollback segment t2
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t2 online;
alter rollback segment t3 offline;
drop public rollback segment t3;
create public rollback segment t3
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t3 online;
alter rollback segment t4 offline;
drop public rollback segment t4;
create public rollback segment t4
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t4 online;
alter rollback segment t5 offline;
drop public rollback segment t5;
create public rollback segment t5
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t5 online;
alter rollback segment t6 offline;
drop public rollback segment t6;
create public rollback segment t6
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t6 online;
alter rollback segment t7 offline;
drop public rollback segment t7;
create public rollback segment t7
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t7 online;
alter rollback segment t8 offline;
drop public rollback segment t8;
create public rollback segment t8
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t8 online;
alter rollback segment t9 offline;
drop public rollback segment t9;
create public rollback segment t9
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t9 online;
alter rollback segment t10 offline;
drop public rollback segment t10;
create public rollback segment t10
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t10 online;
alter rollback segment t11 offline;
```

```
drop public rollback segment t11;
create public rollback segment t11
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t11 online;
alter rollback segment t12 offline;
drop public rollback segment t12;
create public rollback segment t12
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t12 online;
alter rollback segment t13 offline;
drop public rollback segment t13;
create public rollback segment t13
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t13 online;
alter rollback segment t14 offline;
drop public rollback segment t14;
create public rollback segment t14
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t14 online;
alter rollback segment t15 offline;
drop public rollback segment t15;
create public rollback segment t15
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t15 online;
alter rollback segment t16 offline;
drop public rollback segment t16;
create public rollback segment t16
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t16 online;
alter rollback segment t17 offline;
drop public rollback segment t17;
create public rollback segment t17
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t17 online;
alter rollback segment t18 offline;
drop public rollback segment t18;
create public rollback segment t18
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t18 online;
alter rollback segment t19 offline;
drop public rollback segment t19;
create public rollback segment t19
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t19 online;
alter rollback segment t20 offline;
drop public rollback segment t20;
create public rollback segment t20
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t20 online;
alter rollback segment t21 offline;
drop public rollback segment t21;
create public rollback segment t21
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t21 online;
alter rollback segment t22 offline;
drop public rollback segment t22;
create public rollback segment t22
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t22 online;
alter rollback segment t23 offline;
drop public rollback segment t23;
create public rollback segment t23
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t23 online;
alter rollback segment t24 offline;
drop public rollback segment t24;
create public rollback segment t24
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t24 online;
```

```

alter rollback segment t25 offline;
drop public rollback segment t25;
create public rollback segment t25
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t25 online;
alter rollback segment t26 offline;
drop public rollback segment t26;
create public rollback segment t26
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t26 online;
alter rollback segment t27 offline;
drop public rollback segment t27;
create public rollback segment t27
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t27 online;
alter rollback segment t28 offline;
drop public rollback segment t28;
create public rollback segment t28
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t28 online;
alter rollback segment t29 offline;
drop public rollback segment t29;
create public rollback segment t29
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t29 online;
alter rollback segment t30 offline;
drop public rollback segment t30;
create public rollback segment t30
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t30 online;
alter rollback segment t31 offline;
drop public rollback segment t31;
create public rollback segment t31
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t31 online;
alter rollback segment t32 offline;
drop public rollback segment t32;
create public rollback segment t32
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t32 online;
alter rollback segment t33 offline;
drop public rollback segment t33;
create public rollback segment t33
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t33 online;
alter rollback segment t34 offline;
drop public rollback segment t34;
create public rollback segment t34
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t34 online;
alter rollback segment t35 offline;
drop public rollback segment t35;
create public rollback segment t35
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t35 online;
alter rollback segment t36 offline;
drop public rollback segment t36;
create public rollback segment t36
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t36 online;
alter rollback segment t37 offline;
drop public rollback segment t37;
create public rollback segment t37
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t37 online;
alter rollback segment t38 offline;
drop public rollback segment t38;
create public rollback segment t38
  storage (initial 200K minextents 2 next 200K);

```

```

alter rollback segment t38 online;
alter rollback segment t39 offline;
drop public rollback segment t39;
create public rollback segment t39
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t39 online;
alter rollback segment t40 offline;
drop public rollback segment t40;
create public rollback segment t40
  storage (initial 200K minextents 2 next 200K);
alter rollback segment t40 online;

```

```

spool off
set echo off
exit sql.sqlcode

```

```

===== step40_tpcc.sh =====

```

```

$SQLPLUS internal/internal @step40createstats
> junk 2>&1

```

```

if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi

```

```

===== step40createstats.sql =====

```

```

spool step40createstats.log
@orst_cre
@c_stat
@pst_c
spool off
exit sql.sqlcode;

```

```

===== step41_tpcc.sh =====

```

```

$SQLPLUS tpcc/tpcc
@step41createstoreprocs > junk 2>&1

```

```

if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi

```

```

===== step41createstoreprocs.sql
=====

```

```

spool step41createstoreprocs.log
@initpay
@initnew
spool off
exit sql.sqlcode;

```

```

===== step42_tpcc.sh =====

```

```

$SQLPLUS internal/internal
@step42createspacestats > junk 2>&1

```

```

if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi

```

```

===== step42createspacestats.sql
=====

```

```

spool step42createspacestats.log
@space_init
@space_get 12 2800
@space_rpt
spool off
exit sql.sqlcode;

```

```

===== step43_tpcc.sh =====

```

```

$SQLPLUS internal/internal @step43createmisc
> junk 2>&1

```

```

if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi

```

```

===== step43createmisc.sql =====

```

```

spool step43createmisc.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;
@pls_sql_mon
@cre_tab
@create_cache_views
@views
@dml
@extent 2048
@freeext 2048
spool off
exit sql.sqlcode;

```

```

===== step47_tpcc.sh =====

```

```

$SQLPLUS internal/internal
@step47offlinerollsegs > junk 2>&1

```

```

if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi

```

===== step5_WID.sh =====

```

sh addts.sh iware_0 /ora_dev/WID_0_1 150M 4M
> crtiware.log &
sh addts.sh idist_0 /ora_dev/WID_0_2 150M 4M >
crtidist.log &
sh addts.sh iitem_0 /ora_dev/WID_0_3 150M 4M
> crtitem.log &

```

```

sh addts.sh ware_0 /ora_dev/WID_0_4 150M 4M
> crtware.log &
sh addts.sh dist_0 /ora_dev/WID_0_9 300M 4M >
crtidist.log &
sh addts.sh dist_1 /ora_dev/WID_8 300M 4M >
crtidist.log &
sh addts.sh item_0 /ora_dev/WID_0_6 150M 4M
> crtitem.log &
wait

```

```

#sh addts.sh roll_0 /ora_dev/roll_0_1 400M 4M >
crtware.log &
wait

```

===== step5_cust.sh =====

```

sh addts.sh cust_0 /ora_dev/cust_0_1 6000M
5990M > crtcust.log

```

```

l=2
while [ $l -le 90 ]
do
  sh addfile.sh cust_0 /ora_dev/cust_0_${l}
6000M >> crtcust.log &
  l=`expr $l + 1`
  if test $l -eq 11
  then
    wait
  fi
  if test $l -eq 21
  then
    wait
  fi
  if test $l -eq 31
  then
    wait
  fi
  if test $l -eq 41
  then
    wait
  fi
  if test $l -eq 51
  then
    wait
  fi
  if test $l -eq 61
  then

```

```

wait
fi
if test $l -eq 71
then
  wait
fi
if test $l -eq 81
then
  wait
fi

```

```

done
wait
./step11_ika.sh

```

===== step5_ind.sh =====

```

sh addts.sh istok_0 /ora_dev/istock_0_1 3412M
40M > crtistok.log &
sh addts.sh icust1_0 /ora_dev/icust1_0_1 2020M
10M > crticust1.log &
sh addts.sh icust2_0 /ora_dev/icust2_0_1 4922M
10M > crticust2.log &
sh addts.sh iordr1_0 /ora_dev/iordr1_0_1 2335M
10M > crtiord1.log &
sh addts.sh iordr2_0 /ora_dev/iordr2_0_1 1765M
10M > crtiord2.log &
wait

```

```

l=2
while [ $l -le 15 ]
do
  sh addfile.sh istok_0 /ora_dev/istock_0_${l}
3412M >> crtistok.log &
  sh addfile.sh icust1_0 /ora_dev/icust1_0_${l}
2020M >> crticust1.log &
  sh addfile.sh iordr1_0 /ora_dev/iordr1_0_${l}
2335M >> crtiordr1.log &
  sh addfile.sh iordr2_0 /ora_dev/iordr2_0_${l}
1765M >> crtiordr2.log &
  wait
  l=`expr $l + 1`
done

```

```

l=2
while [ $l -le 7 ]
do
  sh addfile.sh icust2_0 /ora_dev/icust2_0_${l}
4922M >> crticust2.log &
  l=`expr $l + 1`
done
wait

```

===== step5_ordl.sh =====

```

sh addts.sh hist_0 /ora_dev/hist_0_1 7500M
100M > crthist.log &
sh addts.sh nord_0 /ora_dev/nord_0_1 6070M
100M > crtnord.log &
sh addts.sh ordr_0 /ora_dev/ordr_0_1 5940M
100M > crtordr.log &
sh addts.sh ordl_0 /ora_dev/ordl_0_1 6820M
100M > crtordl.log &

```

```

wait

l=2
while [ $l -le 7 ]
do
  sh addfile.sh hist_0 /ora_dev/hist_0_${l} 7500M
>> crthist.log &
  l=`expr $l + 1`
done

```

```

sh addfile.sh nord_0 /ora_dev/nord_0_4 6070M
>> crtnord.log &
sh addfile.sh nord_0 /ora_dev/nord_0_7 6070M
>> crtnord.log &
#

```

```

l=2
while [ $l -le 9 ]
do
  sh addfile.sh ordr_0 /ora_dev/ordr_0_${l}
5940M >> crtordr.log &
  l=`expr $l + 1`
done
wait

```

```

l=2
while [ $l -le 90 ]
do
  sh addfile.sh ordl_0 /ora_dev/ordl_0_${l} 6820M
>> crtordl.log &
  l=`expr $l + 1`
  if test $l -eq 31
  then
    wait
  fi
  if test $l -eq 61
  then
    wait
  fi
done

```

```

wait
exit

```

===== step5_roll.sh =====

```

#alter tablespace roll_0 add datafile
'ora_dev/roll_0_2' size 5000M reuse;
svrmgrl <<!
connect internal;
create tablespace roll_0 datafile
'ora_dev/roll_0_1' size 5000M reuse extent
management local uniform size 40K nologging ;
alter tablespace roll_0 add datafile
'ora_dev/roll_0_2' size 5000M reuse;
exit;
!

```

===== step5_stock.sh =====

```

sh addts.sh stock_0 /ora_dev/stock_0_1 2840M
2830M > crtstk.log

```

```

l=1

```

```

while [ $1 -le 210 ]
do
  sh addfile.sh stock_0 /ora_dev/stock_0_${!}
2840M >> crtstk.log &
  l=`expr $1 + 1`
  if test $l -eq 41
  then
    wait
  fi
  if test $l -eq 81
  then
    wait
  fi
  if test $l -eq 121
  then
    wait
  fi
  if test $l -eq 161
  then
    wait
  fi
  if test $l -eq 201
  then
    wait
  fi
done
wait
./step6_ika.sh

===== step5_temp.sh =====

sh addtemp.sh temp_0 /ora_dev/temp_0_1
4420M 20M > crttemp.log

### true 70 #####
l=2
while [ $1 -le 47 ]
do
  sh addtempfile.sh temp_0 /ora_dev/temp_0_${!}
4420M >> crttemp.log &
  l=`expr $1 + 1`
  if test $l -eq 26
  then
    wait
  fi
done

wait
exit

===== step6_tpcc.sh =====

sqlplus internal/internal @step6createddviews >
setup.log

===== step6createddviews.sh =====

#lsh
$SQLPLUS internal/internal
@step6createddviews > junk 2>&1

```

```

if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi

===== step6createddviews.sql
=====

spool step6createddviews.log
@$ORACLE_HOME/rdbms/admin/catalog
@$ORACLE_HOME/rdbms/admin/catproc
spool off
exit sql.sqlcode;

===== step9_tpcc.sh =====

$SQLPLUS tpcc/tpcc @crt_tab_ware > junk
2>&1

if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi

===== step9createware.sql =====

spool step9createware.log;
set echo on;
drop table ware;

set timing on;

create table ware(
  w_id      number(5,0),
  w_ytd     number,
  w_tax     number,
  w_name    varchar2(10),
  w_street_1 varchar2(20),
  w_street_2 varchar2(20),
  w_city    varchar2(20),
  w_state   char(2),
  w_zip     char(9)
)
  initrans 2
  pctfree 98
  pctused 0
  storage ( freelists 22 freelist groups 43 )
  tablespace ware_0;
spool off;
set echo off;
exit sql.sqlcode;

===== stepchkpt.sql =====

spool stepchkpt.log;

```

```

set echo on;

alter system switch logfile;
alter system switch logfile;

set echo off;
spool off;

exit sql.sqlcode;

===== stepcreateuser.sh =====

$SQLPLUS internal/internal @stepcreateuser >
junk 2>&1

if test $? -ne 0
then
  exit 1;
else
  exit 0;
fi

===== stepcreateuser.sql =====

spool stepcreateuser.log;

set echo on;

create user tpcc identified by tpcc;

grant dba to tpcc;

set echo off;
spool off;

exit sql.sqlcode;

===== stepstartb.sql =====

spool stepstartb.log;

set echo on;

startup pfile=p_build.ora open;

set echo off;
spool off;

exit sql.sqlcode;

===== stepusertemp.sh =====

$SQLPLUS internal/internal @stepusertemp >
junk 2>&1

if test $? -ne 0
then
  exit 1;

```



```
else  
  exit 0;  
fi
```

```
===== stepusertemp.sql =====
```

```
spool stepusertemp.log;
```

```
set echo on;
```

```
alter user tpcc temporary tablespace temp_0;
```

```
set echo off;
```

```
spool off;
```

```
exit sql.sqlcode;
```

Appendix F: 180 Day Space Calculation

TPM	138,735.06					
Warehouses	13500					
SEGMENT	TYPE	TSPACE	BLOCKS	FIVE_PCT	DAILY_GROW	TOTAL
CUSTCLUSTER	CLUSTER	CUST	205,480,960	10,274,048	0	215,755,008
DCLUSTER	CLUSTER	WARE	135,168	6,758	0	141,926
HIST	TABLE	HIST	12,288,000	0	2,379,992	14,667,992
ICUST1	INDEX	ICUST1	4,843,520	242,176	0	5,085,696
ICUST2	INDEX	ICUST2	10,920,960	546,048	0	11,467,008
IDIST	INDEX	WARE	32,768	1,638	0	34,406
IITEM	INDEX	ITEMS	32,768	1,638	0	34,406
INORD	INDEX	INORD	1,075,200	53,760	0	1,128,960
IORDL	INDEX	IORDL	162,252,800	0	26,678,699	188,931,499
IORDR1	INDEX	IORDR1	4,838,400	241,920	0	5,080,320
IORDR2	INDEX	IORDR2	8,002,560	400,128	0	8,402,688
ISTOK	INDEX	ISTK	14,622,720	731,136	0	15,353,856
ITEMCLUSTER	CLUSTER	ITEMS	8,192	410	0	8,602
IWARE	INDEX	WARE	24,576	1,229	0	25,805
ORDR	TABLE	ORD	8,499,200	0	1,397,496	9,896,696
ROLL_SEG	SYS	ROLL	5,120,000	0	0	5,120,000
STOCKCLUSTER	CLUSTER	STOCKS	270,955,520	13,547,776	0	284,503,296
SYSTEM	SYS	SYSTEM	1,536,000	0	0	1,536,000
WCLUSTER	CLUSTER	WARE	14,336	717	0	15,053
Total			710,683,648	26,049,382	30,456,186	767,189,217
Dynamic space		183,040,000				
Static space		553,693,030				
Free space		30,456,186				
Daily growth		30,456,186				
Daily spread		0	Oracle may be configured such that daily spread is 0			
180-day space (blk.)		6,035,806,570				
Block size (bytes)		2,048				
180-day (GB)		11,512.39				
Log block size		512				
Log blocks/tpmC		28.33	Number of log blocks used per New-Order			
8-hour log (GB)		899.59				
		Capacity	NumLun		Total GB	
Database Disks		8	203		8,120.00	
		17.8	11		979.00	
					9,099.00	
		Capacity	NumLun			
8-Hr Log Disks		8	34		1,360.00	
		17.8	7		623.00	
					1,983.00	

Appendix G: Price Quotes

VERITAS

VERITAS Software Corporation
 1600 Plymouth Street
 Mountain View, CA 94043
 U.S.
 Phone 650.335.8000 Fax 650.335.8050

DATE: June 26, 2000
 REF: Q001102

QUOTATION VERITAS Volume Manager

TO: Mr. Shunsho Miyazaki
 Project Manager
 Software Division II
 Development Department I
FUJITSU LIMITED
 Tech Bldg, 9-18, Shinyokohama 3-chome,
 Kouhoku-ku, Yokohama 222 Japan

FROM: Masato Iwamoto
 Senior Sales Representative
 OEM Sales
 VERITAS Software K.K.
 Fukoku Seimei Bldg, 2-2-2,
 Chiyoda-ku, Tokyo 100-0011 Japan

Total Price \$63,603.00

Item	Model No.	Description	Q'ty	Unit Price	Disc.	Price
I. License						
	SL-VMG1-1500-4	VERITAS Volume Manager (License Only)	1	\$53,743.00		\$53,743.00
Sub Total Price						\$53,743.00
III. Media & Documentation						
	900-001036	One Media / Documentation Kit	1	\$187.00		\$187.00
Sub Total Price						\$187.00
III. Maintenance (For 1 year)						
	DS-VMG1-1500-4	Direct Assist Support	1	\$9,673.00		\$9,673.00
Sub Total Price						\$9,673.00

III. Remarks:

This quotation is only valid 90days after issuing date.
 All of these products should be ordered at one time with one ordersheet within the effective dates.
 Volume Manager product in this quotation is classified in Tier-4. When the products will be installed in the sites,
 all CPU platform serials or ID numbers to be reported to VERITAS Software Corporation.
 The total price (quotation) will not include withholding tax.
 Fujitsu shall pay the amount by wiretransfer in US\$ by the end of the following month when Fujitsu will receive
 the invoice from VERITAS.

Approved By

 Mikio Kamimura
 President
 VERITAS Japan K.K.

X (Hardware Tiers):

VERITAS products are licensed on a per system basis. The following VERITAS products are licensed based on the hardware tiers below: File System, Volume Manager, FirstWatch, Cluster Server, NetBackup Master and Media Servers, Storage Migrator, TeleBackup, VML, Storage Replicator and all Editions products. Certain configurations restrictions exist when clustering Sun systems and FirstWatch and Cluster Server are not supported on all listed hardware platforms. Please refer to the appropriate product selection in the price book to determine what systems are supported in Cluster configurations.

TIER 0 - No Tier

TIER 1 - Desktop Workstations / Workgroup Servers

Sun	Sun 44xx, SPARCclassic,
Sun SPARCstation	-LX, -4, -5, -10, -20,
Sun SPARCserver	-4, -5, -10, -20, Ultra-1, Ultra-2, Ultra 5, Ultra 10, Ultra-60, Ultra 80, Ultra 450
Sun Enterprise	2, 105, 30, 60, 150, 220R, 420R, 450, NetraT
HP	A-Class, D-Class, E-Class, F-Class, G-Class, L-Class/R-Class, K100
NEC	NX7000/D-Class, NX7000/100
Intel (Windows NT)	Single (1) processor System
Compaq (Tru64 UNIX)	AlphaServer 800, 1200, DS10, DS20, DS20E
Compaq Alpha (Windows NT)	Single (1) processor System
Fujitsu GP7000F/PRIMPOWER	Model 200, Model 200R, Model 400A, Model 400R,
Fujitsu GP7000S	Model 5, Model 10, Model 25, Model 45,
Fujitsu GP400S	Model 5, Model 10, Model 50, Model 450
Fujitsu GP300S	Model 2,
IBM (AIX)	RS/6000 43P Model 140, Model 150, Model 260
NCR (MP-RAS)	3200 Series
SGI (IRIX)	Octane, O2 The SGI 320, 540, and 1400 are Intel NT based machines and are priced accordingly.
Siemens (Reliant UNIX)	RM200

TIER 2 - Departmental Servers

Sun	Sun 48xx,
Sun SPARCserver	-1000
Sun Enterprise	3000, 3500, 4000, 4500, 5000, 5500
HP	H-Class, I-Class, K2XX K3XX, K4XX K5XX, N4000@360MHz
NEC NX7000	/ZXX, /3XX, /4XX, /5XX, /N4000(360MHz)
Intel (Windows NT)	Dual (2) processor System
Compaq (Tru64 UNIX)	AlphaServer 2000, 2100, 4000, 4100, ES40
Compaq Alpha (Windows NT)	Dual (2) processor System
Fujitsu GP7000F/PRIMPOWER	Model 600
Fujitsu GP7000S	Model 800, Model 350, Model 450, Model 550
IBM (AIX)	F40, B50
NCR (MP-RAS)	S20, S24, S26
SGI (IRIX)	Orign 200, ONYX 2
Siemens (Reliant UNIX)	RM300

TIER 3 - Enterprise Servers

Sun Enterprise	6000, 6500
Sun SPARCcenter	2000
HP	V2200, V2250, T-Class, N4000@440MHz
NEC NX7000	/V2200, /V2250, /T-Class, /N4000(440MHz)
Intel (Windows NT)	Quad (4) processor System
Compaq (Tru64 UNIX)	AlphaServer 8200, 8400, GS60E, GS140
Compaq Alpha (Windows NT)	Quad (4) processor System
Fujitsu GP7000F/PRIMPOWER	Model 1000
Fujitsu GP7000S	Model 650
IBM (AIX)	F50, H50, H70, R50, S70
NCR (MP-RAS)	S50, S1501, 4400
SGI (IRIX)	Orign 2000, 2100 series
Siemens (Reliant UNIX)	RM400, RM600

TIER 4 - Global Enterprise Servers

Sun Enterprise	10000
Cray	CS8400
HP	V2500, V2500 SCA
NEC NX7000	/V2500
Intel (Windows NT)	8 or above processor System
Auspax	Auspax 7000/500
Compaq (Tru64 UNIX)	AlphaServer SC
Compaq Alpha (Windows NT)	8 or above processor System
Fujitsu GP7000F/PRIMPOWER	Model 2000
Fujitsu GP7000S	Model 1000
IBM (AIX and DYNIX/ptx)	RS/6000 model S80, and SP (POWER2, PowerPC and POWER3 node types and combinations), Sequent Symmetry
NCR (MP-RAS)	3600, 4700, 4800, 5150, 5200
SGI (IRIX)	Orign 2000, Cray Orign 2000 models (32 or more processors)
Siemens (Reliant UNIX)	RCS II, Pyramid Reliant R8 and POWER3 node types and combinations), Sequent Symmetry

NOTE HP EPS pricing is priced on a per node basis. For example, if an EPS solution is based on a 4 K460 servers then pricing would be to use the K460 Tiers 2 price and multiply by four.

X (NetBackup Client Tiers)

0 - PC Client	Licenses for personal computers, such as Macintosh and Windows 95/98, are sold in 10 packs.
1 - Workstation Client	Used by one person and is not utilized as a file server, database server, or web server.
2 - Server Client	Defined as a system that provides services to more than one person. The system is considered a group or company resource and is controlled by a system administrator or IS organization. a group or company resource and is controlled by a system administrator or IS organization.

Microsoft Corporation 98100 N. 170th Ave. Redmond, WA 98072
Tel: 206.882.8000 Fax: 206.882.8001

August 8, 2000

Ms. Cindy Costa
HAL Computer Systems, Inc.
1315 Dell Avenue
Campbell, CA 95008

Dear Cindy,

Here is the information you requested regarding US pricing of certain Microsoft products:

Microsoft Windows NT Server 4.0, including 5 CALs (24 copies @ \$809)	\$19,416
Visual C++ Professional 6.0	\$549
5-yr maintenance for above software @ \$2095/yr	\$10,475

This quote is valid for the next 60 days. Please let me know if I can be of any further assistance.

Best regards,

Mark Hassall
Product Manager, Microsoft Windows NT Server

4089415904 AUG 08 10:00
PAGE 3/3 13:46 No.008 P.02

HAL COMPUTER SYSTEMS
ID: 19:08 (WED) 09.09.00
PL



THE ECOMMERCE TRANSACTION PLATFORM

July 18, 2000

FUJITSU LIMITED
Tokyo, Japan

Gentlemen:

Per your request I am enclosing the pricing information regarding TUXEDO that you requested. This pricing applies to Tuxedo 6.1, 6.2, 6.3, 6.4 and 6.5, and 7.1. Please note that Tuxedo 7.1 is our most recent version of Tuxedo but that all 6.x releases are generally available. Core functionality services pricing is appropriate for your activities. As per the table below, Fujitsu server systems are classified in one of 5 tiers based on CPU type and capacity.

This pricing quotation will be valid through October 17, 2000.

Tuxedo Core Functionality Services (CFS) Program Product Pricing and Description

TUX-CFS provides a basic level of middleware support for distributed computing, and is best used by organizations with substantial resources and knowledge for advanced distributed computing implementations.

TUX-CFS prices are server only and are based on the overall performance characteristics of the server and uses the same five tier computer classification as TUXEDO 6.x and TUXEDO 7.x. Prices range from \$3,000 for Tier 1 to \$250,000 for Tier 5. Under this pricing option EVERY system running TUX-CFS at the user site must have a TUXEDO license installed and pay the appropriate per server license fees.

BEA Tux/CFS Version 6.x/7.x Unlimited User License Fees Per Server

Unlimited User License fees per server	Number of Users	Dollar Amount	Maintenance (5 x 8) per year	Maintenance (7 x 24) per year
Tier 1 -- PC Servers with 1 or 2 CPUs, entry level RISC Uni-processor workstations and servers	Unlimited	\$3,000.00	\$480.00	\$690.00
Tier 2 -- PC Servers with 3 or 4 CPUs, Midrange RISC Uni-processor servers and workstations	Unlimited	\$12,000.00	\$1,920.00	\$2,760.00
Tier 3 -- Midrange Multiprocessors, up to 8 CPUs per system capacity	Unlimited	\$30,000.00	\$4,800.00	\$6,900.00
Tier 4 -- Large (more than 8, less than 32 CPUs) and Mainframe Systems	Unlimited	\$100,000.00	\$16,000.00	\$23,000.00
Tier 5 -- Massively Parallel Systems, > 32 processors; All Mainframes	Unlimited	\$250,000.00	\$40,000.00	\$57,500.00

10/31/97

BEA SYSTEMS, INC.

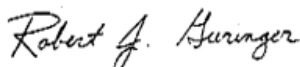
Intel based server tier classifications:

Platform	Operating System	Tier 1	Tier 1	Tier 2	Tier 3	Tier 3
		Class 1	Class 2	Class 3	Class 4	Class 5
Intel 386/486/Pentium/Pentium Pro PCs	SCO UNIX 3.2.2 and 3.2.4 SCO ODT 2.x,3.x Solaris x86 2.X UnixWare, Windows NT	All 386/486 PCs are Class 1	ALL Pentium, Pentium Pro & Pentium II PCs with 1 or 2 CPUs are Class 2/Tier 1	ALL Pentium, Pentium Pro & Pentium PCs with 3 or 4 CPUs are Class 3/Tier 2		ALL Pentium, Pentium Pro & Pentium PCs with 5,6,7, or 8 CPUs are Class 5/Tier 3

Sun Microsystems Server Tier classifications

Tier 1	Tier 2	Tier 3	Tier 3	Tier 4	Tier 5
Class 2	Class 3	Class 4	Class 5	Class 6	Class 7
X-terminal 1 Station 5/85 Station 4 Station 20/50 Station 20/51 Station 20/61 Ultra 1 140/170 Server 470 Server 5/70 Server 20/50 Server 20/51 Server 20/61 Ultra 2 Desktop Ultra 5, 5S Ultra 10, 10S Enterprise 1 &150	Server 5/85 Station 20/71 Server 20/71 Enterprise 220R Enterprise 250 Enterprise 2 - 2100,2200 Ultra 60	Station 20/502 MP 20/612 MP 20/514 MP 20/HS11 20/712 MP Server 1000 Server 1000E Server 20/502 Server 20/712 Server 20/612 Server 20/514 Enterprise 2 -2300 Enterprise 420R	Enterprise 450 SparcCenter 1000 Enterprise 3000 & 3500 Enterprise 4000, 4500, 5000 & 5500 < 8 proc.	SparcCenter 2000 SparcCenter 2000E Enterprise 4000, 4500, 5000 & 5500 Between 8 and 32 proc. Enterprise 6000 & 6500 Between 8 and 32 proc. CRS6400 (< 32 proc.)	CRS6400 (>=32 proc.) Enterprise 6000 & 6500 (>=32 proc.) Enterprise 10000

Very Truly Yours,



Robert J. Gieringer
Worldwide Pricing Manager

1122, n

差出人: Michele Esposito [mesposito@cpx.com]
 送信日時: 2000年7月19日水曜日 12:04
 宛先: cindyc@hal.com
 C C: mesposito@cpx.com
 件名: COMPEX Quotation
 重要度: 高

Quotation for:
 HAL COMPUTERS
<http://www.hal.com/>
 1315 Dell Ave
 Campbell, CA 95008

Attn: Cindy C & Kaz Neomoto
 TP1008C-110V MicroHub 8 Port 10 Base T Ethernet Hub \$28.00 each
 *****Lifetime Warranty
 *****This quote is valid for 90 days

Pricing is based on a staggered delivery and Purchase order
 Committing to a quantity no less than 15,730 units before production
 would begin.

Price may fluctuate down based on if the shipment from Singapore are required by Sea or Air.

Cindy,

Thank you for allowing CompeX the opportunity to submit a quote for this project. Please call or E-mail if your required any additional information, you have a full team of support at COMPEX.

Thank you,
 Michele Esposito
 National Sales Manager
 800 279 8891 x 101
<mailto:mesposito@cpx.com>

- Quotation -

Fujitsu Systems Business of America, Inc.

5200 Patrick Henry Drive
Santa Clara, CA 95054
Tel: (408) 654 2000
Fax: (408) 654 2030

Sales Quote

Quotation valid for 60 days

Date: 7/19/00

Bill To:

Company: HAL COMPUTERS
Address: 1315 DELL AVENUE
CAMPBELL, CA 95008

Phone #: 408.341.5713
FAX #: 408.341.5904
Contact: KAZ MIYAMOTO

Ship To:

Company: HAL COMPUTERS
Address: 1315 DELL AVENUE
CAMPBELL, CA 95008


Phone #: 408.341.5713
FAX #: 408.341.5904
Contact: KAZ MIYAMOTO

NETWORK SECURITY EQUIPMENT

QTY	U/M	Description	Part Number	Tax	Unit Price	Net Price
33	EA	CISCO Catalyst 2924M-RL-A 10/100 Ethernet Switch (24 port)	WS-C2924-XL-EN	Y	1,316.70	43,451.10
33	EA	SMARTNET 1 YR MAINTENANCE 24X7X4 PKG CAT 4	CON-OSP-PKG4	Y	560.00	18,480.00
33	EA	POWER CORD, 110V	CAB-AC	Y	0.00	0.00
Total:						61,931.10

Note:
 (1) Payment term: Net 30 days.
 (2) FOB origin.
 (3) Shipping, taxes, and other applicable fees are not included in the quotation.

Account Manager Signature:  Date: 7/19/00

Prepared by:  Date: 7/19/00

FSBA Engineering Authorized Signature: _____ Date: _____

Customer Authorized Signature: _____ Date: _____

Appendix H: Auditor's attestation letter



<p>Benchmark Sponsor: Makoto Katoh Director, Perf. Engineering Dept. Software Division II Software Group Fujitsu Limited 3-9-18 Shinyokohama Kohoku-ku, Yokohama 222-0033, Japan</p>	<p>Karl Haas Director, OLTP Performance Oracle Corporation 500 Oracle Parkway Redwood Shores, CA 94065 USA</p>
---	---

July 21, 2000

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

Platform:	Fujitsu GP7000F Model 2000
Operating system:	Solaris 7
Database Manager:	Oracle8 Enterprise Edition 8.1.5
Transaction Manager:	BEA Tuxedo 6.4 CFS

The results were:

CPU's Speed	Memory	Disks	NewOrder 90% Response Time	tpmC
Server: GP7000F Model 2000 c/s				
64 x SPARC 64 GP (350 MHz)	80 GB Main (8MB L2 Cache per processor)	1222 x 9 GB 235 x 18 GB	3.58 Seconds	138735.07
Twenty-four (24) Clients: GRANPOWER5000 Model 280 (Specification for each)				
2 x Pentium III (550 MHz)	640 MB Main Cache: 512 KB	1 x 13 GB	n/a	n/a

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

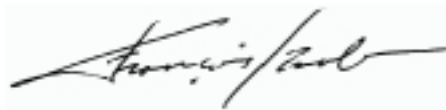
- The database records were the proper size

- The database was properly scaled and populated
- The required ACID properties were met
- The transactions were correctly implemented
- Input data was generated according to the specified percentages
- The transaction cycle times included the required keying and think times
- The reported response times were correctly measured.
- All 90% response times were under the specified maximums
- At least 90% of all delivery transactions met the 80 Second completion time limit
- The reported measurement interval was 30 minutes (1800 seconds)
- The reported measurement interval was representative of steady state conditions
- One checkpoint was taken during the reported measurement interval
- The repeatability of the measured performance was verified
- The 180 day storage requirement was correctly computed
- The system pricing was verified for major components and maintenance

Additional Audit Notes:

Twenty-two of the twenty-four clients on the measured system included one (1) 9 GB Fujitsu MPC3084AT hard disk drive that was substituted by one (1) 13GB Fujitsu MPE3136AT hard disk drive in the priced configuration. Based on the specifications of these disks and on additional performance data collected on these disks, it is our opinion that this substitution does not have a material effect on the reported performance.

Respectfully Yours,



François Raab, President



Bradley J. Askins, Auditor