Bull
Escala PL1660R
Using
AIX 5L Version 5.3
and
DB2 Enterprise 9.5

TPC Benchmark™ C
Full Disclosure Report

First Edition

June 11, 2008

Special Notices

The following terms used in this publication are trademarks of the BULL SAS company in the United States and/or other countries:

BULL
Escala

The following terms used in this publication are trademarks of International Business Machines Corporation in the United States and/or other countries:

IBM System p
IBM System x
AIX
IBM
DB2, DB2 9.5 Enterprise Edition

The following terms used in this publication are trademarks of other companies as follows:

TPC Benchmark, TPC-C, and tpmC are trademarks of the Transaction Processing Performance Council
Microsoft Windows 2003 server and COM+ are registered trademarks of Microsoft Corporation

First Edition: June 11, 2008

The information contained in this document is distributed on an AS IS basis without any warranty either expressed or implied. The use of this information or the implementation of any of these techniques is a customer's responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item has been reviewed by BULL for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environment do so at their own risk.

In this document, any references made to an BULL licensed program are not intended to state or imply that only BULL's licensed program may be used; any functionally equivalent program may be used.

It is possible that this material may contain references to, or information about, BULL products (machines and programs), programming, or services that are not announced in your country. Such references or information must not be construed to mean that BULL intends to announce such products, programming, or services in your country.

All performance data contained in this publication was obtained in a controlled environment, and therefore the results which may be obtained in other operating environments may vary significantly. Users of this document should verify the applicable data in their specific environment.

Request for additional copies of this document should be sent to the following address:

Bull SAS
JF. Lemerre
Rue Jean Jaures , BP68
78340 Les Clayes sous bois
France

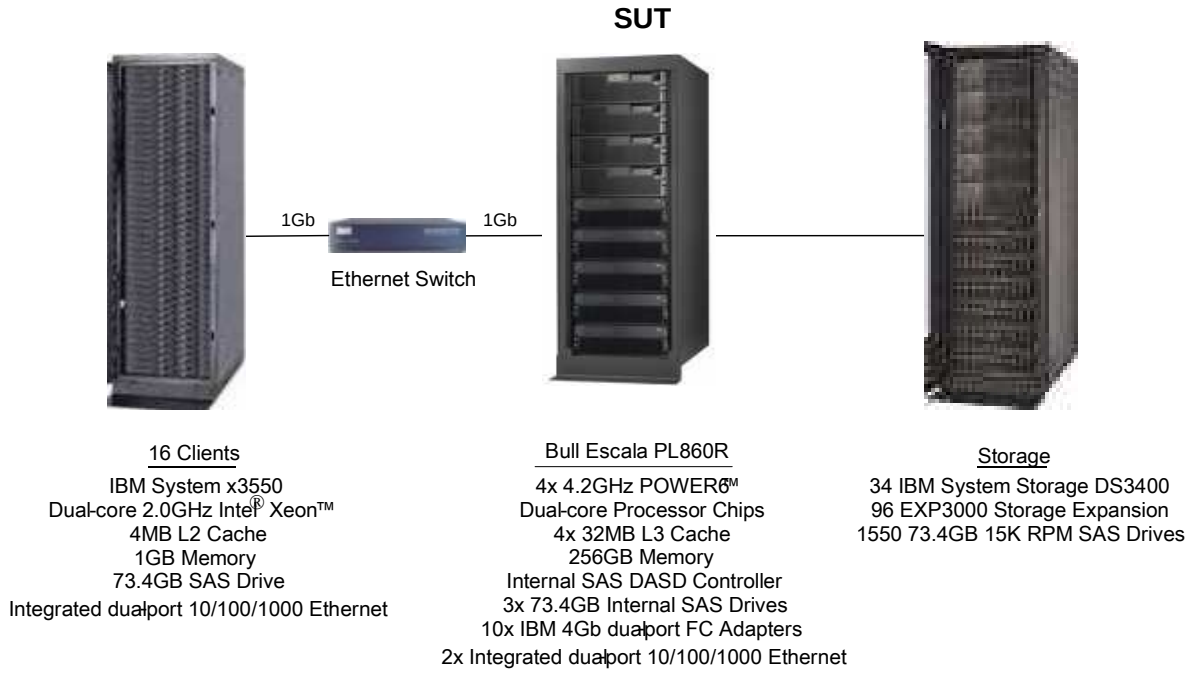
© Copyright Bull SAS, 2008. All rights reserved.

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 on the title page of each item reproduced.

	Bull Escala PL860R	TPC-C Rev. 5.9
		Report Date: June 11, 2008

Total System Cost	TPC-C Throughput	Price/Performance	Availability Date
1,566,664 USD	**629,159.19**	$2.49 USD	**June 11, 2008**

Database Server Processor Chip/Core/Thread	Database Manager	Operating System	Other Software	No. Users
4/8/16 POWER6 4.2 GHz	DB2 9.5	AIX 5L V5.3	Microsoft Visual C++ Microsoft COM+	519,680



System Components	Server		Each of the 16 Clients	
	Quantity	Description	Quantity	Description
Processors Chips /Cores/Threads	4/8/16	4.2GHz POWER6™	1/2/2	2.0GHz 4MB L2 Xeon Processor
Memory	16	16 GB	2	512 MB
Disk Controllers	1 10 34	Integrated SAS 4Gb Dual-port FC Adapters IBM System Storage DS3400	1	SAS Controller
Disk Drives	1553	73.4GB 15K RPM SAS	1	73.4GB 15K RPM SAS
Total Storage		103.60 TB		2,348.8 GB
Terminals	1	System Console	1	System Console

BULL
Escala PL860R

TPC-C Rev. 5.9

Report Date: June11, 2008

Description	Part Number	Brand	Pricing	Unit Price	Quantity	Extended Price	3-year Maint. Price
Server Hardware							
ESCALA NODE PL860R	CPSG069-0000	Bull	1		1	6,298	4829
Indicator SYSTEM PL860R Package	PKHG033-0000	Bull	1		1	0	
4.2GHz P6, 2-Core Processor Card, 0-core active	CPUG159-0000	Bull	1		4	66,748	13553
ONE PROC.ACTIVATION FOR 2-core&4.2 (P6)	CPKG444-0000	Bull	1		8	60,584	
EXPANSION PCI-X I/O DRAWER	DRWG100-0000	Bull	1		2	13,900	13893
73 GB SAS Disk Drive (1"/15krpm)	MSUG251-0000	Bull	1		4	4,824	
2-Port 1GB Ethern.Daught.Board/RJ45 w/i 2sys.port	DCIG015-0000	Bull	1		1	224	
2-Port 10/100/1000 BASE-TX ETHERNET PCI-X ADAPTE	DCCG168-0000	Bull	1		1	860	0
4GB/s Fibre Channel PCI-X Adapter 2-port	DCCG196-0000	Bull	1		10	27,000	
GX Dual-port 12X Channel attach	DCCG206-0000	Bull	1		1	1,059	
IDE Slimline DVD-ROM Drive	CDRG032-0000	Bull	1		1	246	
Rack 25U with one PDU/12 Outlets (1 EIA)	RCKG026-0000	Bull	1		1	3,120	1601
POWER SUPPLY TYPE EUROPE	PSKG004-0002	Bull	1		1	0	
17" FLAT PANEL MONITOR	DMUG024-1000	Bull	1		1	1,040	
QUIET TOUCH KEYBOARD - USB, STEALTH BLACK, FR	KBUG009-000F	Bull	1		1	100	
Power Distribution Unit - Single Phase (1 EIA)	PSSG060-0000	Bull	1		1	920	
AC Power Supply, 1700W, Hot Swap	PSSG067-0000	Bull	1		1	673	
16GB (2x8GB) RDIMMs, 400MHz DRAM	CMMG282-0000	Bull	1		16	314,576	
KIT CABLES/FIRST EXPANSION DRAWER	CKTG308-0000	Bull	1		1	610	
I/O Drawer Mounting Enclosure (Two positions)	CKTG309-0000	Bull	1		1	619	
DASD/Media Back Plane (3.5") with External SAS Port	CKTG313-0000	Bull	1		1	539	
DESKSIDE HMC w/o Monitor/Keyboard	CSKG018-0000	Bull	1		1	2,020	2330
Power Cord PDU/Wall (4.3M), 24A IEC309, 32A Plug	CBLG234-1700	Bull	1		2	530	
Power Cord (9-foot) - Drawer To PDU	CBLG240-1500	Bull	1		2	30	
Power Cord (14-foot) - Drawer To PDU	CBLG240-1700	Bull	1		4	60	
SAS Cable,DASD Bplane w/i Ext.SAS Port to Bulkhead	CBLG250-1000	Bull	1		1	77	
NUMBER OF AIX PARTITIONS	CPFG053-0000	Bull	1		1	0	
PowerVM Express	CPFG073-0000	Bull	1		8	0	
					Subtotal	**506,657**	**36206**
Storage							
IBM System Storage DS3400 Express	1726-42E	IBM	2	8749	34	297,466	
IBM 1M SAS cable	39R6529	IBM	2	119	192	22,848	
IBM System Storage EXP3000	1727-01X	IBM	2	3199	96	307,104	
IBM Hot-Swap 3.5 inch 73.4GB 15K SAS HDD	40K1043	IBM	2	309	1550	478,950	
IBM TotalStorage SAN32B-3	2005-B5K	IBM	2	15810	3	47,430	
8-port Activation	22R5505	IBM	2	8100	6	48,600	
4 Gbps SW SFP Transceivers - 4 pack	2414	IBM	2	550	24	13,200	
IBM S2 42U Standard Rack	93074RX	IBM	2	1489	7	10,423	
ServicePac for 3-Year 24x7x4 Support (DS3400)	44J8073	IBM	2	1300	34		44,200
ServicePac for 3-Year 24x7x4 Support (EXP3000)	41L2768	IBM	2	760	96		72,960
ServicePac for 3-Year 24x7x4 Support (SAN32B-2)	41E9167	IBM	2	3300	3		9,900
ServicePac for 3-Year 24x7x4 Support (Rack)	41L2760	IBM	2	300	7		2,100
					Subtotal	**1,226,021**	**129,160**
Server Software							
BASIC HW DOC POUR EP	MANG031-SPAZ	Bull	1		1	69	
BASIC HARDWARE DOCUMENTATION FOR PL860	MANG130-SP0Z	Bull	1		1	50	
AIX SUBSCRIPTION CLASS E 1 YEAR PER CPU	SBYG013-IN01	Bull	1		8	0	
POWERVM EXPRESS SUBSCRIPTION - 1 YEAR P	SBYG085-IN00	Bull	1		8	0	
VIRTUAL I/O SERVER (MEDIA)	EXSG310-SVCE	Bull	1		1	0	
AIX 5.3 EXPANSION PACK	EXSG312-VA00	Bull	1		1	50	
AIX 5.3 CLASS E - LICENCE (PER CPU)	EXSG319-VE00	Bull	1		8	3,680	9,092
POWERVM EXPRESS - LICENCE (PER CPU)	EXSG371-VA00	Bull	1		8	216	
AIX 5.3 MEDIA KIT	UTSG191-VK00	Bull	1		1	60	
SOFTWARE DELIVERY ON DVDROM	MSDG002-0000	Bull	1		1	0	
AIX SUBSCRIPTION CLASS E 1 YEAR PER CPU	SBYG013-IN01	Bull	1		8		3,132
POWERVM EXPRESS SUBSCRIPTION - 1 YEAR P	SBYG085-IN00	Bull	1		8		194
DB2 9.5 ESE Lic&Mtce (278.52 perVU,p6 VU rating-120, 8 procs)		IBM	2	279	960	267,379	
DB2 9.5 ESE MtceRenewal (13.27 perVU,8 procs, 2 years)		IBM	2	13	1920		25,478
					Subtotal	**271,504**	**37,896**

BULL
Escala PL860R

TPC-C Rev. 5.9

Report Date: June11, 2008

Description	Part Number	Brand	Pricing	Unit Price	Quantity	Extended Price	3-year Maint. Price
Client Hardware							
IBM System x3550 (Dual-core Xeon 2.0GHz 4MB L2	7878AC1	IBM	2	1927	16	30,832	
512MB PC2-5300 CL5 ECC DDR2 Chipkill FBDIMM (	546	IBM	2	115	32	3,680	
73GB 15K Hot-Swap SAS HDD	5161	IBM	2	309	16	4,944	
ServicePac for 3-Year 24x7x4 Support	96P2250	IBM	2	586	16		9,376
NetBAY S24 2U Standard Rack Cabinet	93074RX	IBM	2	1489	1	1,489	
Optical 3-Button Mouse-USB	40K9201	IBM	2	19	1	19	
Preferred Pro FullSize PS/2 Keyboard	40K9584	IBM	2	29	1	29	
IBM C117 17" CRT Monitor	49387NU	IBM	2	149	1	149	
					Subtotal	**41,142**	**9,376**
Third party Hardware/Software							
Visual Studio Standard 2005	127-00012	Microsoft	3	250	1	250	
Microsoft Windows 2003 Server	P70-00275	Microsoft	3	399	16	6,384	
Microsoft Problem Resolution Services		Microsoft	3	245	1		245
Gigabit (+ 2 spares)	512294	3Com	1	290	3	870	
					Subtotal	**7,504**	**245**
					TOTAL	**2,052,828**	**212,883**
					IBM Discount		**-530,067**
					Bull Discount		**-168,980**

Notes:

Pricing: 1-Bull 2-IBM 3-Microsoft

Three-Year Cost of Ownership: $1,566,664

tpmC Rating: 629159

$ / tpmC: 2.49

Audited by Francois Raab of InfoSizing (www.sizing.com)

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

Numerical Quantities Summary for the Escala PL860R

MQTH, computed Maximum Qualified Throughput: 629,159.19 tpmC

Response Times (in seconds)	90th %	Average	Maximum
New Order	1.35	0.89	6.14
Payment	1.37	0.89	5.73
Order-Status	1.28	0.84	4.23
Delivery (interactive)	0.75	0.36	3.33
Delivery (deferred)	0.75	0.32	2.36
Stock-Level	1.55	1.03	4.73
Menu	0.82	0.40	5.87

Response time delay added for emulated components was 0.1 seconds

Transaction Mix, in percent of total transactions	Percent
New Order	44.95%
Payment	43.01%
Order-Status	4.00%
Delivery	4.00%
Stock-Level	4.00%

Keying/Think Times (in seconds)	Min.	Average	Max.
New Order	18.00/0.01	18.01/12.02	18.02/120.20
Payment	3.00/0.01	3.01/12.02	3.02/120.20
Order-Status	2.00/0.01	2.01/10.01	2.02/100.10
Delivery	2.00/0.01	2.01/5.02	2.02/50.20
Stock-Level	2.00/0.01	2.01/5.02	2.02/50.20

Test Duration	
Ramp-up Time	49 minutes
Measurement interval	2 hours 20 minutes
Transactions during measurement interval (all types)	195,917,329
Ramp-down time	15 minutes

Checkpoints	
Number of checkpoints	N/A
Checkpoint interval	N/A

Table of Content

Preface 10
0 General Items 11
0.1. Application Code Disclosure 11
0.2. Benchmark Sponsor 11
0.3. Parameter Settings 11
0.4. Configuration Diagrams 11
1 Clause 1: Logical Data Base Design Related Items 13
1.1. Table Definitions 13
1.2. Database Organization 13
1.3. Insert and/or Delete Operations 13
1.4. Horizontal or Vertical Partitioning 13
2 Clause 2: Transaction & Terminal Profiles Related Items 14
2.1. Verification for the Random Number Generator 14
2.2. Input/Output Screens 14
2.3. Priced Terminal Features 14
2.4. Presentation Managers 14
2.5. Home and Remote Order-lines 14
2.6. New-Order Rollback Transactions 14
2.7. Number of Items per Order 14
2.8. Home and Remote Payment Transactions 15
2.9. Non-Primary Key Transactions 15
2.10. Skipped Delivery Transactions 15
2.11. Mix of Transaction Types 16
2.12. Queuing Mechanism of Delivery 16
3 Clause 3: Transaction and System Properties 17
3.1. Atomicity Requirements 17
3.1.1. Atomicity of Completed Transaction 17
3.1.2. Atomicity of Aborted Transactions 17
3.2. Consistency Requirements 17
3.3. Isolation Requirements 18
3.4. Durability Requirements 18
3.4.1. Permanent Unrecoverable Failure of any Single Durable Medium 18
4 Clause 4: Scaling and Data Base Population Related Items 20
4.1. Cardinality of Tables 20
4.2. Distribution of Tables and Logs 20
4.3. Data Base Model Implemented 20
4.4. Partitions/Replications Mapping 21
4.5. 60-Day Space Calculations 24
5 Clause 5: Performance Metrics and Response Time Related Items 25
5.1. Response Times 25
5.2. Keying and Think Times 25
5.3. Response Time Frequency Distribution 26
5.4. Performance Curve for Response Time versus Throughput 28
5.5. Think Time Frequency Distribution 29
5.6. Throughput versus Elapsed Time 29
5.7. Steady State Determination 30
5.8. Work Performed During Steady State 30
5.8.1. Transaction Flow 30
5.8.2. Database Transaction 31
5.8.3. Checkpoints 31
5.9. Measurement Interval 31
6 Clause 6: SUT, Driver, and Communication Definition Related Items 32
6.1. RTE Availability 32
6.2. Functionality and Performance of Emulated Components 32
6.3. Network Bandwidth 32

6.4. Operator Intervention 32
7 Clause 7: Pricing Related Items 33
7.1. Hardware and Programs Used 33
7.2. Three Year Cost of System Configuration 33
7.3. Availability Dates 34
7.4. Statement of tpmC and Price/Performance 34
7.5. Country-specific pricing 34
7.6. Orderability Date 34
8 Clause 9: Audit Related Items 35
Appendix - A: Client Server Code 38
A.1 Client/Terminal Handler Code 38
A.2 Client Transaction Code 49
Appendix - B: Tunable Parameters 79
B.1 Database Parameters. 79
B.2 Transaction Monitor Parameters 80
B.3 AIX Parameters 81
Appendix - C: Database Setup Code 84
C.1 Database Creation Scripts 84
C.2 Data Generation Code 280
Appendix - D: Pricing Information 292

Abstract

This report documents the full disclosure information required by the TPC Benchmark™ C Standard Specification Revision 5.9 dated June, 2007, for measurements on the Bull Escala PL860R. The software used on the Bull Escala PL860R includes AIX 5L Version 5.3 operating system, DB2 9.5 database manager. Microsoft COM+ is used as transaction manager.

Bull Escala PL860R

Company Name	**System Name**	**Data Base Software**	**Operating System Software**
Bull SAS	Bull Escala PL860R	DB2 9.5	AIX 5L Version 5.3

Total System Cost	**TPC-C Throughput**	**Price/Performance**
• Hardware • Software • 3 Years Maintenance	Sustained maximum throughput of system running TPC-C expressed in transactions per minute	Total system cost/tpmC
1,566,664 USD	629,159.19	$2.49 USD

Preface

TPC Benchmark™ C Standard Specification was developed by the Transaction Processing Performance Council (TPC). It was released on August 13, 1992 and updated with revision 5.9 in June 2007.

This is the full disclosure report for benchmark testing of the Bull Escala PL860R and DB2 9.5 according to the TPC Benchmark™ C Standard Specification.

TPC Benchmark™ C exercises the system components necessary to perform tasks associated with that class of on-line transaction processing (OLTP) environments emphasizing a mixture of read-only and update intensive transactions. This is a complex OLTP application environment exercising a breadth of system components associated by such environments 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
- Data bases consisting of many tables with a wide variety of sizes, attributes, and relationships
- Contention on data access and update

This benchmark defines four on-line transactions and one deferred transaction, intended to emulate functions that are common to many OLTP applications. However, this benchmark does not reflect the entire range of OLTP requirements. The extent to which a customer can achieve the results reported by a vendor is highly dependent on how closely TPC-C approximates the customer application. The relative performance of systems derived from this benchmark does not necessarily hold for other workloads or environments. Extrapolations to any other environment are not recommended.

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

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.

0 General Items

0.1. Application Code Disclosure

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

Appendix A contains the application code for the five TPC Benchmark™ C transactions. Appendix D contains the terminal functions and layouts.

0.2. Benchmark Sponsor

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

This benchmark was sponsored by **Bull SAS.**

0.3. Parameter Settings

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

- *Data Base tuning options*
- *Recovery/commit options*
- *Consistency/locking options*
- *Operating system and application configuration parameters.*

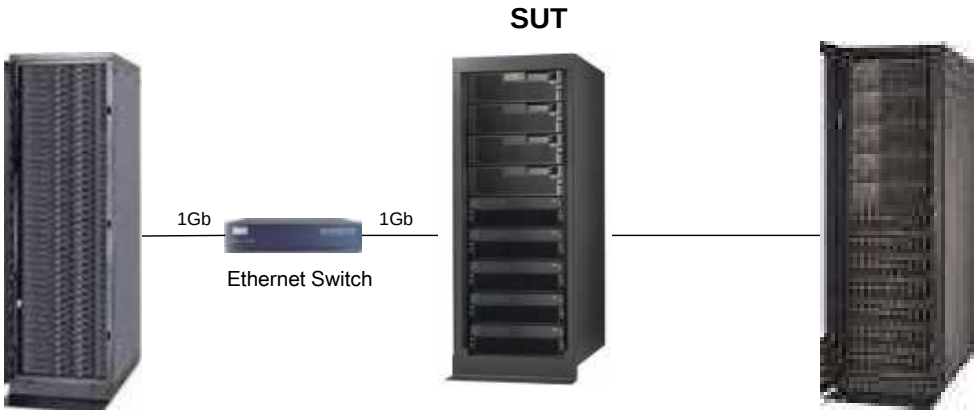
Appendix B contains the system, data base, and application parameters changed from their default values used in these TPC Benchmark™ C tests.

0.4. Configuration Diagrams

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

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

Bull Escala PL860R Benchmark Configuration

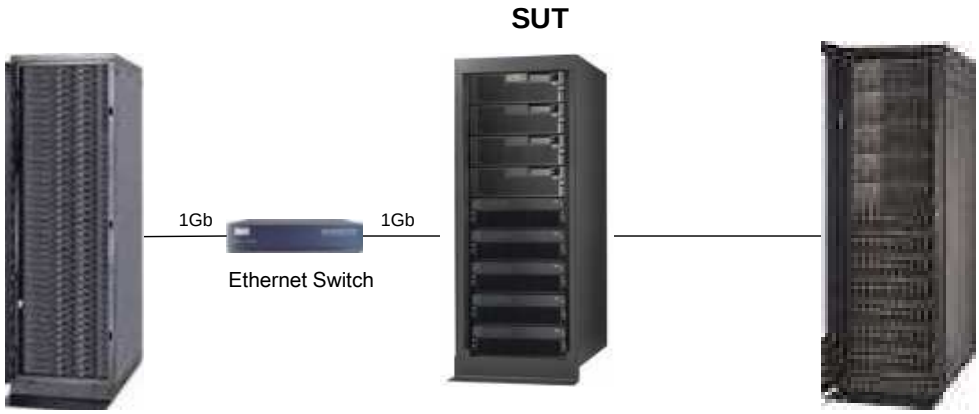


16 Clients
IBM System x3550
Dual-core 2.0GHz Intel® Xeon™
4MB L2 Cache
1GB Memory
73.4GB SAS Drive
Integrated dual-port 10/100/1000 Ethernet

Bull Escala PL860R
4x 4.2GHz POWER6™
Dual-core Processor Chips
4x 32MB L3 Cache
256GB Memory
Internal SAS DASD Controller
3x 73.4GB Internal SAS Drives
10x IBM 4Gb dual-port FC Adapters
2x Integrated dual-port 10/100/1000 Ethernet

Storage
34 IBM System Storage DS3400
96 EXP3000 Storage Expansion
1550 73.4GB 15K RPM SAS Drives

Bull Escala PL860R Priced Configuration



16 Clients
IBM System x3550
Dual-core 2.0GHz Intel® Xeon™
4MB L2 Cache
1GB Memory
73.4GB SAS Drive
Integrated dual-port 10/100/1000 Ethernet

Bull Escala PL860R
4x 4.2GHz POWER6™
Dual-core Processor Chips
4x 32MB L3 Cache
256GB Memory
Internal SAS DASD Controller
3x 73.4GB Internal SAS Drives
10x IBM 4Gb dual-port FC Adapters
2x Integrated dual-port 10/100/1000 Ethernet

Storage
34 IBM System Storage DS3400
96 EXP3000 Storage Expansion
1550 73.4GB 15K RPM SAS Drives

1 Clause 1: Logical Data Base Design Related Items

1.1. Table Definitions

Listings must be provided for all table definition statements and all other statements used to setup the data base.

Appendix C contains the table definitions and the database load programs used to build the data base.

1.2. Database Organization

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

Physical space was allocated to DB2 9.5 on the server disks according to the details provided in Appendix C.

1.3. Insert and/or Delete Operations

It must be ascertained that insert and/or delete operations to any of the tables can occur concurrently with the TPC-C transaction mix. Furthermore, any restriction in the SUT data base 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.

There were no restrictions on insert and/or delete operations to any of the tables. The space required for an additional five percent of the initial table cardinality was allocated to DB2 9.5 and priced as static space.

The insert and delete functions were verified by the auditor. In addition, the auditor verified that the primary key for each database table could be updated outside the range of its initial partition.

1.4. Horizontal or Vertical Partitioning

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

All tables but ITEM were horizontally partitioned into multiple tables.

The WAREHOUSE, DISTRICT, NEW_ORDER, and HISTORY table partitions contain data associated with a range of 1,624 warehouses. The STOCK, CUSTOMER, ORDERS, and ORDER_LINE table partitions contain data associated with a range of 812 warehouses.

For each partitioned table, a view was created over all table partitions to provide full transparency of data manipulation.

No tables were replicated.

2 Clause 2: Transaction & Terminal Profiles Related Items

2.1. Verification for the Random Number Generator

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

The srandom(), getpid() and gettimeofday() functions are used to produce unique random seeds for each driver. The drivers use these seeds to seed the srand(), srandom() and srand48() functions. Random numbers are produced using wrappers around the standard system random number generators.

The negative exponential distribution uses the following function to generate the distribution. This function has the property of producing a negative exponential curve with a specified average and a maximum value 4 times the average.

```
const double RANDOM_4_Z = 0.89837799236185
const double RANDOM_4_K = 0.97249842407114
double neg_exp_4(double average {
     return – average * (1/RANDOM_4_Z * log (1 – RANDOM_4_K * drand48()))};
}
```

The seeds for each user were captured and verified by the auditor to be unique. In addition, the contents of the database were systematically searched and randomly sampled by the auditor for patterns that would indicate the random number generator had affected any kind of a discernible pattern; none were found.

2.2. Input/Output Screens

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

The screen layouts are now presented in HTML 1.0 web pages. Clauses 2.4.3, 2.5.3, 2.6.3, 2.7.3, and 2.8.3 of the TPC-C specifications were used as guidelines for html character placement.

2.3. Priced Terminal Features

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).

Microsoft Internet Explorer was used to verify the compliance with claus 2.2.2.4.

2.4. Presentation Managers

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

The terminal emulation did not involve screen presentations, message bundling or local storage of TPC-C rows. All screen processing was handled by the client system. All data manipulation was handled by the server system.

2.5. Home and Remote Order-lines

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

Table 2-1 shows the percentage of home and remote transactions that occurred during the measurement period for the New-Order transactions.

2.6. New-Order Rollback Transactions

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

Table 2-1 shows the percentage of New-Order transactions that were rolled back due to an illegal item being entered.

2.7. Number of Items per Order

The number of items per order entered by New-Order transactions must be disclosed.

Table 2-1 show the average number of items ordered per New-Order transaction.

2.8. Home and Remote Payment Transactions

The percentage of home and remote Payment transactions must be disclosed.

Table 2-1 shows the percentage of home and remote transactions that occurred during the measurement period for the Payment transactions.

2.9. Non-Primary Key Transactions

The percentage of Payment and Order-Status transactions that used non-primary key (C_LAST) access to the data base must be disclosed.

Table 2-1 shows the percentage of non-primary key accesses to the data base by the Payment and Order-Status transactions.

2.10. Skipped Delivery Transactions

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

Table 2-1 shows the percentage of Delivery transactions missed due to a shortage of supply of rows in the NEW-ORDER table.

2.11. Mix of Transaction Types

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

Table 2-1 shows the mix percentage for each of the transaction types executed by the SUT.

New Order	**Bull Escala PL860R**
Percentage of Home order lines	99.0%
Percentage of Remote order lines	1.00%
Percentage of Rolled Back Transactions	.99%
Average Number of Items per order	10.00
Payment	
Percentage of Home transactions	85.01%
Percentage of Remote transactions	14.99%
Non-Primary Key Access	
Percentage of Payment using C_LAST	60.00%
Percentage of Order-Status using C_LAST	59.96%
Delivery	
Delivery transactions skipped	0
Transaction Mix	
New-Order	44.95%
Payment	43.01%
Order-Status	4.00%
Delivery	4.00%
Stock-Level	4.00%

Table 2-1: Numerical Quantities for Transaction and Terminal Profiles

2.12. Queuing Mechanism of Delivery

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

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

3 Clause 3: Transaction and System Properties

The results of the ACID test must be disclosed along with a description of how the ACID requirements were met.

All ACID tests were conducted according to specification.

3.1. Atomicity Requirements

The system under test must guarantee that data base 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.

3.1.1. Atomicity of Completed Transaction

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

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

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

3.1.2. Atomicity of Aborted Transactions

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

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

1. The Payment application code was implemented with a Perl script that allowed the transaction to be rolled back rather than committed.
2. The balance, BALANCE_3, was retrieved from the Customer table for the same Customer, District, and Warehouse combination used in the completed Payment transaction Atomicity test.
3. The Payment transaction was executed for the Customer, District and Warehouse used in step 2. Rather than commit the transaction, the transaction was rolled back.
4. The balance, BALANCE_4 was retrieved again for the Customer, District, and Warehouse combination used in step 2. It was verified that BALANCE_4 was equal to BALANCE_3, demonstrating that there were no remaining effects of the rolled back Payment transaction.

3.2. Consistency Requirements

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

Verify that the data base is initially consistent by verifying that it meets the consistency conditions defined in Clauses 3.3.2.1 to 3.3.2.4. Describe the steps used to do this in sufficient detail so that the steps are independently repeatable.

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

1. The sum of balances (d_ytd) for all Districts within a specific Warehouse is equal to the balance (w_ytd) of that Warehouse.
2. For each District within a Warehouse, the next available Order ID (d_next_o_id) minus one is equal to the most recent Order ID [max(o_id)] for the Order table associated with the preceeding District and Warehouse.

Additionally, that same relationship exists for the most recent Order ID [max(o_id)] for the New Order table associated with the same District and Warehouse. Those relationships can be illustrated as follows:

d_next_o_id – 1 = max(o_id) = max(no_o_id)

where (d_w_id = o_w_id = no_w_id) and (d_id = o_d_id = no_d_id)

3. For each District within a Warehouse, the value of the most recent Order ID [max(no_o_id)] minus the first Order ID [min(no_o_id)] plus one, for the New Order table associated with the District and Warehouse equals the number of rows in that New Order table. That relationship can be illustrated as follows:

   max(no_o_id) – min(no_o_id) + 1 = number of rows in New Order for the Warehouse/District

4. For each District within a Warehouse, the sum of Order Line counts [sum(o_ol_cnt)] for the Order table associated with the District equals the number of rows in the Order Line table associated with the same District. That relationship can be illustrated as follows:

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

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

3.3. Isolation Requirements

Operations of concurrent data base transactions must yield results which are indistinguishable from the results which would be obtained by forcing each transaction to be serially executed to completion in some order.

The benchmark specification defines nine tests to demonstrate the property of transaction isolation. The tests, described in Clauses 3.4.2.1 – 3.4.2.9 were all successfully executed using a series of scripts. Case A was observed during the execution of Isolation Tests 7-9.

3.4. Durability Requirements

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

3.4.1. Permanent Unrecoverable Failure of any Single Durable Medium

Permanent irrecoverable failure of any single durable medium containing TPC-C data base tables or recovery log data.

Failure of Log Disk and Log Cache:

This test was conducted on a fully scaled database. The following steps were performed successfully.

1. The current count of the total number of orders was determined by the sum of D_NEXT_O_ID of all rows in the DISTRICT table giving SUM_1.
2. A full load test was started and continued to run for several minutes at a throughput well above the 90% of the reported tpmC.
3. One of the disks containing the transaction log was removed. Since the disk was RAID-5, the SUT continued to process the transactions successfully.
4. The test continued for at least another 5 minutes.
5. Since write cache mirroring was enabled for the log device, one of the RAID controllers, which holds one copy of the mirrored cache, was removed. There was a brief pause in I/O while the failover to the remaining log controller occurred. The controller detected a mirror-out-of-sync condition and deactivated write-back cache.
6. The run continued to completion without write-back cache.
7. The disk from step 3 was replaced after the completion of the run

8. Step 1 was performed returning the value for SUM_2. It was verified that SUM_2 was greater than or equal to SUM_1 plus the completed New_Order transactions recorded by the RTE..

Failure of Durable Medium Containing TPC-C Database Tables:

The following steps were successfully performed to demonstrate Durability against the failure of a disk unit with database tables:

1. The contents of the database were backed up in full.
2. The current count of the total number of orders was determined by the sum of D_NEXT_O_ID of all rows in the DISTRICT table giving SUM_1.
3. A scaled-down test was started with about 12.5% of the full load . The test continued to run at about 12.5% of the reported tpmC for 6 minutes.
4. A disk containing the TPCC table was removed causing the SUT to report numerous errors when attempting to access that device
5. The removed disk was replaced and logical volumes were restored to functional state. The full database was restored from the backup copy in step 1.
6. The database was restarted and the transactions in the log were applied to the database.
7. Step 2 was performed returning SUM_2. It was verified that SUM_2 was greater than or equal to SUM_1 plus the completed New_Order transactions recorded by the RTE.
8. Consistency condition 3 was verified.

Instantaneous Interruption and Memory Failure:

The following steps were conducted on a fully scaled database:

1. The current count of the total number of orders was determined by the sum of D_NEXT_O_ID of all rows in the DISTRICT table giving SUM_1.
2. A full load test was started and continued to run for several minutes at a throughput level well above 90% of the reported tpmC.
3. The system was powered off, which removed power from all system components, including memory.
4. The system was powered back on and the database completed the recovery process.
5. Step 1 was performed returning SUM_2. It was verified that SUM_2 was greater than or equal to SUM_1 plus the completed New_Order transactions recorded by the RTE.
6. Consistency condition 3 was verified.

4 Clause 4: Scaling and Data Base Population Related Items

4.1. Cardinality of Tables

The cardinality (e.g., the number of rows) of each table, as it existed at the start of the benchmark run, must be disclosed.

Table 4-1 portrays the TPC Benchmark™ C defined tables and the number of rows for each table as they were built initially.

All tables are based on 51,968 warehouses, the number of active warehouses during the benchmark.

Table Name	Number of Rows
Warehouse	51,968
District	519,680
Customer	1,559,040,000
History	1,559,040,000
Order	1,559,040,000
New Order	467,712,000
Order Line	15,590,385,921
Stock	5,196,800,000
Item	100,000

Table 4-1: Initial Cardinality of Tables

4.2. Distribution of Tables and Logs

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

Two FC adapters connected to two DS3400 storage controllers were used for the log. Each storage controller contained one RAID5 array with 7 disk drives.The log logical volume was striped across the two arrays (hdisks). Each of the disks used for the log had 73GB of storage capacity and the RAID5 LUN was 407.19GB in size.

There were 8 dual-port FC adapters connected to 32 DS3400 storage controllers through 3 SAN switches. Each of the storage controllers contained 48 disks for a total of 1536 disks. All storage controllers were used evenly.

All database data was evenly distributed on storage arrays. Each array is created using 24 disks. Each array contains 12 LUNs and each LUN corresponds to one DB2 container.

RAID0 was used to create the disk arrays. All the arrays used 73GB disks. The distribution of data across all arrays was identical.

4.3. Data Base Model Implemented

A statement must be provided that describes the data base model implemented by the DBMS used.

The database manager used for this testing was DB2 9.5. DB2 9.5 is a relational DBMS. DB2 remote stored procedures and embedded SQL statements were used. The DB2 stored procedures were invoked via SQL CALL statements. Both the client application and stored procedures were written in embedded C code.

4.4. Partitions/Replications Mapping

The mapping of data base partitions/replications must be explicitly described.

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

DATA DISTRIBUTION		
ARRAY GROUP NAME	DATABASE PARTITION	LUNS
F01V1	1	D1F01V1ITEM, D1F01V1WARE, D1F01V1DIST, D1F01V1CSTI, D1F01V1NORA, D1F01V1ORL, D1F01V1STK, D1F01V1CST, D1F01V1ORDI, D1F01V1ORD, D1F01V1HIST, D1F01V1NORB
F01V1	2	D1F01V2ITEM, D1F01V2WARE, D1F01V2DIST, D1F01V2CSTI, D1F01V2NORA, D1F01V2ORL, D1F01V2STK, D1F01V2CST, D1F01V2ORDI, D1F01V2ORD, D1F01V2HIST, D1F01V2NORB
F02V1	3	D1F02V1ITEM, D1F02V1WARE, D1F02V1DIST, D1F02V1CSTI, D1F02V1NORA, D1F02V1ORL, D1F02V1STK, D1F02V1CST, D1F02V1ORDI, D1F02V1ORD, D1F02V1HIST, D1F02V1NORB
F02V1	4	D1F02V2ITEM, D1F02V2WARE, D1F02V2DIST, D1F02V2CSTI, D1F02V2NORA, D1F02V2ORL, D1F02V2STK, D1F02V2CST, D1F02V2ORDI, D1F02V2ORD, D1F02V2HIST, D1F02V2NORB
F03V1	5	D1F03V1ITEM, D1F03V1WARE, D1F03V1DIST, D1F03V1CSTI, D1F03V1NORA, D1F03V1ORL, D1F03V1STK, D1F03V1CST, D1F03V1ORDI, D1F03V1ORD, D1F03V1HIST, D1F03V1NORB
F03V1	6	D1F03V2ITEM, D1F03V2WARE, D1F03V2DIST, D1F03V2CSTI, D1F03V2NORA, D1F03V2ORL, D1F03V2STK, D1F03V2CST, D1F03V2ORDI, D1F03V2ORD, D1F03V2HIST, D1F03V2NORB
F04V1	7	D1F04V1ITEM, D1F04V1WARE, D1F04V1DIST, D1F04V1CSTI, D1F04V1NORA, D1F04V1ORL, D1F04V1STK, D1F04V1CST, D1F04V1ORDI, D1F04V1ORD, D1F04V1HIST, D1F04V1NORB
F04V1	8	D1F04V2ITEM, D1F04V2WARE, D1F04V2DIST, D1F04V2CSTI, D1F04V2NORA, D1F04V2ORL, D1F04V2STK, D1F04V2CST, D1F04V2ORDI, D1F04V2ORD, D1F04V2HIST, D1F04V2NORB
F05V1	9	D1F05V1ITEM, D1F05V1WARE, D1F05V1DIST, D1F05V1CSTI, D1F05V1NORA, D1F05V1ORL, D1F05V1STK, D1F05V1CST, D1F05V1ORDI, D1F05V1ORD, D1F05V1HIST, D1F05V1NORB
F05V1	10	D1F05V2ITEM, D1F05V2WARE, D1F05V2DIST, D1F05V2CSTI, D1F05V2NORA, D1F05V2ORL, D1F05V2STK, D1F05V2CST, D1F05V2ORDI, D1F05V2ORD, D1F05V2HIST, D1F05V2NORB
F06V1	11	D1F06V1ITEM, D1F06V1WARE, D1F06V1DIST, D1F06V1CSTI, D1F06V1NORA, D1F06V1ORL, D1F06V1STK, D1F06V1CST, D1F06V1ORDI, D1F06V1ORD, D1F06V1HIST, D1F06V1NORB
F06V1	12	D1F06V2ITEM, D1F06V2WARE, D1F06V2DIST, D1F06V2CSTI, D1F06V2NORA, D1F06V2ORL, D1F06V2STK, D1F06V2CST, D1F06V2ORDI, D1F06V2ORD, D1F06V2HIST, D1F06V2NORB
F07V1	13	D1F07V1ITEM, D1F07V1WARE, D1F07V1DIST, D1F07V1CSTI, D1F07V1NORA, D1F07V1ORL, D1F07V1STK, D1F07V1CST, D1F07V1ORDI, D1F07V1ORD, D1F07V1HIST, D1F07V1NORB
F07V1	14	D1F07V2ITEM, D1F07V2WARE, D1F07V2DIST, D1F07V2CSTI, D1F07V2NORA, D1F07V2ORL, D1F07V2STK, D1F07V2CST, D1F07V2ORDI, D1F07V2ORD, D1F07V2HIST, D1F07V2NORB
F08V1	15	D1F08V1ITEM, D1F08V1WARE, D1F08V1DIST, D1F08V1CSTI, D1F08V1NORA, D1F08V1ORL, D1F08V1STK, D1F08V1CST, D1F08V1ORDI, D1F08V1ORD, D1F08V1HIST, D1F08V1NORB
F08V1	16	D1F08V2ITEM, D1F08V2WARE, D1F08V2DIST, D1F08V2CSTI, D1F08V2NORA, D1F08V2ORL, D1F08V2STK, D1F08V2CST, D1F08V2ORDI, D1F08V2ORD, D1F08V2HIST, D1F08V2NORB
F09V1	17	D1F09V1ITEM, D1F09V1WARE, D1F09V1DIST, D1F09V1CSTI, D1F09V1NORA, D1F09V1ORL, D1F09V1STK, D1F09V1CST, D1F09V1ORDI, D1F09V1ORD, D1F09V1HIST, D1F09V1NORB
F09V1	18	D1F09V2ITEM, D1F09V2WARE, D1F09V2DIST, D1F09V2CSTI, D1F09V2NORA, D1F09V2ORL, D1F09V2STK, D1F09V2CST, D1F09V2ORDI, D1F09V2ORD, D1F09V2HIST, D1F09V2NORB

F10V1	19	D1F10V1ITEM, D1F10V1WARE, D1F10V1DIST, D1F10V1CSTI, D1F10V1NORA, D1F10V1ORL, D1F10V1STK, D1F10V1CST, D1F10V1ORDI, D1F10V1ORD, D1F10V1HIST, D1F10V1NORB
F10V1	20	D1F10V2ITEM, D1F10V2WARE, D1F10V2DIST, D1F10V2CSTI, D1F10V2NORA, D1F10V2ORL, D1F10V2STK, D1F10V2CST, D1F10V2ORDI, D1F10V2ORD, D1F10V2HIST, D1F10V2NORB
F11V1	21	D1F11V1ITEM, D1F11V1WARE, D1F11V1DIST, D1F11V1CSTI, D1F11V1NORA, D1F11V1ORL, D1F11V1STK, D1F11V1CST, D1F11V1ORDI, D1F11V1ORD, D1F11V1HIST, D1F11V1NORB
F11V1	22	D1F11V2ITEM, D1F11V2WARE, D1F11V2DIST, D1F11V2CSTI, D1F11V2NORA, D1F11V2ORL, D1F11V2STK, D1F11V2CST, D1F11V2ORDI, D1F11V2ORD, D1F11V2HIST, D1F11V2NORB
F12V1	23	D1F12V1ITEM, D1F12V1WARE, D1F12V1DIST, D1F12V1CSTI, D1F12V1NORA, D1F12V1ORL, D1F12V1STK, D1F12V1CST, D1F12V1ORDI, D1F12V1ORD, D1F12V1HIST, D1F12V1NORB
F12V1	23	D1F12V2ITEM, D1F12V2WARE, D1F12V2DIST, D1F12V2CSTI, D1F12V2NORA, D1F12V2ORL, D1F12V2STK, D1F12V2CST, D1F12V2ORDI, D1F12V2ORD, D1F12V2HIST, D1F12V2NORB
F13V1	25	D1F13V1ITEM, D1F13V1WARE, D1F13V1DIST, D1F13V1CSTI, D1F13V1NORA, D1F13V1ORL, D1F13V1STK, D1F13V1CST, D1F13V1ORDI, D1F13V1ORD, D1F13V1HIST, D1F13V1NORB
F13V1	26	D1F13V2ITEM, D1F13V2WARE, D1F13V2DIST, D1F13V2CSTI, D1F13V2NORA, D1F13V2ORL, D1F13V2STK, D1F13V2CST, D1F13V2ORDI, D1F13V2ORD, D1F13V2HIST, D1F13V2NORB
F14V1	27	D1F14V1ITEM, D1F14V1WARE, D1F14V1DIST, D1F14V1CSTI, D1F14V1NORA, D1F14V1ORL, D1F14V1STK, D1F14V1CST, D1F14V1ORDI, D1F14V1ORD, D1F14V1HIST, D1F14V1NORB
F14V1	28	D1F14V2ITEM, D1F14V2WARE, D1F14V2DIST, D1F14V2CSTI, D1F14V2NORA, D1F14V2ORL, D1F14V2STK, D1F14V2CST, D1F14V2ORDI, D1F14V2ORD, D1F14V2HIST, D1F14V2NORB
F15V1	29	D1F15V1ITEM, D1F15V1WARE, D1F15V1DIST, D1F15V1CSTI, D1F15V1NORA, D1F15V1ORL, D1F15V1STK, D1F15V1CST, D1F15V1ORDI, D1F15V1ORD, D1F15V1HIST, D1F15V1NORB
F15V1	30	D1F15V2ITEM, D1F15V2WARE, D1F15V2DIST, D1F15V2CSTI, D1F15V2NORA, D1F15V2ORL, D1F15V2STK, D1F15V2CST, D1F15V2ORDI, D1F15V2ORD, D1F15V2HIST, D1F15V2NORB
F16V1	31	D1F16V1ITEM, D1F16V1WARE, D1F16V1DIST, D1F16V1CSTI, D1F16V1NORA, D1F16V1ORL, D1F16V1STK, D1F16V1CST, D1F16V1ORDI, D1F16V1ORD, D1F16V1HIST, D1F16V1NORB
F16V1	32	D1F16V2ITEM, D1F16V2WARE, D1F16V2DIST, D1F16V2CSTI, D1F16V2NORA, D1F16V2ORL, D1F16V2STK, D1F16V2CST, D1F16V2ORDI, D1F16V2ORD, D1F16V2HIST, D1F16V2NORB
F17V1	33	D1F17V1ITEM, D1F17V1WARE, D1F17V1DIST, D1F17V1CSTI, D1F17V1NORA, D1F17V1ORL, D1F17V1STK, D1F17V1CST, D1F17V1ORDI, D1F17V1ORD, D1F17V1HIST, D1F17V1NORB
F17V1	34	D1F17V2ITEM, D1F17V2WARE, D1F17V2DIST, D1F17V2CSTI, D1F17V2NORA, D1F17V2ORL, D1F17V2STK, D1F17V2CST, D1F17V2ORDI, D1F17V2ORD, D1F17V2HIST, D1F17V2NORB
F18V1	35	D1F18V1ITEM, D1F18V1WARE, D1F18V1DIST, D1F18V1CSTI, D1F18V1NORA, D1F18V1ORL, D1F18V1STK, D1F18V1CST, D1F18V1ORDI, D1F18V1ORD, D1F18V1HIST, D1F18V1NORB
F18V1	36	D1F18V2ITEM, D1F18V2WARE, D1F18V2DIST, D1F18V2CSTI, D1F18V2NORA, D1F18V2ORL, D1F18V2STK, D1F18V2CST, D1F18V2ORDI, D1F18V2ORD, D1F18V2HIST, D1F18V2NORB
F19V1	37	D1F19V1ITEM, D1F19V1WARE, D1F19V1DIST, D1F19V1CSTI, D1F19V1NORA, D1F19V1ORL, D1F19V1STK, D1F19V1CST, D1F19V1ORDI, D1F19V1ORD, D1F19V1HIST, D1F19V1NORB
F19V1	38	D1F19V2ITEM, D1F19V2WARE, D1F19V2DIST, D1F19V2CSTI, D1F19V2NORA, D1F19V2ORL, D1F19V2STK, D1F19V2CST, D1F19V2ORDI, D1F19V2ORD, D1F19V2HIST, D1F19V2NORB
F20V1	39	D1F20V1ITEM, D1F20V1WARE, D1F20V1DIST, D1F20V1CSTI, D1F20V1NORA, D1F20V1ORL, D1F20V1STK, D1F20V1CST, D1F20V1ORDI, D1F20V1ORD, D1F20V1HIST, D1F20V1NORB
F20V1	40	D1F20V2ITEM, D1F20V2WARE, D1F20V2DIST, D1F20V2CSTI, D1F20V2NORA, D1F20V2ORL, D1F20V2STK, D1F20V2CST, D1F20V2ORDI, D1F20V2ORD, D1F20V2HIST, D1F20V2NORB
F21V1	41	D1F21V1ITEM, D1F21V1WARE, D1F21V1DIST, D1F21V1CSTI, D1F21V1NORA, D1F21V1ORL, D1F21V1STK, D1F21V1CST, D1F21V1ORDI, D1F21V1ORD, D1F21V1HIST, D1F21V1NORB

F21V1	42	D1F21V2ITEM, D1F21V2WARE, D1F21V2DIST, D1F21V2CSTI, D1F21V2NORA, D1F21V2ORL, D1F21V2STK, D1F21V2CST, D1F21V2ORDI, D1F21V2ORD, D1F21V2HIST, D1F21V2NORB
F22V1	43	D1F22V1ITEM, D1F22V1WARE, D1F22V1DIST, D1F22V1CSTI, D1F22V1NORA, D1F22V1ORL, D1F22V1STK, D1F22V1CST, D1F22V1ORDI, D1F22V1ORD, D1F22V1HIST, D1F22V1NORB
F22V1	44	D1F22V2ITEM, D1F22V2WARE, D1F22V2DIST, D1F22V2CSTI, D1F22V2NORA, D1F22V2ORL, D1F22V2STK, D1F22V2CST, D1F22V2ORDI, D1F22V2ORD, D1F22V2HIST, D1F22V2NORB
F23V1	45	D1F23V1ITEM, D1F23V1WARE, D1F23V1DIST, D1F23V1CSTI, D1F23V1NORA, D1F23V1ORL, D1F23V1STK, D1F23V1CST, D1F23V1ORDI, D1F23V1ORD, D1F23V1HIST, D1F23V1NORB
F23V1	46	D1F23V2ITEM, D1F23V2WARE, D1F23V2DIST, D1F23V2CSTI, D1F23V2NORA, D1F23V2ORL, D1F23V2STK, D1F23V2CST, D1F23V2ORDI, D1F23V2ORD, D1F23V2HIST, D1F23V2NORB
F24V1	47	D1F24V1ITEM, D1F24V1WARE, D1F24V1DIST, D1F24V1CSTI, D1F24V1NORA, D1F24V1ORL, D1F24V1STK, D1F24V1CST, D1F24V1ORDI, D1F24V1ORD, D1F24V1HIST, D1F24V1NORB
F24V1	48	D1F24V2ITEM, D1F24V2WARE, D1F24V2DIST, D1F24V2CSTI, D1F24V2NORA, D1F24V2ORL, D1F24V2STK, D1F24V2CST, D1F24V2ORDI, D1F24V2ORD, D1F24V2HIST, D1F24V2NORB
F25V1	49	D1F25V1ITEM, D1F25V1WARE, D1F25V1DIST, D1F25V1CSTI, D1F25V1NORA, D1F25V1ORL, D1F25V1STK, D1F25V1CST, D1F25V1ORDI, D1F25V1ORD, D1F25V1HIST, D1F25V1NORB
F25V1	50	D1F25V2ITEM, D1F25V2WARE, D1F25V2DIST, D1F25V2CSTI, D1F25V2NORA, D1F25V2ORL, D1F25V2STK, D1F25V2CST, D1F25V2ORDI, D1F25V2ORD, D1F25V2HIST, D1F25V2NORB
F26V1	51	D1F26V1ITEM, D1F26V1WARE, D1F26V1DIST, D1F26V1CSTI, D1F26V1NORA, D1F26V1ORL, D1F26V1STK, D1F26V1CST, D1F26V1ORDI, D1F26V1ORD, D1F26V1HIST, D1F26V1NORB
F26V1	52	D1F26V2ITEM, D1F26V2WARE, D1F26V2DIST, D1F26V2CSTI, D1F26V2NORA, D1F26V2ORL, D1F26V2STK, D1F26V2CST, D1F26V2ORDI, D1F26V2ORD, D1F26V2HIST, D1F26V2NORB
F27V1	53	D1F27V1ITEM, D1F27V1WARE, D1F27V1DIST, D1F27V1CSTI, D1F27V1NORA, D1F27V1ORL, D1F27V1STK, D1F27V1CST, D1F27V1ORDI, D1F27V1ORD, D1F27V1HIST, D1F27V1NORB
F27V1	54	D1F27V2ITEM, D1F27V2WARE, D1F27V2DIST, D1F27V2CSTI, D1F27V2NORA, D1F27V2ORL, D1F27V2STK, D1F27V2CST, D1F27V2ORDI, D1F27V2ORD, D1F27V2HIST, D1F27V2NORB
F28V1	55	D1F28V1ITEM, D1F28V1WARE, D1F28V1DIST, D1F28V1CSTI, D1F28V1NORA, D1F28V1ORL, D1F28V1STK, D1F28V1CST, D1F28V1ORDI, D1F28V1ORD, D1F28V1HIST, D1F28V1NORB
F28V1	56	D1F28V2ITEM, D1F28V2WARE, D1F28V2DIST, D1F28V2CSTI, D1F28V2NORA, D1F28V2ORL, D1F28V2STK, D1F28V2CST, D1F28V2ORDI, D1F28V2ORD, D1F28V2HIST, D1F28V2NORB
F29V1	57	D1F29V1ITEM, D1F29V1WARE, D1F29V1DIST, D1F29V1CSTI, D1F29V1NORA, D1F29V1ORL, D1F29V1STK, D1F29V1CST, D1F29V1ORDI, D1F29V1ORD, D1F29V1HIST, D1F29V1NORB
F29V1	58	D1F29V2ITEM, D1F29V2WARE, D1F29V2DIST, D1F29V2CSTI, D1F29V2NORA, D1F29V2ORL, D1F29V2STK, D1F29V2CST, D1F29V2ORDI, D1F29V2ORD, D1F29V2HIST, D1F29V2NORB
F30V1	59	D1F30V1ITEM, D1F30V1WARE, D1F30V1DIST, D1F30V1CSTI, D1F30V1NORA, D1F30V1ORL, D1F30V1STK, D1F30V1CST, D1F30V1ORDI, D1F30V1ORD, D1F30V1HIST, D1F30V1NORB
F30V1	60	D1F30V2ITEM, D1F30V2WARE, D1F30V2DIST, D1F30V2CSTI, D1F30V2NORA, D1F30V2ORL, D1F30V2STK, D1F30V2CST, D1F30V2ORDI, D1F30V2ORD, D1F30V2HIST, D1F30V2NORB
F31V1	61	D1F31V1ITEM, D1F31V1WARE, D1F31V1DIST, D1F31V1CSTI, D1F31V1NORA, D1F31V1ORL, D1F31V1STK, D1F31V1CST, D1F31V1ORDI, D1F31V1ORD, D1F31V1HIST, D1F31V1NORB
F31V1	62	D1F31V2ITEM, D1F31V2WARE, D1F31V2DIST, D1F31V2CSTI, D1F31V2NORA, D1F31V2ORL, D1F31V2STK, D1F31V2CST, D1F31V2ORDI, D1F31V2ORD, D1F31V2HIST, D1F31V2NORB
F32V1	63	D1F32V1ITEM, D1F32V1WARE, D1F32V1DIST, D1F32V1CSTI, D1F32V1NORA, D1F32V1ORL, D1F32V1STK, D1F32V1CST, D1F32V1ORDI, D1F32V1ORD, D1F32V1HIST, D1F32V1NORB
F32V1	64	D1F32V2ITEM, D1F32V2WARE, D1F32V2DIST, D1F32V2CSTI, D1F32V2NORA, D1F32V2ORL, D1F32V2STK, D1F32V2CST, D1F32V2ORDI, D1F32V2ORD, D1F32V2HIST, D1F32V2NORB

Table 4-2: Bull Escala PL860R Data Distribution Benchmark Configuration

4.5. 60-Day Space Calculations

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

60-Day Space Computation
All data sizes in MB unless otherwise stated

Warehouses	51,968
Measured TpmC	629,159

Table	**Rows**	**Table**	**Index**	**5% Space**	**Total Space**
Warehouse	51,968	24	0	1	25
District	519,680	96	0	5	101
Item	100,000	10	0	1	11
Stock	5,196,800,000	1,691,968	0	84,598	1,776,566
Customer	1,559,040,000	1,218,304	37,760	62,803	1,318,867
New-Order	467,712,000	36,544	0	1,827	38,371
Orders	1,559,040,000	60,492	44,032	0	104,524
Order-Line	15,590,400,000	1,050,470	0	0	1,050,470
History	1,559,040,000	96,512	0	0	96,512
Additional Overhead		904,014			904,014

Free Space	183,507		
Dynamic Space	1,207,474	30 Minute log Computations	
Static Space	4,081,988	Log Written (KB)	45,657,036
Daily Growth	233,896	New-Order Txns	18,874,770
Daily Spread	0	Log Written per New-Order (KB)	2.42

Data Storage Requirement	
60 Days (MB)	18,115,724
60 Days (GB)	17,691

Log Storage Requirement	
8 Hours (GB)	696.67

Disk Sizing

	Formatted	**SUT**		**Priced**	
Disk Type	**Capacity (GB)**	**# of Disks**	**Capacity (GB)**	**# of Disks**	**Capacity (GB)**
DB DS3400 RAID0	68.40	1,536	105,062	1,536	105,062
LOG DS3400 RAID5	68.40	14	814	14	814
OS SAS	68.40	3	205	3	205
Total Capacity					106,082

5 Clause 5: Performance Metrics and Response Time Related Items

5.1. Response Times

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

Table 5-1 lists the response times and the ninetieth percentiles for each of the transaction types for the measured system.

5.2. Keying and Think Times

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

Table 5-1 lists the TPC-C keying and think times for the measured system.

Response Times	New Order	Payment	Order Status	Delivery (int./def.)	Stock Level	Menus
90 %	1.35	1.37	1.28	0.75/0.75	1.55	0.82
Average	0.89	0.89	0.84	0.36/0.32	1.03	0.40
Maximum	6.14	5.73	4.23	3.33/2.36	4.73	5.87
Think Times						
Minimum	0.01	0.01	0.01	0.01	0.01	N/A
Average	12.02	12.02	10.01	5.02	5.02	N/A
Maximum	120.20	120.20	100.10	50.20	50.20	N/A
Keying Times						
Minimum	18.00	3.00	2.00	2.00	2.00	N/A
Average	18.01	3.01	2.01	2.01	2.01	N/A
Maximum	18.02	3.02	2.02	2.02	2.02	N/A

Table 5-1: Think and Keying Times

5.3. Response Time Frequency Distribution

Response time frequency distribution curves must be reported for each transaction type.

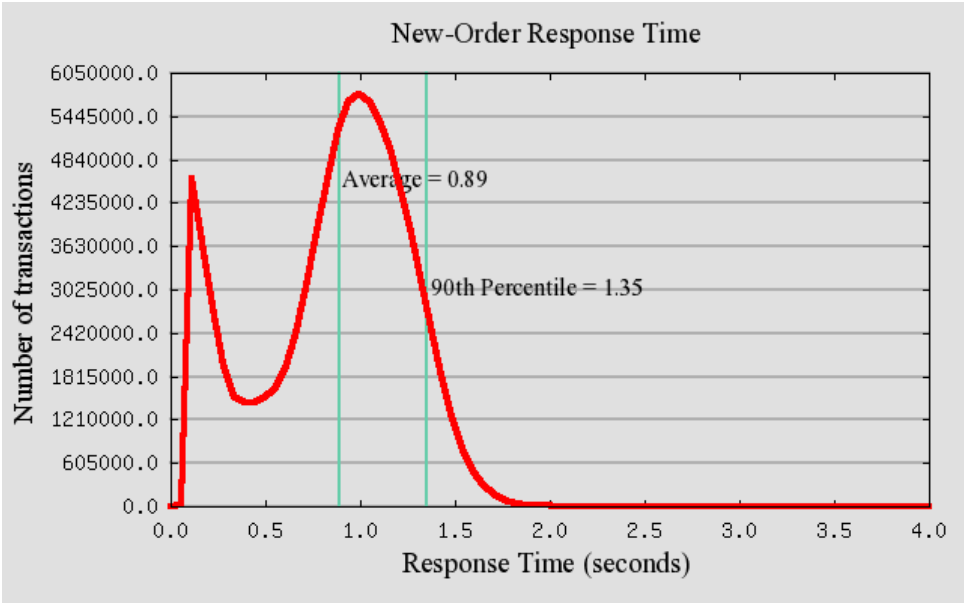


Figure 5-1: New-Order Response Time Distribution

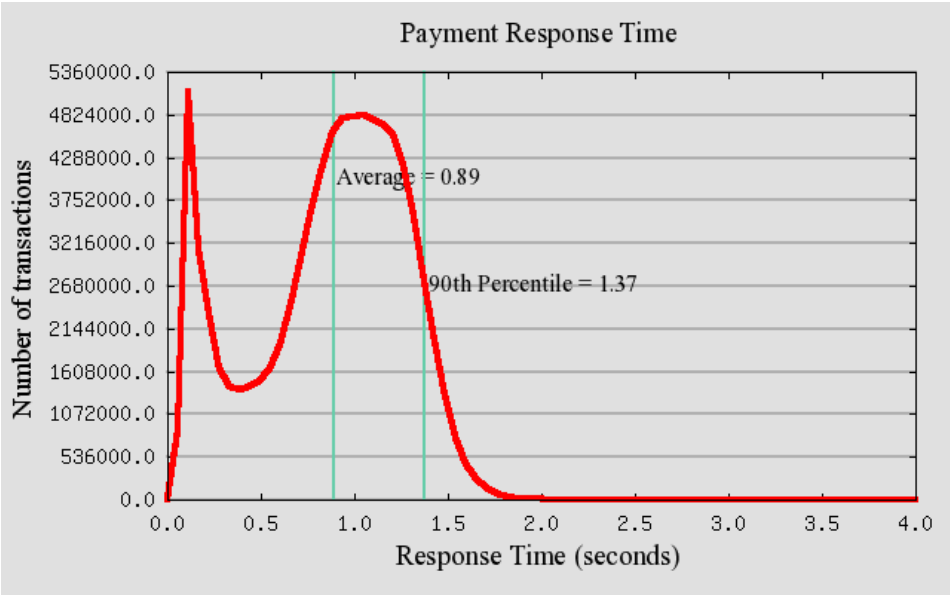


Figure 5-2: Payment Response Time Distribution

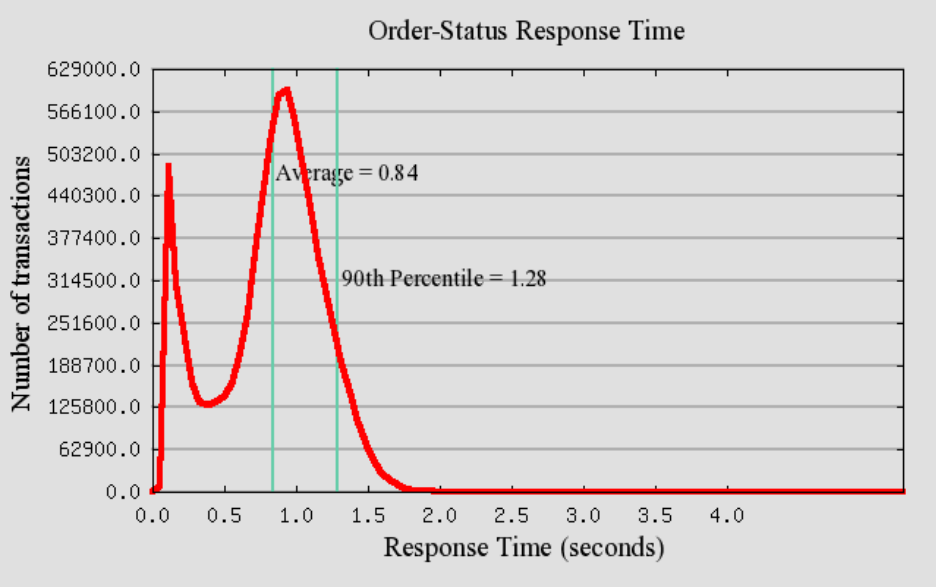


Figure 5-3: Order-Status Response Time Distribution

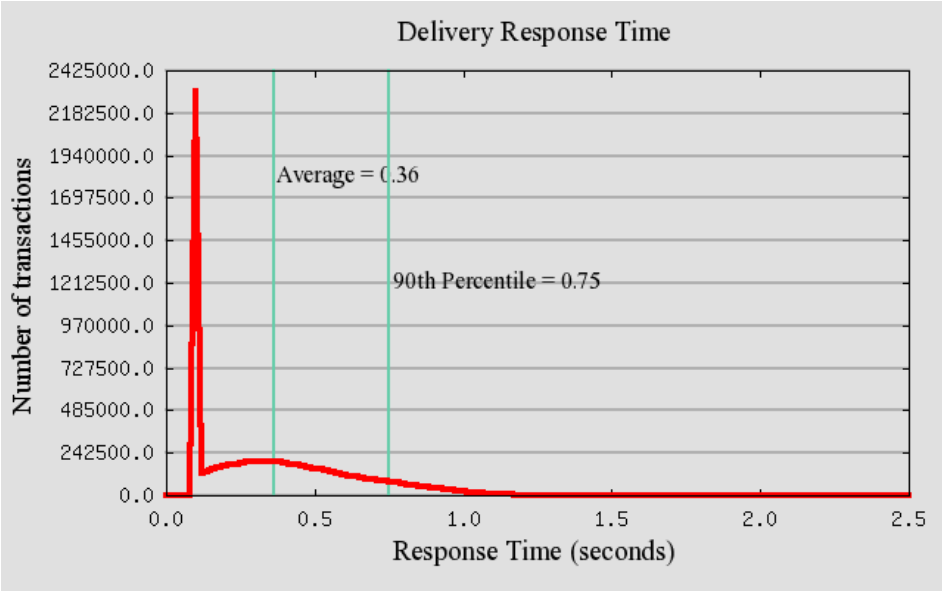


Figure 5-4: Delivery (Interactive) Response Time Distribution

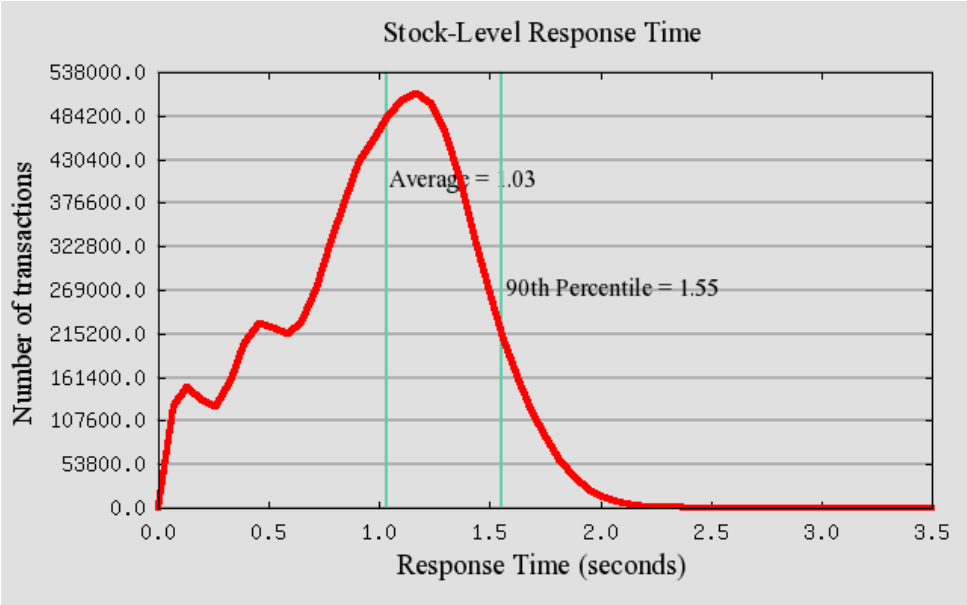


Figure 5-5: Stock Level Response Time Distribution

5.4. Performance Curve for Response Time versus Throughput

The performance curve for response times versus throughput must be reported for the New-Order transaction.

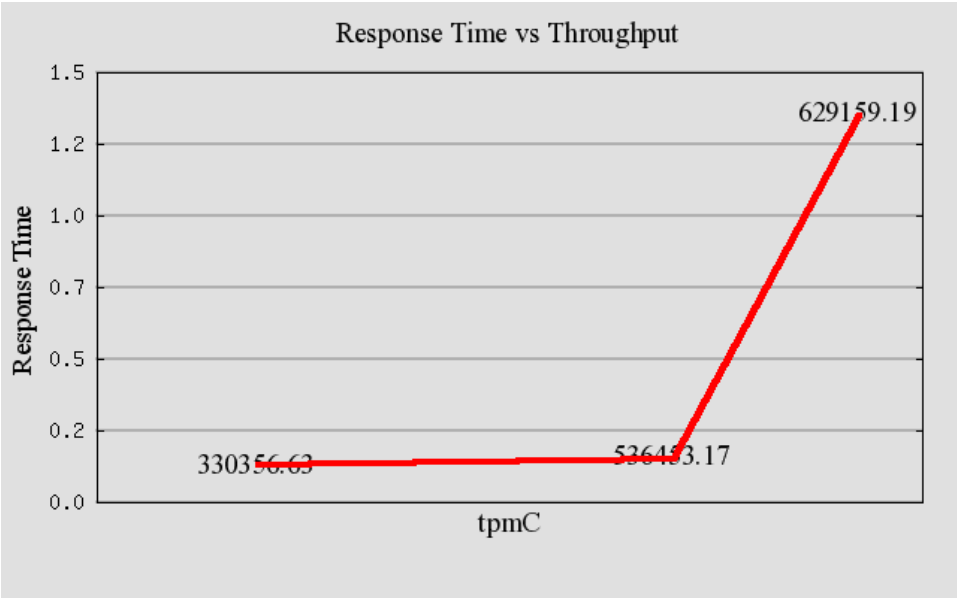


Figure 5-6: New-Order Response Time vs. Throughput

5.5. Think Time Frequency Distribution

A graph of the think time frequency distribution must be reported for the New-Order transaction.

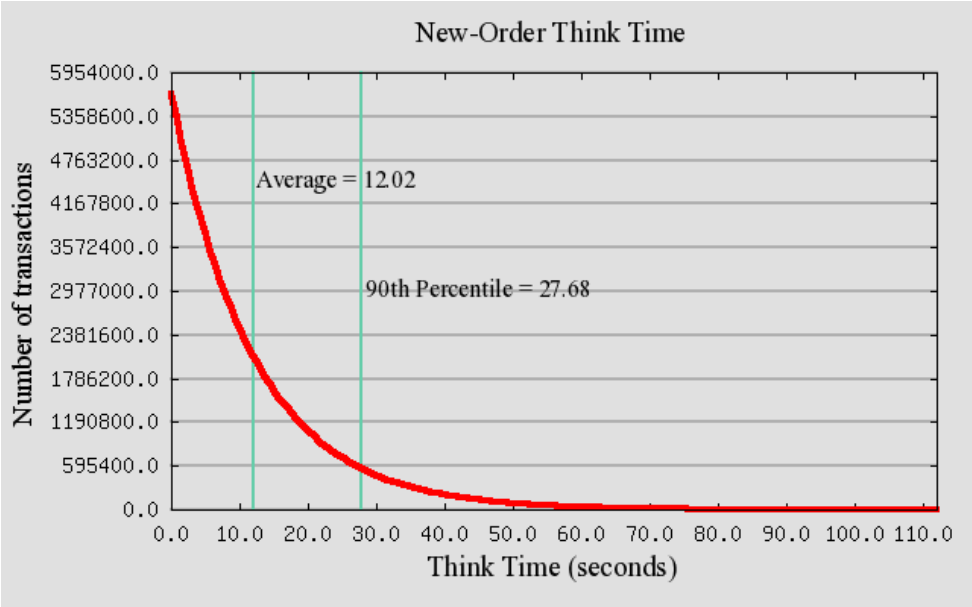


Figure 5-7: New-Order Think Time Distribution

5.6. Throughput versus Elapsed Time

A graph of throughput versus elapsed time must be reported for the New-Order transaction.

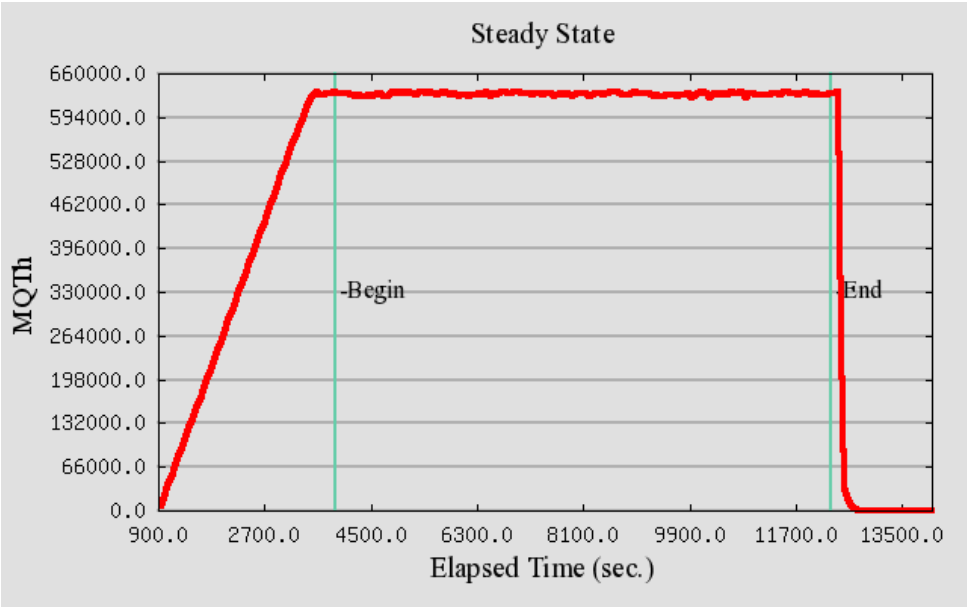


Figure 5-8: New-Order Throughput vs. Elapsed Time

5.7. Steady State Determination

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

All the emulated users were allowed to logon and do transactions. The user ramp-up phase is clearly visible on the graph above. Refer to the Numerical Quantities Summary pages for the rampup time. Figure 5-8 New-Order throughput versus Elapsed Time graph shows that the system maintained a steady state during the measurement interval

5.8. Work Performed During Steady State

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

A 2-hour 20-minute measurement interval was used to guaranty that all work normally performed during an 8-hour sustained test are included in the reported throughput.

5.8.1. Transaction Flow

Each of the 4 (non-delivery) transactions is serviced by 2 individual programs, Internet Information System 5.1 (IIS) and a Microsoft COM+ 1.0 Queued Component Server, used as the transaction manager (COM+). Both programs are running on the client system:

- The initial HTML 1.0 request is serviced by an ISAPI custom-written handler running on Internet Information System 5.1(IIS). IIS is responsible for handling all HTML requests. The web server communicates to the COM+ server through a Microsoft COM+ api interface.
- COM+ communicates with the Server system over Ethernet and handles all database operations**,** using DB2 embedded SQL calls.
  When the COM+ server boots up, it creates a configurable amount of connections to the Server (listed in application settings).
  COM+ routes the transaction and balances the load according to the options defined in the Component Services GUI for the COM+ server application and settings in the Windows 2000 Registry. The configuration file and registry variables are listed in Appendix B.2.
  At the beginning, each TPC-C user sends a pair of HTML 1.0 requests submitting the its unique warehouse and district to the IIS ISAPI handler. Upon successful validation of user's login, IIS the displays an HTML form which encapsulates the TPC-C transaction menu.
  The transaction flow is described below:
  - The TPC-C user requests the transaction type's HTML form and proceeds to generate (fill in) a GET request with the required files for the transaction.
  - IIS accepts the filled in GET request , parses, and validates all values entered by the user.
  - It then proceeds to transmit those values to the COM+ server through an transaction type specific COM+ api interface.
  - The COM+ Pool Manager receives the request and first decides if there is a connection object in the pool available to service it.
    If so, the connection is used to send the transaction request to the Server.
    If no connection is available, the request will enter a COM+ internal queue and will be serviced by the next available connection.
  - Once the connection is available to be used, a COM+ pool thread receives the transaction and calls a TPC-C back end DB2 client api to execute all database operations related to the transaction type. (All the transaction information entered on the HTML form is available in a data structure provided by the ISAPI caller).
  - The transaction is committed and the DB2 back end client returns control back to the COM pool thread.
  - COM pool thread returns control to the ISAPI caller.
    (All transaction results are inside the data structure that the ISAPI caller provided to the COM+ api in the parameter list).
  - ISAPI caller returns control to the "screen application" by doing a PUT request.

5.8.2. Database Transaction

All database operations are performed by the TPC-C back-end programs. The process is described below:

> Using embedded SQL calls, the TPC-C back-end program interacts with DB2 9.5 to perform SQL data manipulations such as update, select, delete and insert, as required by the transaction. After all database operations are performed for a transaction, the transaction is committed.
>
> DB2 9.5 proceeds to update the database as follows:
>
> When DB2 9.5 changes a database table with an update, insert, or delete operation, the change is initially made in memory, not on disk. When there is not enough space in the memory buffer to read in or write additional data pages, DB2 9.5 will make space by flushing some modified pages to disk. Modified pages are also written to disk as part of the "Soft" checkpoint to ensure that no updates remain unflushed for longer than the allowed time. Before a change is made to the database, it is first recorded in the transaction log. This ensures that the database can be recovered completely in the event of a failure. Using the transaction log, transactions that started but did not complete prior to a failure can be undone, and transactions recorded as complete in the transaction log but not yet written to disk can be redone.

5.8.3. Checkpoints

DB2 9.5 uses a write-ahead-logging protocol to guarantee recovery. This protocol uses "Soft" checkpoint to write least-recently-used database pages to disk independent of transaction commit. However, enough log information to redo/undo the change to a database pages is committed to disk before the database page itself is written. This protocol therefore renders checkpoint unnecessary for DB2 9.5. For a more detailed description of the general principles of the write-ahead-logging protocol, see the IBM research paper, "ARIES: A Transaction Recovery Method Supporting Fine Granularity Locking and Partial Rollbacks Using Write-Ahead Logging," by C. Mohan, Database Technology Institute, IBM Almaden Research Center.
(http:// portal.acm.org/citation.cfm?id=128770&coll=portal&dl=ACM&CFID=10343790&CFTOKEN=42047146)

5.9. Measurement Interval

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

A 2-hour 20-minute measurement interval was used. No connections were lost during the run.

6 Clause 6: SUT, Driver, and Communication Definition Related Items

6.1. RTE Availability

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

Bull used a RTE developed by IBM for these tests. Appendix D contains the scripts used in the testing.

6.2. Functionality and Performance of Emulated Components

It must be demonstrated that the functionality and performance of the components being emulated in the Driver System are equivalent to that of the priced system.

No components were emulated.

6.3. Network Bandwidth

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

The database system was connected to a Gigabit Ethernet switch with a rate of 1000Mbits full-duplex. The 16 clients were connected to the Gigabit Ethernet switch at 1000Mbits full-duplex rate.

6.4. Operator Intervention

If the configuration requires operator intervention, the mechanism and the frequency of this intervention must be disclosed.

No operator intervention is required to sustain the reported throughput during the eight-hour period.

7 Clause 7: Pricing Related Items

7.1. Hardware and Programs Used

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

The detailed list of all hardware and software for the priced configuration is listed in the pricing sheets as part of the executive summary. Third Party Pricing Information is provided inAppendix - D:.

7.2. Three Year Cost of System Configuration

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

The pricing details for this disclosure is contained in the executive summary pages. All 3rd party quotations are included at the end of this report in Appendix - D:. All prices are based on Bull and IBM US list prices.

The components (hardware and software) and the maintenances supplied by Bull are discounted by 30% from list price, (37% for Software)only for similar quantities and configuration and in case of a prepay cash.

A discount of $530,067 (30% on Hardware, 37% on Software) was based on the overall value of the specific components from IBM in the quotation provided in Appendix - D:. Discounts for similarly sized configurations with similar quantities and configurations will be similar to those quoted here.

The three years support pricing for Bull S.A. consists of one year warranty (two years for disks) included in the system package price and two years support price, defined as full care in the “GlobalCare” Bull maintenance offer.
This offer is presented in Three levels of service: Bronze, Silver and Gold. The customer may opt, according to their requirements, for the level of their choice. The level of service chosen must be the same for all of the products of the configuration.

Contents of the Gold service:
The Gold service offers a high quality of service, permanent prevention and accelerated response time. It is composed of the following services:
Hardware maintenance
• The reception of 'break-down' calls 24hrs 7/7 through a number allocated by Bull or upon the automatic telephone server or being recorded on the web.
• Telephone product assistance from 8 AM to 8 PM (*) Monday to Friday except bank holidays.
• The customer is called back by an expert within 1 hour of the call being received, not included during the intervention stages.
• Remote maintenance and remote diagnostics of the system (*)
• System controlled remotely (*)
• Access to technical product data via the web.
• Breakdown intervention carried out during working days, between 8 AM and 8 PM Monday to Friday with a delay inferior to 2 hours for a subsystem stops, or inferior to 4 hours for a partly non-functioning system. This consists of the on-site repair and replacement if necessary of parts or faulty equipment broken due to ordinary usage, by new or equivalent parts or equipment. Carrying out these interventions is included in the price of the service.
• The correction of faulty products is carried out within 8 hours of receiving the initial call from the customer.
• The creation and implementation of hardware technical status applicable to Bull customers.
In choosing the Gold level of service, the equivalent level of software support is chosen automatically.
Other extended coverage options are available
(*)The 'Alarm' (Autocalls) service authorises warning messages to be sent by the server to a Bull remote maintenance centre. These alarms can be received 24h 7/7, but the reaction depends wholly upon the contractual service cover period. This means that if action is necessary following an alarm, action will only be taken during the periods stated in the maintenance contract. It is mandatory to have the material necessary to alarming and remote maintenance to benefit this functionnality.

The Gold Service is choosen for the system configuration.

7.3. Availability Dates

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

All components of the SUT will be available on: May 20, 2008.

7.4. Statement of tpmC and Price/Performance

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

.System	tpmC	3-year System Cost	$/tpmC	Availability Date
Bull Escala PL860R	629,159.19	1,566,664 USD	$2.49 USD	June 11, 2008

Please refer to the price list on the Executive Summary page for details.

7.5. 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. All prices are based on Bull and IBM US list prices.

7.6. Orderability Date

For each of the components that are not orderable on the report date of the FDR, the following information must be included in the FDR:

- *Name and part number of the item that is not orderable*
- *The date when the component can be ordered (on or before the Availability Date)*
- *The method to be used to order the component (at or below the quoted price) when that date arrives*
- *The method for verifying the price*

All components used in this benchmark are orderable at the time of this publication.

Prices for all items used in this benchmark can be verified through the contact information provided in the pricing quote for the appropriate vendor. Price quotes are included in Appendix - D:

8 Clause 9: Audit Related Items

If the benchmark has been independently audited, then the auditor's name, address, phone number, and a brief audit summary report indicating compliance must be included in the Full Disclosure Report. A statement should be included, specifying when the complete audit report will become available and who to contact in order to obtain a copy.

The auditor's attestation letter is included in this section of this report:

INFO SIZING

TPC TRANSACTION PROCESSING PERFORMANCE COUNCIL CERTIFIED AUDITOR

Jean-Francois Lemerre
Bull S.A.S.
Rue Jean Jaures, BP68
78340 Les Clayes-sous-Bois
France

Berni Schiefer
IBM Information Management Performance
8200 Warden Avenue
Markham, Ontario L6G1C7
Canada

June 10, 2008

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

Platform: Bull Escala PL860R
Operating system: AIX 5L Version 5.3
Database Manager: DB2 Enterprise 9.5
Transaction Manager: Microsoft COM+

The results were:

CPU's Speed	Memory	Disks	New Order 90% Response Time	tpmC
Server: Bull Escala PL860R				
4 x POWER6 Dual-Core (4.2GHz)	256 GB (4 x 32MB L3)	1553 x 73.4 GB 15K rpm SAS	1.35 Seconds	629,159.19
16 Client: IBM System x3550 (each with)				
1 x Intel Xeon Dual-core (2.0 GHz)	1 GB (4 MB L2 cache)	1 x 73.4 GB 15K rpm SAS	n/a	n/a

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

125 West Monroe Street • Colorado Springs, CO 80907 • 719-473-7555 • www.sizing.com

The following verification items were given special attention:

- The transactions were correctly implemented
- The database records were the proper size
- The database was properly scaled and populated
- The ACID properties were met
- Input data was generated according to the specified percentages
- The transaction cycle times included the required keying and think times
- The reported response times were correctly measured.
- At least 90% of all delivery transactions met the 80 Second completion time limit
- All 90% response times were under the specified maximums
- The measurement interval was representative of steady state conditions
- The reported measurement interval was 140 minutes
- Write-ahead-logging was active during the measurement interval
- The 60 day storage requirement was correctly computed
- The system pricing was verified for major components and maintenance

Additional Audit Notes:

None.

Respectfully Yours,

François Raab, President

125 West Monroe Street • Colorado Springs, CO 80907 • 719-473-7555 • www.sizing.com

Appendix - A: Client Server Code

A.1 Client/Terminal Handler Code

Makefile.config

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

#
# Makefile.config - NT/Win2000 Makefile Configuration
#
#

# Make Configuration (MSVC)
MAKE=nmake.exe

# Compiler Configuration (MSVC).
# CFLAGS_DEBUG may be set to "-Zi -Od", "-DDEBUGIT" "-Zi -Od -DDEBUGIT" or left blank
CC=cl.exe
CFLAGS_OS=-DSQLWINT -MT -DWIN32 -J -Zp8 -DREG_KIT_METHOD -DSWAP_ENDIAN
CFLAGS_OUT=/Fo
CFLAGS_DEBUG=

# Linker Configuration (MSVC)
LD_EXEC=link.exe
LD_STORP=link.exe
LDFLAGS_EXEC=
LDFLAGS_SHLIB=/DLL
LDFLAGS_STORP=$(LDFLAGS_SHLIB) /DEF:rpctpcc.def
LDFLAGS_LIB=/LIBPATH:$(TPCC_SQLLIB)\lib /LIBPATH:"C:\Program Files\Microsoft Visual
Studio\VC98\Lib" db2api.lib winmm.lib
LDFLAGS_OUT=/OUT:

# Library Configuration
AR=lib.exe
ARFLAGS=
ARFLAGS_LIB=
ARFLAGS_OUT=/OUT:

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

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

Src.Cli/Makefile

```
##########################################################################
## Licensed Materials - Property of IBM
##
## Governed under the terms of the International
## License Agreement for Non-Warranted Sample Code.
##
## (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
## All Rights Reserved.
##
## US Government Users Restricted Rights - Use, duplication or
## disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
##########################################################################

#
# Makefile - Makefile for Src.Cli (RTE/Driver Interface)
#
#

!include $(TPCC_ROOT)/Makefile.config

# ########################################################################
# Preprocessor, Compiler and Linker Flags
# ########################################################################

PRP_OPTS =    PACKAGE \
              ISOLATION RR \
              QUERYOPT 7 \
              EXPLAIN ALL \
              MESSAGES $*.prep.msg \
              LEVEL $(TPCC_VERSION) \
              NOLINEMACRO

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

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

OBJS =        $(TPCC_ROOT)/Src.Common/tpccdbg$(OBJEXT) \
              $(TPCC_ROOT)/Src.Common/tpccctx$(OBJEXT) \
              tpcccli$(OBJEXT)

LIBS =        tpcccli$(LIBEXT)

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

all:    connect $(OBJS) plan $(LIBS) disconnect
        $(AR) $(ARFLAGS) $(ARFLAGS_OUT)tpcccli$(LIBEXT) $(OBJS) $(ARFLAGS_LIB)
        @echo "-----------------------------------------------------------"
        @echo "Please copy Ival.h, db2tpcc.h, and tpcccli$(LIBEXT) to"
        @echo "a place where they can be #included and linked with the"
        @echo "RTE/driver code."
        @echo "-----------------------------------------------------------"

clean:
        - $(ERASE) *.msg *.bnd *.plan *$(OBJEXT) *$(LIBEXT) tpcccli.c

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

connect:
        - db2 connect to $(TPCC_DBNAME)

disconnect:
        - db2 connect reset
        - db2 terminate

plan:
        - db2exfmt -d $(TPCC_DBNAME) -e $(TPCC_SCHEMA) -s $(TPCC_SCHEMA) -w -1 -n TPCCCLI
-g -# 0 -o TPCCCLI.exfmt.plan
        - db2expln -d $(TPCC_DBNAME) -c $(TPCC_SCHEMA) -p TPCCCLI -s 0 -g -o
TPCCCLI.expln.plan

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

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

tpcccli.c:
        @echo "Prepping $*.sqc"
        db2 prep $*.sqc $(PRP_OPTS)
        db2 grant execute on package TPCCCLI to public

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

# Client Library:
tpcccli$(LIBEXT):       $(OBJS)

# Source
tpcccli$(OBJEXT):       tpcccli.c

# Headers
tpcccli.c:              $(TPCC_ROOT)/include/db2tpcc.h $(TPCC_ROOT)/include/Ival.h
```

Src.Cli/tpcccli.sqc

```
/**************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
**************************************************************************/

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

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

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

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

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

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

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

  EXEC SQL BEGIN DECLARE SECTION;

    struct vc_new_in
    {
      short len;
      char data[ 262 ] ;
    } * pHostvarInput ;

    struct vc_new_out
    {
      short len;
      char data[ 682 ] ;
    } * pHostvarOutput ;

  EXEC SQL END DECLARE SECTION;

  int clientRc = TRAN_OK ;

  int itemIndex = 0 ;

  in_neword->s_all_local  = 1 ;
  for ( itemIndex = 0 ;
        itemIndex < 15 && in_neword->in_item[ itemIndex ].s_OL_I_ID != UNUSED_ITEM_ID ;
        itemIndex++
      )
  {
    if ( in_neword->in_item[ itemIndex ].s_OL_SUPPLY_W_ID != in_neword->s_W_ID )
    {
      in_neword->s_all_local = 0 ;
    }
  }

  in_neword->s_O_OL_CNT = itemIndex ;

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

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

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

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

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

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

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

  SWAP_BYTE(neword->s_W_TAX);
  SWAP_BYTE(neword->s_D_TAX);
  SWAP_BYTE(neword->s_C_DISCOUNT);
  SWAP_BYTE(neword->s_total_amount);
  SWAP_BYTE(neword->s_O_ID);
  SWAP_BYTE(neword->s_O_OL_CNT);
  SWAP_BYTE(neword->s_transtatus);
  SWAP_BYTE(neword->deadlocks);
  for (itemIndex=0; itemIndex<in_neword->s_O_OL_CNT; itemIndex++)
  {
    SWAP_BYTE(neword->item[ itemIndex ].s_I_PRICE);
    SWAP_BYTE(neword->item[ itemIndex ].s_OL_AMOUNT);
    SWAP_BYTE(neword->item[ itemIndex ].s_S_QUANTITY);
  }
#endif  //SWAP_ENDIAN

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

    // Compute order total

    neword->s_total_amount = 0 ;

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

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

      }
    }

    neword->s_total_amount *= factor;
  }
  else
  {
    sqlerror( NEWORD_SQL, "NEW", __FILE__, __LINE__, &sqlca) ;
    neword->s_transtatus = FATAL_SQLERROR ;
    clientRc = FATAL_SQLERROR ;
  }

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

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

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

  return ( clientRc ) ;
}

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

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

  int clientRc = TRAN_OK ;

  EXEC SQL BEGIN DECLARE SECTION;

    // Inputs

    float h_amount ;
    sqlint32 in_c_id ;

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

    sqlint32 w_id ;
    sqlint32 c_w_id ;
    short    d_id ;
    short    c_d_id ;

    // Outputs

    sqlint32 c_id ;

    double  c_credit_lim ;
    float   c_discount ;
    double  c_balance ;

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

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

    char    c_last [ 16 ] ;

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

    char    c_credit [ 2 ] ;

    char    c_since [27];

    char    c_data [ 200 ] ;
    short   c_data_indicator = 0 ;

    char    h_date [27];
```

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


  EXEC SQL END DECLARE SECTION;


  // Input redirects

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

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

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

  // Output redirects

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

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

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

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

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

  #define h_date            payment->s_H_DATE_time

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

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

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

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

  if ( in_c_id == 0 )
  {

    c_data_prefix_c_last.len = sprintf( c_data_prefix_c_last.data, " %2.2d %6.6d %2.2d %6.6d %06.2f",
c_d_id , c_w_id , d_id , w_id , h_amount ) ;

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

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

  } else {

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


    c_data_prefix_c_id.len = sprintf( c_data_prefix_c_id.data, " %5.5d %2.2d %6.6d %2.2d %6.6d %06.2f",
c_id , c_d_id , c_w_id , d_id , w_id , h_amount) ;

  }

retry_tran:

  payment->deadlocks ++ ;

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

        SELECT   W_STREET_1, W_STREET_2, W_CITY, W_STATE,  W_ZIP
              , D_STREET_1, D_STREET_2, D_CITY, D_STATE, D_ZIP
              , C_ID, C_FIRST, C_MIDDLE, C_STREET_1, C_STREET_2
              , C_CITY, C_STATE, C_ZIP, C_PHONE, C_SINCE, C_CREDIT, C_CREDIT_LIM
              , C_DISCOUNT, C_BALANCE, C_DATA, H_DATE

          INTO   :w_street_1 , :w_street_2 , :w_city , :w_state , :w_zip
              , :d_street_1 , :d_street_2 , :d_city , :d_state , :d_zip
              , :c_id , :c_first , :c_middle ,   :c_street_1 , :c_street_2 , :c_city , :c_state
              , :c_zip , :c_phone , :c_since , :c_credit , :c_credit_lim
              , :c_discount , :c_balance, :c_data :c_data_indicator, :h_date
          FROM TABLE ( PAY_C_LAST(   :w_id
                                  , :d_id
                                  , :c_w_id
                                  , :c_d_id
                                  , :c_last_input
                                  , CAST(:h_amount AS DECIMAL(6,2))
                                  , :c_data_prefix_c_last
                                 )

                  ) AS T (   W_STREET_1, W_STREET_2, W_CITY, W_STATE,  W_ZIP
                          , D_STREET_1, D_STREET_2, D_CITY, D_STATE, D_ZIP
                          , C_ID, C_FIRST, C_MIDDLE, C_STREET_1, C_STREET_2
                          , C_CITY, C_STATE, C_ZIP, C_PHONE, C_SINCE, C_CREDIT, C_CREDIT_LIM
                          , C_DISCOUNT, C_BALANCE, C_DATA, H_DATE
                         )
        ;

        COMMIT ;

      END COMPOUND ;

  }
  else
  {
      EXEC SQL BEGIN COMPOUND NOT ATOMIC STATIC

        SELECT   W_STREET_1, W_STREET_2, W_CITY, W_STATE,  W_ZIP
              , D_STREET_1, D_STREET_2, D_CITY, D_STATE, D_ZIP
              , C_LAST, C_FIRST, C_MIDDLE, C_STREET_1, C_STREET_2
              , C_CITY, C_STATE, C_ZIP, C_PHONE, C_SINCE, C_CREDIT, C_CREDIT_LIM
              , C_DISCOUNT, C_BALANCE, C_DATA, H_DATE

          INTO   :w_street_1 , :w_street_2 , :w_city , :w_state , :w_zip
              , :d_street_1 , :d_street_2 , :d_city , :d_state , :d_zip
              , :c_last, :c_first , :c_middle , :c_street_1 , :c_street_2 , :c_city , :c_state
              , :c_zip , :c_phone , :c_since , :c_credit , :c_credit_lim
              , :c_discount , :c_balance, :c_data :c_data_indicator, :h_date
          FROM TABLE ( PAY_C_ID(   :w_id
                                  , :d_id
                                  , :c_w_id
                                  , :c_d_id
                                  , :in_c_id
                                  , CAST(:h_amount AS DECIMAL(6,2))
                                  , :c_data_prefix_c_id
                                 )

                  ) AS T(   W_STREET_1, W_STREET_2, W_CITY, W_STATE,  W_ZIP
                          , D_STREET_1, D_STREET_2, D_CITY, D_STATE, D_ZIP
                          , C_LAST, C_FIRST, C_MIDDLE, C_STREET_1, C_STREET_2
                          , C_CITY, C_STATE, C_ZIP, C_PHONE, C_SINCE, C_CREDIT, C_CREDIT_LIM
                          , C_DISCOUNT, C_BALANCE, C_DATA, H_DATE
                         )
        ;

        COMMIT ;

      END COMPOUND ;

  }

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

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

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

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

    EXEC SQL ROLLBACK WORK ;

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

  return ( clientRc ) ;
}

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

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

  EXEC SQL BEGIN DECLARE SECTION;

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

    struct vc_ord_out
    {
      short len ;
      char data[ 822 ] ;
    } * out_ord ;

  EXEC SQL END DECLARE SECTION;
```

```
  int clientRc = TRAN_OK ;
  int itemIndex = 0 ;

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

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

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

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

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

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

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

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

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

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

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

  return ( clientRc ) ;
}

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

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

  EXEC SQL BEGIN DECLARE SECTION;

    struct vc_del_in
    {
      short len ;
      char data[ 14 ] ;
    } * in_del ;

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

  EXEC SQL END DECLARE SECTION;

  int clientRc = TRAN_OK ;
  int orderIndex = 0 ;

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

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

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

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

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

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

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

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

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

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

  return ( clientRc ) ;
}

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

#undef w_id
#undef d_id

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

  int clientRc = TRAN_OK ;

  EXEC SQL BEGIN DECLARE SECTION;

    // input

    sqlint32      threshold ;

    // output

    sqlint32    low_stock ;

  EXEC SQL END DECLARE SECTION;

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

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

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

retry_tran:

  stocklev->deadlocks ++ ;

  EXEC SQL BEGIN COMPOUND NOT ATOMIC STATIC

    SELECT COUNT( S_I_ID ) INTO :low_stock

      FROM ( SELECT DISTINCT S_I_ID

             FROM ORDER_LINE , STOCK , DISTRICT

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

           ) OLS

      WITH CS
    ;

    COMMIT ;
```

```
  END COMPOUND ;

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

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

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

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

    EXEC SQL ROLLBACK WORK ;

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

  return ( clientRc ) ;
}
```

Src.Common/Makefile

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

#
# Makefile - Makefile for Src.Common
#
#

!include $(TPCC_ROOT)/Makefile.config

# ######################################################################
# Preprocessor, Compiler and LInker Flags
# ######################################################################

PRP_OPTS =    PACKAGE \
              OPTLEVEL 1 \
              ISOLATION RR \
              MESSAGES $*.prep.msg \
              LEVEL $(TPCC_VERSION) \
              NOLINEMACRO

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

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

UTIL_OBJ_DBG =      tpccdbg$(OBJEXT)
UTIL_OBJ_GEN =      tpccmisc$(OBJEXT)
UTIL_OBJ_DB2 =      tpccctx$(OBJEXT)

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

all:      $(UTIL_OBJ_DBG) $(UTIL_OBJ_GEN) connect $(UTIL_OBJ_DB2) disconnect

dbgen:  $(UTIL_OBJ_GEN)

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

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

connect:
          - db2 connect to $(TPCC_DBNAME)

disconnect:
          - db2 connect reset
          - db2 terminate

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

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

.sqc.c:
          @echo "Prepping $*.sqc"
          db2 prep $*.sqc $(PRP_OPTS)
          db2 grant execute on package TPCCCTX to public

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

# Source
tpccdbg$(OBJEXT):     tpccdbg.c
tpccctx$(OBJEXT):     tpccctx.c
tpccmisc$(OBJEXT):    tpccmisc.c

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

Src.Common/tpccctx.sqc

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

/*
 *
 * tpccctx.sqc - TPCC context code
 *
 */

#include <string.h>
#include <sqlutil.h>
#include "db2tpcc.h"
#include "tpccdbg.h"

int connect_to_TM(char *in_dbname);
int connect_to_TM_auth(char *in_dbname, char *in_username, char *in_password);
int disconnect_from_TM(void);
int create_context();
int destroy_context();
int attach_context(void*);
int detach_context(void*);
int get_context(void**);

int connect_to_TM(char *in_dbname)
{
  return connect_to_TM_auth(in_dbname, "", "");
}

int connect_to_TM_auth(char *in_dbname, char *in_username, char *in_password)
{
  SQL_STRUCTURE sqlca  sqlca;
  int ConnectSQLCODE = 0;

  EXEC SQL BEGIN DECLARE SECTION;
  char dbname[9];
  char username[129];
  char password[15];
  EXEC SQL END DECLARE SECTION;

  SQLCODE = create_context();
  if (SQLCODE != 0) { return SQLCODE; }

  /* Copy 9 characters - 8 for dbname, 1 for NULL */
  strncpy(dbname,in_dbname,9);
  if (strcmp(in_username,"") == 0)
  {
    EXEC SQL CONNECT TO :dbname IN SHARE MODE;
  } else {
    strncpy(username,in_username,128);
    strncpy(password,in_password,14);
    EXEC SQL CONNECT TO :dbname IN SHARE MODE USER :username USING :password;
  }

  ConnectSQLCODE = SQLCODE;
  if (ConnectSQLCODE != 0)
  {
    sqlerror( CLIENT_SQL, "CONNECT", __FILE__, __LINE__, &sqlca);

    SQLCODE = destroy_context();
    if (SQLCODE != 0) { return SQLCODE; }

    return ConnectSQLCODE;
  }

  return 0;
}

int disconnect_from_TM(void)
{
  SQL_STRUCTURE sqlca  sqlca;
  int DisconnectSQLCODE = 0;

  EXEC SQL CONNECT RESET;

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

  SQLCODE = destroy_context();
  if (SQLCODE != 0) { return SQLCODE; }

  if (DisconnectSQLCODE) {
```

```
    return DisconnectSQLCODE;
  }
  return 0;
}

int create_context(void)
{
  SQL_STRUCTURE sqlca sqlca;
  void *ctx;

  sqleSetTypeCtx(SQL_CTX_MULTI_MANUAL);
  sqleBeginCtx(&ctx, SQL_CTX_BEGIN_ALL, NULL, &sqlca);

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

  return 0;
}

int attach_context(void *ctx)
{
  SQL_STRUCTURE sqlca sqlca;

  sqleAttachToCtx(ctx, NULL, &sqlca);

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

  return 0;
}

int detach_context(void *ctx)
{
  SQL_STRUCTURE sqlca sqlca;

  sqleDetachFromCtx(ctx, NULL, &sqlca);

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

  return 0;
}

int destroy_context(void)
{
  SQL_STRUCTURE sqlca sqlca;
  void *ctx;

  SQLCODE = get_context(&ctx);
  if (SQLCODE) { return SQLCODE; }

  sqleEndCtx(&ctx, SQL_CTX_END_ALL, NULL, &sqlca);

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

  return 0;
}

int get_context(void **ctx)
{
  SQL_STRUCTURE sqlca sqlca;

  sqleGetCurrentCtx(ctx, NULL, &sqlca);

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

  return 0;
}
```

Src.Common/tpccdbg.c

```
/*****************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****************************************************************************/

/*
 * tccdbg.c - Debugging Routines
 *
 */

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

#include "sqlca.h"
#include "sql.h"
#include "db2tpcc.h"
#include "tpccdbg.h"

#define DEBUG_FILENAME_SZ 128
#define DEBUG_PATH_SIZE 128

void      del_print();
void      new_print();
void      ord_print();
void      pay_print();
void      stk_print();

void current_tmstmp(char *buf);

static int debugInit = 0;
static char debugPath[DEBUG_PATH_SIZE] = "";

/*-----------------------------------------------------------------*/
/*       InitializeDebug                                           */
/*-----------------------------------------------------------------*/
__inline void InitializeDebug(void) {
  if (debugInit == 0) {
      char *p = getenv("TPCC_DEBUGDIR");
      if (p) {
        strncpy(debugPath, p, DEBUG_PATH_SIZE);
      } else {
        strcpy(debugPath, "C:\\temp");
      }
      strcat(debugPath, "\\");
  }
  debugInit = 1;
}

/*-----------------------------------------------------------------*/
/*       sqlerror                                                  */
/*-----------------------------------------------------------------*/
void sqlerror(int tranType, char *msg, char *file, int line, SQL_STRUCTURE sqlca *psqlca)
{
  FILE *err_fp = NULL;
  char  err_fn[DEBUG_PATH_SIZE + DEBUG_FILENAME_SZ];
  char  tranName[16];
  int j,k;
  char  timeStamp[27];
  char  errStr[512] = "";

  InitializeDebug();
  strncpy(err_fn, debugPath, DEBUG_PATH_SIZE);
  current_tmstmp(&timeStamp[0]);
  timeStamp[19] = (char)NULL;

  switch(tranType)
  {
    case NEWORD_SQL:
      // sprintf(err_fn, "%d.err.out", getpid());
      strcat(err_fn, "new.err.out");
      strcpy(tranName, "NEW_ORDER");
      break;

    case DELIVERY_SQL:
      // sprintf(err_fn, "%d.err.out", getpid());
      strcat(err_fn, "del.err.out");
      strcpy(tranName, "DELIVERY");
      break;

    case PAYMENT_SQL:
      // sprintf(err_fn, "%d.err.out", getpid());
      strcat(err_fn, "pay.err.out");
      strcpy(tranName, "PAYMENT");
      break;

    case ORDSTAT_SQL:
      // sprintf(err_fn, "%d.err.out", getpid());
      strcat(err_fn, "ord.err.out");
      strcpy(tranName, "ORDER_STAT");
      break;

    case STOCKLEV_SQL:
      //sprintf(err_fn, "%d.err.out", getpid());
      strcat(err_fn, "stk.err.out");
      strcpy(tranName, "STOCK_LVL");
      break;

    case 0:
      strcat(err_fn, "cli.err.out");
      strcpy(tranName, "CLIENT");
      break;

    default:
      return;

  }

  /* Generate Formatted Error Message */
  sqlaintp(errStr, 512, 78, psqlca);

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

  fprintf(err_fp, "-----------------------------------------\n");
  fprintf(err_fp, "Transaction: %s (%s)\n", tranName, msg);
  fprintf(err_fp, "FILE %s (%u)\n", file, line);
  fprintf(err_fp, "SQLCODE %d ", psqlca->sqlcode);
  fprintf(err_fp, "TIME %s\n", timeStamp);
```

```
  fprintf(err_fp, "----------------------------------------\n");
  fprintf(err_fp, "%s", errStr);
  fprintf(err_fp, "----------------------------------------\n");

  if (psqlca->sqlerrmc[0] != ' ' || psqlca->sqlerrmc[1] != ' ')
  {
    fprintf(err_fp, "slerrmc:  ");

    for(j = 0; j < 5; j++)
    {
      for(k = 0; k < 16; k++) {
        int pos = j * 16 + k;
        if (pos < 70) fprintf(err_fp, "%02x ", psqlca->sqlerrmc[pos]);
        else fprintf(err_fp, "   ");
      }
      fprintf(err_fp,"   |");
      for(k = 0; k < 16; k++) {
        int pos = j * 16 + k;
        char c = ' ';
        if (pos < 70) {
          c = psqlca->sqlerrmc[pos];
          if (!isprint(c)) c = ' ';
        }
        fprintf(err_fp,"%c", c);
      }
      fprintf(err_fp,"|\n");
      if (j < 4) fprintf(err_fp,"          ");
    }
  }

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

  fprintf(err_fp, "sqlerrd: ");
  for(j = 0; j < 6; j++)
    fprintf(err_fp, " %d", psqlca->sqlerrd[j]);
  fprintf(err_fp,"\n");

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

  fprintf(err_fp, "\n");

  fclose(err_fp);
}


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

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

/*----------------------------------------------------------------*/
/* del_print                                                      */
/*----------------------------------------------------------------*/
void del_print (struct out_delivery_struct *delivery_ptr,
                struct in_delivery_struct  *in_delivery,
                char *filename,
                char *msg)
{
  FILE *debug_fp;
  char  timeStamp[27];
  int   j;

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

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

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

  fprintf(debug_fp,"in_delivery_struct {\n");
  fprintf(debug_fp,"\ts_W_ID         = %d (%X)\n",
          in_delivery->s_W_ID, in_delivery->s_W_ID);
  fprintf(debug_fp,"\ts_O_CARRIER_ID = %d (%X)\n",
          in_delivery->s_O_CARRIER_ID, in_delivery->s_O_CARRIER_ID);
  fprintf(debug_fp,"}\n\n");

  fprintf(debug_fp,"out_delivery_struct {\n");
  fprintf(debug_fp,"\ts_transtatus   = %d (%X)\n",
          delivery_ptr->s_transtatus,delivery_ptr->s_transtatus);
  fprintf(debug_fp,"\tdeadlocks      = %d (%X)\n",
          delivery_ptr->deadlocks,delivery_ptr->deadlocks);

  for (j = 0; j < 10; j++) {
    fprintf(debug_fp,"\t\ts_O_ID[%d]        = %d\n",
            j, delivery_ptr->s_O_ID[j]);
  }
  fprintf(debug_fp,"\t}\n}\n\n");
  fclose(debug_fp);
}


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

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

/*----------------------------------------------------------------*/
/* new_print                                                      */
/*----------------------------------------------------------------*/
void new_print (struct out_neword_struct *neword_ptr,
                struct in_neword_struct *in_neword,
                char *filename,
                char *msg)
{
  FILE *debug_fp;
  char  timeStamp[27];
  int   j, items;

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

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

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

  fprintf(debug_fp,"in_neword_struct {\n");

  fprintf(debug_fp,"\ts_C_ID         = %d (%X)\n",
          in_neword->s_C_ID, in_neword->s_C_ID);
  fprintf(debug_fp,"\ts_W_ID         = %d (%X)\n",
          in_neword->s_W_ID, in_neword->s_W_ID);
  fprintf(debug_fp,"\ts_D_ID         = %d (%X)\n",
          in_neword->s_D_ID, in_neword->s_D_ID);
  fprintf(debug_fp,"\ts_O_OL_CNT     = %d (%X)\n",
          in_neword->s_O_OL_CNT, in_neword->s_O_OL_CNT);
  fprintf(debug_fp,"\ts_all_local    = %d (%X)\n",
          in_neword->s_all_local, in_neword->s_all_local);
// fprintf(debug_fp,"\ts_transtatus   = %d (%X)\n",
//         in_neword->s_transtatus, in_neword->s_transtatus);
// fprintf(debug_fp,"\tduplicate_items= %d (%X)\n",
//         in_neword->duplicate_items, in_neword->duplicate_items);

  fprintf(debug_fp,"\titems {\n");
  items = in_neword->s_O_OL_CNT;
  for (j=0; j<items; j++) {
    if(j != 0)
      fprintf(debug_fp,"\n");
    fprintf(debug_fp,"\t\ts_OL_I_ID[%d]        = %d (%X)\n",
            j, in_neword->in_item[j].s_OL_I_ID, in_neword->in_item[j].s_OL_I_ID);
    fprintf(debug_fp,"\t\ts_OL_SUPPLY_W_ID[%d] = %d (%X)\n",
            j, in_neword->in_item[j].s_OL_SUPPLY_W_ID, in_neword->in_item[j].s_OL_SUPPLY_W_ID);
    fprintf(debug_fp,"\t\ts_OL_QUANTITY[%d]    = %d (%X)\n",
            j, in_neword->in_item[j].s_OL_QUANTITY, in_neword->in_item[j].s_OL_QUANTITY);
  }
  fprintf(debug_fp,"\t}\n}\n\n");

  fprintf(debug_fp,"out_neword_struct {\n");
  fprintf(debug_fp,"\ts_C_LAST       = %s\n",
          neword_ptr->s_C_LAST);
  fprintf(debug_fp,"\ts_C_CREDIT     = %s\n",
          neword_ptr->s_C_CREDIT);
  fprintf(debug_fp,"\ts_W_TAX        = %04.4f \n",
          neword_ptr->s_W_TAX);
  fprintf(debug_fp,"\ts_D_TAX        = %04.4f \n",
          neword_ptr->s_D_TAX);
  fprintf(debug_fp,"\ts_C_DISCOUNT   = %04.4f \n",
          neword_ptr->s_C_DISCOUNT);
  fprintf(debug_fp,"\ts_O_ID         = %d (%X)\n",
          neword_ptr->s_O_ID, neword_ptr->s_O_ID);
  fprintf(debug_fp,"\ts_O_OL_CNT     = %d (%X)\n",
          neword_ptr->s_O_OL_CNT, neword_ptr->s_O_OL_CNT);
  fprintf(debug_fp,"\ts_O_ENTRY_D    = %s \n",
          neword_ptr->s_O_ENTRY_D_time);
  fprintf(debug_fp,"\ts_total_amount = %.2f \n",
          neword_ptr->s_total_amount);
  fprintf(debug_fp,"\ts_transtatus   = %d (%X)\n",
          neword_ptr->s_transtatus, neword_ptr->s_transtatus);
  fprintf(debug_fp,"\tdeadlocks      = %d (%X)\n",
          neword_ptr->deadlocks, neword_ptr->deadlocks);

// fprintf(debug_fp,"\ts_W_ID         = %d (%X)\n",
//         neword_ptr->s_W_ID, neword_ptr->s_W_ID);
// fprintf(debug_fp,"\ts_D_ID         = %d (%X)\n",
//         neword_ptr->s_D_ID, neword_ptr->s_D_ID);
// fprintf(debug_fp,"\ts_all_local    = %d (%X)\n",
//         neword_ptr->s_all_local, neword_ptr->s_all_local);
// fprintf(debug_fp,"\tduplicate_items= %d (%X)\n",
//         neword_ptr->duplicate_items, neword_ptr->duplicate_items);

  fprintf(debug_fp,"\titems {\n");
  items = neword_ptr->s_O_OL_CNT;
  for (j=0; j<items; j++) {
```

```
    if(j != 0)
      fprintf(debug_fp,"\n");
    fprintf(debug_fp,"\t\ts_I_NAME[%d]        = %s\n",
            j, neword_ptr->item[j].s_I_NAME);
    fprintf(debug_fp,"\t\ts_I_PRICE[%d]       = %.2f \n",
            j, neword_ptr->item[j].s_I_PRICE);
    fprintf(debug_fp,"\t\ts_OL_AMOUNT[%d]     = %.2f \n",
            j, neword_ptr->item[j].s_OL_AMOUNT);
    fprintf(debug_fp,"\t\ts_S_QUANTITY[%d]    = %d (%X)\n",
            j, neword_ptr->item[j].s_S_QUANTITY, neword_ptr->item[j].s_S_QUANTITY);
    fprintf(debug_fp,"\t\ts_brand_generic[%d]  = %c\n",
            j, neword_ptr->item[j].s_brand_generic);
  }
  fprintf(debug_fp,"\t}\n}\n\n");
  fclose(debug_fp);
}



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

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

/*----------------------------------------------------------------*/
/* ord_print                                                      */
/*----------------------------------------------------------------*/
void ord_print (struct out_ordstat_struct *ordstat_ptr,
                struct in_ordstat_struct  *in_ordstat,
                char *filename,
                char *msg)
{
  FILE *debug_fp;
  char  timeStamp[27];
  int   j, items;

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

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

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

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

  fprintf(debug_fp,"out_ordstat_struct {\n");
  fprintf(debug_fp,"\ts_C_ID         = %d (%X)\n",
          ordstat_ptr->s_C_ID, ordstat_ptr->s_C_ID);
  fprintf(debug_fp,"\ts_C_FIRST      = %s\n",
          ordstat_ptr->s_C_FIRST);
  fprintf(debug_fp,"\ts_C_MIDDLE     = %s\n",
          ordstat_ptr->s_C_MIDDLE);
  fprintf(debug_fp,"\ts_C_LAST       = %s\n",
          ordstat_ptr->s_C_LAST);
  fprintf(debug_fp,"\ts_C_BALANCE    = %.2f\n",
          ordstat_ptr->s_C_BALANCE);
  fprintf(debug_fp,"\ts_O_ID         = %d (%X)\n",
          ordstat_ptr->s_O_ID, ordstat_ptr->s_O_ID);
  fprintf(debug_fp,"\ts_O_ENTRY_D    = %s \n",
          ordstat_ptr->s_O_ENTRY_D_time);
  fprintf(debug_fp,"\ts_O_CARRIER_ID = %d (%X)\n",
          ordstat_ptr->s_O_CARRIER_ID, ordstat_ptr->s_O_CARRIER_ID);
  fprintf(debug_fp,"\ts_ol_cnt       = %d (%X)\n",
          ordstat_ptr->s_ol_cnt, ordstat_ptr->s_ol_cnt);
  fprintf(debug_fp,"\ts_transtatus   = %d (%X)\n",
          ordstat_ptr->s_transtatus, ordstat_ptr->s_transtatus);
  fprintf(debug_fp,"\tdeadlocks      = %d (%X)\n",
          ordstat_ptr->deadlocks, ordstat_ptr->deadlocks);

  fprintf(debug_fp,"\titems {\n");
  items = ordstat_ptr->s_ol_cnt;
  for (j = 0; j < items; j++) {
    if(j != 0)
      fprintf(debug_fp,"\n");
    fprintf(debug_fp,"\t\ts_OL_SUPPLY_W_ID[%d] = %d (%X)\n",
            j, ordstat_ptr->item[j].s_OL_SUPPLY_W_ID, ordstat_ptr->item[j].s_OL_SUPPLY_W_ID);
    fprintf(debug_fp,"\t\ts_OL_I_ID[%d]       = %d (%X)\n",
            j, ordstat_ptr->item[j].s_OL_I_ID, ordstat_ptr->item[j].s_OL_I_ID);
    fprintf(debug_fp,"\t\ts_OL_QUANTITY[%d]   = %d (%X)\n",
            j, ordstat_ptr->item[j].s_OL_QUANTITY, ordstat_ptr->item[j].s_OL_QUANTITY);
    fprintf(debug_fp,"\t\ts_OL_AMOUNT[%d]     = %.2f\n",
            j, ordstat_ptr->item[j].s_OL_AMOUNT);
    fprintf(debug_fp,"\t\ts_OL_DELIVERY_D[%d]  = %s \n",
            j,ordstat_ptr->item[j].s_OL_DELIVERY_D_time);
  }
  fprintf(debug_fp,"\t}\n}\n\n");
  fclose(debug_fp);
}


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

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

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

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

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

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

  fprintf(debug_fp,"in_payment_struct {\n");
  fprintf(debug_fp,"\ts_H_AMOUNT     = %.2f \n",
          in_payment->s_H_AMOUNT);
  fprintf(debug_fp,"\ts_C_ID         = %d (%X)\n",
          in_payment->s_C_ID, in_payment->s_C_ID);
  fprintf(debug_fp,"\ts_W_ID         = %d (%X)\n",
          in_payment->s_W_ID, in_payment->s_W_ID);
  fprintf(debug_fp,"\ts_D_ID         = %d (%X)\n",
          in_payment->s_D_ID, in_payment->s_D_ID);
  fprintf(debug_fp,"\ts_C_D_ID       = %d (%X)\n",
          in_payment->s_C_D_ID, in_payment->s_C_D_ID);
  fprintf(debug_fp,"\ts_C_W_ID       = %d (%X)\n",
          in_payment->s_C_W_ID, in_payment->s_C_W_ID);
  fprintf(debug_fp,"\ts_C_LAST       = %s\n",
          in_payment->s_C_LAST);
  fprintf(debug_fp,"\n}\n\n");

  fprintf(debug_fp,"out_payment_struct {\n");
  fprintf(debug_fp,"\ts_C_CREDIT_LIM = %.2f\n",
          payment_ptr->s_C_CREDIT_LIM);
  fprintf(debug_fp,"\ts_C_DISCOUNT   = %04.4f\n",
          payment_ptr->s_C_DISCOUNT);
  fprintf(debug_fp,"\ts_C_BALANCE    = %.2f\n",
          payment_ptr->s_C_BALANCE);
  fprintf(debug_fp,"\ts_C_ID         = %d (%X)\n",
          payment_ptr->s_C_ID, payment_ptr->s_C_ID);
  fprintf(debug_fp,"\ts_W_STREET_1   = %s\n",
          payment_ptr->s_W_STREET_1);
  fprintf(debug_fp,"\ts_W_STREET_2   = %s\n",
          payment_ptr->s_W_STREET_2);
  fprintf(debug_fp,"\ts_W_CITY       = %s\n",
          payment_ptr->s_W_CITY);
  fprintf(debug_fp,"\ts_W_STATE      = %s\n",
          payment_ptr->s_W_STATE);
  fprintf(debug_fp,"\ts_W_ZIP        = %s\n",
          payment_ptr->s_W_ZIP);
  fprintf(debug_fp,"\ts_D_STREET_1   = %s\n",
          payment_ptr->s_D_STREET_1);
  fprintf(debug_fp,"\ts_D_STREET_2   = %s\n",
          payment_ptr->s_D_STREET_2);
  fprintf(debug_fp,"\ts_D_CITY       = %s\n",
          payment_ptr->s_D_CITY);
  fprintf(debug_fp,"\ts_D_STATE      = %s\n",
          payment_ptr->s_D_STATE);
  fprintf(debug_fp,"\ts_D_ZIP        = %s\n",
          payment_ptr->s_D_ZIP);
  fprintf(debug_fp,"\ts_C_FIRST      = %s\n",
          payment_ptr->s_C_FIRST);
  fprintf(debug_fp,"\ts_C_MIDDLE     = %s\n",
          payment_ptr->s_C_MIDDLE);
  fprintf(debug_fp,"\ts_C_LAST       = %s\n",
          payment_ptr->s_C_LAST);
  fprintf(debug_fp,"\ts_C_STREET_1   = %s\n",
          payment_ptr->s_C_STREET_1);
  fprintf(debug_fp,"\ts_C_STREET_2   = %s\n",
          payment_ptr->s_C_STREET_2);
  fprintf(debug_fp,"\ts_C_CITY       = %s\n",
          payment_ptr->s_C_CITY);
  fprintf(debug_fp,"\ts_C_STATE      = %s\n",
          payment_ptr->s_C_STATE);
  fprintf(debug_fp,"\ts_C_ZIP        = %s\n",
          payment_ptr->s_C_ZIP);
  fprintf(debug_fp,"\ts_C_PHONE      = %s\n",
          payment_ptr->s_C_PHONE);
  fprintf(debug_fp,"\ts_C_SINCE      = %s \n",
          payment_ptr->s_C_SINCE_time);
  fprintf(debug_fp,"\ts_C_CREDIT     = %s\n",
          payment_ptr->s_C_CREDIT);
```

```
  fprintf(debug_fp,"\ts_C_DATA       = %s\n",
          payment_ptr->s_C_DATA);
  fprintf(debug_fp,"\ts_transtatus   = %d (%X)\n",
          payment_ptr->s_transtatus,payment_ptr->s_transtatus);
  fprintf(debug_fp,"\tdeadlocks      = %d (%X)\n",
          payment_ptr->deadlocks,payment_ptr->deadlocks);
  fprintf(debug_fp,"\n}\n\n");
  fclose(debug_fp);
}


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

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

/*-----------------------------------------------------------------*/
/* stk_print                                                       */
/*-----------------------------------------------------------------*/
void stk_print (struct out_stocklev_struct *stocklev,
                struct in_stocklev_struct  *in_stocklev,
                char *filename,
                char *msg)
{
  FILE *debug_fp;
  char  timeStamp[27];

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

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

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

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

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

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

Src.Common/tpccmisc.c

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

/*
 *
 * tpccmisc.c - Miscellaneous routines
 *
 */

#include <windows.h>

#define RAND_A  16807
#define RAND_M  2147483647
#define RAND_M1 2147483646
#define RAND_MD 2147483647.0
#define RAND_Q  127773
#define RAND_R  2836

static int seed = 1;
static int seedflag = 0;

void srandom(int);
int random(void);
double current_time_ms(void);
double current_time(void);

void srandom (int initial_seed)
{
  seed = initial_seed;
  if ((seed < 1) || (seed > RAND_M1)) seed = 1;
}

int random (void)
{
  int lo;
  int hi;
  int test;

  hi = seed / RAND_Q;
  lo = seed % RAND_Q;
  test = RAND_A * lo - RAND_R * hi;
  if (test > 0) seed = test;
  else seed = test + RAND_M;

  return (seed);
}

/* Current time in SECONDS, precision SECONDS */
double current_time(void)
{
  /* truncate fractional seconds -> seconds */
  return (double)((int)(current_time_ms()));
}

/* Current time in SECONDS, precision MILLISECONDS */
double current_time_ms(void)
{
  /* GetCurrentTime() returns ms */
  /* convert to fractional seconds */
  return (GetCurrentTime() / 1000);
}
```

include/db2tpcc.h

```
/*****************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****************************************************************************/

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

#ifndef __DB2TPCC_H
#define __DB2TPCC_H

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

#include "lval.h"

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

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

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

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

#define MIN_WAREHOUSE 1
#define MAX_WAREHOUSE WAREHOUSES

/*****************************************************************************/
/* NURand Constants                                                         */
/* C_C_LAST_RUN and C_C_LAST_LOAD must adhere to clause 2.1.6.              */
/* Analysis indicates that a C_LAST delta of 85 is optimal.                 */
/*****************************************************************************/
#define C_C_LAST_RUN          88
#define C_C_LAST_LOAD         173
#define C_C_ID            319
```

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

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

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

#define SPGENERAL_PAD 3
#define SPGENERAL_ADJUST sizeof(int16_t)

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

struct out_neword_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  struct   items_struct {
    float  s_I_PRICE;
    float  s_OL_AMOUNT;
    int16_t  s_S_QUANTITY;
    int16_t  pad2;
    char     s_I_NAME[25];
    char     s_brand_generic;
  } item[15];
  float  s_W_TAX;
  float  s_D_TAX;
  float  s_C_DISCOUNT;
  float  s_total_amount;
  int32_t  s_O_ID;
  int16_t  s_O_OL_CNT;
  int16_t  s_transtatus;
  int16_t  deadlocks;
  char     s_C_LAST[17];
  char     s_C_CREDIT[3];
  char     s_O_ENTRY_D_time[27];

};

struct in_payment_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  float  s_H_AMOUNT;
  int32_t  s_W_ID;
  int32_t  s_C_W_ID;
  int32_t  s_C_ID;
  int16_t  s_C_D_ID;
  int16_t  s_D_ID;
  char     s_C_LAST[17];
};

struct out_payment_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  double  s_C_CREDIT_LIM;
  double  s_C_BALANCE;
  float  s_C_DISCOUNT;
  int32_t  s_C_ID;
  int16_t  s_transtatus;
  int16_t  deadlocks;
  char     s_W_STREET_1[21];
  char     s_W_STREET_2[21];
  char     s_W_CITY[21];
  char     s_W_STATE[3];
  char     s_W_ZIP[10];
  char     s_D_STREET_1[21];
  char     s_D_STREET_2[21];
  char     s_D_CITY[21];
  char     s_D_STATE[3];
  char     s_D_ZIP[10];
  char     s_C_FIRST[17];
  char     s_C_MIDDLE[3];
  char     s_C_LAST[17];
  char     s_C_STREET_1[21];
  char     s_C_STREET_2[21];
  char     s_C_CITY[21];
  char     s_C_STATE[3];
  char     s_C_ZIP[10];
  char     s_C_PHONE[17];
  char     s_C_CREDIT[3];
  char     s_C_DATA[201];
  char     s_H_DATE_time[27];
  char     s_C_SINCE_time[27];
};

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

struct out_ordstat_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  double  s_C_BALANCE;
  int32_t  s_C_ID;
  int32_t  s_O_ID;
  int16_t  s_O_CARRIER_ID;
  int16_t  s_ol_cnt;
  int16_t  pad1[2];
  struct   oitems_struct {
    double  s_OL_AMOUNT;
    int32_t  s_OL_I_ID;
    int32_t  s_OL_SUPPLY_W_ID;
    int16_t  s_OL_QUANTITY;
    int16_t  pad2;
    char     s_OL_DELIVERY_D_time[27];
  } item[15];
  int16_t  s_transtatus;
  int16_t  deadlocks;
  char     s_C_FIRST[17];
  char     s_C_MIDDLE[3];
  char     s_C_LAST[17];
  char     s_O_ENTRY_D_time[27];
  int16_t  pad3[2];
};

struct in_delivery_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  int32_t  s_W_ID;
  int16_t  s_O_CARRIER_ID;
};

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

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

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

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

#ifdef __cplusplus
extern "C" {
#endif

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

#ifdef __cplusplus
}
#endif

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

#ifdef __cplusplus
extern "C" {
#endif

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

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

#ifdef __cplusplus
}
#endif

#endif // __DB2TPCC_H
```

include/lval.h

```
/* lval.h - generated automatically at 20080316.1411 */

#ifndef __LVAL_H
#define __LVAL_H
#define WAREHOUSES 55980
#define DISTRICTS_PER_WAREHOUSE 10
#define CUSTOMERS_PER_DISTRICT 3000
#define ITEMS 100000
#define STOCK_PER_WAREHOUSE 100000
#define MIN_OL_PER_ORDER 5
#define MAX_OL_PER_ORDER 15
#define NU_ORDERS_PER_DISTRICT 900
#endif // __LVAL_H
```

include/tpccapp.h

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

/*
 * tpccapp.h - Application Macros
 *
 */

#ifndef __TPCCAPP_H
#define __TPCCAPP_H

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

#include "sqlenv.h"
#define daricall __stdcall

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

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

/*****************************************************************************
  FUNCTION: SwapEndian
  PURPOSE: Swap the byte order of a structure
  EXAMPLE: int I=0x12345678; SWAP_BYTE(I); I => 0x78563412;
  IMPLEMENTATION: Fold Addr in half, swap header & tail by XOR op
          e.g.: *a = 0x12 [ Addr + 0];
                *b = 0x78 [ Add + 4 - 0 - 1 = Addr+3];
                *a ^= *b;        // sets *a to 0x6A
                *b ^= *a;        // sets *b to 0x12
                *a ^= *b;        // sets *a to 0x78

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

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

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


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

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

/* ************************************************************************* */
/* In NOT ATOMIC COMPOUND SQL, all statements will be executed, but not    */
/* all will neccessarily complete successfully.  We can use sqlerrd(4) to  */
/* determine how many statements succeeded, but this won't tell us what    */
/* statements failed.  In order to determine this, we need to look at      */
/* sqlerrmc, which has the following structure:  HHHXNNNSSSSSXNNNSSSSS...  */
/* (See the docs for more details.)  Since we're interested in the first   */
/* failing statement, we can look at elements 5 and 6, which will contain  */
/* the first two digits of NNN (which is right-padded with spaces).  We    */
/* need to look at the first two digits since some of our compound blocks  */
/* have > 9 statements.  We convert these digits from ASCII to an int and  */
/* set 'last' to this value.                                                */
/* ************************************************************************* */

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

#endif // __TPCCAPP_H
```

include/tpccdbg.h

```
/*****************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****************************************************************************/

/*
 * tpccdbg.h - Debugging Macros
 *
 */

#ifndef __TPCCDBG_H
#define __TPCCDBG_H

#ifdef __cplusplus
extern "C" {
#endif

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

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

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

#ifdef __cplusplus
}
#endif

#endif // __TPCCDBG_H
```

tpccenv.bat

```
@REM ***********************************************************************
@REM  Licensed Materials - Property of IBM
@REM
@REM  Governed under the terms of the International
@REM  License Agreement for Non-Warranted Sample Code.
@REM
@REM  (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
@REM  All Rights Reserved.
@REM
@REM  US Government Users Restricted Rights - Use, duplication or
@REM  disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
@REM ***********************************************************************
@REM
@REM tpccenv.bat - Windows Environment Setup
@REM

@REM The Kit Version
set TPCC_VERSION=CK080131
```

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

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

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

@REM Specifies whether or not to use dari stored proc's for the TPC-C driver.  Set to either
DARIVERSION or NONDARI;
@REM set TPCC_SPTYPE=NOSP
@REM set TPCC_SPTYPE=SPGENERAL2
set TPCC_SPTYPE=SPGENERAL
@REM set TPCC_SPTYPE=DARI2SQLDA

set DB2VERSION=v8

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

@REM DB2 EE/EEE Configuration
set DB2EDITION=EE
@REM set DB2EDITION=EEE
set DB2NODE=0
@REM set to the number of nodes you have.  Set to 1 for EE.
set DB2NODES=1

@REM TPCC General Configuration
@REM ** IMPORTANT NOTE **
@REM The kit is not guaranteed to work properly if TPCC_ROOT or TPCC_SQLLIB
@REM have spaces in them.  If you absolutely must use paths with spaces,
@REM then the entire path must be surrounded by double quotes.
@REM For example:  HOME="C:\Program Files\IBM"
set HOME=c:\home\tpcc
set TPCC_DBNAME=TPCC
set TPCC_ROOT=c:\home\tpcc\tpc-c.ibm
set TPCC_SQLLIB=c:\Progra~1\IBM\SQLLIB
set TPCC_RUNDATA=c:\home\tpcc\tpccdata

@REM TPCC Debug Configuration
@REM This is the path where all error and debug logs are placed.
@REM To get debugging from within the stored procedures, you must
@REM set DB2ENVLIST="TPCC_DEBUGDIR" in tpcc.config.
set TPCC_DEBUGDIR=c:\temp

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

A.2 Client Transaction Code

Makefile

Src.Srv/Makefile

```
- ignore failures from 'db2 bind' command
```

Makefile.config

```
##############################################################################
## Licensed Materials - Property of IBM
##
## Governed under the terms of the International
## License Agreement for Non-Warranted Sample Code.
##
## (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
## All Rights Reserved.
##
## US Government Users Restricted Rights - Use, duplication or
## disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
##############################################################################

#
# Makefile.config - AIX 64-bit
#
#

# Make Configuration
MAKE=make

# Compiler Configuration.
# CFLAGS_DEBUG may be set to "-g", "-DDEBUGIT" "-g -DDEBUGIT" or left blank
CC=xlc
CFLAGS_OS=-qflag=i:i -qlanglvl=ansi -qcpluscmt -DSQLUNIX -DSQLAIX -q64 -O3 -D_LARGE_FILES
CFLAGS_OUT=-o
CFLAGS_DEBUG=

# Linker Configuration
LD_EXEC=xlc
LD_STORP=xlc
LDFLAGS_EXEC=-lm -q64
LDFLAGS_SHLIB=-qmkshrobj
LDFLAGS_STORP=$(LDFLAGS_SHLIB) -bE:$@.exp -lc -b64
LDFLAGS_LIB=-L$(TPCC_SQLLIB)/lib -ldb2
LDFLAGS_OUT=-o

# Library Configuration
AR=ar
ARFLAGS=-r -v -X64
ARFLAGS_LIB=
ARFLAGS_OUT=

# OS Commands
ERASE=rm -f
ERASEDIR=$(ERASE) -R
MOVE=mv
COPY=cp

# OS File Extensions & Path Separators
OBJEXT=.o
LIBEXT=.a
SHLIBEXT=.a
BINEXT=
SLASH=/
CMDSEP=;
```

Src.Common/Makefile

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

#
# Makefile - Makefile for Src.Common
#
#

include $(TPCC_ROOT)/Makefile.config

# ###########################################################################
# Preprocessor, Compiler and LInker Flags
# ###########################################################################

BND_OPTS =   GRANT PUBLIC \
             MESSAGES $*.bnd.msg
PRP_OPTS =   BINDFILE \
             OPTLEVEL 1 \
             ISOLATION RR \
             MESSAGES $*.prep.msg \
             LEVEL $(TPCC_VERSION) \
             NOLINEMACRO

INCLUDE =    -I$(TPCC_SQLLIB)/include -I$(TPCC_ROOT)/include

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

UTIL_OBJ_DBG =     tpccdbg$(OBJEXT)
UTIL_OBJ_GEN =     tpccmisc$(OBJEXT)
UTIL_OBJ_DB2 =     tpccctx$(OBJEXT)

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

all:     $(UTIL_OBJ_DBG) $(UTIL_OBJ_GEN) connect $(UTIL_OBJ_DB2) disconnect

dbgen:  $(UTIL_OBJ_GEN)

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

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

connect:
         - db2 connect to $(TPCC_DBNAME)

disconnect:
         - db2 connect reset
         - db2 terminate

rebind:  connect
         db2 bind tpccctx.bnd $(BND_OPTS)

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

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

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

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

# Source
tpccdbg$(OBJEXT):	tpccdbg.c
tpccctx$(OBJEXT):	tpccctx.c
tpccmisc$(OBJEXT):	tpccmisc.c

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

Src.Common/tpccctx.sqc

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

/*
 *
 * tpccctx.sqc - TPCC context code
 *
 */

#include <string.h>
#include <sqlutil.h>
#include "db2tpcc.h"
#include "tpccdbg.h"

int connect_to_TM(char *in_dbname);
int connect_to_TM_auth(char *in_dbname, char *in_username, char *in_password);
int disconnect_from_TM(void);

int connect_to_TM(char *in_dbname)
{
  return connect_to_TM_auth(in_dbname, "", "");
}

int connect_to_TM_auth(char *in_dbname, char *in_username, char *in_password)
{
  SQL_STRUCTURE sqlca  sqlca;
  int ConnectSQLCODE = 0;

  EXEC SQL BEGIN DECLARE SECTION;
  char dbname[9];
  char username[129];
  char password[15];
  EXEC SQL END DECLARE SECTION;

  /* Copy 9 characters - 8 for dbname, 1 for NULL */
  strncpy(dbname,in_dbname,9);
  if (strcmp(in_username,"") == 0)
  {
    EXEC SQL CONNECT TO :dbname IN SHARE MODE;
  } else {
    strncpy(username,in_username,128);
    strncpy(password,in_password,14);
    EXEC SQL CONNECT TO :dbname IN SHARE MODE USER :username USING :password;
  }

  ConnectSQLCODE = SQLCODE;
  if (ConnectSQLCODE != 0)
  {
    sqlerror( CLIENT_SQL, "CONNECT", __FILE__, __LINE__, &sqlca);

    return ConnectSQLCODE;
  }

  return 0;
}

int disconnect_from_TM(void)
{
  SQL_STRUCTURE sqlca  sqlca;
  int DisconnectSQLCODE = 0;

  EXEC SQL CONNECT RESET;

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

  if (DisconnectSQLCODE) {
    return DisconnectSQLCODE;
  }
  return 0;
}
```

Src.Common/tpccdbg.c

```
/*************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*************************************************************************/

/*
 * tccdbg.c - Debugging Routines
 *
 */

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

#include "sqlca.h"
#include "sql.h"
#include "db2tpcc.h"
#include "tpccdbg.h"

#define DEBUG_FILENAME_SZ 128
#define DEBUG_PATH_SIZE 128

void	del_print();
void	new_print();
void	ord_print();
void	pay_print();
void	stk_print();

void current_tmstmp(char *buf);

static int debugInit = 0;
static char debugPath[DEBUG_PATH_SIZE] = "";

/*-----------------------------------------------------------------*/
/*	InitializeDebug	*/
/*-----------------------------------------------------------------*/
__inline void InitializeDebug(void) {
  if (debugInit == 0) {
    char *p = getenv("TPCC_DEBUGDIR");
    if (p) {
      strncpy(debugPath, p, DEBUG_PATH_SIZE);
    } else {
      strcpy(debugPath, "/tmp");
    }
    strcat(debugPath, "/");
  }
  debugInit = 1;
}

/*-----------------------------------------------------------------*/
/*	sqlerror	*/
/*-----------------------------------------------------------------*/
void sqlerror(int tranType, char *msg, char *file, int line, SQL_STRUCTURE sqlca *psqlca)
{
  FILE *err_fp = NULL;
  char  err_fn[DEBUG_PATH_SIZE + DEBUG_FILENAME_SZ];
  char  tranName[16];
  int j,k;
  char  timeStamp[27];
  char  errStr[512] = "";

  InitializeDebug();
  strncpy(err_fn, debugPath, DEBUG_PATH_SIZE);
  current_tmstmp(&timeStamp[0]);
  timeStamp[19] = (char)NULL;

  switch(tranType)
  {
    case NEWORD_SQL:
      // sprintf(err_fn, "%d.err.out", getpid());
      strcat(err_fn, "new.err.out");
      strcpy(tranName, "NEW_ORDER");
      break;

    case DELIVERY_SQL:
      // sprintf(err_fn, "%d.err.out", getpid());
      strcat(err_fn, "del.err.out");
      strcpy(tranName, "DELIVERY");
      break;

    case PAYMENT_SQL:
      // sprintf(err_fn, "%d.err.out", getpid());
      strcat(err_fn, "pay.err.out");
      strcpy(tranName, "PAYMENT");
      break;

    case ORDSTAT_SQL:
      // sprintf(err_fn, "%d.err.out", getpid());
      strcat(err_fn, "ord.err.out");
      strcpy(tranName, "ORDER_STAT");
      break;

    case STOCKLEV_SQL:
```

```
      //sprintf(err_fn, "%d.err.out", getpid());
      strcat(err_fn, "stk.err.out");
      strcpy(tranName, "STOCK_LVL");
      break;

    case 0:
      strcat(err_fn, "cli.err.out");
      strcpy(tranName, "CLIENT");
      break;

    default:
      return;

  }

  /* Generate Formatted Error Message */
  sqlaintp(errStr, 512, 78, psqlca);

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

  fprintf(err_fp, "----------------------------------------\n");
  fprintf(err_fp, "Transaction: %s (%s)\n", tranName, msg);
  fprintf(err_fp, "FILE %s (%u)\n", file, line);
  fprintf(err_fp, "SQLCODE %d ", psqlca->sqlcode);
  fprintf(err_fp, "PID %d ", getpid());
  fprintf(err_fp, "TIME %s\n", timeStamp);
  fprintf(err_fp, "----------------------------------------\n");
  fprintf(err_fp, "%s", errStr);
  fprintf(err_fp, "----------------------------------------\n");

  if (psqlca->sqlerrmc[0] != ' ' || psqlca->sqlerrmc[1] != ' ')
  {
    fprintf(err_fp, "slerrmc:  ");

    for(j = 0; j < 5; j++)
    {
      for(k = 0; k < 16; k++) {
        int pos = j * 16 + k;
        if (pos < 70) fprintf(err_fp, "%02x ", psqlca->sqlerrmc[pos]);
        else fprintf(err_fp, "   ");
      }
      fprintf(err_fp,"   |");
      for(k = 0; k < 16; k++) {
        int pos = j * 16 + k;
        char c = ' ';
        if (pos < 70) {
          c = psqlca->sqlerrmc[pos];
          if (!isprint(c)) c = ' ';
        }
        fprintf(err_fp,"%c", c);
      }
      fprintf(err_fp,"|\n");
      if (j < 4) fprintf(err_fp,"          ");
    }
  }

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

  fprintf(err_fp, "sqlerrd: ");
  for(j = 0; j < 6; j++)
    fprintf(err_fp, " %d", psqlca->sqlerrd[j]);
  fprintf(err_fp,"\n");

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

  fprintf(err_fp, "\n");

  fclose(err_fp);
}


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

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

/*----------------------------------------------------------*/
/* del_print                                                */
/*----------------------------------------------------------*/
void del_print (struct out_delivery_struct *delivery_ptr,
                struct in_delivery_struct  *in_delivery,
                char *filename,
                char *msg)
{
  FILE *debug_fp;
  char  timeStamp[27];
  int   j;

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

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

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

  fprintf(debug_fp,"in_delivery_struct {\n");
  fprintf(debug_fp,"\ts_W_ID         = %d (%X)\n",
          in_delivery->s_W_ID, in_delivery->s_W_ID);
  fprintf(debug_fp,"\ts_O_CARRIER_ID = %d (%X)\n",
          in_delivery->s_O_CARRIER_ID, in_delivery->s_O_CARRIER_ID);
  fprintf(debug_fp,"}\n\n");

  fprintf(debug_fp,"out_delivery_struct {\n");
  fprintf(debug_fp,"\ts_transtatus   = %d (%X)\n",
          delivery_ptr->s_transtatus,delivery_ptr->s_transtatus);
  fprintf(debug_fp,"\tdeadlocks      = %d (%X)\n",
          delivery_ptr->deadlocks,delivery_ptr->deadlocks);

  for (j = 0; j < 10; j++) {
    fprintf(debug_fp,"\t\ts_O_ID[%d]        = %d\n",
            j, delivery_ptr->s_O_ID[j]);
  }
  fprintf(debug_fp,"\t}\n}\n\n");
  fclose(debug_fp);
}


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

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

/*----------------------------------------------------------*/
/* new_print                                                */
/*----------------------------------------------------------*/
void new_print (struct out_neword_struct *neword_ptr,
                struct in_neword_struct *in_neword,
                char *filename,
                char *msg)
{
  FILE *debug_fp;
  char  timeStamp[27];
  int   j, items;

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

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

  fprintf(debug_fp,"New order debug information follows %s (%s)\n", timeStamp, msg);
  fprintf(debug_fp, " PID %d ", getpid());
  fprintf(debug_fp,"\n==================================================\n");

  fprintf(debug_fp,"in_neword_struct {\n");

  fprintf(debug_fp,"\ts_C_ID        = %d (%X)\n",
          in_neword->s_C_ID, in_neword->s_C_ID);
  fprintf(debug_fp,"\ts_W_ID        = %d (%X)\n",
          in_neword->s_W_ID, in_neword->s_W_ID);
  fprintf(debug_fp,"\ts_D_ID        = %d (%X)\n",
          in_neword->s_D_ID, in_neword->s_D_ID);
  fprintf(debug_fp,"\ts_O_OL_CNT    = %d (%X)\n",
          in_neword->s_O_OL_CNT, in_neword->s_O_OL_CNT);
  fprintf(debug_fp,"\ts_all_local   = %d (%X)\n",
          in_neword->s_all_local, in_neword->s_all_local);
// fprintf(debug_fp,"\ts_transtatus   = %d (%X)\n",
//         in_neword->s_transtatus, in_neword->s_transtatus);
// fprintf(debug_fp,"\tduplicate_items= %d (%X)\n",
//         in_neword->duplicate_items, in_neword->duplicate_items);

  fprintf(debug_fp,"\titems {\n");
  items = in_neword->s_O_OL_CNT;
  for (j=0; j<items; j++) {
    if(j != 0)
      fprintf(debug_fp,"\n");
    fprintf(debug_fp,"\t\ts_OL_I_ID[%d]       = %d (%X)\n",
            j, in_neword->in_item[j].s_OL_I_ID, in_neword->in_item[j].s_OL_I_ID);
    fprintf(debug_fp,"\t\ts_OL_SUPPLY_W_ID[%d] = %d (%X)\n",
            j, in_neword->in_item[j].s_OL_SUPPLY_W_ID, in_neword->in_item[j].s_OL_SUPPLY_W_ID);
    fprintf(debug_fp,"\t\ts_OL_QUANTITY[%d]    = %d (%X)\n",
            j, in_neword->in_item[j].s_OL_QUANTITY, in_neword->in_item[j].s_OL_QUANTITY);
  }
  fprintf(debug_fp,"\t}\n}\n\n");

  fprintf(debug_fp,"out_neword_struct {\n");
  fprintf(debug_fp,"\ts_C_LAST      = %s\n",
          neword_ptr->s_C_LAST);
  fprintf(debug_fp,"\ts_C_CREDIT    = %s\n",
          neword_ptr->s_C_CREDIT);
```

```
  fprintf(debug_fp,"\ts_W_TAX       = %04.4f \n",
          neword_ptr->s_W_TAX);
  fprintf(debug_fp,"\ts_D_TAX       = %04.4f \n",
          neword_ptr->s_D_TAX);
  fprintf(debug_fp,"\ts_C_DISCOUNT   = %04.4f \n",
          neword_ptr->s_C_DISCOUNT);
  fprintf(debug_fp,"\ts_O_ID        = %d (%X)\n",
          neword_ptr->s_O_ID, neword_ptr->s_O_ID);
  fprintf(debug_fp,"\ts_O_OL_CNT     = %d (%X)\n",
          neword_ptr->s_O_OL_CNT, neword_ptr->s_O_OL_CNT);
  fprintf(debug_fp,"\ts_O_ENTRY_D    = %s \n",
          neword_ptr->s_O_ENTRY_D_time);
  fprintf(debug_fp,"\ts_total_amount = %.2f \n",
          neword_ptr->s_total_amount);
  fprintf(debug_fp,"\ts_transtatus   = %d (%X)\n",
          neword_ptr->s_transtatus, neword_ptr->s_transtatus);
  fprintf(debug_fp,"\tdeadlocks      = %d (%X)\n",
          neword_ptr->deadlocks, neword_ptr->deadlocks);

// fprintf(debug_fp,"\ts_W_ID        = %d (%X)\n",
//         neword_ptr->s_W_ID, neword_ptr->s_W_ID);
// fprintf(debug_fp,"\ts_D_ID        = %d (%X)\n",
//         neword_ptr->s_D_ID, neword_ptr->s_D_ID);
// fprintf(debug_fp,"\ts_all_local   = %d (%X)\n",
//         neword_ptr->s_all_local, neword_ptr->s_all_local);
// fprintf(debug_fp,"\tduplicate_items= %d (%X)\n",
//         neword_ptr->duplicate_items, neword_ptr->duplicate_items);

  fprintf(debug_fp,"\titems {\n");
  items = neword_ptr->s_O_OL_CNT;
  for (j=0; j<items; j++) {
    if(j != 0)
      fprintf(debug_fp,"\n");
    fprintf(debug_fp,"\t\ts_I_NAME[%d]        = %s\n",
            j, neword_ptr->item[j].s_I_NAME);
    fprintf(debug_fp,"\t\ts_I_PRICE[%d]       = %.2f \n",
            j, neword_ptr->item[j].s_I_PRICE);
    fprintf(debug_fp,"\t\ts_OL_AMOUNT[%d]     = %.2f \n",
            j, neword_ptr->item[j].s_OL_AMOUNT);
    fprintf(debug_fp,"\t\ts_S_QUANTITY[%d]    = %d (%X)\n",
            j, neword_ptr->item[j].s_S_QUANTITY, neword_ptr->item[j].s_S_QUANTITY);
    fprintf(debug_fp,"\t\ts_brand_generic[%d]  = %c\n",
            j, neword_ptr->item[j].s_brand_generic);
  }
  fprintf(debug_fp,"\t}\n}\n\n");
  fclose(debug_fp);
}



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

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

/*-----------------------------------------------------------------*/
/* ord_print                                                       */
/*-----------------------------------------------------------------*/
void ord_print (struct out_ordstat_struct *ordstat_ptr,
                struct in_ordstat_struct  *in_ordstat,
                char *filename,
                char *msg)
{
  FILE *debug_fp;
  char  timeStamp[27];
  int   j, items;

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

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

  fprintf(debug_fp,"Order status debug information follows %s (%s)\n", timeStamp, msg);
  fprintf(debug_fp, " PID %d ", getpid());
  fprintf(debug_fp,"\n==================================================\n");

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

  fprintf(debug_fp,"out_ordstat_struct {\n");
  fprintf(debug_fp,"\ts_C_ID        = %d (%X)\n",
          ordstat_ptr->s_C_ID, ordstat_ptr->s_C_ID);
  fprintf(debug_fp,"\ts_C_FIRST     = %s\n",
          ordstat_ptr->s_C_FIRST);
  fprintf(debug_fp,"\ts_C_MIDDLE    = %s\n",
          ordstat_ptr->s_C_MIDDLE);
  fprintf(debug_fp,"\ts_C_LAST      = %s\n",
          ordstat_ptr->s_C_LAST);
  fprintf(debug_fp,"\ts_C_BALANCE   = %.2f\n",
          ordstat_ptr->s_C_BALANCE);
  fprintf(debug_fp,"\ts_O_ID        = %d (%X)\n",
          ordstat_ptr->s_O_ID, ordstat_ptr->s_O_ID);
  fprintf(debug_fp,"\ts_O_ENTRY_D   = %s \n",
          ordstat_ptr->s_O_ENTRY_D_time);
  fprintf(debug_fp,"\ts_O_CARRIER_ID = %d (%X)\n",
          ordstat_ptr->s_O_CARRIER_ID, ordstat_ptr->s_O_CARRIER_ID);
  fprintf(debug_fp,"\ts_ol_cnt      = %d (%X)\n",
          ordstat_ptr->s_ol_cnt, ordstat_ptr->s_ol_cnt);
  fprintf(debug_fp,"\ts_transtatus  = %d (%X)\n",
          ordstat_ptr->s_transtatus, ordstat_ptr->s_transtatus);
  fprintf(debug_fp,"\tdeadlocks     = %d (%X)\n",
          ordstat_ptr->deadlocks, ordstat_ptr->deadlocks);

  fprintf(debug_fp,"\titems {\n");
  items = ordstat_ptr->s_ol_cnt;
  for (j = 0; j < items; j++) {
    if(j != 0)
      fprintf(debug_fp,"\n");
    fprintf(debug_fp,"\t\ts_OL_SUPPLY_W_ID[%d] = %d (%X)\n",
            j, ordstat_ptr->item[j].s_OL_SUPPLY_W_ID, ordstat_ptr->item[j].s_OL_SUPPLY_W_ID);
    fprintf(debug_fp,"\t\ts_OL_I_ID[%d]       = %d (%X)\n",
            j, ordstat_ptr->item[j].s_OL_I_ID, ordstat_ptr->item[j].s_OL_I_ID);
    fprintf(debug_fp,"\t\ts_OL_QUANTITY[%d]   = %d (%X)\n",
            j, ordstat_ptr->item[j].s_OL_QUANTITY, ordstat_ptr->item[j].s_OL_QUANTITY);
    fprintf(debug_fp,"\t\ts_OL_AMOUNT[%d]     = %.2f\n",
            j, ordstat_ptr->item[j].s_OL_AMOUNT);
    fprintf(debug_fp,"\t\ts_OL_DELIVERY_D[%d]  = %s \n",
            j,ordstat_ptr->item[j].s_OL_DELIVERY_D_time);
  }
  fprintf(debug_fp,"\t}\n}\n\n");
  fclose(debug_fp);
}

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

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

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

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

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

  fprintf(debug_fp,"Payment debug information follows %s (%s)\n", timeStamp, msg);
  fprintf(debug_fp, " PID %d ", getpid());
  fprintf(debug_fp,"\n==================================================\n");

  fprintf(debug_fp,"in_payment_struct {\n");
  fprintf(debug_fp,"\ts_H_AMOUNT     = %.2f \n",
          in_payment->s_H_AMOUNT);
  fprintf(debug_fp,"\ts_C_ID        = %d (%X)\n",
          in_payment->s_C_ID, in_payment->s_C_ID);
  fprintf(debug_fp,"\ts_W_ID        = %d (%X)\n",
          in_payment->s_W_ID, in_payment->s_W_ID);
  fprintf(debug_fp,"\ts_D_ID        = %d (%X)\n",
          in_payment->s_D_ID, in_payment->s_D_ID);
  fprintf(debug_fp,"\ts_C_D_ID      = %d (%X)\n",
          in_payment->s_C_D_ID, in_payment->s_C_D_ID);
  fprintf(debug_fp,"\ts_C_W_ID      = %d (%X)\n",
          in_payment->s_C_W_ID, in_payment->s_C_W_ID);
  fprintf(debug_fp,"\ts_C_LAST      = %s\n",
          in_payment->s_C_LAST);
  fprintf(debug_fp,"\n}\n\n");

  fprintf(debug_fp,"out_payment_struct {\n");
  fprintf(debug_fp,"\ts_C_CREDIT_LIM = %.2f\n",
          payment_ptr->s_C_CREDIT_LIM);
  fprintf(debug_fp,"\ts_C_DISCOUNT   = %04.4f\n",
          payment_ptr->s_C_DISCOUNT);
  fprintf(debug_fp,"\ts_C_BALANCE    = %.2f\n",
          payment_ptr->s_C_BALANCE);
  fprintf(debug_fp,"\ts_C_ID        = %d (%X)\n",
          payment_ptr->s_C_ID, payment_ptr->s_C_ID);
  fprintf(debug_fp,"\ts_W_STREET_1   = %s\n",
          payment_ptr->s_W_STREET_1);
  fprintf(debug_fp,"\ts_W_STREET_2   = %s\n",
          payment_ptr->s_W_STREET_2);
  fprintf(debug_fp,"\ts_W_CITY      = %s\n",
          payment_ptr->s_W_CITY);
  fprintf(debug_fp,"\ts_W_STATE     = %s\n",
          payment_ptr->s_W_STATE);
  fprintf(debug_fp,"\ts_W_ZIP       = %s\n",
```

```
        payment_ptr->s_W_ZIP);
  fprintf(debug_fp,"\ts_D_STREET_1   = %s\n",
        payment_ptr->s_D_STREET_1);
  fprintf(debug_fp,"\ts_D_STREET_2   = %s\n",
        payment_ptr->s_D_STREET_2);
  fprintf(debug_fp,"\ts_D_CITY      = %s\n",
        payment_ptr->s_D_CITY);
  fprintf(debug_fp,"\ts_D_STATE     = %s\n",
        payment_ptr->s_D_STATE);
  fprintf(debug_fp,"\ts_D_ZIP       = %s\n",
        payment_ptr->s_D_ZIP);
  fprintf(debug_fp,"\ts_C_FIRST     = %s\n",
        payment_ptr->s_C_FIRST);
  fprintf(debug_fp,"\ts_C_MIDDLE    = %s\n",
        payment_ptr->s_C_MIDDLE);
  fprintf(debug_fp,"\ts_C_LAST      = %s\n",
        payment_ptr->s_C_LAST);
  fprintf(debug_fp,"\ts_C_STREET_1   = %s\n",
        payment_ptr->s_C_STREET_1);
  fprintf(debug_fp,"\ts_C_STREET_2   = %s\n",
        payment_ptr->s_C_STREET_2);
  fprintf(debug_fp,"\ts_C_CITY      = %s\n",
        payment_ptr->s_C_CITY);
  fprintf(debug_fp,"\ts_C_STATE     = %s\n",
        payment_ptr->s_C_STATE);
  fprintf(debug_fp,"\ts_C_ZIP       = %s\n",
        payment_ptr->s_C_ZIP);
  fprintf(debug_fp,"\ts_C_PHONE     = %s\n",
        payment_ptr->s_C_PHONE);
  fprintf(debug_fp,"\ts_C_SINCE     = %s \n",
        payment_ptr->s_C_SINCE_time);
  fprintf(debug_fp,"\ts_C_CREDIT    = %s\n",
        payment_ptr->s_C_CREDIT);
  fprintf(debug_fp,"\ts_C_DATA      = %s\n",
        payment_ptr->s_C_DATA);
  fprintf(debug_fp,"\ts_transtatus   = %d (%X)\n",
        payment_ptr->s_transtatus,payment_ptr->s_transtatus);
  fprintf(debug_fp,"\tdeadlocks     = %d (%X)\n",
        payment_ptr->deadlocks,payment_ptr->deadlocks);
  fprintf(debug_fp,"\n}\n\n");
  fclose(debug_fp);
}


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

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

/*-----------------------------------------------------------*/
/* stk_print                                                 */
/*-----------------------------------------------------------*/
void stk_print (struct out_stocklev_struct *stocklev,
            struct in_stocklev_struct  *in_stocklev,
            char *filename,
            char *msg)
{
  FILE *debug_fp;
  char  timeStamp[27];

  current_tmstmp(&timeStamp[0]);
  timeStamp[19] = (char)NULL;
  if ((debug_fp = fopen(filename, "a+")) == NULL)
  {
    return;
  }

  fprintf(debug_fp,"Stock level debug information follows %s (%s)\n", timeStamp, msg);
  fprintf(debug_fp, " PID %d ", getpid());
  fprintf(debug_fp,"\n==============================================\n");

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

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

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

Src.Common/tpccmisc.c

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

/*
 *
 * tpccmisc.c - Miscellaneous routines
 *
 */

#include <stdlib.h>
#include <sys/types.h>
#include <sys/time.h>

double current_time_ms(void);
double current_time(void);


/* Current time in SECONDS, precision SECONDS */
double current_time(void)
{
 /* use time() to get seconds */
 return(time(NULL));
}

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

Src.Srv/Makefile

```
#############################################################################
## Licensed Materials - Property of IBM
##
## Governed under the terms of the International
## License Agreement for Non-Warranted Sample Code.
##
## (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
## All Rights Reserved.
##
## US Government Users Restricted Rights - Use, duplication or
## disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#############################################################################

#
# Makefile - Makefile for Src.Srv
#
#

include $(TPCC_ROOT)/Makefile.config

# ###########################################################################
# Preprocessor, Compiler and Linker Flags
# ###########################################################################

BND_OPTS =   GRANT PUBLIC \
             MESSAGES $*.bnd.msg
PRP_OPTS =   BINDFILE \
             EXPLAIN ALL \
             MESSAGES $*.prep.msg

INCLUDE =    -I$(TPCC_SQLLIB)/include -I$(TPCC_ROOT)/include

CFLAGS =     $(CFLAGS_OS) $(INCLUDE) $(CFLAGS_DEBUG) \
             -D$(DB2EDITION) -D$(DB2VERSION) \
             -DSQLA_NOLINES -DLINT_ARGS

LDFLAGS =    $(LDFLAGS_STORP) $(LDFLAGS_LIB)

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

STORED_PROCS =  new ord del

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

EXE =        news ords dels

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

all:     connect explain catalog $(EXE) install plan disconnect
```

```
clean:   connect uncatalog unexplain disconnect
         - $(ERASE) $(TPCC_SPDIR)$(SLASH)news
         - $(ERASE) $(TPCC_SPDIR)$(SLASH)ords
         - $(ERASE) $(TPCC_SPDIR)$(SLASH)dels
         - $(ERASE) *.bnd *.msg *.out *$(OBJEXT) $(EXE) tpcc_all_sql.c
         - $(ERASE) TPCC_ALL.*.plan

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

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

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

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

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

connect:
         - db2 connect to $(TPCC_DBNAME)

disconnect:
         - db2 connect reset
         - db2 terminate

plan:
         - db2exfmt -d $(TPCC_DBNAME) -e $(TPCC_SCHEMA) -s $(TPCC_SCHEMA) -w -1 -n
TPCC_ALL -g -# 0 -o TPCC_ALL.exfmt.plan
         - (export DB2EXPLN_BUFFER=3000000; db2expln -d $(TPCC_DBNAME) -c $(TPCC_SCHEMA)
-p TPCC_ALL -s 0 -g -o TPCC_ALL.expln.plan )

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

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

install:  $(EXE)
         - mkdir $(TPCC_SPDIR)
         $(COPY) ords $(TPCC_SPDIR)
         $(COPY) news $(TPCC_SPDIR)
         $(COPY) dels $(TPCC_SPDIR)

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

.SUFFIXES: $(OBJEXT) .c .sqc

# d230437mte: QUERYOPT 7 required for UNION ALL
# Only stock needs CS , and that can be specified on the SELECT statement
tpcc_all_sql.c:
         @echo "Prepping $*.sqc"
         -db2 prep $*.sqc $(PRP_OPTS) ISOLATION RR
         @echo "Binding $*.bnd"
         -db2 bind $*.bnd $(BND_OPTS) QUERYOPT 7

# Stored procedures are built in a special way

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

$(EXE): $(UTIL_OBJ) tpcc_all_sql.o
         $(LD_STORP) $(LDFLAGS) $(UTIL_OBJ) tpcc_all_sql.o $(LDFLAGS_OUT)$@

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

# Executables (Stored Procedures)
$(EXE): $(UTIL_OBJ) tpcc_all_sql.o

# Source
tpcc_all_sql$(OBJEXT): tpcc_all_sql.c

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

Src.Srv/cat-func.ddl

```
-----------------------------------------------------------------------
-- Licensed Materials - Property of IBM
--
-- Governed under the terms of the International
-- License Agreement for Non-Warranted Sample Code.
--
-- (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
-- All Rights Reserved.
--
-- US Government Users Restricted Rights - Use, duplication or
-- disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
-----------------------------------------------------------------------
--
-- cat-func.ddl - Create table functions
--
--


--
-- DELIVERY
--

CREATE FUNCTION DEL(   W_ID          INTEGER
                     , D_ID          SMALLINT
                     , CARRIER_ID    SMALLINT
                    )

RETURNS TABLE ( O_ID INTEGER )

SPECIFIC DELIVERY

MODIFIES SQL DATA DETERMINISTIC NO EXTERNAL ACTION LANGUAGE SQL

VAR: BEGIN ATOMIC

 DECLARE O_ID   INTEGER ;
 DECLARE C_ID   INTEGER ;
 DECLARE AMOUNT DECIMAL(12,2) ;

 /* Delete the order from new order table */

 SET VAR.O_ID = ( SELECT NO_O_ID

                 FROM OLD TABLE ( DELETE

                         FROM ( SELECT NO_O_ID

                                FROM NEW_ORDER

                                WHERE NO_W_ID = DEL.W_ID

                                  AND NO_D_ID = DEL.D_ID

                                ORDER BY NO_O_ID ASC

                                FETCH FIRST 1 ROW ONLY
                              ) AS NEW_ORDER
                         ) AS D
                 )
 ;

 /* Update the order as delivered and retrieve the customer id */

 SET VAR.C_ID = ( SELECT O_C_ID

                 FROM OLD TABLE ( UPDATE ORDERS

                         SET O_CARRIER_ID = DEL.CARRIER_ID

                        WHERE O_W_ID = DEL.W_ID
                          AND O_D_ID = DEL.D_ID
                          AND O_ID   = VAR.O_ID
                         ) AS U
                 )
 ;

 SET VAR.AMOUNT = ( SELECT SUM( OL_AMOUNT )

                 FROM OLD TABLE ( UPDATE ORDER_LINE

                         SET OL_DELIVERY_D = CURRENT TIMESTAMP

                        WHERE OL_W_ID = DEL.W_ID
                          AND OL_D_ID = DEL.D_ID
                          AND OL_O_ID = VAR.O_ID
                         ) AS U
                 )
 ;

 /* Charge the customer */

 UPDATE CUSTOMER

   SET  C_BALANCE      = C_BALANCE + VAR.AMOUNT
      , C_DELIVERY_CNT = C_DELIVERY_CNT + SMALLINT( 1 )

  WHERE C_W_ID = DEL.W_ID
    AND C_D_ID = DEL.D_ID
    AND C_ID   = VAR.C_ID
 ;

 /* Return the order id to the caller (or NULL) */

 RETURN VALUES VAR.O_ID ;

END
%


--
-- ORDER STATUS
--

CREATE FUNCTION ORD_C_LAST(   W_ID INTEGER
                            , D_ID SMALLINT
                            , C_LAST VARCHAR(16)
                           )

RETURNS TABLE(   O_ID        INTEGER
               , O_CARRIER_ID SMALLINT
               , O_ENTRY_D    TIMESTAMP
               , C_BALANCE    DECIMAL(12,2)
               , C_FIRST      VARCHAR(16)
               , C_MIDDLE     CHAR(2)
```

```
            , C_ID        INTEGER
          )

SPECIFIC ORD_C_LAST

READS SQL DATA NO EXTERNAL ACTION DETERMINISTIC LANGUAGE SQL

VAR: BEGIN ATOMIC

 DECLARE C_BALANCE    DECIMAL(12,2) ;
 DECLARE C_FIRST      VARCHAR(16) ;
 DECLARE C_MIDDLE     CHAR(2) ;
 DECLARE C_ID         INTEGER ;
 DECLARE O_ID         INTEGER;
 DECLARE O_CARRIER_ID SMALLINT;
 DECLARE O_ENTRY_D    TIMESTAMP;

 /* Retrieve the Customer information */

 SET ( C_BALANCE, C_FIRST, C_MIDDLE, C_ID )

  = ( SELECT  C_BALANCE, C_FIRST, C_MIDDLE , C_ID

       FROM ( SELECT   C_ID
                    , C_BALANCE
                    , C_FIRST
                    , C_MIDDLE
                    , NUM
                    , MAX(NUM) AS COUNT

              FROM ( SELECT C_ID, C_BALANCE, C_FIRST, C_MIDDLE, ROWNUMBER() OVER
(ORDER BY C_FIRST) AS NUM
                      FROM CUSTOMER
                      WHERE C_W_ID = ORD_C_LAST.W_ID
                      AND C_D_ID = ORD_C_LAST.D_ID
                      AND C_LAST = ORD_C_LAST.C_LAST
                    ) AS V2

              GROUP BY C_ID, C_BALANCE, C_FIRST, C_MIDDLE, NUM

            ) AS V1

       WHERE NUM = (COUNT + BIGINT( 1 ) ) / BIGINT( 2 )
    )
;

 SET ( O_ID , O_CARRIER_ID , O_ENTRY_D )
  =    ( SELECT   O_ID
             , O_CARRIER_ID
             , O_ENTRY_D

          FROM ORDERS

         WHERE O_W_ID = ORD_C_LAST.W_ID
           AND O_D_ID = ORD_C_LAST.D_ID
           AND O_C_ID = VAR.C_ID

         ORDER BY O_ID DESC
         FETCH FIRST 1 ROW ONLY

      )
;


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

```
END
%

CREATE FUNCTION ORD_C_ID(   W_ID INTEGER
                          , D_ID SMALLINT
                          , C_ID INTEGER
                         )

RETURNS TABLE(   O_ID         INTEGER
            , O_CARRIER_ID SMALLINT
            , O_ENTRY_D    TIMESTAMP
            , C_BALANCE    DECIMAL(12,2)
            , C_FIRST      VARCHAR(16)
            , C_MIDDLE     CHAR(2)
            , C_LAST       VARCHAR(16)
          )

SPECIFIC ORD_C_ID

READS SQL DATA NO EXTERNAL ACTION DETERMINISTIC LANGUAGE SQL

VAR: BEGIN ATOMIC

 DECLARE C_BALANCE    DECIMAL(12,2) ;
 DECLARE C_FIRST      VARCHAR(16) ;
 DECLARE C_MIDDLE     CHAR(2) ;
 DECLARE C_LAST       VARCHAR(16) ;
 DECLARE O_ID         INTEGER;
 DECLARE O_CARRIER_ID SMALLINT;
 DECLARE O_ENTRY_D    TIMESTAMP;

 /* Retrieve the Customer information */

 SET ( C_BALANCE, C_FIRST, C_MIDDLE, C_LAST )

  = ( SELECT  C_BALANCE, C_FIRST, C_MIDDLE, C_LAST

       FROM CUSTOMER

      WHERE C_ID   = ORD_C_ID.C_ID
        AND C_W_ID = ORD_C_ID.W_ID
        AND C_D_ID = ORD_C_ID.D_ID
    )
;

 SET (O_ID, O_CARRIER_ID, O_ENTRY_D)

  =    ( SELECT   O_ID
             , O_CARRIER_ID
             , O_ENTRY_D

          FROM ORDERS

              WHERE O_W_ID = ORD_C_ID.W_ID
                AND O_D_ID = ORD_C_ID.D_ID
                AND O_C_ID = ORD_C_ID.C_ID

         ORDER BY O_ID DESC
         FETCH FIRST 1 ROW ONLY
       )
;

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

```
END
%



--
-- PAYMENT
--

CREATE FUNCTION PAY_C_LAST(  W_ID INTEGER
                          , D_ID SMALLINT
                          , C_W_ID INTEGER
                          , C_D_ID SMALLINT
                          , C_LAST VARCHAR(16)
                          , H_AMOUNT DECIMAL(6,2)
                          , BAD_CREDIT_PREFIX VARCHAR(28)
                         )

RETURNS TABLE(   W_STREET_1 CHAR(20)
            , W_STREET_2 CHAR(20)
            , W_CITY CHAR(20)
            , W_STATE CHAR(2)
            , W_ZIP CHAR(9)
            , D_STREET_1 CHAR(20)
            , D_STREET_2 CHAR(20)
            , D_CITY CHAR(20)
            , D_STATE CHAR(2)
            , D_ZIP CHAR(9)
            , C_ID INTEGER
            , C_FIRST VARCHAR(16)
            , C_MIDDLE CHAR(2)
            , C_STREET_1 VARCHAR(20)
            , C_STREET_2 VARCHAR(20)
            , C_CITY VARCHAR(20)
            , C_STATE CHAR(2)
            , C_ZIP CHAR(9)
            , C_PHONE CHAR(16)
            , C_SINCE TIMESTAMP
            , C_CREDIT CHAR(2)
            , C_CREDIT_LIM DECIMAL(12,2)
            , C_DISCOUNT INTEGER
            , C_BALANCE DECIMAL(12,2)
            , C_DATA CHAR(200)
            , H_DATE TIMESTAMP
          )

SPECIFIC PAY_C_LAST

MODIFIES SQL DATA DETERMINISTIC NO EXTERNAL ACTION LANGUAGE SQL

VAR: BEGIN ATOMIC

 DECLARE  W_NAME CHAR(10) ;
 DECLARE  D_NAME CHAR(10) ;

 DECLARE  W_STREET_1 CHAR(20) ;
 DECLARE  W_STREET_2 CHAR(20) ;
 DECLARE  W_CITY CHAR(20) ;
 DECLARE  W_STATE CHAR(2) ;
 DECLARE  W_ZIP CHAR(9) ;

 DECLARE  D_STREET_1 CHAR(20) ;
 DECLARE  D_STREET_2 CHAR(20) ;
 DECLARE  D_CITY CHAR(20) ;
 DECLARE  D_STATE CHAR(2) ;
 DECLARE  D_ZIP CHAR(9) ;

 DECLARE  C_ID INTEGER ;

 DECLARE  C_FIRST VARCHAR(16) ;
 DECLARE  C_MIDDLE CHAR(2) ;
 DECLARE  C_STREET_1 VARCHAR(20) ;
 DECLARE  C_STREET_2 VARCHAR(20) ;
```

```
  DECLARE  C_CITY VARCHAR(20) ;
  DECLARE  C_STATE CHAR(2) ;
  DECLARE  C_ZIP CHAR(9) ;
  DECLARE  C_PHONE CHAR(16) ;
  DECLARE  C_SINCE TIMESTAMP ;
  DECLARE  C_CREDIT CHAR(2) ;
  DECLARE  C_CREDIT_LIM DECIMAL(12,2) ;
  DECLARE  C_DISCOUNT INTEGER ;
  DECLARE  C_BALANCE DECIMAL(12,2) ;
  DECLARE  C_DATA CHAR(200) ;

  DECLARE  H_DATE TIMESTAMP ;

  /* Generate the current date and time for the payment date */
  SET H_DATE = CURRENT TIMESTAMP ;

  /* Update District and retieve its data */

  SET ( D_NAME, D_STREET_1, D_STREET_2, D_CITY, D_STATE, D_ZIP)

    =  ( SELECT   D_NAME, D_STREET_1, D_STREET_2, D_CITY, D_STATE, D_ZIP

        FROM OLD TABLE ( UPDATE DISTRICT

                        SET D_YTD = D_YTD + PAY_C_LAST.H_AMOUNT

                       WHERE D_W_ID = PAY_C_LAST.W_ID
                         AND D_ID   = PAY_C_LAST.D_ID
                     ) AS U
     )
;

  /* Determine the C_ID */

  SET ( C_ID )

    = ( SELECT C_ID
        FROM ( SELECT   C_ID
                  , NUM
                  , MAX(NUM) AS COUNT

             FROM ( SELECT C_ID, C_BALANCE, C_FIRST, C_MIDDLE, ROWNUMBER() OVER (ORDER
BY C_FIRST) AS NUM
                  FROM CUSTOMER
                  WHERE C_W_ID = PAY_C_LAST.W_ID
                  AND C_D_ID = PAY_C_LAST.D_ID
                  AND C_LAST = PAY_C_LAST.C_LAST
                ) AS V2

             GROUP BY C_ID, C_BALANCE, C_FIRST, C_MIDDLE, NUM

            ) AS V1

        WHERE NUM = (COUNT + BIGINT( 1 ) ) / BIGINT( 2 )
     )
;


  /* Update the middle customer */

  SET (   C_ID, C_FIRST, C_MIDDLE, C_STREET_1, C_STREET_2
       , C_CITY, C_STATE, C_ZIP, C_PHONE, C_SINCE, C_CREDIT, C_CREDIT_LIM
       , C_DISCOUNT, C_BALANCE, C_DATA )

    = ( SELECT   C_ID, C_FIRST, C_MIDDLE, C_STREET_1, C_STREET_2
            , C_CITY, C_STATE, C_ZIP, C_PHONE, C_SINCE, C_CREDIT, C_CREDIT_LIM
            , C_DISCOUNT, C_BALANCE
            , CASE WHEN C_CREDIT = 'BC' THEN SUBSTR(C_DATA, 1, 200) ELSE NULL END AS
C_DATA

        FROM NEW TABLE ( UPDATE CUSTOMER

                        SET   C_BALANCE     = C_BALANCE - PAY_C_LAST.H_AMOUNT
                            , C_YTD_PAYMENT = C_YTD_PAYMENT + PAY_C_LAST.H_AMOUNT
                            , C_PAYMENT_CNT = C_PAYMENT_CNT + SMALLINT( 1 )

                            , C_DATA = CASE WHEN C_CREDIT = 'BC'
                                      THEN    CHAR( C_ID )            -- 11 bytes long
                                          || BAD_CREDIT_PREFIX        -- 28 bytes long
                                          || SUBSTR( C_DATA, 1, 461 ) -- 461 + 39 = 500
                                      ELSE C_DATA
                                    END

                       WHERE C_W_ID = PAY_C_LAST.C_W_ID
                         AND C_D_ID = PAY_C_LAST.C_D_ID
                         AND C_ID   = VAR.C_ID
                     ) AS U
     )
;

  /* Update the warehouse */

  SET ( W_NAME, W_STREET_1, W_STREET_2, W_CITY, W_STATE,  W_ZIP )

   = ( SELECT W_NAME, W_STREET_1, W_STREET_2, W_CITY, W_STATE,  W_ZIP

        FROM OLD TABLE ( UPDATE WAREHOUSE

                        SET W_YTD = W_YTD + PAY_C_LAST.H_AMOUNT

                       WHERE W_ID = PAY_C_LAST.W_ID
                     ) AS U
     )
;

  /* Finally insert into the warehouse */

  INSERT

    INTO HISTORY ( H_C_ID, H_C_D_ID, H_C_W_ID, H_D_ID, H_W_ID, H_DATA, H_DATE,
H_AMOUNT )

    VALUES (   VAR.C_ID
             , PAY_C_LAST.C_D_ID
             , PAY_C_LAST.C_W_ID
             , PAY_C_LAST.D_ID
             , PAY_C_LAST.W_ID
             , VAR.W_NAME || CHAR( '    ', 4 )  || VAR.D_NAME
             , VAR.H_DATE
             , PAY_C_LAST.H_AMOUNT
           )
;

  /* Done - return the collected data */

  RETURN VALUES (   W_STREET_1, W_STREET_2, W_CITY, W_STATE,  W_ZIP
                  , D_STREET_1, D_STREET_2, D_CITY, D_STATE, D_ZIP
                  , C_ID, C_FIRST, C_MIDDLE, C_STREET_1, C_STREET_2
                  , C_CITY, C_STATE, C_ZIP, C_PHONE, C_SINCE, C_CREDIT, C_CREDIT_LIM
                  , C_DISCOUNT, C_BALANCE, C_DATA, H_DATE
                )
;

END
%

CREATE FUNCTION PAY_C_ID(  W_ID INTEGER
                  , D_ID SMALLINT
                  , C_W_ID INTEGER
                  , C_D_ID SMALLINT
                  , C_ID INTEGER
                  , H_AMOUNT DECIMAL(6,2)
                  , BAD_CREDIT_PREFIX VARCHAR(34)
                  )

RETURNS TABLE(   W_STREET_1 CHAR(20)
                  , W_STREET_2 CHAR(20)
                  , W_CITY CHAR(20)
                  , W_STATE CHAR(2)
                  , W_ZIP CHAR(9)
                  , D_STREET_1 CHAR(20)
                  , D_STREET_2 CHAR(20)
                  , D_CITY CHAR(20)
                  , D_STATE CHAR(2)
                  , D_ZIP CHAR(9)
                  , C_LAST VARCHAR(16)
                  , C_FIRST VARCHAR(16)
                  , C_MIDDLE CHAR(2)
                  , C_STREET_1 VARCHAR(20)
                  , C_STREET_2 VARCHAR(20)
                  , C_CITY VARCHAR(20)
                  , C_STATE CHAR(2)
                  , C_ZIP CHAR(9)
                  , C_PHONE CHAR(16)
                  , C_SINCE TIMESTAMP
                  , C_CREDIT CHAR(2)
                  , C_CREDIT_LIM DECIMAL(12,2)
                  , C_DISCOUNT REAL
                  , C_BALANCE DECIMAL(12,2)
                  , C_DATA CHAR(200)
                  , H_DATE TIMESTAMP
                 )
SPECIFIC PAY_C_ID

MODIFIES SQL DATA DETERMINISTIC NO EXTERNAL ACTION LANGUAGE SQL

VAR: BEGIN ATOMIC

  DECLARE  W_NAME CHAR(10) ;
  DECLARE  D_NAME CHAR(10) ;

  DECLARE  W_STREET_1 CHAR(20) ;
  DECLARE  W_STREET_2 CHAR(20) ;
  DECLARE  W_CITY CHAR(20) ;
  DECLARE  W_STATE CHAR(2) ;
  DECLARE  W_ZIP CHAR(9) ;

  DECLARE  D_STREET_1 CHAR(20) ;
  DECLARE  D_STREET_2 CHAR(20) ;
  DECLARE  D_CITY CHAR(20) ;
  DECLARE  D_STATE CHAR(2) ;
  DECLARE  D_ZIP CHAR(9) ;

  DECLARE  C_LAST VARCHAR(16) ;

  DECLARE  C_FIRST VARCHAR(16) ;
  DECLARE  C_MIDDLE CHAR(2) ;
  DECLARE  C_STREET_1 VARCHAR(20) ;
  DECLARE  C_STREET_2 VARCHAR(20) ;
  DECLARE  C_CITY VARCHAR(20) ;
  DECLARE  C_STATE CHAR(2) ;
  DECLARE  C_ZIP CHAR(9) ;
  DECLARE  C_PHONE CHAR(16) ;
  DECLARE  C_SINCE TIMESTAMP ;
  DECLARE  C_CREDIT CHAR(2) ;
  DECLARE  C_CREDIT_LIM DECIMAL(12,2) ;
  DECLARE  C_DISCOUNT REAL ;
  DECLARE  C_BALANCE DECIMAL(12,2) ;
  DECLARE  C_DATA CHAR(200) ;
  DECLARE  H_DATE TIMESTAMP;

  /* Generate the current date and time for the payment date */
  SET H_DATE = CURRENT TIMESTAMP;

  /* Update District and retieve its data */

  SET ( D_NAME, D_STREET_1, D_STREET_2, D_CITY, D_STATE, D_ZIP )

    =  ( SELECT   D_NAME, D_STREET_1, D_STREET_2, D_CITY, D_STATE, D_ZIP
```

```
        FROM OLD TABLE ( UPDATE DISTRICT

                        SET D_YTD = D_YTD + PAY_C_ID.H_AMOUNT

                       WHERE D_W_ID = PAY_C_ID.W_ID
                        AND D_ID   = PAY_C_ID.D_ID
                     ) AS U
    )
;

 /* Update the middle customer */

 SET (   C_LAST, C_FIRST, C_MIDDLE, C_STREET_1, C_STREET_2
      , C_CITY, C_STATE, C_ZIP, C_PHONE, C_SINCE, C_CREDIT, C_CREDIT_LIM
      , C_DISCOUNT, C_BALANCE, C_DATA )

  = ( SELECT   C_LAST, C_FIRST, C_MIDDLE, C_STREET_1, C_STREET_2
            , C_CITY, C_STATE, C_ZIP, C_PHONE, C_SINCE, C_CREDIT, C_CREDIT_LIM
            , C_DISCOUNT, C_BALANCE
            , CASE WHEN C_CREDIT = 'BC' THEN SUBSTR(C_DATA, 1, 200) ELSE NULL END AS
C_DATA

        FROM NEW TABLE ( UPDATE CUSTOMER

                       SET   C_BALANCE     = C_BALANCE - PAY_C_ID.H_AMOUNT
                          , C_YTD_PAYMENT = C_YTD_PAYMENT + PAY_C_ID.H_AMOUNT
                          , C_PAYMENT_CNT = C_PAYMENT_CNT + SMALLINT( 1 )

                          , C_DATA = CASE WHEN C_CREDIT = 'BC'
                                     THEN    BAD_CREDIT_PREFIX        -- 34 bytes long
                                        || SUBSTR( C_DATA, 1, 466 ) -- 466 + 34 = 500 bytes
                                     ELSE C_DATA
                                   END

                      WHERE C_W_ID = PAY_C_ID.C_W_ID
                       AND C_D_ID = PAY_C_ID.C_D_ID
                       AND C_ID   = PAY_C_ID.C_ID
                    ) AS U
    )
;

 /* Update the warehouse */

 SET ( W_NAME, W_STREET_1, W_STREET_2, W_CITY, W_STATE,  W_ZIP )

  = ( SELECT W_NAME, W_STREET_1, W_STREET_2, W_CITY, W_STATE,  W_ZIP

        FROM OLD TABLE ( UPDATE WAREHOUSE

                        SET W_YTD = W_YTD + PAY_C_ID.H_AMOUNT

                       WHERE W_ID = PAY_C_ID.W_ID
                     ) AS U
    )
;

 /* Finally insert into the warehouse */

 INSERT

   INTO HISTORY ( H_C_ID, H_C_D_ID, H_C_W_ID, H_D_ID, H_W_ID, H_DATA, H_DATE,
H_AMOUNT )

   VALUES (   PAY_C_ID.C_ID
          , PAY_C_ID.C_D_ID
          , PAY_C_ID.C_W_ID
          , PAY_C_ID.D_ID
          , PAY_C_ID.W_ID
          , VAR.W_NAME || CHAR( '    ', 4 ) || VAR.D_NAME
          , VAR.H_DATE
          , PAY_C_ID.H_AMOUNT
        )
 ;

 /* Done - return the collected data */

 RETURN VALUES (   W_STREET_1, W_STREET_2, W_CITY, W_STATE,  W_ZIP
              , D_STREET_1, D_STREET_2, D_CITY, D_STATE, D_ZIP
              , C_LAST, C_FIRST, C_MIDDLE, C_STREET_1, C_STREET_2
              , C_CITY, C_STATE, C_ZIP, C_PHONE, C_SINCE, C_CREDIT, C_CREDIT_LIM
              , C_DISCOUNT, C_BALANCE, C_DATA, H_DATE
             )
 ;

END
%

--
-- NEW ORDER
--

CREATE FUNCTION NEW_OL_ALL(   I_ID      INT
                            , I_QTY     SMALLINT
                            , W_ID      INT
                            , SUPP_W_ID INT
                            , O_ID      INT
                            , D_ID      SMALLINT
                           )

RETURNS TABLE(   I_PRICE       DECIMAL(5,2)
               , I_NAME        CHAR(24)
               , I_DATA        VARCHAR(50)
               , OL_DIST_INFO  CHAR(24)
               , S_DATA        VARCHAR(50)
               , S_QUANTITY    SMALLINT
              )

SPECIFIC NEW_OL_ALL

MODIFIES SQL DATA DETERMINISTIC NO EXTERNAL ACTION LANGUAGE SQL

VAR: BEGIN ATOMIC

 DECLARE I_PRICE       DECIMAL(5,2) ;
 DECLARE I_NAME        CHAR(24)  ;
 DECLARE I_DATA        VARCHAR(50)  ;
 DECLARE OL_DIST_INFO  CHAR(24)  ;
 DECLARE S_DATA        VARCHAR(50)  ;
 DECLARE S_QUANTITY    SMALLINT  ;


 SET   ( I_PRICE , I_NAME , I_DATA )

  =  ( SELECT
            I_PRICE
          , I_NAME
          , I_DATA

       FROM ITEM

      WHERE ITEM.I_ID = NEW_OL_ALL.I_ID
    ) ;

  SET ( OL_DIST_INFO , S_DATA , S_QUANTITY )

   = ( SELECT   OL_DIST_INFO
          , S_DATA
          , S_QUANTITY

        FROM NEW TABLE (    UPDATE STOCK

                       INCLUDE (  OL_DIST_INFO CHAR( 24 )  )

                         SET S_QUANTITY = CASE WHEN S_QUANTITY - NEW_OL_ALL.I_QTY >= 10
                                               THEN S_QUANTITY - NEW_OL_ALL.I_QTY
                                               ELSE S_QUANTITY - NEW_OL_ALL.I_QTY + 91
                                          END

                         , S_ORDER_CNT = S_ORDER_CNT + SMALLINT( 1 )

                         , S_YTD = S_YTD + NEW_OL_ALL.I_QTY

                         , S_REMOTE_CNT = CASE WHEN NEW_OL_ALL.SUPP_W_ID =
NEW_OL_ALL.W_ID
                                               THEN S_REMOTE_CNT
                                               ELSE S_REMOTE_CNT + SMALLINT( 1 )
                                          END

                         , OL_DIST_INFO = CASE D_ID WHEN SMALLINT(  1 )  THEN
S_DIST_01
                                                 WHEN SMALLINT(  2 )  THEN S_DIST_02
                                                 WHEN SMALLINT(  3 )  THEN S_DIST_03
                                                 WHEN SMALLINT(  4 )  THEN S_DIST_04
                                                 WHEN SMALLINT(  5 )  THEN S_DIST_05
                                                 WHEN SMALLINT(  6 )  THEN S_DIST_06
                                                 WHEN SMALLINT(  7 )  THEN S_DIST_07
                                                 WHEN SMALLINT(  8 )  THEN S_DIST_08
                                                 WHEN SMALLINT(  9 )  THEN S_DIST_09
                                                 WHEN SMALLINT( 10 )  THEN S_DIST_10
                                            END
                      WHERE S_I_ID = NEW_OL_ALL.I_ID
                       AND S_W_ID = NEW_OL_ALL.SUPP_W_ID
                    ) AS U
    )
  ;

  RETURN VALUES(   VAR.I_PRICE
                 , VAR.I_NAME
                 , VAR.I_DATA
                 , VAR.OL_DIST_INFO
                 , VAR.S_DATA
                 , VAR.S_QUANTITY
                )
  ;

END
%

CREATE FUNCTION NEW_OL_LOCAL(   I_ID      INT
                              , I_QTY     SMALLINT
                              , W_ID      INT
                              , O_ID      INT
                              , D_ID      SMALLINT
                             )

RETURNS TABLE(   I_PRICE       DECIMAL(5,2)
               , I_NAME        CHAR(24)
               , I_DATA        VARCHAR(50)
               , OL_DIST_INFO  CHAR(24)
               , S_DATA        VARCHAR(50)
               , S_QUANTITY    SMALLINT
              )

SPECIFIC NEW_OL_LOCAL

MODIFIES SQL DATA DETERMINISTIC NO EXTERNAL ACTION LANGUAGE SQL

VAR: BEGIN ATOMIC

 DECLARE I_PRICE       DECIMAL(5,2)  ;
 DECLARE I_NAME        CHAR(24)  ;
 DECLARE I_DATA        VARCHAR(50)  ;
 DECLARE OL_DIST_INFO  CHAR(24)  ;
 DECLARE S_DATA        VARCHAR(50)  ;
 DECLARE S_QUANTITY    SMALLINT  ;


 SET   ( I_PRICE , I_NAME , I_DATA )
```

```
  =  ( SELECT
           I_PRICE
         , I_NAME
         , I_DATA

       FROM ITEM

      WHERE ITEM.I_ID = NEW_OL_LOCAL.I_ID
    ) ;

  SET ( OL_DIST_INFO , S_DATA , S_QUANTITY )

   = ( SELECT   OL_DIST_INFO
         , S_DATA
         , S_QUANTITY

       FROM NEW TABLE (    UPDATE STOCK

                    INCLUDE (  OL_DIST_INFO CHAR( 24 )  )

                      SET S_QUANTITY = CASE WHEN S_QUANTITY - NEW_OL_LOCAL.I_QTY >= 10
                                  THEN S_QUANTITY - NEW_OL_LOCAL.I_QTY
                                  ELSE S_QUANTITY - NEW_OL_LOCAL.I_QTY + 91
                                END

                                , S_ORDER_CNT = S_ORDER_CNT + SMALLINT( 1 )

                                , S_YTD = S_YTD + NEW_OL_LOCAL.I_QTY

                                , OL_DIST_INFO = CASE D_ID WHEN SMALLINT(  1 )  THEN
S_DIST_01
                                            WHEN SMALLINT(  2 )  THEN S_DIST_02
                                            WHEN SMALLINT(  3 )  THEN S_DIST_03
                                            WHEN SMALLINT(  4 )  THEN S_DIST_04
                                            WHEN SMALLINT(  5 )  THEN S_DIST_05
                                            WHEN SMALLINT(  6 )  THEN S_DIST_06
                                            WHEN SMALLINT(  7 )  THEN S_DIST_07
                                            WHEN SMALLINT(  8 )  THEN S_DIST_08
                                            WHEN SMALLINT(  9 )  THEN S_DIST_09
                                            WHEN SMALLINT( 10 )  THEN S_DIST_10
                                        END
                      WHERE S_I_ID = NEW_OL_LOCAL.I_ID
                       AND S_W_ID = NEW_OL_LOCAL.W_ID
                  ) AS U
    )
  ;

  RETURN VALUES(   VAR.I_PRICE
             , VAR.I_NAME
             , VAR.I_DATA
             , VAR.OL_DIST_INFO
             , VAR.S_DATA
             , VAR.S_QUANTITY
           )
  ;

END
%

CREATE FUNCTION NEW_WH (   O_ID       INTEGER
                  , W_ID       INTEGER
                  , D_ID       SMALLINT
                  , C_ID       INTEGER
                  , O_OL_CNT    SMALLINT
                  , O_ALL_LOCAL SMALLINT
                 )

RETURNS TABLE (   W_TAX       REAL
            , C_DISCOUNT  REAL
            , C_LAST      VARCHAR(16)
            , C_CREDIT    CHAR(2)
            , O_ENTRY_D   TIMESTAMP
            )

SPECIFIC NEW_WH

MODIFIES SQL DATA DETERMINISTIC NO EXTERNAL ACTION LANGUAGE SQL

VAR: BEGIN ATOMIC

 DECLARE C_DISCOUNT  REAL ;
 DECLARE C_LAST      VARCHAR(16) ;
 DECLARE C_CREDIT    CHAR(2) ;
 DECLARE W_TAX       REAL ;
 DECLARE O_ENTRY_D   TIMESTAMP;

 SET O_ENTRY_D = CURRENT TIMESTAMP;

 INSERT

  INTO NEW_ORDER ( NO_O_ID, NO_D_ID, NO_W_ID )

       VALUES (   O_ID
                , D_ID
                , W_ID
              )
 ;


 INSERT

  INTO ORDERS ( O_C_ID, O_ENTRY_D, O_CARRIER_ID, O_OL_CNT, O_ALL_LOCAL, O_ID,
O_W_ID, O_D_ID )

      VALUES (   C_ID
               , O_ENTRY_D
               , 0
               , O_OL_CNT
               , O_ALL_LOCAL
               , O_ID
               , W_ID
               , D_ID
             )
 ;

 SET ( C_DISCOUNT, C_LAST, C_CREDIT )

  =  ( SELECT  C_DISCOUNT, C_LAST, C_CREDIT

       FROM  CUSTOMER

      WHERE  C_ID = NEW_WH.C_ID
       AND  C_W_ID = W_ID
       AND  C_D_ID = D_ID
    )
 ;


 SET W_TAX
  =   ( SELECT W_TAX

        FROM WAREHOUSE

       WHERE W_ID = NEW_WH.W_ID
     )
 ;

 RETURN VALUES ( W_TAX , C_DISCOUNT , C_LAST , C_CREDIT, O_ENTRY_D ) ;

END
%
```

Src.Srv/cat-proc.ddl

```
CREATE PROCEDURE news
            (in new_in   varchar(262) FOR BIT DATA,
             out new_out varchar(682) FOR BIT DATA)
            LANGUAGE C
            PARAMETER STYLE GENERAL
            EXTERNAL NAME '/home/tpcc/sqllib/function/news!news'
            not fenced;

CREATE PROCEDURE ords
            (in ord_in   varchar(42) FOR BIT DATA,
             out ord_out varchar(822) FOR BIT DATA)
            LANGUAGE C
            PARAMETER STYLE GENERAL
            EXTERNAL NAME '/home/tpcc/sqllib/function/ords!ords'
            not fenced;

CREATE PROCEDURE dels
            (in del_in   varchar(14) FOR BIT DATA,
             out del_out varchar(50) FOR BIT DATA)
            LANGUAGE C
            PARAMETER STYLE GENERAL
            EXTERNAL NAME '/home/tpcc/sqllib/function/dels!dels'
            not fenced;
```

Src.Srv/dels.exp

```
#! Export file
dels
```

Src.Srv/news.exp

```
#! Export file
news
```

Src.Srv/ords.exp

```
#! Export file
ords
```

Src.Srv/pays.exp

```
#! Export file
pays
```

Src.Srv/stks.exp

```
#! Export file
stks
```

Src.Srv/tpcc_all_sql.sqc

```
/*******************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*******************************************************************************/

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

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

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

// ---------------------------------------------------------------
// New Order  SERVER
// ---------------------------------------------------------------

int  static is_ORIGINAL( char *string, short length ) ;

SQL_API_RC new_order_internal( char *pin, char *pout )
{

  struct out_neword_struct *neword;

  struct in_neword_struct  *in_neword;

  struct sqlca sqlca ;

  int     fbadItemDetected = 0 ;

  EXEC SQL BEGIN DECLARE SECTION;

    char    c_last [ 16 ] ;
    char    c_credit [ 2 ] ;
    float  c_discount ;
    float  dist_tax ;
    float  ware_tax ;

    sqlint32   w_id ;
    short      d_id ;
    sqlint32   c_id ;

    sqlint32   next_o_id ;

    short      s_quantity ;

    sqlint32   supply_w_id ;

    short   inputItemCount ;

    char    stockDistrictInformation [ 24 ];
    char    item_name[ 24 ] ;

    char    o_entry_d [27];

    short   allLocal ;

    float    item_price ;

    struct i_data_type { short len ; char data[ 50 ] ; } i_data ;
    struct s_data_type { short len ; char data[ 50 ] ; } s_data ;

    sqlint32    id0, id1, id2,  id3,  id4,  id5,  id6,  id7;
    sqlint32    id8, id9, id10, id11, id12, id13, id14;

    sqlint32   supply_w_id0,  supply_w_id1, supply_w_id2,  supply_w_id3;
    sqlint32   supply_w_id4,  supply_w_id5, supply_w_id6,  supply_w_id7;
    sqlint32   supply_w_id8,  supply_w_id9, supply_w_id10,  supply_w_id11;
    sqlint32   supply_w_id12,  supply_w_id13, supply_w_id14;

    short   ol_quantity0,  ol_quantity1, ol_quantity2,  ol_quantity3;
    short   ol_quantity4,  ol_quantity5, ol_quantity6,  ol_quantity7;
    short   ol_quantity8,  ol_quantity9, ol_quantity10,  ol_quantity11;
    short   ol_quantity12,  ol_quantity13, ol_quantity14;


  EXEC SQL END DECLARE SECTION;

  int      storedProcRc ;
  int      inputItemArrayIndex ;

  char    stockDistrictInformationArray [15][25];

  #define stockDistrictInformation  stockDistrictInformationArray[ inputItemArrayIndex ]

  // Redirected input fields

  #define w_id in_neword->s_W_ID
  #define d_id in_neword->s_D_ID
  #define c_id in_neword->s_C_ID

  #define inputItemCount in_neword->s_O_OL_CNT

  #define allLocal in_neword->s_all_local

  // Redirected output fields

  #define  c_last       neword->s_C_LAST
  #define  c_credit     neword->s_C_CREDIT
  #define  c_discount   neword->s_C_DISCOUNT
  #define  ware_tax     neword->s_W_TAX
  #define  dist_tax     neword->s_D_TAX
  #define  s_quantity   neword->item[ inputItemArrayIndex ].s_S_QUANTITY
  #define o_entry_d     neword->s_O_ENTRY_D_time

  // This output field becomes an input field to order_line

  #define next_o_id neword->s_O_ID

  // item price/name

  #define  item_name    neword->item[ inputItemArrayIndex ].s_I_NAME

  float    i_priceArray[ 15 ] ;

  #define  item_price   i_priceArray[ inputItemArrayIndex ]

  // Handle the generic/brand distinction

  struct  i_data_type i_dataArray[ 15 ] ;
  struct  s_data_type s_dataArray[ 15 ] ;

  #define  i_data       i_dataArray[ inputItemArrayIndex ]
  #define  s_data       s_dataArray[ inputItemArrayIndex ]

  // Redirect hostvars to input structure

  #define  id0  in_neword->in_item[0].s_OL_I_ID
  #define  id1  in_neword->in_item[1].s_OL_I_ID
  #define  id2  in_neword->in_item[2].s_OL_I_ID
  #define  id3  in_neword->in_item[3].s_OL_I_ID
  #define  id4  in_neword->in_item[4].s_OL_I_ID
  #define  id5  in_neword->in_item[5].s_OL_I_ID
  #define  id6  in_neword->in_item[6].s_OL_I_ID
  #define  id7  in_neword->in_item[7].s_OL_I_ID
  #define  id8  in_neword->in_item[8].s_OL_I_ID
  #define  id9  in_neword->in_item[9].s_OL_I_ID
  #define  id10 in_neword->in_item[10].s_OL_I_ID
  #define  id11 in_neword->in_item[11].s_OL_I_ID
  #define  id12 in_neword->in_item[12].s_OL_I_ID
  #define  id13 in_neword->in_item[13].s_OL_I_ID
  #define  id14 in_neword->in_item[14].s_OL_I_ID

  #define ol_quantity0  in_neword->in_item[  0 ].s_OL_QUANTITY
  #define ol_quantity1  in_neword->in_item[  1 ].s_OL_QUANTITY
  #define ol_quantity2  in_neword->in_item[  2 ].s_OL_QUANTITY
  #define ol_quantity3  in_neword->in_item[  3 ].s_OL_QUANTITY
  #define ol_quantity4  in_neword->in_item[  4 ].s_OL_QUANTITY
  #define ol_quantity5  in_neword->in_item[  5 ].s_OL_QUANTITY
  #define ol_quantity6  in_neword->in_item[  6 ].s_OL_QUANTITY
  #define ol_quantity7  in_neword->in_item[  7 ].s_OL_QUANTITY
  #define ol_quantity8  in_neword->in_item[  8 ].s_OL_QUANTITY
  #define ol_quantity9  in_neword->in_item[  9 ].s_OL_QUANTITY
  #define ol_quantity10 in_neword->in_item[ 10 ].s_OL_QUANTITY
  #define ol_quantity11 in_neword->in_item[ 11 ].s_OL_QUANTITY
  #define ol_quantity12 in_neword->in_item[ 12 ].s_OL_QUANTITY
  #define ol_quantity13 in_neword->in_item[ 13 ].s_OL_QUANTITY
  #define ol_quantity14 in_neword->in_item[ 14 ].s_OL_QUANTITY

  #define supply_w_id0  in_neword->in_item[  0 ].s_OL_SUPPLY_W_ID
  #define supply_w_id1  in_neword->in_item[  1 ].s_OL_SUPPLY_W_ID
  #define supply_w_id2  in_neword->in_item[  2 ].s_OL_SUPPLY_W_ID
  #define supply_w_id3  in_neword->in_item[  3 ].s_OL_SUPPLY_W_ID
  #define supply_w_id4  in_neword->in_item[  4 ].s_OL_SUPPLY_W_ID
  #define supply_w_id5  in_neword->in_item[  5 ].s_OL_SUPPLY_W_ID
  #define supply_w_id6  in_neword->in_item[  6 ].s_OL_SUPPLY_W_ID
  #define supply_w_id7  in_neword->in_item[  7 ].s_OL_SUPPLY_W_ID
  #define supply_w_id8  in_neword->in_item[  8 ].s_OL_SUPPLY_W_ID
  #define supply_w_id9  in_neword->in_item[  9 ].s_OL_SUPPLY_W_ID
  #define supply_w_id10 in_neword->in_item[ 10 ].s_OL_SUPPLY_W_ID
  #define supply_w_id11 in_neword->in_item[ 11 ].s_OL_SUPPLY_W_ID
  #define supply_w_id12 in_neword->in_item[ 12 ].s_OL_SUPPLY_W_ID
  #define supply_w_id13 in_neword->in_item[ 13 ].s_OL_SUPPLY_W_ID
  #define supply_w_id14 in_neword->in_item[ 14 ].s_OL_SUPPLY_W_ID

  EXEC SQL DECLARE ISOL_Remote_1 CURSOR FOR

    WITH DATA AS (  SELECT  O_ID
                  , D_ID
                  , W_ID
                  , OL_NUMBER
                  , I_ID
                  , I_SUPPLY_W_ID
                  , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
                  , I_QTY
                  , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
                  , OL_DIST_INFO
                  , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                      FROM ( SELECT   :next_o_id as O_ID
                                   , :w_id AS W_ID
                                   , :d_id as D_ID
                                   , OL_NUMBER
                                   , I_ID
                                   , I_SUPPLY_W_ID
                                   , I_QTY

                            FROM  Table( VALUES

                                  (  SMALLINT( 1 )        , :id0  , :ol_quantity0  , :supply_w_id0  )
```

```
                    ) AS X ( OL_NUMBER , I_ID  , I_QTY        , I_SUPPLY_W_ID  )
                ) AS ITEMLIST

              , TABLE( NEW_OL_ALL(   I_ID
                                   , I_QTY
                                   , W_ID
                                   , I_SUPPLY_W_ID
                                   , O_ID
                                   , D_ID
                                   )
                ) AS NEW_OL_ALL

            WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
        )


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

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

          (   OL_O_ID
            , OL_D_ID
            , OL_W_ID
            , OL_NUMBER
            , OL_I_ID
            , OL_SUPPLY_W_ID
            , OL_DELIVERY_D
            , OL_QUANTITY
            , OL_AMOUNT
            , OL_DIST_INFO
          )

          INCLUDE (   I_PRICE    DECIMAL(5,2)
                    , I_NAME     CHAR(24)
                    , I_DATA     VARCHAR(50)
                    , S_DATA     VARCHAR(50)
                    , S_QUANTITY SMALLINT )

              SELECT   O_ID
                     , D_ID
                     , W_ID
                     , OL_NUMBER
                     , I_ID
                     , I_SUPPLY_W_ID
                     , OL_DELIVERY_D
                     , I_QTY
                     , TOTAL_PRICE
                     , OL_DIST_INFO
                     , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

              FROM DATA

          ) AS INS
;

EXEC SQL DECLARE ISOL_Remote_2 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
                , D_ID
                , W_ID
                , OL_NUMBER
                , I_ID
                , I_SUPPLY_W_ID
                , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
                , I_QTY
                , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
                , OL_DIST_INFO
                , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM ( SELECT   :next_o_id as O_ID
                               , :w_id AS W_ID
                               , :d_id as D_ID
                               , OL_NUMBER
                               , I_ID
                               , I_SUPPLY_W_ID
                               , I_QTY

                            FROM  Table( VALUES

                                   (  SMALLINT( 1 )        , :id0  , :ol_quantity0  , :supply_w_id0  )
                                  , (  SMALLINT( 2 )        , :id1  , :ol_quantity1  , :supply_w_id1  )

                                   ) AS X ( OL_NUMBER , I_ID  , I_QTY        , I_SUPPLY_W_ID  )
                        ) AS ITEMLIST

                      , TABLE( NEW_OL_ALL(   I_ID
                                           , I_QTY
                                           , W_ID
                                           , I_SUPPLY_W_ID
                                           , O_ID
                                           , D_ID
                                           )
                        ) AS NEW_OL_ALL

                    WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
          )


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

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

          (   OL_O_ID
            , OL_D_ID
            , OL_W_ID
            , OL_NUMBER
            , OL_I_ID
            , OL_SUPPLY_W_ID
            , OL_DELIVERY_D
            , OL_QUANTITY
            , OL_AMOUNT
            , OL_DIST_INFO
          )

          INCLUDE (   I_PRICE    DECIMAL(5,2)
                    , I_NAME     CHAR(24)
                    , I_DATA     VARCHAR(50)
                    , S_DATA     VARCHAR(50)
                    , S_QUANTITY SMALLINT )

              SELECT   O_ID
                     , D_ID
                     , W_ID
                     , OL_NUMBER
                     , I_ID
                     , I_SUPPLY_W_ID
                     , OL_DELIVERY_D
                     , I_QTY
                     , TOTAL_PRICE
                     , OL_DIST_INFO
                     , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

              FROM DATA

          ) AS INS
;

EXEC SQL DECLARE ISOL_Remote_3 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
                , D_ID
                , W_ID
                , OL_NUMBER
                , I_ID
                , I_SUPPLY_W_ID
                , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
                , I_QTY
                , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
                , OL_DIST_INFO
                , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM ( SELECT   :next_o_id as O_ID
                               , :w_id AS W_ID
                               , :d_id as D_ID
                               , OL_NUMBER
                               , I_ID
                               , I_SUPPLY_W_ID
                               , I_QTY

                            FROM  Table( VALUES

                                   (  SMALLINT( 1 )        , :id0  , :ol_quantity0  , :supply_w_id0  )
                                  , (  SMALLINT( 2 )        , :id1  , :ol_quantity1  , :supply_w_id1  )
                                  , (  SMALLINT( 3 )        , :id2  , :ol_quantity2  , :supply_w_id2  )

                                   ) AS X ( OL_NUMBER , I_ID  , I_QTY        , I_SUPPLY_W_ID  )
                        ) AS ITEMLIST

                      , TABLE( NEW_OL_ALL(   I_ID
                                           , I_QTY
                                           , W_ID
                                           , I_SUPPLY_W_ID
                                           , O_ID
                                           , D_ID
                                           )
                        ) AS NEW_OL_ALL

                    WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
          )


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

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

          (   OL_O_ID
            , OL_D_ID
            , OL_W_ID
            , OL_NUMBER
            , OL_I_ID
            , OL_SUPPLY_W_ID
            , OL_DELIVERY_D
            , OL_QUANTITY
            , OL_AMOUNT
            , OL_DIST_INFO
          )

          INCLUDE (   I_PRICE    DECIMAL(5,2)
                    , I_NAME     CHAR(24)
                    , I_DATA     VARCHAR(50)
                    , S_DATA     VARCHAR(50)
                    , S_QUANTITY SMALLINT )

              SELECT   O_ID
                     , D_ID
                     , W_ID
                     , OL_NUMBER
                     , I_ID
                     , I_SUPPLY_W_ID
                     , OL_DELIVERY_D
                     , I_QTY
                     , TOTAL_PRICE
                     , OL_DIST_INFO
                     , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

              FROM DATA

          ) AS INS
;
```

```
EXEC SQL DECLARE ISOL_Remote_4 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM ( SELECT   :next_o_id as O_ID
                           , :w_id AS W_ID
                           , :d_id as D_ID
                           , OL_NUMBER
                           , I_ID
                           , I_SUPPLY_W_ID
                           , I_QTY

                        FROM  Table( VALUES

                              (  SMALLINT( 1 )        , :id0  , :ol_quantity0  , :supply_w_id0  )
                             , (  SMALLINT( 2 )        , :id1  , :ol_quantity1  , :supply_w_id1  )
                             , (  SMALLINT( 3 )        , :id2  , :ol_quantity2  , :supply_w_id2  )
                             , (  SMALLINT( 4 )        , :id3  , :ol_quantity3  , :supply_w_id3  )

                             ) AS X ( OL_NUMBER , I_ID  , I_QTY         , I_SUPPLY_W_ID  )
                        ) AS ITEMLIST

                       , TABLE( NEW_OL_ALL(   I_ID
                                  , I_QTY
                                  , W_ID
                                  , I_SUPPLY_W_ID
                                  , O_ID
                                  , D_ID
                                  )
                        ) AS NEW_OL_ALL

                  WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
          )


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

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

            (  OL_O_ID
             , OL_D_ID
             , OL_W_ID
             , OL_NUMBER
             , OL_I_ID
             , OL_SUPPLY_W_ID
             , OL_DELIVERY_D
             , OL_QUANTITY
             , OL_AMOUNT
             , OL_DIST_INFO
            )

            INCLUDE (   I_PRICE    DECIMAL(5,2)
                  , I_NAME     CHAR(24)
                  , I_DATA     VARCHAR(50)
                  , S_DATA     VARCHAR(50)
                  , S_QUANTITY SMALLINT )

                SELECT   O_ID
                     , D_ID
                     , W_ID
                     , OL_NUMBER
                     , I_ID
                     , I_SUPPLY_W_ID
                     , OL_DELIVERY_D
                     , I_QTY
                     , TOTAL_PRICE
                     , OL_DIST_INFO
                     , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM DATA

            ) AS INS
;

EXEC SQL DECLARE ISOL_Remote_5 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM ( SELECT   :next_o_id as O_ID
                           , :w_id AS W_ID
                           , :d_id as D_ID
                           , OL_NUMBER
                           , I_ID
                           , I_SUPPLY_W_ID
                           , I_QTY

                        FROM  Table( VALUES

                              (  SMALLINT( 1 )        , :id0  , :ol_quantity0  , :supply_w_id0  )
                             , (  SMALLINT( 2 )        , :id1  , :ol_quantity1  , :supply_w_id1  )
                             , (  SMALLINT( 3 )        , :id2  , :ol_quantity2  , :supply_w_id2  )
                             , (  SMALLINT( 4 )        , :id3  , :ol_quantity3  , :supply_w_id3  )
                             , (  SMALLINT( 5 )        , :id4  , :ol_quantity4  , :supply_w_id4  )

                             ) AS X ( OL_NUMBER , I_ID  , I_QTY         , I_SUPPLY_W_ID  )
                        ) AS ITEMLIST

                       , TABLE( NEW_OL_ALL(   I_ID
                                  , I_QTY
                                  , W_ID
                                  , I_SUPPLY_W_ID
                                  , O_ID
                                  , D_ID
                                  )
                        ) AS NEW_OL_ALL

                  WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
          )


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

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

            (  OL_O_ID
             , OL_D_ID
             , OL_W_ID
             , OL_NUMBER
             , OL_I_ID
             , OL_SUPPLY_W_ID
             , OL_DELIVERY_D
             , OL_QUANTITY
             , OL_AMOUNT
             , OL_DIST_INFO
            )

            INCLUDE (   I_PRICE    DECIMAL(5,2)
                  , I_NAME     CHAR(24)
                  , I_DATA     VARCHAR(50)
                  , S_DATA     VARCHAR(50)
                  , S_QUANTITY SMALLINT )

                SELECT   O_ID
                     , D_ID
                     , W_ID
                     , OL_NUMBER
                     , I_ID
                     , I_SUPPLY_W_ID
                     , OL_DELIVERY_D
                     , I_QTY
                     , TOTAL_PRICE
                     , OL_DIST_INFO
                     , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM DATA

            ) AS INS
;

EXEC SQL DECLARE ISOL_Remote_6 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM ( SELECT   :next_o_id as O_ID
                           , :w_id AS W_ID
                           , :d_id as D_ID
                           , OL_NUMBER
                           , I_ID
                           , I_SUPPLY_W_ID
                           , I_QTY

                        FROM  Table( VALUES

                              (  SMALLINT( 1 )        , :id0  , :ol_quantity0  , :supply_w_id0  )
                             , (  SMALLINT( 2 )        , :id1  , :ol_quantity1  , :supply_w_id1  )
                             , (  SMALLINT( 3 )        , :id2  , :ol_quantity2  , :supply_w_id2  )
                             , (  SMALLINT( 4 )        , :id3  , :ol_quantity3  , :supply_w_id3  )
                             , (  SMALLINT( 5 )        , :id4  , :ol_quantity4  , :supply_w_id4  )
                             , (  SMALLINT( 6 )        , :id5  , :ol_quantity5  , :supply_w_id5  )

                             ) AS X ( OL_NUMBER , I_ID  , I_QTY         , I_SUPPLY_W_ID  )
                        ) AS ITEMLIST

                       , TABLE( NEW_OL_ALL(   I_ID
                                  , I_QTY
                                  , W_ID
                                  , I_SUPPLY_W_ID
                                  , O_ID
                                  , D_ID
                                  )
                        ) AS NEW_OL_ALL

                  WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
          )


  SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY
```

```
  FROM NEW TABLE ( INSERT INTO ORDER_LINE

            (  OL_O_ID
             , OL_D_ID
             , OL_W_ID
             , OL_NUMBER
             , OL_I_ID
             , OL_SUPPLY_W_ID
             , OL_DELIVERY_D
             , OL_QUANTITY
             , OL_AMOUNT
             , OL_DIST_INFO
            )

            INCLUDE (  I_PRICE    DECIMAL(5,2)
                     , I_NAME     CHAR(24)
                     , I_DATA     VARCHAR(50)
                     , S_DATA     VARCHAR(50)
                     , S_QUANTITY SMALLINT )

                SELECT  O_ID
                      , D_ID
                      , W_ID
                      , OL_NUMBER
                      , I_ID
                      , I_SUPPLY_W_ID
                      , OL_DELIVERY_D
                      , I_QTY
                      , TOTAL_PRICE
                      , OL_DIST_INFO
                      , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                FROM DATA

            ) AS INS
;

EXEC SQL DECLARE ISOL_Remote_7 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
                , D_ID
                , W_ID
                , OL_NUMBER
                , I_ID
                , I_SUPPLY_W_ID
                , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
                , I_QTY
                , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
                , OL_DIST_INFO
                , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM ( SELECT   :next_o_id as O_ID
                               , :w_id AS W_ID
                               , :d_id as D_ID
                               , OL_NUMBER
                               , I_ID
                               , I_SUPPLY_W_ID
                               , I_QTY

                    FROM  Table( VALUES

                            (  SMALLINT( 1 )       , :id0  , :ol_quantity0  , :supply_w_id0  )
                          , (  SMALLINT( 2 )       , :id1  , :ol_quantity1  , :supply_w_id1  )
                          , (  SMALLINT( 3 )       , :id2  , :ol_quantity2  , :supply_w_id2  )
                          , (  SMALLINT( 4 )       , :id3  , :ol_quantity3  , :supply_w_id3  )
                          , (  SMALLINT( 5 )       , :id4  , :ol_quantity4  , :supply_w_id4  )
                          , (  SMALLINT( 6 )       , :id5  , :ol_quantity5  , :supply_w_id5  )
                          , (  SMALLINT( 7 )       , :id6  , :ol_quantity6  , :supply_w_id6  )

                          ) AS X ( OL_NUMBER , I_ID  , I_QTY         , I_SUPPLY_W_ID  )
                  ) AS ITEMLIST

                , TABLE( NEW_OL_ALL(   I_ID
                                     , I_QTY
                                     , W_ID
                                     , I_SUPPLY_W_ID
                                     , O_ID
                                     , D_ID
                                     )
                        ) AS NEW_OL_ALL

                  WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
            )


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

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

            (  OL_O_ID
             , OL_D_ID
             , OL_W_ID
             , OL_NUMBER
             , OL_I_ID
             , OL_SUPPLY_W_ID
             , OL_DELIVERY_D
             , OL_QUANTITY
             , OL_AMOUNT
             , OL_DIST_INFO
            )

            INCLUDE (  I_PRICE    DECIMAL(5,2)
                     , I_NAME     CHAR(24)
                     , I_DATA     VARCHAR(50)
                     , S_DATA     VARCHAR(50)
                     , S_QUANTITY SMALLINT )

                SELECT  O_ID
                      , D_ID
                      , W_ID
                      , OL_NUMBER
                      , I_ID
                      , I_SUPPLY_W_ID
                      , OL_DELIVERY_D
                      , I_QTY
                      , TOTAL_PRICE
                      , OL_DIST_INFO
                      , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                FROM DATA

            ) AS INS
;

EXEC SQL DECLARE ISOL_Remote_8 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
                , D_ID
                , W_ID
                , OL_NUMBER
                , I_ID
                , I_SUPPLY_W_ID
                , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
                , I_QTY
                , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
                , OL_DIST_INFO
                , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM ( SELECT   :next_o_id as O_ID
                               , :w_id AS W_ID
                               , :d_id as D_ID
                               , OL_NUMBER
                               , I_ID
                               , I_SUPPLY_W_ID
                               , I_QTY

                    FROM  Table( VALUES

                            (  SMALLINT( 1 )       , :id0  , :ol_quantity0  , :supply_w_id0  )
                          , (  SMALLINT( 2 )       , :id1  , :ol_quantity1  , :supply_w_id1  )
                          , (  SMALLINT( 3 )       , :id2  , :ol_quantity2  , :supply_w_id2  )
                          , (  SMALLINT( 4 )       , :id3  , :ol_quantity3  , :supply_w_id3  )
                          , (  SMALLINT( 5 )       , :id4  , :ol_quantity4  , :supply_w_id4  )
                          , (  SMALLINT( 6 )       , :id5  , :ol_quantity5  , :supply_w_id5  )
                          , (  SMALLINT( 7 )       , :id6  , :ol_quantity6  , :supply_w_id6  )
                          , (  SMALLINT( 8 )       , :id7  , :ol_quantity7  , :supply_w_id7  )

                          ) AS X ( OL_NUMBER , I_ID  , I_QTY         , I_SUPPLY_W_ID  )
                  ) AS ITEMLIST

                , TABLE( NEW_OL_ALL(   I_ID
                                     , I_QTY
                                     , W_ID
                                     , I_SUPPLY_W_ID
                                     , O_ID
                                     , D_ID
                                     )
                        ) AS NEW_OL_ALL

                  WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
            )


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

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

            (  OL_O_ID
             , OL_D_ID
             , OL_W_ID
             , OL_NUMBER
             , OL_I_ID
             , OL_SUPPLY_W_ID
             , OL_DELIVERY_D
             , OL_QUANTITY
             , OL_AMOUNT
             , OL_DIST_INFO
            )

            INCLUDE (  I_PRICE    DECIMAL(5,2)
                     , I_NAME     CHAR(24)
                     , I_DATA     VARCHAR(50)
                     , S_DATA     VARCHAR(50)
                     , S_QUANTITY SMALLINT )

                SELECT  O_ID
                      , D_ID
                      , W_ID
                      , OL_NUMBER
                      , I_ID
                      , I_SUPPLY_W_ID
                      , OL_DELIVERY_D
                      , I_QTY
                      , TOTAL_PRICE
                      , OL_DIST_INFO
                      , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                FROM DATA

            ) AS INS
;

EXEC SQL DECLARE ISOL_Remote_9 CURSOR FOR

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

```
        , I_SUPPLY_W_ID
        , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
        , I_QTY
        , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
        , OL_DIST_INFO
        , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

            FROM ( SELECT   :next_o_id as O_ID
                    , :w_id AS W_ID
                    , :d_id as D_ID
                    , OL_NUMBER
                    , I_ID
                    , I_SUPPLY_W_ID
                    , I_QTY

                FROM  Table( VALUES

                        (  SMALLINT( 1 )       , :id0  , :ol_quantity0  , :supply_w_id0  )
                       , (  SMALLINT( 2 )       , :id1  , :ol_quantity1  , :supply_w_id1  )
                       , (  SMALLINT( 3 )       , :id2  , :ol_quantity2  , :supply_w_id2  )
                       , (  SMALLINT( 4 )       , :id3  , :ol_quantity3  , :supply_w_id3  )
                       , (  SMALLINT( 5 )       , :id4  , :ol_quantity4  , :supply_w_id4  )
                       , (  SMALLINT( 6 )       , :id5  , :ol_quantity5  , :supply_w_id5  )
                       , (  SMALLINT( 7 )       , :id6  , :ol_quantity6  , :supply_w_id6  )
                       , (  SMALLINT( 8 )       , :id7  , :ol_quantity7  , :supply_w_id7  )
                       , (  SMALLINT( 9 )       , :id8  , :ol_quantity8  , :supply_w_id8  )

                      ) AS X ( OL_NUMBER , I_ID  , I_QTY         , I_SUPPLY_W_ID  )
            ) AS ITEMLIST

            , TABLE( NEW_OL_ALL(   I_ID
                    , I_QTY
                    , W_ID
                    , I_SUPPLY_W_ID
                    , O_ID
                    , D_ID
                    )
            ) AS NEW_OL_ALL

            WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
    )


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

 FROM NEW TABLE ( INSERT INTO ORDER_LINE

        (  OL_O_ID
         , OL_D_ID
         , OL_W_ID
         , OL_NUMBER
         , OL_I_ID
         , OL_SUPPLY_W_ID
         , OL_DELIVERY_D
         , OL_QUANTITY
         , OL_AMOUNT
         , OL_DIST_INFO
        )

        INCLUDE (   I_PRICE    DECIMAL(5,2)
                , I_NAME     CHAR(24)
                , I_DATA     VARCHAR(50)
                , S_DATA     VARCHAR(50)
                , S_QUANTITY SMALLINT )

            SELECT   O_ID
                , D_ID
                , W_ID
                , OL_NUMBER
                , I_ID
                , I_SUPPLY_W_ID
                , OL_DELIVERY_D
                , I_QTY
                , TOTAL_PRICE
                , OL_DIST_INFO
                , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

            FROM DATA

        ) AS INS
;

EXEC SQL DECLARE ISOL_Remote_10 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
        , D_ID
        , W_ID
        , OL_NUMBER
        , I_ID
        , I_SUPPLY_W_ID
        , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
        , I_QTY
        , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
        , OL_DIST_INFO
        , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

            FROM ( SELECT   :next_o_id as O_ID
                    , :w_id AS W_ID
                    , :d_id as D_ID
                    , OL_NUMBER
                    , I_ID
                    , I_SUPPLY_W_ID
                    , I_QTY

                FROM  Table( VALUES

                        (  SMALLINT( 1 )       , :id0  , :ol_quantity0  , :supply_w_id0  )
                       , (  SMALLINT( 2 )       , :id1  , :ol_quantity1  , :supply_w_id1  )
                       , (  SMALLINT( 3 )       , :id2  , :ol_quantity2  , :supply_w_id2  )
                       , (  SMALLINT( 4 )       , :id3  , :ol_quantity3  , :supply_w_id3  )
                       , (  SMALLINT( 5 )       , :id4  , :ol_quantity4  , :supply_w_id4  )
                       , (  SMALLINT( 6 )       , :id5  , :ol_quantity5  , :supply_w_id5  )
                       , (  SMALLINT( 7 )       , :id6  , :ol_quantity6  , :supply_w_id6  )
                       , (  SMALLINT( 8 )       , :id7  , :ol_quantity7  , :supply_w_id7  )
                       , (  SMALLINT( 9 )       , :id8  , :ol_quantity8  , :supply_w_id8  )
                       , (  SMALLINT( 10 )       , :id9  , :ol_quantity9  , :supply_w_id9   )

                      ) AS X ( OL_NUMBER , I_ID  , I_QTY         , I_SUPPLY_W_ID  )
            ) AS ITEMLIST

            , TABLE( NEW_OL_ALL(   I_ID
                    , I_QTY
                    , W_ID
                    , I_SUPPLY_W_ID
                    , O_ID
                    , D_ID
                    )
            ) AS NEW_OL_ALL

            WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
    )


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

 FROM NEW TABLE ( INSERT INTO ORDER_LINE

        (  OL_O_ID
         , OL_D_ID
         , OL_W_ID
         , OL_NUMBER
         , OL_I_ID
         , OL_SUPPLY_W_ID
         , OL_DELIVERY_D
         , OL_QUANTITY
         , OL_AMOUNT
         , OL_DIST_INFO
        )

        INCLUDE (   I_PRICE    DECIMAL(5,2)
                , I_NAME     CHAR(24)
                , I_DATA     VARCHAR(50)
                , S_DATA     VARCHAR(50)
                , S_QUANTITY SMALLINT )

            SELECT   O_ID
                , D_ID
                , W_ID
                , OL_NUMBER
                , I_ID
                , I_SUPPLY_W_ID
                , OL_DELIVERY_D
                , I_QTY
                , TOTAL_PRICE
                , OL_DIST_INFO
                , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

            FROM DATA

        ) AS INS
;

EXEC SQL DECLARE ISOL_Remote_11 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
        , D_ID
        , W_ID
        , OL_NUMBER
        , I_ID
        , I_SUPPLY_W_ID
        , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
        , I_QTY
        , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
        , OL_DIST_INFO
        , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

            FROM ( SELECT   :next_o_id as O_ID
                    , :w_id AS W_ID
                    , :d_id as D_ID
                    , OL_NUMBER
                    , I_ID
                    , I_SUPPLY_W_ID
                    , I_QTY

                FROM  Table( VALUES

                        (  SMALLINT( 1 )       , :id0  , :ol_quantity0  , :supply_w_id0  )
                       , (  SMALLINT( 2 )       , :id1  , :ol_quantity1  , :supply_w_id1  )
                       , (  SMALLINT( 3 )       , :id2  , :ol_quantity2  , :supply_w_id2  )
                       , (  SMALLINT( 4 )       , :id3  , :ol_quantity3  , :supply_w_id3  )
                       , (  SMALLINT( 5 )       , :id4  , :ol_quantity4  , :supply_w_id4  )
                       , (  SMALLINT( 6 )       , :id5  , :ol_quantity5  , :supply_w_id5  )
                       , (  SMALLINT( 7 )       , :id6  , :ol_quantity6  , :supply_w_id6  )
                       , (  SMALLINT( 8 )       , :id7  , :ol_quantity7  , :supply_w_id7  )
                       , (  SMALLINT( 9 )       , :id8  , :ol_quantity8  , :supply_w_id8  )
                       , (  SMALLINT( 10 )       , :id9  , :ol_quantity9  , :supply_w_id9   )
                       , (  SMALLINT( 11 )       , :id10 , :ol_quantity10 , :supply_w_id10  )

                      ) AS X ( OL_NUMBER , I_ID  , I_QTY         , I_SUPPLY_W_ID  )
            ) AS ITEMLIST

            , TABLE( NEW_OL_ALL(   I_ID
                    , I_QTY
                    , W_ID
                    , I_SUPPLY_W_ID
                    , O_ID
                    , D_ID
                    )
            ) AS NEW_OL_ALL
```

```
                    WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
            )


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

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

                (   OL_O_ID
                  , OL_D_ID
                  , OL_W_ID
                  , OL_NUMBER
                  , OL_I_ID
                  , OL_SUPPLY_W_ID
                  , OL_DELIVERY_D
                  , OL_QUANTITY
                  , OL_AMOUNT
                  , OL_DIST_INFO
                )

                INCLUDE (   I_PRICE    DECIMAL(5,2)
                        , I_NAME     CHAR(24)
                        , I_DATA     VARCHAR(50)
                        , S_DATA     VARCHAR(50)
                        , S_QUANTITY SMALLINT )

                    SELECT   O_ID
                        , D_ID
                        , W_ID
                        , OL_NUMBER
                        , I_ID
                        , I_SUPPLY_W_ID
                        , OL_DELIVERY_D
                        , I_QTY
                        , TOTAL_PRICE
                        , OL_DIST_INFO
                        , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                    FROM DATA

                ) AS INS
;

EXEC SQL DECLARE ISOL_Remote_12 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                FROM ( SELECT   :next_o_id as O_ID
                        , :w_id AS W_ID
                        , :d_id as D_ID
                        , OL_NUMBER
                        , I_ID
                        , I_SUPPLY_W_ID
                        , I_QTY

                    FROM  Table( VALUES

                        (  SMALLINT( 1 )       , :id0  , :ol_quantity0  , :supply_w_id0  )
                       , (  SMALLINT( 2 )       , :id1  , :ol_quantity1  , :supply_w_id1  )
                       , (  SMALLINT( 3 )       , :id2  , :ol_quantity2  , :supply_w_id2  )
                       , (  SMALLINT( 4 )       , :id3  , :ol_quantity3  , :supply_w_id3  )
                       , (  SMALLINT( 5 )       , :id4  , :ol_quantity4  , :supply_w_id4  )
                       , (  SMALLINT( 6 )       , :id5  , :ol_quantity5  , :supply_w_id5  )
                       , (  SMALLINT( 7 )       , :id6  , :ol_quantity6  , :supply_w_id6  )
                       , (  SMALLINT( 8 )       , :id7  , :ol_quantity7  , :supply_w_id7  )
                       , (  SMALLINT( 9 )       , :id8  , :ol_quantity8  , :supply_w_id8  )
                       , (  SMALLINT( 10 )      , :id9  , :ol_quantity9  , :supply_w_id9  )
                       , (  SMALLINT( 11 )      , :id10 , :ol_quantity10 , :supply_w_id10 )
                       , (  SMALLINT( 12 )      , :id11 , :ol_quantity11 , :supply_w_id11 )

                        ) AS X ( OL_NUMBER , I_ID  , I_QTY         , I_SUPPLY_W_ID  )
                    ) AS ITEMLIST

                  , TABLE( NEW_OL_ALL(   I_ID
                              , I_QTY
                              , W_ID
                              , I_SUPPLY_W_ID
                              , O_ID
                              , D_ID
                              )
                    ) AS NEW_OL_ALL

                WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
            )


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

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

                (   OL_O_ID
                  , OL_D_ID
                  , OL_W_ID
                  , OL_NUMBER
                  , OL_I_ID
                  , OL_SUPPLY_W_ID
                  , OL_DELIVERY_D
                  , OL_QUANTITY
                  , OL_AMOUNT
                  , OL_DIST_INFO
                )

                INCLUDE (   I_PRICE    DECIMAL(5,2)
                        , I_NAME     CHAR(24)
                        , I_DATA     VARCHAR(50)
                        , S_DATA     VARCHAR(50)
                        , S_QUANTITY SMALLINT )

                    SELECT   O_ID
                        , D_ID
                        , W_ID
                        , OL_NUMBER
                        , I_ID
                        , I_SUPPLY_W_ID
                        , OL_DELIVERY_D
                        , I_QTY
                        , TOTAL_PRICE
                        , OL_DIST_INFO
                        , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                    FROM DATA

                ) AS INS
;

EXEC SQL DECLARE ISOL_Remote_13 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                FROM ( SELECT   :next_o_id as O_ID
                        , :w_id AS W_ID
                        , :d_id as D_ID
                        , OL_NUMBER
                        , I_ID
                        , I_SUPPLY_W_ID
                        , I_QTY

                    FROM  Table( VALUES

                        (  SMALLINT( 1 )       , :id0  , :ol_quantity0  , :supply_w_id0  )
                       , (  SMALLINT( 2 )       , :id1  , :ol_quantity1  , :supply_w_id1  )
                       , (  SMALLINT( 3 )       , :id2  , :ol_quantity2  , :supply_w_id2  )
                       , (  SMALLINT( 4 )       , :id3  , :ol_quantity3  , :supply_w_id3  )
                       , (  SMALLINT( 5 )       , :id4  , :ol_quantity4  , :supply_w_id4  )
                       , (  SMALLINT( 6 )       , :id5  , :ol_quantity5  , :supply_w_id5  )
                       , (  SMALLINT( 7 )       , :id6  , :ol_quantity6  , :supply_w_id6  )
                       , (  SMALLINT( 8 )       , :id7  , :ol_quantity7  , :supply_w_id7  )
                       , (  SMALLINT( 9 )       , :id8  , :ol_quantity8  , :supply_w_id8  )
                       , (  SMALLINT( 10 )      , :id9  , :ol_quantity9  , :supply_w_id9  )
                       , (  SMALLINT( 11 )      , :id10 , :ol_quantity10 , :supply_w_id10 )
                       , (  SMALLINT( 12 )      , :id11 , :ol_quantity11 , :supply_w_id11 )
                       , (  SMALLINT( 13 )      , :id12 , :ol_quantity12 , :supply_w_id12 )

                        ) AS X ( OL_NUMBER , I_ID  , I_QTY         , I_SUPPLY_W_ID  )
                    ) AS ITEMLIST

                  , TABLE( NEW_OL_ALL(   I_ID
                              , I_QTY
                              , W_ID
                              , I_SUPPLY_W_ID
                              , O_ID
                              , D_ID
                              )
                    ) AS NEW_OL_ALL

                WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
            )


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

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

                (   OL_O_ID
                  , OL_D_ID
                  , OL_W_ID
                  , OL_NUMBER
                  , OL_I_ID
                  , OL_SUPPLY_W_ID
                  , OL_DELIVERY_D
                  , OL_QUANTITY
                  , OL_AMOUNT
                  , OL_DIST_INFO
                )

                INCLUDE (   I_PRICE    DECIMAL(5,2)
                        , I_NAME     CHAR(24)
                        , I_DATA     VARCHAR(50)
                        , S_DATA     VARCHAR(50)
                        , S_QUANTITY SMALLINT )

                    SELECT   O_ID
                        , D_ID
                        , W_ID
                        , OL_NUMBER
                        , I_ID
                        , I_SUPPLY_W_ID
                        , OL_DELIVERY_D
```

```
                    , I_QTY
                    , TOTAL_PRICE
                    , OL_DIST_INFO
                    , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                FROM DATA

            ) AS INS
;

EXEC SQL DECLARE ISOL_Remote_14 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                FROM ( SELECT   :next_o_id as O_ID
                        , :w_id AS W_ID
                        , :d_id as D_ID
                        , OL_NUMBER
                        , I_ID
                        , I_SUPPLY_W_ID
                        , I_QTY

                  FROM  Table( VALUES

                        (  SMALLINT( 1 )       , :id0  , :ol_quantity0  , :supply_w_id0  )
                       , (  SMALLINT( 2 )       , :id1  , :ol_quantity1  , :supply_w_id1  )
                       , (  SMALLINT( 3 )       , :id2  , :ol_quantity2  , :supply_w_id2  )
                       , (  SMALLINT( 4 )       , :id3  , :ol_quantity3  , :supply_w_id3  )
                       , (  SMALLINT( 5 )       , :id4  , :ol_quantity4  , :supply_w_id4  )
                       , (  SMALLINT( 6 )       , :id5  , :ol_quantity5  , :supply_w_id5  )
                       , (  SMALLINT( 7 )       , :id6  , :ol_quantity6  , :supply_w_id6  )
                       , (  SMALLINT( 8 )       , :id7  , :ol_quantity7  , :supply_w_id7  )
                       , (  SMALLINT( 9 )       , :id8  , :ol_quantity8  , :supply_w_id8  )
                       , (  SMALLINT( 10 )       , :id9  , :ol_quantity9  , :supply_w_id9   )
                       , (  SMALLINT( 11 )       , :id10 , :ol_quantity10 , :supply_w_id10  )
                       , (  SMALLINT( 12 )       , :id11 , :ol_quantity11 , :supply_w_id11  )
                       , (  SMALLINT( 13 )       , :id12 , :ol_quantity12 , :supply_w_id12  )
                       , (  SMALLINT( 14 )       , :id13 , :ol_quantity13 , :supply_w_id13  )

                       ) AS X ( OL_NUMBER , I_ID  , I_QTY         , I_SUPPLY_W_ID  )
                ) AS ITEMLIST

                , TABLE( NEW_OL_ALL(   I_ID
                        , I_QTY
                        , W_ID
                        , I_SUPPLY_W_ID
                        , O_ID
                        , D_ID
                        )
                  ) AS NEW_OL_ALL

                WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
          )


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

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

            (  OL_O_ID
             , OL_D_ID
             , OL_W_ID
             , OL_NUMBER
             , OL_I_ID
             , OL_SUPPLY_W_ID
             , OL_DELIVERY_D
             , OL_QUANTITY
             , OL_AMOUNT
             , OL_DIST_INFO
            )

            INCLUDE (   I_PRICE    DECIMAL(5,2)
                  , I_NAME     CHAR(24)
                  , I_DATA     VARCHAR(50)
                  , S_DATA     VARCHAR(50)
                  , S_QUANTITY SMALLINT )

              SELECT   O_ID
                  , D_ID
                  , W_ID
                  , OL_NUMBER
                  , I_ID
                  , I_SUPPLY_W_ID
                  , OL_DELIVERY_D
                  , I_QTY
                  , TOTAL_PRICE
                  , OL_DIST_INFO
                  , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

              FROM DATA

            ) AS INS
;

EXEC SQL DECLARE ISOL_Remote_15 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                FROM ( SELECT   :next_o_id as O_ID
                        , :w_id AS W_ID
                        , :d_id as D_ID
                        , OL_NUMBER
                        , I_ID
                        , I_SUPPLY_W_ID
                        , I_QTY

                  FROM  Table( VALUES

                        (  SMALLINT( 1 )       , :id0  , :ol_quantity0  , :supply_w_id0  )
                       , (  SMALLINT( 2 )       , :id1  , :ol_quantity1  , :supply_w_id1  )
                       , (  SMALLINT( 3 )       , :id2  , :ol_quantity2  , :supply_w_id2  )
                       , (  SMALLINT( 4 )       , :id3  , :ol_quantity3  , :supply_w_id3  )
                       , (  SMALLINT( 5 )       , :id4  , :ol_quantity4  , :supply_w_id4  )
                       , (  SMALLINT( 6 )       , :id5  , :ol_quantity5  , :supply_w_id5  )
                       , (  SMALLINT( 7 )       , :id6  , :ol_quantity6  , :supply_w_id6  )
                       , (  SMALLINT( 8 )       , :id7  , :ol_quantity7  , :supply_w_id7  )
                       , (  SMALLINT( 9 )       , :id8  , :ol_quantity8  , :supply_w_id8  )
                       , (  SMALLINT( 10 )       , :id9  , :ol_quantity9  , :supply_w_id9   )
                       , (  SMALLINT( 11 )       , :id10 , :ol_quantity10 , :supply_w_id10  )
                       , (  SMALLINT( 12 )       , :id11 , :ol_quantity11 , :supply_w_id11  )
                       , (  SMALLINT( 13 )       , :id12 , :ol_quantity12 , :supply_w_id12  )
                       , (  SMALLINT( 14 )       , :id13 , :ol_quantity13 , :supply_w_id13  )
                       , (  SMALLINT( 15 )       , :id14 , :ol_quantity14 , :supply_w_id14  )

                       ) AS X ( OL_NUMBER , I_ID  , I_QTY         , I_SUPPLY_W_ID  )
                ) AS ITEMLIST

                , TABLE( NEW_OL_ALL(   I_ID
                        , I_QTY
                        , W_ID
                        , I_SUPPLY_W_ID
                        , O_ID
                        , D_ID
                        )
                  ) AS NEW_OL_ALL

                WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
          )


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

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

            (  OL_O_ID
             , OL_D_ID
             , OL_W_ID
             , OL_NUMBER
             , OL_I_ID
             , OL_SUPPLY_W_ID
             , OL_DELIVERY_D
             , OL_QUANTITY
             , OL_AMOUNT
             , OL_DIST_INFO
            )

            INCLUDE (   I_PRICE    DECIMAL(5,2)
                  , I_NAME     CHAR(24)
                  , I_DATA     VARCHAR(50)
                  , S_DATA     VARCHAR(50)
                  , S_QUANTITY SMALLINT )

              SELECT   O_ID
                  , D_ID
                  , W_ID
                  , OL_NUMBER
                  , I_ID
                  , I_SUPPLY_W_ID
                  , OL_DELIVERY_D
                  , I_QTY
                  , TOTAL_PRICE
                  , OL_DIST_INFO
                  , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

              FROM DATA

            ) AS INS
;

EXEC SQL DECLARE ISOL_Local_1 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , W_ID AS I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                FROM ( SELECT   :next_o_id as O_ID
                        , :w_id AS W_ID
                        , :d_id as D_ID
                        , OL_NUMBER
                        , I_ID
                        , I_QTY
```

```
                    FROM  Table( VALUES

                          (  SMALLINT( 1 )  , :id0  , :ol_quantity0  )

                          ) AS X ( OL_NUMBER , I_ID  , I_QTY          )
                  ) AS ITEMLIST

                , TABLE( NEW_OL_LOCAL(   I_ID
                              , I_QTY
                              , W_ID
                              , O_ID
                              , D_ID
                              )
                  ) AS NEW_OL_LOCAL

              WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
        )


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

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

            (   OL_O_ID
              , OL_D_ID
              , OL_W_ID
              , OL_NUMBER
              , OL_I_ID
              , OL_SUPPLY_W_ID
              , OL_DELIVERY_D
              , OL_QUANTITY
              , OL_AMOUNT
              , OL_DIST_INFO
            )

            INCLUDE (   I_PRICE    DECIMAL(5,2)
                  , I_NAME     CHAR(24)
                  , I_DATA     VARCHAR(50)
                  , S_DATA     VARCHAR(50)
                  , S_QUANTITY SMALLINT )

                SELECT   O_ID
                     , D_ID
                     , W_ID
                     , OL_NUMBER
                     , I_ID
                     , I_SUPPLY_W_ID
                     , OL_DELIVERY_D
                     , I_QTY
                     , TOTAL_PRICE
                     , OL_DIST_INFO
                     , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                FROM DATA

            ) AS INS
;

EXEC SQL DECLARE ISOL_Local_2 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , W_ID AS I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY
                    FROM ( SELECT   :next_o_id as O_ID
                          , :w_id AS W_ID
                          , :d_id as D_ID
                          , OL_NUMBER
                          , I_ID
                          , I_QTY

                      FROM  Table( VALUES

                            (  SMALLINT( 1 )  , :id0  , :ol_quantity0  )
                          , (  SMALLINT( 2 )  , :id1  , :ol_quantity1  )

                          ) AS X ( OL_NUMBER , I_ID  , I_QTY          )
                    ) AS ITEMLIST

                  , TABLE( NEW_OL_LOCAL(   I_ID
                              , I_QTY
                              , W_ID
                              , O_ID
                              , D_ID
                              )
                    ) AS NEW_OL_LOCAL

                WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
        )


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

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

            (   OL_O_ID
              , OL_D_ID
              , OL_W_ID
              , OL_NUMBER
              , OL_I_ID
              , OL_SUPPLY_W_ID
              , OL_DELIVERY_D
              , OL_QUANTITY
              , OL_AMOUNT
              , OL_DIST_INFO
            )

            INCLUDE (   I_PRICE    DECIMAL(5,2)
                  , I_NAME     CHAR(24)
                  , I_DATA     VARCHAR(50)
                  , S_DATA     VARCHAR(50)
                  , S_QUANTITY SMALLINT )

                SELECT   O_ID
                     , D_ID
                     , W_ID
                     , OL_NUMBER
                     , I_ID
                     , I_SUPPLY_W_ID
                     , OL_DELIVERY_D
                     , I_QTY
                     , TOTAL_PRICE
                     , OL_DIST_INFO
                     , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                FROM DATA

            ) AS INS
;

EXEC SQL DECLARE ISOL_Local_3 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , W_ID AS I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY
                    FROM ( SELECT   :next_o_id as O_ID
                          , :w_id AS W_ID
                          , :d_id as D_ID
                          , OL_NUMBER
                          , I_ID
                          , I_QTY

                      FROM  Table( VALUES

                            (  SMALLINT( 1 )  , :id0  , :ol_quantity0  )
                          , (  SMALLINT( 2 )  , :id1  , :ol_quantity1  )
                          , (  SMALLINT( 3 )  , :id2  , :ol_quantity2  )

                          ) AS X ( OL_NUMBER , I_ID  , I_QTY          )
                    ) AS ITEMLIST

                  , TABLE( NEW_OL_LOCAL(   I_ID
                              , I_QTY
                              , W_ID
                              , O_ID
                              , D_ID
                              )
                    ) AS NEW_OL_LOCAL

                WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
        )


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

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

            (   OL_O_ID
              , OL_D_ID
              , OL_W_ID
              , OL_NUMBER
              , OL_I_ID
              , OL_SUPPLY_W_ID
              , OL_DELIVERY_D
              , OL_QUANTITY
              , OL_AMOUNT
              , OL_DIST_INFO
            )

            INCLUDE (   I_PRICE    DECIMAL(5,2)
                  , I_NAME     CHAR(24)
                  , I_DATA     VARCHAR(50)
                  , S_DATA     VARCHAR(50)
                  , S_QUANTITY SMALLINT )

                SELECT   O_ID
                     , D_ID
                     , W_ID
                     , OL_NUMBER
                     , I_ID
                     , I_SUPPLY_W_ID
                     , OL_DELIVERY_D
                     , I_QTY
                     , TOTAL_PRICE
                     , OL_DIST_INFO
                     , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                FROM DATA

            ) AS INS
;
```

```
EXEC SQL DECLARE ISOL_Local_4 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , W_ID AS I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM ( SELECT   :next_o_id as O_ID
                           , :w_id AS W_ID
                           , :d_id as D_ID
                           , OL_NUMBER
                           , I_ID
                           , I_QTY

                        FROM  Table( VALUES

                              (  SMALLINT( 1 )  , :id0  , :ol_quantity0  )
                             , (  SMALLINT( 2 )  , :id1  , :ol_quantity1  )
                             , (  SMALLINT( 3 )  , :id2  , :ol_quantity2  )
                             , (  SMALLINT( 4 )  , :id3  , :ol_quantity3  )

                              ) AS X ( OL_NUMBER , I_ID  , I_QTY          )
                        ) AS ITEMLIST

                       , TABLE( NEW_OL_LOCAL(   I_ID
                                  , I_QTY
                                  , W_ID
                                  , O_ID
                                  , D_ID
                                 )
                        ) AS NEW_OL_LOCAL

                  WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
            )


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

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

              (  OL_O_ID
               , OL_D_ID
               , OL_W_ID
               , OL_NUMBER
               , OL_I_ID
               , OL_SUPPLY_W_ID
               , OL_DELIVERY_D
               , OL_QUANTITY
               , OL_AMOUNT
               , OL_DIST_INFO
              )

              INCLUDE (   I_PRICE    DECIMAL(5,2)
                     , I_NAME     CHAR(24)
                     , I_DATA     VARCHAR(50)
                     , S_DATA     VARCHAR(50)
                     , S_QUANTITY SMALLINT )

                  SELECT   O_ID
                       , D_ID
                       , W_ID
                       , OL_NUMBER
                       , I_ID
                       , I_SUPPLY_W_ID
                       , OL_DELIVERY_D
                       , I_QTY
                       , TOTAL_PRICE
                       , OL_DIST_INFO
                       , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM DATA

            ) AS INS
;

EXEC SQL DECLARE ISOL_Local_5 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , W_ID AS I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM ( SELECT   :next_o_id as O_ID
                           , :w_id AS W_ID
                           , :d_id as D_ID
                           , OL_NUMBER
                           , I_ID
                           , I_QTY

                        FROM  Table( VALUES

                              (  SMALLINT( 1 )  , :id0  , :ol_quantity0  )
                             , (  SMALLINT( 2 )  , :id1  , :ol_quantity1  )
                             , (  SMALLINT( 3 )  , :id2  , :ol_quantity2  )
                             , (  SMALLINT( 4 )  , :id3  , :ol_quantity3  )
                             , (  SMALLINT( 5 )  , :id4  , :ol_quantity4  )

                              ) AS X ( OL_NUMBER , I_ID  , I_QTY          )
                        ) AS ITEMLIST

                       , TABLE( NEW_OL_LOCAL(   I_ID
                                  , I_QTY
                                  , W_ID
                                  , O_ID
                                  , D_ID
                                 )
                        ) AS NEW_OL_LOCAL

                  WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
            )


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

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

              (  OL_O_ID
               , OL_D_ID
               , OL_W_ID
               , OL_NUMBER
               , OL_I_ID
               , OL_SUPPLY_W_ID
               , OL_DELIVERY_D
               , OL_QUANTITY
               , OL_AMOUNT
               , OL_DIST_INFO
              )

              INCLUDE (   I_PRICE    DECIMAL(5,2)
                     , I_NAME     CHAR(24)
                     , I_DATA     VARCHAR(50)
                     , S_DATA     VARCHAR(50)
                     , S_QUANTITY SMALLINT )

                  SELECT   O_ID
                       , D_ID
                       , W_ID
                       , OL_NUMBER
                       , I_ID
                       , I_SUPPLY_W_ID
                       , OL_DELIVERY_D
                       , I_QTY
                       , TOTAL_PRICE
                       , OL_DIST_INFO
                       , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM DATA

            ) AS INS
;

EXEC SQL DECLARE ISOL_Local_6 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , W_ID AS I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM ( SELECT   :next_o_id as O_ID
                           , :w_id AS W_ID
                           , :d_id as D_ID
                           , OL_NUMBER
                           , I_ID
                           , I_QTY

                        FROM  Table( VALUES

                              (  SMALLINT( 1 )  , :id0  , :ol_quantity0  )
                             , (  SMALLINT( 2 )  , :id1  , :ol_quantity1  )
                             , (  SMALLINT( 3 )  , :id2  , :ol_quantity2  )
                             , (  SMALLINT( 4 )  , :id3  , :ol_quantity3  )
                             , (  SMALLINT( 5 )  , :id4  , :ol_quantity4  )
                             , (  SMALLINT( 6 )  , :id5  , :ol_quantity5  )

                              ) AS X ( OL_NUMBER , I_ID  , I_QTY          )
                        ) AS ITEMLIST

                       , TABLE( NEW_OL_LOCAL(   I_ID
                                  , I_QTY
                                  , W_ID
                                  , O_ID
                                  , D_ID
                                 )
                        ) AS NEW_OL_LOCAL

                  WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
            )


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

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

              (  OL_O_ID
               , OL_D_ID
               , OL_W_ID
               , OL_NUMBER
```

```
                , OL_I_ID
                , OL_SUPPLY_W_ID
                , OL_DELIVERY_D
                , OL_QUANTITY
                , OL_AMOUNT
                , OL_DIST_INFO
               )

               INCLUDE (   I_PRICE    DECIMAL(5,2)
                     , I_NAME     CHAR(24)
                     , I_DATA     VARCHAR(50)
                     , S_DATA     VARCHAR(50)
                     , S_QUANTITY SMALLINT )

                     SELECT   O_ID
                          , D_ID
                          , W_ID
                          , OL_NUMBER
                          , I_ID
                          , I_SUPPLY_W_ID
                          , OL_DELIVERY_D
                          , I_QTY
                          , TOTAL_PRICE
                          , OL_DIST_INFO
                          , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                     FROM DATA

               ) AS INS
;

EXEC SQL DECLARE ISOL_Local_7 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , W_ID AS I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM ( SELECT   :next_o_id as O_ID
                           , :w_id AS W_ID
                           , :d_id as D_ID
                           , OL_NUMBER
                           , I_ID
                           , I_QTY

                        FROM  Table( VALUES

                              (  SMALLINT( 1 )  , :id0  , :ol_quantity0  )
                             , (  SMALLINT( 2 )  , :id1  , :ol_quantity1  )
                             , (  SMALLINT( 3 )  , :id2  , :ol_quantity2  )
                             , (  SMALLINT( 4 )  , :id3  , :ol_quantity3  )
                             , (  SMALLINT( 5 )  , :id4  , :ol_quantity4  )
                             , (  SMALLINT( 6 )  , :id5  , :ol_quantity5  )
                             , (  SMALLINT( 7 )  , :id6  , :ol_quantity6  )

                             ) AS X ( OL_NUMBER , I_ID  , I_QTY          )
                        ) AS ITEMLIST

                       , TABLE( NEW_OL_LOCAL(   I_ID
                                  , I_QTY
                                  , W_ID
                                  , O_ID
                                  , D_ID
                                 )
                        ) AS NEW_OL_LOCAL

                              WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
                  )


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

    FROM NEW TABLE ( INSERT INTO ORDER_LINE

                 (   OL_O_ID
                   , OL_D_ID
                   , OL_W_ID
                   , OL_NUMBER
                   , OL_I_ID
                   , OL_SUPPLY_W_ID
                   , OL_DELIVERY_D
                   , OL_QUANTITY
                   , OL_AMOUNT
                   , OL_DIST_INFO
                 )

                 INCLUDE (   I_PRICE    DECIMAL(5,2)
                       , I_NAME     CHAR(24)
                       , I_DATA     VARCHAR(50)
                       , S_DATA     VARCHAR(50)
                       , S_QUANTITY SMALLINT )

                       SELECT   O_ID
                            , D_ID
                            , W_ID
                            , OL_NUMBER
                            , I_ID
                            , I_SUPPLY_W_ID
                            , OL_DELIVERY_D
                            , I_QTY
                            , TOTAL_PRICE
                            , OL_DIST_INFO
                            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                       FROM DATA

                 ) AS INS
;

EXEC SQL DECLARE ISOL_Local_8 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , W_ID AS I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM ( SELECT   :next_o_id as O_ID
                           , :w_id AS W_ID
                           , :d_id as D_ID
                           , OL_NUMBER
                           , I_ID
                           , I_QTY

                        FROM  Table( VALUES

                              (  SMALLINT( 1 )  , :id0  , :ol_quantity0  )
                             , (  SMALLINT( 2 )  , :id1  , :ol_quantity1  )
                             , (  SMALLINT( 3 )  , :id2  , :ol_quantity2  )
                             , (  SMALLINT( 4 )  , :id3  , :ol_quantity3  )
                             , (  SMALLINT( 5 )  , :id4  , :ol_quantity4  )
                             , (  SMALLINT( 6 )  , :id5  , :ol_quantity5  )
                             , (  SMALLINT( 7 )  , :id6  , :ol_quantity6  )
                             , (  SMALLINT( 8 )  , :id7  , :ol_quantity7  )

                             ) AS X ( OL_NUMBER , I_ID  , I_QTY          )
                        ) AS ITEMLIST

                       , TABLE( NEW_OL_LOCAL(   I_ID
                                  , I_QTY
                                  , W_ID
                                  , O_ID
                                  , D_ID
                                 )
                        ) AS NEW_OL_LOCAL

                              WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
                  )


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

    FROM NEW TABLE ( INSERT INTO ORDER_LINE

                 (   OL_O_ID
                   , OL_D_ID
                   , OL_W_ID
                   , OL_NUMBER
                   , OL_I_ID
                   , OL_SUPPLY_W_ID
                   , OL_DELIVERY_D
                   , OL_QUANTITY
                   , OL_AMOUNT
                   , OL_DIST_INFO
                 )

                 INCLUDE (   I_PRICE    DECIMAL(5,2)
                       , I_NAME     CHAR(24)
                       , I_DATA     VARCHAR(50)
                       , S_DATA     VARCHAR(50)
                       , S_QUANTITY SMALLINT )

                       SELECT   O_ID
                            , D_ID
                            , W_ID
                            , OL_NUMBER
                            , I_ID
                            , I_SUPPLY_W_ID
                            , OL_DELIVERY_D
                            , I_QTY
                            , TOTAL_PRICE
                            , OL_DIST_INFO
                            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                       FROM DATA

                 ) AS INS
;

EXEC SQL DECLARE ISOL_Local_9 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , W_ID AS I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM ( SELECT   :next_o_id as O_ID
                           , :w_id AS W_ID
                           , :d_id as D_ID
```

```
                    , OL_NUMBER
                    , I_ID
                    , I_QTY

                  FROM  Table( VALUES

                        (  SMALLINT( 1 )  , :id0  , :ol_quantity0  )
                       , (  SMALLINT( 2 )  , :id1  , :ol_quantity1  )
                       , (  SMALLINT( 3 )  , :id2  , :ol_quantity2  )
                       , (  SMALLINT( 4 )  , :id3  , :ol_quantity3  )
                       , (  SMALLINT( 5 )  , :id4  , :ol_quantity4  )
                       , (  SMALLINT( 6 )  , :id5  , :ol_quantity5  )
                       , (  SMALLINT( 7 )  , :id6  , :ol_quantity6  )
                       , (  SMALLINT( 8 )  , :id7  , :ol_quantity7  )
                       , (  SMALLINT( 9 )  , :id8  , :ol_quantity8  )

                       ) AS X ( OL_NUMBER , I_ID  , I_QTY          )
                 ) AS ITEMLIST

                , TABLE( NEW_OL_LOCAL(   I_ID
                            , I_QTY
                            , W_ID
                            , O_ID
                            , D_ID
                            )
                  ) AS NEW_OL_LOCAL

              WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
        )


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

 FROM NEW TABLE ( INSERT INTO ORDER_LINE

          (   OL_O_ID
            , OL_D_ID
            , OL_W_ID
            , OL_NUMBER
            , OL_I_ID
            , OL_SUPPLY_W_ID
            , OL_DELIVERY_D
            , OL_QUANTITY
            , OL_AMOUNT
            , OL_DIST_INFO
          )

          INCLUDE (   I_PRICE    DECIMAL(5,2)
                    , I_NAME     CHAR(24)
                    , I_DATA     VARCHAR(50)
                    , S_DATA     VARCHAR(50)
                    , S_QUANTITY SMALLINT )

                SELECT   O_ID
                       , D_ID
                       , W_ID
                       , OL_NUMBER
                       , I_ID
                       , I_SUPPLY_W_ID
                       , OL_DELIVERY_D
                       , I_QTY
                       , TOTAL_PRICE
                       , OL_DIST_INFO
                       , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                FROM DATA

          ) AS INS
;

EXEC SQL DECLARE ISOL_Local_10 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
                , D_ID
                , W_ID
                , OL_NUMBER
                , I_ID
                , W_ID AS I_SUPPLY_W_ID
                , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
                , I_QTY
                , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
                , OL_DIST_INFO
                , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM ( SELECT   :next_o_id as O_ID
                               , :w_id AS W_ID
                               , :d_id as D_ID
                               , OL_NUMBER
                               , I_ID
                               , I_QTY

                        FROM  Table( VALUES

                              (  SMALLINT( 1 )  , :id0  , :ol_quantity0  )
                             , (  SMALLINT( 2 )  , :id1  , :ol_quantity1  )
                             , (  SMALLINT( 3 )  , :id2  , :ol_quantity2  )
                             , (  SMALLINT( 4 )  , :id3  , :ol_quantity3  )
                             , (  SMALLINT( 5 )  , :id4  , :ol_quantity4  )
                             , (  SMALLINT( 6 )  , :id5  , :ol_quantity5  )
                             , (  SMALLINT( 7 )  , :id6  , :ol_quantity6  )
                             , (  SMALLINT( 8 )  , :id7  , :ol_quantity7  )
                             , (  SMALLINT( 9 )  , :id8  , :ol_quantity8  )
                             , (  SMALLINT( 10 ) , :id9  , :ol_quantity9  )

                             ) AS X ( OL_NUMBER , I_ID  , I_QTY          )
                        ) AS ITEMLIST

                      , TABLE( NEW_OL_LOCAL(   I_ID
                                  , I_QTY
                                  , W_ID
                                  , O_ID
                                  , D_ID
                                  )
                        ) AS NEW_OL_LOCAL

                    WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
          )


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

 FROM NEW TABLE ( INSERT INTO ORDER_LINE

          (   OL_O_ID
            , OL_D_ID
            , OL_W_ID
            , OL_NUMBER
            , OL_I_ID
            , OL_SUPPLY_W_ID
            , OL_DELIVERY_D
            , OL_QUANTITY
            , OL_AMOUNT
            , OL_DIST_INFO
          )

          INCLUDE (   I_PRICE    DECIMAL(5,2)
                    , I_NAME     CHAR(24)
                    , I_DATA     VARCHAR(50)
                    , S_DATA     VARCHAR(50)
                    , S_QUANTITY SMALLINT )

                SELECT   O_ID
                       , D_ID
                       , W_ID
                       , OL_NUMBER
                       , I_ID
                       , I_SUPPLY_W_ID
                       , OL_DELIVERY_D
                       , I_QTY
                       , TOTAL_PRICE
                       , OL_DIST_INFO
                       , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                FROM DATA

          ) AS INS
;

EXEC SQL DECLARE ISOL_Local_11 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
                , D_ID
                , W_ID
                , OL_NUMBER
                , I_ID
                , W_ID AS I_SUPPLY_W_ID
                , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
                , I_QTY
                , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
                , OL_DIST_INFO
                , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM ( SELECT   :next_o_id as O_ID
                               , :w_id AS W_ID
                               , :d_id as D_ID
                               , OL_NUMBER
                               , I_ID
                               , I_QTY

                        FROM  Table( VALUES

                              (  SMALLINT( 1 )  , :id0  , :ol_quantity0  )
                             , (  SMALLINT( 2 )  , :id1  , :ol_quantity1  )
                             , (  SMALLINT( 3 )  , :id2  , :ol_quantity2  )
                             , (  SMALLINT( 4 )  , :id3  , :ol_quantity3  )
                             , (  SMALLINT( 5 )  , :id4  , :ol_quantity4  )
                             , (  SMALLINT( 6 )  , :id5  , :ol_quantity5  )
                             , (  SMALLINT( 7 )  , :id6  , :ol_quantity6  )
                             , (  SMALLINT( 8 )  , :id7  , :ol_quantity7  )
                             , (  SMALLINT( 9 )  , :id8  , :ol_quantity8  )
                             , (  SMALLINT( 10 ) , :id9  , :ol_quantity9  )
                             , (  SMALLINT( 11 ) , :id10 , :ol_quantity10 )

                             ) AS X ( OL_NUMBER , I_ID  , I_QTY          )
                        ) AS ITEMLIST

                      , TABLE( NEW_OL_LOCAL(   I_ID
                                  , I_QTY
                                  , W_ID
                                  , O_ID
                                  , D_ID
                                  )
                        ) AS NEW_OL_LOCAL

                    WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
          )


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

 FROM NEW TABLE ( INSERT INTO ORDER_LINE

          (   OL_O_ID
            , OL_D_ID
            , OL_W_ID
            , OL_NUMBER
            , OL_I_ID
            , OL_SUPPLY_W_ID
            , OL_DELIVERY_D
```

```
                , OL_QUANTITY
                , OL_AMOUNT
                , OL_DIST_INFO
               )

               INCLUDE (   I_PRICE    DECIMAL(5,2)
                      , I_NAME     CHAR(24)
                      , I_DATA     VARCHAR(50)
                      , S_DATA     VARCHAR(50)
                      , S_QUANTITY SMALLINT )

                    SELECT   O_ID
                         , D_ID
                         , W_ID
                         , OL_NUMBER
                         , I_ID
                         , I_SUPPLY_W_ID
                         , OL_DELIVERY_D
                         , I_QTY
                         , TOTAL_PRICE
                         , OL_DIST_INFO
                         , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                    FROM DATA

               ) AS INS
;

EXEC SQL DECLARE ISOL_Local_12 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , W_ID AS I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM ( SELECT   :next_o_id as O_ID
                           , :w_id AS W_ID
                           , :d_id as D_ID
                           , OL_NUMBER
                           , I_ID
                           , I_QTY

                        FROM  Table( VALUES

                              (  SMALLINT( 1 )  , :id0  , :ol_quantity0  )
                             , (  SMALLINT( 2 )  , :id1  , :ol_quantity1  )
                             , (  SMALLINT( 3 )  , :id2  , :ol_quantity2  )
                             , (  SMALLINT( 4 )  , :id3  , :ol_quantity3  )
                             , (  SMALLINT( 5 )  , :id4  , :ol_quantity4  )
                             , (  SMALLINT( 6 )  , :id5  , :ol_quantity5  )
                             , (  SMALLINT( 7 )  , :id6  , :ol_quantity6  )
                             , (  SMALLINT( 8 )  , :id7  , :ol_quantity7  )
                             , (  SMALLINT( 9 )  , :id8  , :ol_quantity8  )
                             , (  SMALLINT( 10 ) , :id9  , :ol_quantity9  )
                             , (  SMALLINT( 11 ) , :id10 , :ol_quantity10 )
                             , (  SMALLINT( 12 ) , :id11 , :ol_quantity11 )

                             ) AS X ( OL_NUMBER , I_ID  , I_QTY          )
                        ) AS ITEMLIST

                      , TABLE( NEW_OL_LOCAL(   I_ID
                                  , I_QTY
                                  , W_ID
                                  , O_ID
                                  , D_ID
                                 )
                             ) AS NEW_OL_LOCAL

                        WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
            )


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

    FROM NEW TABLE ( INSERT INTO ORDER_LINE

               (  OL_O_ID
                , OL_D_ID
                , OL_W_ID
                , OL_NUMBER
                , OL_I_ID
                , OL_SUPPLY_W_ID
                , OL_DELIVERY_D
                , OL_QUANTITY
                , OL_AMOUNT
                , OL_DIST_INFO
               )

               INCLUDE (   I_PRICE    DECIMAL(5,2)
                      , I_NAME     CHAR(24)
                      , I_DATA     VARCHAR(50)
                      , S_DATA     VARCHAR(50)
                      , S_QUANTITY SMALLINT )

                    SELECT   O_ID
                         , D_ID
                         , W_ID
                         , OL_NUMBER
                         , I_ID
                         , I_SUPPLY_W_ID
                         , OL_DELIVERY_D
                         , I_QTY
                         , TOTAL_PRICE
                         , OL_DIST_INFO
                         , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                    FROM DATA

               ) AS INS
;

EXEC SQL DECLARE ISOL_Local_13 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , W_ID AS I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                  FROM ( SELECT   :next_o_id as O_ID
                           , :w_id AS W_ID
                           , :d_id as D_ID
                           , OL_NUMBER
                           , I_ID
                           , I_QTY

                        FROM  Table( VALUES

                              (  SMALLINT( 1 )  , :id0  , :ol_quantity0  )
                             , (  SMALLINT( 2 )  , :id1  , :ol_quantity1  )
                             , (  SMALLINT( 3 )  , :id2  , :ol_quantity2  )
                             , (  SMALLINT( 4 )  , :id3  , :ol_quantity3  )
                             , (  SMALLINT( 5 )  , :id4  , :ol_quantity4  )
                             , (  SMALLINT( 6 )  , :id5  , :ol_quantity5  )
                             , (  SMALLINT( 7 )  , :id6  , :ol_quantity6  )
                             , (  SMALLINT( 8 )  , :id7  , :ol_quantity7  )
                             , (  SMALLINT( 9 )  , :id8  , :ol_quantity8  )
                             , (  SMALLINT( 10 ) , :id9  , :ol_quantity9  )
                             , (  SMALLINT( 11 ) , :id10 , :ol_quantity10 )
                             , (  SMALLINT( 12 ) , :id11 , :ol_quantity11 )
                             , (  SMALLINT( 13 ) , :id12 , :ol_quantity12 )

                             ) AS X ( OL_NUMBER , I_ID  , I_QTY          )
                        ) AS ITEMLIST

                      , TABLE( NEW_OL_LOCAL(   I_ID
                                  , I_QTY
                                  , W_ID
                                  , O_ID
                                  , D_ID
                                 )
                             ) AS NEW_OL_LOCAL

                        WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
            )


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

    FROM NEW TABLE ( INSERT INTO ORDER_LINE

               (  OL_O_ID
                , OL_D_ID
                , OL_W_ID
                , OL_NUMBER
                , OL_I_ID
                , OL_SUPPLY_W_ID
                , OL_DELIVERY_D
                , OL_QUANTITY
                , OL_AMOUNT
                , OL_DIST_INFO
               )

               INCLUDE (   I_PRICE    DECIMAL(5,2)
                      , I_NAME     CHAR(24)
                      , I_DATA     VARCHAR(50)
                      , S_DATA     VARCHAR(50)
                      , S_QUANTITY SMALLINT )

                    SELECT   O_ID
                         , D_ID
                         , W_ID
                         , OL_NUMBER
                         , I_ID
                         , I_SUPPLY_W_ID
                         , OL_DELIVERY_D
                         , I_QTY
                         , TOTAL_PRICE
                         , OL_DIST_INFO
                         , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                    FROM DATA

               ) AS INS
;

EXEC SQL DECLARE ISOL_Local_14 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , W_ID AS I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
```

```
        , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
        , OL_DIST_INFO
        , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                FROM ( SELECT   :next_o_id as O_ID
                        , :w_id AS W_ID
                        , :d_id as D_ID
                        , OL_NUMBER
                        , I_ID
                        , I_QTY

                    FROM  Table( VALUES

                            (  SMALLINT( 1 )  , :id0  , :ol_quantity0  )
                            , (  SMALLINT( 2 )  , :id1  , :ol_quantity1  )
                            , (  SMALLINT( 3 )  , :id2  , :ol_quantity2  )
                            , (  SMALLINT( 4 )  , :id3  , :ol_quantity3  )
                            , (  SMALLINT( 5 )  , :id4  , :ol_quantity4  )
                            , (  SMALLINT( 6 )  , :id5  , :ol_quantity5  )
                            , (  SMALLINT( 7 )  , :id6  , :ol_quantity6  )
                            , (  SMALLINT( 8 )  , :id7  , :ol_quantity7  )
                            , (  SMALLINT( 9 )  , :id8  , :ol_quantity8  )
                            , (  SMALLINT( 10 ) , :id9  , :ol_quantity9  )
                            , (  SMALLINT( 11 ) , :id10 , :ol_quantity10 )
                            , (  SMALLINT( 12 ) , :id11 , :ol_quantity11 )
                            , (  SMALLINT( 13 ) , :id12 , :ol_quantity12 )
                            , (  SMALLINT( 14 ) , :id13 , :ol_quantity13 )

                           ) AS X ( OL_NUMBER , I_ID  , I_QTY          )
                  ) AS ITEMLIST

                 , TABLE( NEW_OL_LOCAL(   I_ID
                              , I_QTY
                              , W_ID
                              , O_ID
                              , D_ID
                              )
                    ) AS NEW_OL_LOCAL

                WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
        )


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

 FROM NEW TABLE ( INSERT INTO ORDER_LINE

            (   OL_O_ID
              , OL_D_ID
              , OL_W_ID
              , OL_NUMBER
              , OL_I_ID
              , OL_SUPPLY_W_ID
              , OL_DELIVERY_D
              , OL_QUANTITY
              , OL_AMOUNT
              , OL_DIST_INFO
            )

            INCLUDE (   I_PRICE    DECIMAL(5,2)
                  , I_NAME     CHAR(24)
                  , I_DATA     VARCHAR(50)
                  , S_DATA     VARCHAR(50)
                  , S_QUANTITY SMALLINT )

                SELECT   O_ID
                    , D_ID
                    , W_ID
                    , OL_NUMBER
                    , I_ID
                    , I_SUPPLY_W_ID
                    , OL_DELIVERY_D
                    , I_QTY
                    , TOTAL_PRICE
                    , OL_DIST_INFO
                    , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                FROM DATA

            ) AS INS
;

EXEC SQL DECLARE ISOL_Local_15 CURSOR FOR

  WITH DATA AS (  SELECT  O_ID
            , D_ID
            , W_ID
            , OL_NUMBER
            , I_ID
            , W_ID AS I_SUPPLY_W_ID
            , (TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
            , I_QTY
            , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
            , OL_DIST_INFO
            , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                FROM ( SELECT   :next_o_id as O_ID
                        , :w_id AS W_ID
                        , :d_id as D_ID
                        , OL_NUMBER
                        , I_ID
                        , I_QTY

                    FROM  Table( VALUES

                            (  SMALLINT( 1 )  , :id0  , :ol_quantity0  )
                            , (  SMALLINT( 2 )  , :id1  , :ol_quantity1  )
                            , (  SMALLINT( 3 )  , :id2  , :ol_quantity2  )
                            , (  SMALLINT( 4 )  , :id3  , :ol_quantity3  )
                            , (  SMALLINT( 5 )  , :id4  , :ol_quantity4  )
                            , (  SMALLINT( 6 )  , :id5  , :ol_quantity5  )
                            , (  SMALLINT( 7 )  , :id6  , :ol_quantity6  )
                            , (  SMALLINT( 8 )  , :id7  , :ol_quantity7  )
                            , (  SMALLINT( 9 )  , :id8  , :ol_quantity8  )
                            , (  SMALLINT( 10 ) , :id9  , :ol_quantity9  )
                            , (  SMALLINT( 11 ) , :id10 , :ol_quantity10 )
                            , (  SMALLINT( 12 ) , :id11 , :ol_quantity11 )
                            , (  SMALLINT( 13 ) , :id12 , :ol_quantity12 )
                            , (  SMALLINT( 14 ) , :id13 , :ol_quantity13 )
                            , (  SMALLINT( 15 ) , :id14 , :ol_quantity14 )

                           ) AS X ( OL_NUMBER , I_ID  , I_QTY          )
                  ) AS ITEMLIST

                 , TABLE( NEW_OL_LOCAL(   I_ID
                              , I_QTY
                              , W_ID
                              , O_ID
                              , D_ID
                              )
                    ) AS NEW_OL_LOCAL

                WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
        )


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

   FROM NEW TABLE ( INSERT INTO ORDER_LINE

            (   OL_O_ID
              , OL_D_ID
              , OL_W_ID
              , OL_NUMBER
              , OL_I_ID
              , OL_SUPPLY_W_ID
              , OL_DELIVERY_D
              , OL_QUANTITY
              , OL_AMOUNT
              , OL_DIST_INFO
            )

            INCLUDE (   I_PRICE    DECIMAL(5,2)
                  , I_NAME     CHAR(24)
                  , I_DATA     VARCHAR(50)
                  , S_DATA     VARCHAR(50)
                  , S_QUANTITY SMALLINT )

                SELECT   O_ID
                    , D_ID
                    , W_ID
                    , OL_NUMBER
                    , I_ID
                    , I_SUPPLY_W_ID
                    , OL_DELIVERY_D
                    , I_QTY
                    , TOTAL_PRICE
                    , OL_DIST_INFO
                    , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

                FROM DATA

            ) AS INS
  ;

  // Start processing

  in_neword = (struct in_neword_struct *)  pin ;
  neword    = (struct out_neword_struct *) pout ;

#ifdef DEBUGIT
  new_debug( neword, in_neword, "SP upon entry");
#endif

  // Using I_PRICE == 0 as a flag to the client that the ITEM was not fetched (hence bad).

  for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < in_neword->s_O_OL_CNT ;
inputItemArrayIndex++ )
  {
    i_priceArray[ inputItemArrayIndex ] = 0 ;
  }

  neword->deadlocks = -1 ;

 retry_tran:

  neword->deadlocks++ ;

  EXEC SQL

    SELECT D_TAX, D_NEXT_O_ID  INTO :dist_tax , :next_o_id

      FROM OLD TABLE ( UPDATE DISTRICT

                SET  D_NEXT_O_ID = D_NEXT_O_ID + 1

               WHERE D_W_ID = :w_id
                 AND D_ID = :d_id
            ) AS OT
  ;

  if ( sqlca.sqlcode != 0 )
  {
    DLCHK( retry_tran );
    sqlerror( NEWORD_SQL, "DISTRICT", __FILE__, __LINE__, &sqlca ) ;
    goto ferror;
  }

  #define NEW_CURSOR_OPEN_ERROR                                    \
```

```
  {                                                    \
    if( sqlca.sqlcode != 0    )                              \
    {                                                  \
      goto sql_error ;                                     \
    }                                                  \
  }

  #define NEW_CURSOR_ERROR                                          \
  {                                                    \
    if( sqlca.sqlcode == 0    )                              \
    {                                                  \
      neword->s_O_OL_CNT ++ ;                                   \
    }                                                  \
    else                                                 \
    if( sqlca.sqlcode == +100 )                              \
    {                                                  \
      break ;                                            \
    }                                                  \
    else                                                 \
      goto sql_error ;                                     \
  }

  if ( allLocal )
  {
    switch( inputItemCount )
    {
      case 1:
        EXEC SQL OPEN ISOL_Local_1 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Local_1
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 2:
        EXEC SQL OPEN ISOL_Local_2 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Local_2
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 3:
        EXEC SQL OPEN ISOL_Local_3 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Local_3
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 4:
        EXEC SQL OPEN ISOL_Local_4 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Local_4
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 5:
        EXEC SQL OPEN ISOL_Local_5 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Local_5
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 6:
        EXEC SQL OPEN ISOL_Local_6 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Local_6
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 7:
        EXEC SQL OPEN ISOL_Local_7 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Local_7
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 8:
        EXEC SQL OPEN ISOL_Local_8 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Local_8
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 9:
        EXEC SQL OPEN ISOL_Local_9 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Local_9
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 10:
        EXEC SQL OPEN ISOL_Local_10 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Local_10
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 11:
        EXEC SQL OPEN ISOL_Local_11 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Local_11
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 12:
        EXEC SQL OPEN ISOL_Local_12 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Local_12
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 13:
        EXEC SQL OPEN ISOL_Local_13 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Local_13
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 14:
        EXEC SQL OPEN ISOL_Local_14 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Local_14
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 15:
        EXEC SQL OPEN ISOL_Local_15 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Local_15
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;

      default:
        sqlerror(NEWORD_SQL, "Default switch on local  orderline/stock/index", __FILE__, __LINE__,
&sqlca );
        goto ferror;
    }
  }
  else
  {
    switch( inputItemCount )
    {
      case 1:
        EXEC SQL OPEN ISOL_Remote_1 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Remote_1
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 2:
        EXEC SQL OPEN ISOL_Remote_2 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Remote_2
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 3:
        EXEC SQL OPEN ISOL_Remote_3 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Remote_3
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 4:
        EXEC SQL OPEN ISOL_Remote_4 ;
```

```
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Remote_4
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 5:
        EXEC SQL OPEN ISOL_Remote_5 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Remote_5
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 6:
        EXEC SQL OPEN ISOL_Remote_6 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Remote_6
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 7:
        EXEC SQL OPEN ISOL_Remote_7 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Remote_7
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 8:
        EXEC SQL OPEN ISOL_Remote_8 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Remote_8
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 9:
        EXEC SQL OPEN ISOL_Remote_9 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Remote_9
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 10:
        EXEC SQL OPEN ISOL_Remote_10 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Remote_10
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 11:
        EXEC SQL OPEN ISOL_Remote_11 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Remote_11
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 12:
        EXEC SQL OPEN ISOL_Remote_12 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Remote_12
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 13:
        EXEC SQL OPEN ISOL_Remote_13 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Remote_13
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 14:
        EXEC SQL OPEN ISOL_Remote_14 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Remote_14
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;
      case 15:
        EXEC SQL OPEN ISOL_Remote_15 ;
        NEW_CURSOR_OPEN_ERROR
        for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )
        {
          EXEC SQL FETCH ISOL_Remote_15
INTO :item_price, :item_name, :i_data, :stockDistrictInformation , :s_data , :s_quantity ;
          NEW_CURSOR_ERROR
        }
        break ;

      default:
        sqlerror(NEWORD_SQL, "Default switch on remote orderline/stock/index", __FILE__, __LINE__,
&sqlca);
        goto ferror;
    }
  }

  for ( inputItemArrayIndex = 0 ;
        inputItemArrayIndex < in_neword->s_O_OL_CNT  // from input
      && i_priceArray[ inputItemArrayIndex ] != 0 ;
      inputItemArrayIndex++ )
  {
    // s_I_NAME, and s_S_QUANTITY already set as output host variables

    neword->item[ inputItemArrayIndex ].s_I_PRICE = i_priceArray[ inputItemArrayIndex ] ;

    if (   is_ORIGINAL( s_dataArray[ inputItemArrayIndex ].data, s_dataArray[ inputItemArrayIndex ].len )
        && is_ORIGINAL( i_dataArray[ inputItemArrayIndex ].data,
i_dataArray[ inputItemArrayIndex ].len )    )
    {
      neword->item[ inputItemArrayIndex ].s_brand_generic = 'B';
    }
    else
    {
      neword->item[ inputItemArrayIndex ].s_brand_generic = 'G';
    }
  }

  EXEC SQL

    SELECT W_TAX, C_DISCOUNT, C_LAST, C_CREDIT, O_ENTRY_D

     INTO :ware_tax, :c_discount, :c_last, :c_credit, :o_entry_d

     FROM TABLE ( NEW_WH (   :next_o_id
                           , :w_id
                           , :d_id
                           , :c_id
                           , :inputItemCount
                           , :allLocal
                         )
                ) AS NEW_WH_TABLE
  ;

  if ( sqlca.sqlcode == 0 )
  {
    if ( neword->s_O_OL_CNT == in_neword->s_O_OL_CNT  )
    {
      neword->s_transtatus = TRAN_OK ;

      EXEC SQL COMMIT;

      if( sqlca.sqlcode != 0 )
      {
        sqlerror(NEWORD_SQL, "COMMIT", __FILE__, __LINE__ , &sqlca ) ;
        goto ferror;
      }
    }
    else
    {
      neword->s_transtatus = INVALID_ITEM ;

      EXEC SQL ROLLBACK WORK ;

      if ( sqlca.sqlcode != 0 )
      {
        neword->s_transtatus = FATAL_SQLERROR;

        sqlerror(NEWORD_SQL, "ROLLBACK FAILED (INVALID ITEM)", __FILE__, __LINE__ , &sqlca);
        // no point in ferror
      }
    }
  }
  else
  {
    DLCHK( retry_tran );

    sqlerror( NEWORD_SQL, "NEW_WH", __FILE__, __LINE__ , &sqlca);
    goto ferror;
  }

  /*-----------------------------------------*/
  /*  Return to client                       */
  /*-----------------------------------------*/

mexit:

  if ( sqlca.sqlcode >= 0 )
  {
    storedProcRc = SQLZ_HOLD_PROC ;
  }
  else
  {
    storedProcRc = SQLZ_DISCONNECT_PROC ;
  }

#ifdef DEBUGIT
  new_debug( neword, in_neword, "SP prior to return");
#endif
```

```
  return ( storedProcRc ) ;

sql_error:

  {
    char tempstr[ 4096 ] ;

    DLCHK( retry_tran ) ;

    sprintf( tempstr, "inputItemCount=%d, :next_o_id=%d, :d_id=%d, :w_id=%d", inputItemCount,
next_o_id, d_id, w_id ) ;
    sqlerror( NEWORD_SQL, tempstr , __FILE__, __LINE__, &sqlca ) ;
  }

ferror:

  neword->s_transtatus = FATAL_SQLERROR;

  EXEC SQL ROLLBACK WORK;

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


  goto mexit ;
}

/*
** A little function to search for the string "ORIGINAL" given a string and
** it's length
*/
static unsigned char  skip[256] = {8,8,8,8,8,8,8,8,8,8,  /*0-9*/
                                   8,8,8,8,8,8,8,8,8,8,  /*10-19*/
                                   8,8,8,8,8,8,8,8,8,8,  /*20-29*/
                                   8,8,8,8,8,8,8,8,8,8,  /*30-39*/
                                   8,8,8,8,8,8,8,8,8,8,  /*40-49*/
                                   8,8,8,8,8,8,8,8,8,8,  /*50-59*/
                                   8,8,8,8,8,1,8,8,8,8,  /*60-69*/
                                   8,4,8,3,8,8,0,8,2,7,  /*70-79*/
                                   8,8,6,8,8,8,8,8,8,8,  /*80-89*/
                                   8,8,8,8,8,8,8,8,8,8,  /*90-99*/
                                   8,8,8,8,8,8,8,8,8,8,  /*100-109*/
                                   8,8,8,8,8,8,8,8,8,8,  /*110-119*/
                                   8,8,8,8,8,8,8,8,8,8,  /*120-129*/
                                   8,8,8,8,8,8,8,8,8,8,  /*130-139*/
                                   8,8,8,8,8,8,8,8,8,8,  /*140-149*/
                                   8,8,8,8,8,8,8,8,8,8,  /*150-159*/
                                   8,8,8,8,8,8,8,8,8,8,  /*160-169*/
                                   8,8,8,8,8,8,8,8,8,8,  /*170-179*/
                                   8,8,8,8,8,8,8,8,8,8,  /*180-189*/
                                   8,8,8,8,8,8,8,8,8,8,  /*190-199*/
                                   8,8,8,8,8,8,8,8,8,8,  /*200-209*/
                                   8,8,8,8,8,8,8,8,8,8,  /*210-219*/
                                   8,8,8,8,8,8,8,8,8,8,  /*220-229*/
                                   8,8,8,8,8,8,8,8,8,8,  /*230-239*/
                                   8,8,8,8,8,8,8,8,8,8,  /*240-249*/
                                   8,8,8,8,8};           /*250-254*/

static int is_ORIGINAL( char *string, short length )
{
  char          *cur_string;
  char          *end_string;
  unsigned char   *skips;
  int           skip_dist;
  int           result = 0;

  cur_string = string+7;
  end_string = string + length;
  skips = skip;

  while (cur_string < end_string)
  {
    skip_dist = skips[*cur_string];
    while ( (skip_dist > 0) && (cur_string < end_string) )
      {
        skip_dist = skips[*(cur_string += skip_dist)];
      }

    if (cur_string >= end_string)
      goto exit;

    if ( cur_string[-4] != 'G' )
      goto noMatch;

    if ( memcmp( cur_string-7, "ORIGINAL", 8 ) == 0 )
      {
        result = 1;
        goto exit;
      }
noMatch:
    cur_string += 8;
  } /* end while */

exit:
  return ( result ) ;
}

// ---------------------------------------------------------------
// Order Status SERVER
// ---------------------------------------------------------------

#undef w_id
#undef d_id
#undef c_id_input
#undef o_id
#undef o_entry_d
#undef o_carrier_d
#undef c_id
#undef c_first
#undef c_middle
#undef c_last
#undef c_balance

SQL_API_RC order_status_internal( char *pin, char *pout )
{
  struct in_ordstat_struct  * in_ordstat = (struct in_ordstat_struct *) pin ;
  struct out_ordstat_struct * ordstat = (struct out_ordstat_struct *) pout ;

  struct sqlca sqlca ;

  EXEC SQL BEGIN DECLARE SECTION;

    // From input values

    //##sqlint32   w_id ;
    //##short      d_id;
    sqlint32   c_id_input ;

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

    // From queries

    // From initial query

    sqlint32   o_id ;
    //##sqlint32   c_id ;
    short      o_carrier_id ;
    //##sqlint64   o_entry_d ;

    char   c_first[ 16 ] ;
    char   c_middle[ 2 ] ;
    //##char   c_last[ 16 ] ;
    double c_balance ;

    // From cursor

    sqlint32   ol_i_id ;
    sqlint32   ol_supply_w_id ;
    short      ol_quantity ;
    float   ol_amount ;
    char    ol_delivery_d [27] ;
    //##char    o_entry_d[ 27 ] ;

  EXEC SQL END DECLARE SECTION;

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

  int     storedProcRc ;
  int     itemArrayIndex = 0 ;

  #define w_id                in_ordstat->s_W_ID ;
  #define d_id                in_ordstat->s_D_ID ;
  #define c_id_input          in_ordstat->s_C_ID
  #define o_id                ordstat->s_O_ID
  #define o_entry_d           ordstat->s_O_ENTRY_D_time
  #define o_carrier_id        ordstat->s_O_CARRIER_ID
  #define c_id                ordstat->s_C_ID
  #define c_first             ordstat->s_C_FIRST
  #define c_middle            ordstat->s_C_MIDDLE
  #define c_last              ordstat->s_C_LAST
  #define c_balance           ordstat->s_C_BALANCE

  EXEC SQL DECLARE read_orderline_cur CURSOR FOR

    SELECT OL_I_ID, OL_SUPPLY_W_ID, OL_QUANTITY, OL_AMOUNT, OL_DELIVERY_D

      FROM ORDER_LINE

      WHERE OL_W_ID = :w_id
        AND OL_D_ID = :d_id
        AND OL_O_ID = :o_id

      FOR FETCH ONLY ;

  ordstat->deadlocks = -1 ;

#ifdef DEBUGIT
  ord_debug(ordstat, in_ordstat, "SP upon entry");
#endif

retry_tran:

  ordstat->deadlocks ++ ;

  if ( c_id_input == 0 )
  {
    c_last_input.len = strlen( in_ordstat->s_C_LAST ) ;
    memcpy( c_last_input.data , in_ordstat->s_C_LAST , c_last_input.len ) ;

    EXEC SQL

      SELECT O_ID, O_CARRIER_ID, O_ENTRY_D, C_BALANCE, C_FIRST, C_MIDDLE, C_ID

        INTO :o_id, :o_carrier_id , :o_entry_d , :c_balance, :c_first, :c_middle, :c_id

        FROM TABLE ( ORD_C_LAST(   :w_id
                                 , :d_id
                                 , :c_last_input
                                )
                   ) AS ORD_C_LAST
    ;
  }
  else
  {
    EXEC SQL

      SELECT O_ID, O_CARRIER_ID, O_ENTRY_D , C_BALANCE, C_FIRST, C_MIDDLE ,C_LAST
```

```
        INTO :o_id, :o_carrier_id , :o_entry_d , :c_balance, :c_first, :c_middle, :c_last

        FROM TABLE ( ORD_C_ID(   :w_id
                               , :d_id
                               , :c_id_input
                               )
                   ) AS ORD_C_ID
    ;
  }

  if ( sqlca.sqlcode != 0 )
  {
    DLCHK( retry_tran );
    sqlerror( ORDSTAT_SQL, "READ CUST and ORDERS", __FILE__, __LINE__ ,  &sqlca ) ;
    goto ferror;
  }

  /*-----------------------------------------------*/
  /*  Read ORDER_LINEs                             */
  /*-----------------------------------------------*/

  EXEC SQL OPEN read_orderline_cur ;

  if ( sqlca.sqlcode != 0 )
  {
    DLCHK( retry_tran );
    sqlerror(ORDSTAT_SQL, "OPEN CURSOR read_orderline_cur", __FILE__, __LINE__ , &sqlca ) ;
    goto ferror;
  }

  itemArrayIndex = 0 ;
  {
    do
    {
      EXEC SQL FETCH read_orderline_cur

              INTO :ol_i_id , :ol_supply_w_id , :ol_quantity , :ol_amount , :ol_delivery_d ;

      if ( sqlca.sqlcode == 0 )
      {
        ordstat->item[ itemArrayIndex ].s_OL_I_ID           = ol_i_id ;
        ordstat->item[ itemArrayIndex ].s_OL_SUPPLY_W_ID    = ol_supply_w_id ;
        ordstat->item[ itemArrayIndex ].s_OL_QUANTITY       = ol_quantity ;
        ordstat->item[ itemArrayIndex ].s_OL_AMOUNT         = ol_amount ;
        strcpy(ordstat->item[ itemArrayIndex ].s_OL_DELIVERY_D_time, ol_delivery_d) ;

        itemArrayIndex++;
      }
      else
      if (sqlca.sqlcode < 0 )
      {
        DLCHK( retry_tran ) ;
        sqlerror( ORDSTAT_SQL, "FETCH CURSOR read_orderline_cur" , __FILE__, __LINE__ ,
&sqlca ) ;
        goto ferror ;
      }
    }
    while ( sqlca.sqlcode == 0 ) ;
  }

  ordstat->s_ol_cnt = itemArrayIndex ;

  EXEC SQL COMMIT ;

  if ( sqlca.sqlcode == 0 )
  {
    ordstat->s_transtatus = TRAN_OK ;
  }
  else
  {
    DLCHK( retry_tran );
    sqlerror(ORDSTAT_SQL, "COMMIT", __FILE__, __LINE__ , &sqlca);
    goto ferror ;
  }

mexit:

  if ( sqlca.sqlcode >= 0 )
  {
    storedProcRc = SQLZ_HOLD_PROC ;
  }
  else
  {
    storedProcRc = SQLZ_DISCONNECT_PROC ;
  }

#ifdef DEBUGIT
  ord_debug(ordstat, in_ordstat, "SP prior to return");
#endif

  return ( storedProcRc ) ;

ferror:

  ordstat->s_transtatus = FATAL_SQLERROR ;

  EXEC SQL ROLLBACK WORK ;

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

  goto mexit;
}

// ---------------------------------------------------------------
// Delivery SERVER
// ---------------------------------------------------------------

#undef d_id
#undef c_id
#undef w_id
#undef o_carrier_id
#undef ol_delivery_d

SQL_API_RC delivery_internal ( char * pin, char * pout )
{
  struct in_delivery_struct * in_delivery = (struct in_delivery_struct *)  pin ;
  struct out_delivery_struct * delivery   = (struct out_delivery_struct *) pout ;

  struct sqlca sqlca ;

  int storedProcRc ;

  short district_id ;
  sqlint32  customer_id ;

  EXEC SQL BEGIN DECLARE SECTION;

    // input

    //##sqlint32   w_id ;
    //##short      d_id ;
    //##sqlint32   c_id ;
    //##short      o_carrier_id ;
    //##sqlint64   ol_delivery_d ;

    // output

    short      no_o_id_indicator = 0 ;
    sqlint32   no_o_id ;

  EXEC SQL END DECLARE SECTION;

  #define d_id         district_id
  #define c_id         customer_id

  #define w_id         in_delivery->s_W_ID
  #define o_carrier_id  in_delivery->s_O_CARRIER_ID
  #define ol_delivery_d in_delivery->s_O_DELIVERY_D_time

  delivery->deadlocks = -1 ;

#ifdef DEBUGIT
  del_debug( delivery, in_delivery, "SP upon entry");
#endif

  // Deadlock Handling
  // ----------------
  // Since we COMMIT inside the for() loop, we must take special
  // care while handling deadlocks.  This is best explained by
  // an example.
  //
  // Assume we deadlock on d_id=6.  This means that an order from the
  // first 5 districts have already been delivered.  We will then
  // restart the loop (retry_tran).  However, the loop will restart
  // at d_id = 1!  This means that the second (and all subsequent)
  // time through the loop, we will deliver orders for districts that
  // have already been delivered, with the net result being more than
  // 10 orders being delivered.
  //
  // The solution to this problem is to initialize the starting point
  // of the loop *before* the retry_tran label.  This will ensure that
  // if we deadlock, we will restart the loop with the same district
  // that we deadlocked on, and we won't deliver any extra orders.
  //
  // NOTE: If we ever change this back to one COMMIT per transaction
  // (instead of one COMMIT per iteration), then the initialization
  // of d_id must be moved back into the for loop.  (A rollback due
  // to deadlock in this case would rollback all delivered orders so
  // far, so we'd need to re-deliver them all on the next iteration.)

  d_id = 1;

retry_tran:

  delivery->deadlocks++;

  for ( ;  d_id <= DISTRICTS_PER_WAREHOUSE ; d_id++ )
  {
    no_o_id = 0 ;
    no_o_id_indicator = 0 ;

      EXEC SQL BEGIN COMPOUND NOT ATOMIC STATIC

        SELECT O_ID

              INTO :no_o_id :no_o_id_indicator

          FROM TABLE (  DEL( :w_id , :d_id , :o_carrier_id ) ) AS T ;

        COMMIT ;

      END COMPOUND ;

    if ( sqlca.sqlcode == 0 )
    {
      delivery->s_O_ID[ d_id - 1 ] = no_o_id ;
    }
    else
    {
      DLCHK( retry_tran );

      sqlerror( DELIVERY_SQL , "DELIVERY", __FILE__, __LINE__ , &sqlca);
      goto ferror ;
    }
  }
```

```
  delivery->s_transtatus = TRAN_OK ;

mexit:

  if ( sqlca.sqlcode >= 0 )
  {
    storedProcRc = SQLZ_HOLD_PROC ;
  }
  else
  {
    storedProcRc = SQLZ_DISCONNECT_PROC ;
  }

#ifdef DEBUGIT
  del_debug( delivery, in_delivery, "SP prior to return");
#endif

  return ( storedProcRc ) ;

ferror:

  delivery->s_transtatus = FATAL_SQLERROR ;

  EXEC SQL ROLLBACK WORK ;

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


  goto mexit ;
}

// ---------------------------------------------------------------
// Stored Procedure Stubs
// ---------------------------------------------------------------

SQL_API_RC SQL_API_FN news( char *pin, char *pout )
{
  return new_order_internal( pin, pout ) ;
}

SQL_API_RC SQL_API_FN ords( char *pin, char *pout )
{
  return order_status_internal( pin, pout ) ;
}

SQL_API_RC SQL_API_FN dels ( char * pin, char * pout )
{
  return delivery_internal( pin, pout ) ;
}
```

include/db2tpcc.h

```
/*****************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****************************************************************************/

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

#ifndef __DB2TPCC_H
#define __DB2TPCC_H

#include <sys/types.h>

#include "lval.h"

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

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

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

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

#define MIN_WAREHOUSE 1
#define MAX_WAREHOUSE WAREHOUSES

/*********************************************************************/
/* NURand Constants                                                  */
/* C_C_LAST_RUN and C_C_LAST_LOAD must adhere to clause 2.1.6.       */
/* Analysis indicates that a C_LAST delta of 85 is optimal.          */
/*********************************************************************/
#define C_C_LAST_RUN          88
#define C_C_LAST_LOAD         173
#define C_C_ID                319
#define C_OL_I_ID             3849
#define A_C_LAST              255
#define A_C_ID                1023
#define A_OL_I_ID             8191

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

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

#define SPGENERAL_PAD 3
#define SPGENERAL_ADJUST sizeof(int16_t)

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

struct out_neword_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  struct   items_struct {
    float  s_I_PRICE;
    float  s_OL_AMOUNT;
    int16_t  s_S_QUANTITY;
    int16_t  pad2;
    char     s_I_NAME[25];
    char     s_brand_generic;
  } item[15];
  float  s_W_TAX;
  float  s_D_TAX;
  float  s_C_DISCOUNT;
  float  s_total_amount;
  int32_t  s_O_ID;
  int16_t  s_O_OL_CNT;
  int16_t  s_transtatus;
  int16_t  deadlocks;
  char     s_C_LAST[17];
  char     s_C_CREDIT[3];
  char     s_O_ENTRY_D_time[27];

};

struct in_payment_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  float  s_H_AMOUNT;
  int32_t  s_W_ID;
  int32_t  s_C_W_ID;
  int32_t  s_C_ID;
  int16_t  s_C_D_ID;
  int16_t  s_D_ID;
  char     s_C_LAST[17];
};

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

```
  char     s_C_DATA[201];
  char     s_H_DATE_time[27];
  char     s_C_SINCE_time[27];
};

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

struct out_ordstat_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  double  s_C_BALANCE;
  int32_t  s_C_ID;
  int32_t  s_O_ID;
  int16_t  s_O_CARRIER_ID;
  int16_t  s_ol_cnt;
  int16_t  pad1[2];
  struct   oitems_struct {
    double  s_OL_AMOUNT;
    int32_t  s_OL_I_ID;
    int32_t  s_OL_SUPPLY_W_ID;
    int16_t  s_OL_QUANTITY;
    int16_t  pad2;
    char     s_OL_DELIVERY_D_time[27];
  } item[15];
  int16_t  s_transtatus;
  int16_t  deadlocks;
  char     s_C_FIRST[17];
  char     s_C_MIDDLE[3];
  char     s_C_LAST[17];
  char     s_O_ENTRY_D_time[27];
  int16_t  pad3[2];
};

struct in_delivery_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  int32_t  s_W_ID;
  int16_t  s_O_CARRIER_ID;
};

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

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

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

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

#ifdef __cplusplus
extern "C" {
#endif

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

#ifdef __cplusplus
}
#endif

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

#ifdef __cplusplus
extern "C" {
#endif

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


#ifdef __cplusplus
}
#endif

#endif // __DB2TPCC_H
```

include/lval.h

```
/* lval.h - generated automatically at 20080316.1422 */

#ifndef __LVAL_H
#define __LVAL_H
#define WAREHOUSES 51968
#define DISTRICTS_PER_WAREHOUSE 10
#define CUSTOMERS_PER_DISTRICT 3000
#define ITEMS 100000
#define STOCK_PER_WAREHOUSE 100000
#define MIN_OL_PER_ORDER 5
#define MAX_OL_PER_ORDER 15
#define NU_ORDERS_PER_DISTRICT 900
#endif // __LVAL_H
```

include/tpccapp.h

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

/*
 * tpccapp.h - Application Macros
 *
 */

#ifndef __TPCCAPP_H
#define __TPCCAPP_H

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

#define darcall

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

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

/*******************************************************************************
  FUNCTION: SwapEndian
  PURPOSE: Swap the byte order of a structure
  EXAMPLE: int I=0x12345678; SWAP_BYTE(I); I => 0x78563412;
  IMPLEMENTATION: Fold Addr in half, swap header & tail by XOR op
          e.g.: *a = 0x12 [ Addr + 0];
                *b = 0x78 [ Add + 4 - 0 - 1 = Addr+3];
                *a ^= *b;        // sets *a to 0x6A
                *b ^= *a;        // sets *b to 0x12
                *a ^= *b;        // sets *a to 0x78

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

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

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


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

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

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

#endif // __TPCCAPP_H
```

include/tpccdbg.h

```
/*****************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****************************************************************************/

/*
 * tpccdbg.h - Debugging Macros
 *
 */

#ifndef __TPCCDBG_H
#define __TPCCDBG_H

#ifdef __cplusplus
extern "C" {
#endif

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

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

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

#ifdef __cplusplus
}
#endif
#endif // __TPCCDBG_H
```

tpccenv.sh

```
##############################################################################
## Licensed Materials - Property of IBM
##
## Governed under the terms of the International
## License Agreement for Non-Warranted Sample Code.
##
## (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
## All Rights Reserved.
##
## US Government Users Restricted Rights - Use, duplication or
## disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
##############################################################################

#
# tpccenv.sh - UNIX Environment Setup
#

# The Kit Version
export TPCC_VERSION=CK080131

# The DB2 Instance Name (for DB2)
export DB2INSTANCE=${USER}

# The OS being used (i.e. "UNIX", "LINUX", "WINDOWS")
export PLATFORM=UNIX
export SERVER_PLATFORM=UNIX

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

# Specifies whether or not to use dari stored proc's for the TPC-C driver.  Set to either DARIVERSION or
NONDARI;
#export TPCC_SPTYPE=NOSP
#export TPCC_SPTYPE=SPGENERAL2
export TPCC_SPTYPE=SPGENERAL
#export TPCC_SPTYPE=DARI2SQLDA

export DB2VERSION=v9

# The schema name is typically the SQL autorization ID (or username).
# This is required for runstats and EEE.
export TPCC_SCHEMA=${USER}
export SERVER_TPCC_SCHEMA=${USER}

# DB2 EE/EEE Configuration
export DB2EDITION=EE
#export DB2EDITION=EEE
export DB2NODE=0
export DB2NODES=1;              # set to the number of nodes you have.  Set to 1 for EE.

# TPCC General Configuration
export TPCC_DBNAME=TPCC
export TPCC_ROOT=${HOME}/tpc-c.ibm
export TPCC_SQLLIB=${HOME}/sqllib
export TPCC_RUNDATA=${HOME}/tpccdata

# TPCC Debug Configuration
# This is the path where all error and debug logs are placed.
# To get debugging from within the stored procedures, you must
# set DB2ENVLIST="TPCC_DEBUGDIR" in tpcc.config.
export TPCC_DEBUGDIR=/tmp

# Specifies where stored procedures should be placed and if they should
# be fenced.
export TPCC_SPDIR=${TPCC_SQLLIB}/function
export TPCC_FENCED=NO
```

Appendix - B: Tunable Parameters

B.1 Database Parameters.

db.cfg.out

```
Database Configuration for Database TPCC

Database configuration release level                = 0x0c00
Database release level                              = 0x0c00

Database territory                                  = US
Database code page                                  = 819
Database code set                                   = ISO8859-1
Database country/region code                        = 1
Database collating sequence                         = IDENTITY
Alternate collating sequence            (ALT_COLLATE) =
Number compatibility                                = OFF
Varchar2 compatibility                              = OFF
Database page size                                  = 4096

Dynamic SQL Query management          (DYN_QUERY_MGMT) = DISABLE

Discovery support for this database      (DISCOVER_DB) = ENABLE

Restrict access                                     = NO
Default query optimization class        (DFT_QUERYOPT) = 5
Degree of parallelism                   (DFT_DEGREE) = 1
Continue upon arithmetic exceptions   (DFT_SQLMATHWARN) = NO
Default refresh age                  (DFT_REFRESH_AGE) = 0
Default maintained table types for opt (DFT_MTTB_TYPES) = SYSTEM
Number of frequent values retained     (NUM_FREQVALUES) = 10
Number of quantiles retained           (NUM_QUANTILES) = 20

Decimal floating point rounding mode  (DECFLT_ROUNDING) = ROUND_HALF_EVEN

Backup pending                                      = NO

Database is consistent                              = YES
Rollforward pending                                 = NO
Restore pending                                     = NO

Multi-page file allocation enabled                  = YES

Log retain for recovery status                      = RECOVERY
User exit for logging status                        = NO

Self tuning memory                  (SELF_TUNING_MEM) = OFF
Size of database shared memory (4KB)  (DATABASE_MEMORY) = 64318072
Database memory threshold             (DB_MEM_THRESH) = 10
Max storage for lock list (4KB)            (LOCKLIST) = 13000
Percent. of lock lists per application      (MAXLOCKS) = 20
Package cache size (4KB)                (PCKCACHESZ) = 12000
Sort heap thres for shared sorts (4KB) (SHEAPTHRES_SHR) = 5000
Sort list heap (4KB)                       (SORTHEAP) = 16

Database heap (4KB)                         (DBHEAP) = 524288
Catalog cache size (4KB)            (CATALOGCACHE_SZ) = (MAXAPPLS*4)
Log buffer size (4KB)                      (LOGBUFSZ) = 60000
Utilities heap size (4KB)               (UTIL_HEAP_SZ) = 5000
Buffer pool size (pages)                   (BUFFPAGE) = 1000
SQL statement heap (4KB)                   (STMTHEAP) = 65000
Default application heap (4KB)           (APPLHEAPSZ) = 2500
Application Memory Size (4KB)           (APPL_MEMORY) = AUTOMATIC
Statistics heap size (4KB)             (STAT_HEAP_SZ) = AUTOMATIC

Interval for checking deadlock (ms)        (DLCHKTIME) = 3000
Lock timeout (sec)                      (LOCKTIMEOUT) = -1

Changed pages threshold                (CHNGPGS_THRESH) = 99
Number of asynchronous page cleaners   (NUM_IOCLEANERS) = 1
Number of I/O servers                    (NUM_IOSERVERS) = 1
Index sort flag                          (INDEXSORT) = YES
Sequential detect flag                   (SEQDETECT) = NO
Default prefetch size (pages)        (DFT_PREFETCH_SZ) = AUTOMATIC

Track modified pages                      (TRACKMOD) = OFF

Default number of containers                        = 1
Default tablespace extentsize (pages)   (DFT_EXTENT_SZ) = 32

Max number of active applications          (MAXAPPLS) = 5050
Average number of active applications       (AVG_APPLS) = 1
Max DB files open per application          (MAXFILOP) = 61440

Log file size (4KB)                      (LOGFILSIZ) = 262144
Number of primary log files              (LOGPRIMARY) = 250
Number of secondary log files            (LOGSECOND) = 0
Changed path to log files                (NEWLOGPATH) =
Path to log files                                   = /dev/rdbloglv
Overflow log path                   (OVERFLOWLOGPATH) =
Mirror log path                     (MIRRORLOGPATH) =
First active log file                               = S0000001.LOG
Block log on disk full              (BLK_LOG_DSK_FUL) = NO
Percent max primary log space by transaction  (MAX_LOG) = 0
Num. of active log files for 1 active UOW(NUM_LOG_SPAN) = 0

Group commit count                        (MINCOMMIT) = 1
Percent log file reclaimed before soft chckpt (SOFTMAX) = 4128
Log retain for recovery enabled           (LOGRETAIN) = RECOVERY
User exit for logging enabled             (USEREXIT) = OFF

HADR database role                                  = STANDARD
HADR local host name                (HADR_LOCAL_HOST) =
HADR local service name              (HADR_LOCAL_SVC) =
HADR remote host name               (HADR_REMOTE_HOST) =
HADR remote service name             (HADR_REMOTE_SVC) =
HADR instance name of remote server  (HADR_REMOTE_INST) =
HADR timeout value                    (HADR_TIMEOUT) = 120
HADR log write synchronization mode     (HADR_SYNCMODE) = NEARSYNC
HADR peer window duration (seconds)  (HADR_PEER_WINDOW) = 0

First log archive method              (LOGARCHMETH1) = LOGRETAIN
Options for logarchmeth1                (LOGARCHOPT1) =
Second log archive method             (LOGARCHMETH2) = OFF
Options for logarchmeth2                (LOGARCHOPT2) =
Failover log archive path               (FAILARCHPATH) =
Number of log archive retries on error   (NUMARCHRETRY) = 5
Log archive retry Delay (secs)         (ARCHRETRYDELAY) = 20
Vendor options                          (VENDOROPT) =

Auto restart enabled                    (AUTORESTART) = ON
Index re-creation time and redo index build  (INDEXREC) = SYSTEM (RESTART)
Log pages during index build            (LOGINDEXBUILD) = OFF
Default number of loadrec sessions    (DFT_LOADREC_SES) = 1
Number of database backups to retain   (NUM_DB_BACKUPS) = 12
Recovery history retention (days)     (REC_HIS_RETENTN) = 366
Auto deletion of recovery objects    (AUTO_DEL_REC_OBJ) = OFF

TSM management class                   (TSM_MGMTCLASS) =
TSM node name                          (TSM_NODENAME) =
TSM owner                              (TSM_OWNER) =
TSM password                           (TSM_PASSWORD) =

Automatic maintenance                   (AUTO_MAINT) = OFF
  Automatic database backup            (AUTO_DB_BACKUP) = OFF
  Automatic table maintenance          (AUTO_TBL_MAINT) = OFF
  Automatic runstats                   (AUTO_RUNSTATS) = OFF
    Automatic statement statistics  (AUTO_STMT_STATS) = OFF
  Automatic statistics profiling     (AUTO_STATS_PROF) = OFF
    Automatic profile updates         (AUTO_PROF_UPD) = OFF
  Automatic reorganization             (AUTO_REORG) = OFF

Enable XML Character operations        (ENABLE_XMLCHAR) = YES
WLM Collection Interval (minutes)     (WLM_COLLECT_INT) = 0
```

db2set.cfg.out

```
[i] DB2_KEEP_AS_AND_DMS_CONTAINERS_OPEN=ON
[i] DB2_LARGE_PAGE_MEM=DB:16GB
[i] DB2_RESOURCE_POLICY=/home/tpcc/tpc-c.ibm/cfg/affinity.cfg
[i] DB2_SELUDI_COMM_BUFFER=Y
[i] DB2_USE_ALTERNATE_PAGE_CLEANING=YES
[i] DB2_MAX_NON_TABLE_LOCKS=1000
[i] DB2_TRUSTED_BINDIN=ON
[i] DB2_KEEPTABLELOCK=CONNECTION
[i] DB2_NUM_CKPW_DAEMONS=0
[i] DB2_NO_FORK_CHECK=ON
[i] DB2_ALLOCATION_SIZE=16777216
[i] DB2_APM_PERFORMANCE=ALL
[i] DB2_PINNED_BP=YES
[i] DB2_SELECTIVITY=ON
[i] DB2ASSUMEUPDATE=ON
[i] DB2CHECKCLIENTINTERVAL=0
[i] DB2_HASH_JOIN=OFF
[i] DB2CHKSQLDA=OFF
[i] DB2ENVLIST=MEMORY_AFFINITY LDR_CNTRL
[i] DB2_UOW_LOGSPACE_CACHE=18432
[i] DB2_COLLECT_TS_REC_INFO=false
[i] DB2COMM=tcpip
[i] DB2CHKPTR=OFF
```

dbm.cfg.out

```
        Database Manager Configuration

    Node type = Database Server with local clients

Database manager configuration release level          = 0x0c00

CPU speed (millisec/instruction)          (CPUSPEED) = 2.834065e-07

Max number of concurrently active databases     (NUMDB) = 1
Federated Database System Support          (FEDERATED) = NO
Transaction processor monitor name       (TP_MON_NAME) =

Default charge-back account          (DFT_ACCOUNT_STR) =

Java Development Kit installation path      (JDK_PATH) = /home/tpcc/sqllib/java/jdk64

Diagnostic error capture level            (DIAGLEVEL) = 1
Notify Level                            (NOTIFYLEVEL) = 1
Diagnostic data directory path            (DIAGPATH) =

Default database monitor switches
  Buffer pool                      (DFT_MON_BUFPOOL) = OFF
  Lock                               (DFT_MON_LOCK) = OFF
  Sort                              (DFT_MON_SORT) = OFF
  Statement                          (DFT_MON_STMT) = OFF
  Table                             (DFT_MON_TABLE) = OFF
  Timestamp                       (DFT_MON_TIMESTAMP) = OFF
  Unit of work                        (DFT_MON_UOW) = OFF
Monitor health of instance and databases   (HEALTH_MON) = OFF
```

```
SYSADM group name                    (SYSADM_GROUP) = STAFF
SYSCTRL group name                   (SYSCTRL_GROUP) =
SYSMAINT group name                  (SYSMAINT_GROUP) =
SYSMON group name                    (SYSMON_GROUP) =

Client Userid-Password Plugin        (CLNT_PW_PLUGIN) =
Client Kerberos Plugin               (CLNT_KRB_PLUGIN) =
Group Plugin                         (GROUP_PLUGIN) =
GSS Plugin for Local Authorization   (LOCAL_GSSPLUGIN) =
Server Plugin Mode                   (SRV_PLUGIN_MODE) = UNFENCED
Server List of GSS Plugins           (SRVCON_GSSPLUGIN_LIST) =
Server Userid-Password Plugin        (SRVCON_PW_PLUGIN) =
Server Connection Authentication     (SRVCON_AUTH) = NOT_SPECIFIED
Cluster manager                      (CLUSTER_MGR) =

Database manager authentication      (AUTHENTICATION) = CLIENT
Cataloging allowed without authority (CATALOG_NOAUTH) = YES
Trust all clients                    (TRUST_ALLCLNTS) = YES
Trusted client authentication        (TRUST_CLNTAUTH) = CLIENT
Bypass federated authentication      (FED_NOAUTH) = NO

Default database path                (DFTDBPATH) = /home/tpcc

Database monitor heap size (4KB)     (MON_HEAP_SZ) = 4096
Java Virtual Machine heap size (4KB) (JAVA_HEAP_SZ) = 2048
Audit buffer size (4KB)              (AUDIT_BUF_SZ) = 0
Size of instance shared memory (4KB) (INSTANCE_MEMORY) = 65068072
Backup buffer default size (4KB)     (BACKBUFSZ) = 1024
Restore buffer default size (4KB)    (RESTBUFSZ) = 1024

Agent stack size                     (AGENT_STACK_SZ) = 1024
Sort heap threshold (4KB)            (SHEAPTHRES) = 0

Directory cache support              (DIR_CACHE) = YES

Application support layer heap size (4KB) (ASLHEAPSZ) = 15
Max requester I/O block size (bytes)      (RQRIOBLK) = 4096
Query heap size (4KB)                     (QUERY_HEAP_SZ) = 1000

Workload impact by throttled utilities(UTIL_IMPACT_LIM) = 10

Priority of agents                   (AGENTPRI) = 60
Agent pool size                      (NUM_POOLAGENTS) = 0
Initial number of agents in pool     (NUM_INITAGENTS) = 0
Max number of coordinating agents    (MAX_COORDAGENTS) = AUTOMATIC
Max number of client connections     (MAX_CONNECTIONS) = AUTOMATIC

Keep fenced process                  (KEEPFENCED) = YES
Number of pooled fenced processes    (FENCED_POOL) = MAX_COORDAGENTS
Initial number of fenced processes   (NUM_INITFENCED) = 0

Index re-creation time and redo index build (INDEXREC) = RESTART

Transaction manager database name    (TM_DATABASE) = 1ST_CONN
Transaction resync interval (sec)    (RESYNC_INTERVAL) = 180

SPM name                             (SPM_NAME) =
SPM log size                         (SPM_LOG_FILE_SZ) = 256
SPM resync agent limit               (SPM_MAX_RESYNC) = 20
SPM log path                         (SPM_LOG_PATH) =

TCP/IP Service name                  (SVCENAME) =
Discovery mode                       (DISCOVER) = SEARCH
Discover server instance             (DISCOVER_INST) = ENABLE

Maximum query degree of parallelism  (MAX_QUERYDEGREE) = ANY
Enable intra-partition parallelism   (INTRA_PARALLEL) = NO

No. of int. communication buffers(4KB)(FCM_NUM_BUFFERS) = AUTOMATIC
No. of int. communication channels   (FCM_NUM_CHANNELS) = AUTOMATIC
db2start/db2stop timeout (min)       (START_STOP_TIME) = 10
```

affinity.cfg

```
<RESOURCE_POLICY>
<DATABASE_RESOURCE_POLICY>
<DBNAME>TPCC</DBNAME>
<METHOD>RSET</METHOD>
<RESOURCE_BINDING>
 <RESOURCE>sys/node.03.00000</RESOURCE>
 <DBMEM_PERCENTAGE>25</DBMEM_PERCENTAGE>
  <SERVICE_NAME>xtpcc0</SERVICE_NAME>
  <BUFFERPOOL_BINDING>
   <NUM_CLEANERS>2</NUM_CLEANERS>
   <BUFFERPOOL_ID>13</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>17</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>21</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>25</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>29</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>33</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>37</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>4</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>41</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>8</BUFFERPOOL_ID>
  </BUFFERPOOL_BINDING>
</RESOURCE_BINDING>
<RESOURCE_BINDING>
 <RESOURCE>sys/node.03.00001</RESOURCE>
 <DBMEM_PERCENTAGE>25</DBMEM_PERCENTAGE>
  <SERVICE_NAME>xtpcc1</SERVICE_NAME>
  <BUFFERPOOL_BINDING>
   <NUM_CLEANERS>2</NUM_CLEANERS>
   <BUFFERPOOL_ID>14</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>18</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>22</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>26</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>30</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>34</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>38</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>42</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>5</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>9</BUFFERPOOL_ID>
  </BUFFERPOOL_BINDING>
</RESOURCE_BINDING>
<RESOURCE_BINDING>
 <RESOURCE>sys/node.03.00002</RESOURCE>
 <DBMEM_PERCENTAGE>25</DBMEM_PERCENTAGE>
  <SERVICE_NAME>xtpcc2</SERVICE_NAME>
  <BUFFERPOOL_BINDING>
   <NUM_CLEANERS>2</NUM_CLEANERS>
   <BUFFERPOOL_ID>10</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>15</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>19</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>23</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>27</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>31</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>35</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>39</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>43</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>6</BUFFERPOOL_ID>
  </BUFFERPOOL_BINDING>
</RESOURCE_BINDING>
<RESOURCE_BINDING>
 <RESOURCE>sys/node.03.00003</RESOURCE>
 <DBMEM_PERCENTAGE>25</DBMEM_PERCENTAGE>
  <SERVICE_NAME>xtpcc3</SERVICE_NAME>
  <BUFFERPOOL_BINDING>
   <NUM_CLEANERS>2</NUM_CLEANERS>
   <BUFFERPOOL_ID>11</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>16</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>20</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>24</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>28</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>32</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>36</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>40</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>44</BUFFERPOOL_ID>
   <BUFFERPOOL_ID>7</BUFFERPOOL_ID>
  </BUFFERPOOL_BINDING>
</RESOURCE_BINDING>
</DATABASE_RESOURCE_POLICY>
</RESOURCE_POLICY>
```

B.2 Transaction Monitor Parameters

tpccCom.tpcc_com_settings.txt

```
Transactions not supported
Enable object pooling
Minimum pool size 50
Maximum pool size 50
Creation timeout 1900000000
Enable Object Construction
Enable Just in time activation
Concurrency Required
```

InetInfo_registry.reg

```
Windows Registry Editor Version 5.00

[[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Parameters]
"ListenBackLog"=dword:00000040
"DispatchEntries"=hex(7):00,00,13,04,9d,3d,83,7c,40,9d,88,7c,cc,8b,13,04,00,00,\
00,00
"PoolThreadLimit"=dword:00000320
"MaxPoolThreads"=dword:000000c8
"MaxConcurrency"=dword:00000320
```

tcpip_parameters_registry.reg

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters]
"NV Hostname"="client1"
"DataBasePath"=hex(2):25,00,53,00,79,00,73,00,74,00,65,00,6d,00,52,00,6f,00,6f,\
  00,74,00,25,00,5c,00,53,00,79,00,73,00,74,00,65,00,6d,00,33,00,32,00,5c,00,\
  64,00,72,00,69,00,76,00,65,00,72,00,73,00,5c,00,65,00,74,00,63,00,00,00
"NameServer"=""
"ForwardBroadcasts"=dword:00000000
"IPEnableRouter"=dword:00000000
"Domain"=""
"Hostname"="client1"
"SearchList"=""
"UseDomainNameDevolution"=dword:00000001
"EnableICMPRedirect"=dword:00000001
"DeadGWDetectDefault"=dword:00000001
"DontAddDefaultGatewayDefault"=dword:00000000
"EnableSecurityFilters"=dword:00000000
"SynAttackProtect"=dword:00000000


[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\Interfaces\{409E0751-
A13A-4346-BE16-1574316AD3D5}]
"UseZeroBroadcast"=dword:00000000
"EnableDeadGWDetect"=dword:00000001
"EnableDHCP"=dword:00000000
"IPAddress"=hex(7):31,00,30,00,2e,00,31,00,2e,00,31,00,2e,00,32,00,00,00,00,00
```

```
"SubnetMask"=hex(7):32,00,35,00,35,00,2e,00,32,00,35,00,35,00,2e,00,32,00,35,\
  00,35,00,2e,00,30,00,00,00,00,00
"DefaultGateway"=hex(7):00,00
"DefaultGatewayMetric"=hex(7):00,00
"NameServer"=""
"Domain"=""
"RegistrationEnabled"=dword:00000001
"RegisterAdapterName"=dword:00000000
"TCPAllowedPorts"=hex(7):30,00,00,00,00,00
"UDPAllowedPorts"=hex(7):30,00,00,00,00,00
"RawIPAllowedProtocols"=hex(7):30,00,00,00,00,00
"NTEContextList"=hex(7):30,00,78,00,30,00,30,00,30,00,30,00,30,00,30,00,30,00,\
  34,00,00,00,00,00
"DhcpClassIdBin"=hex:
"DhcpServer"="255.255.255.255"
"Lease"=dword:00000e10
"LeaseObtainedTime"=dword:478640d6
"T1"=dword:478647de
"T2"=dword:47864d24
"LeaseTerminatesTime"=dword:47864ee6
"IPAutoconfigurationAddress"="0.0.0.0"
"IPAutoconfigurationMask"="255.255.0.0"
"IPAutoconfigurationSeed"=dword:00000000
"AddressType"=dword:00000000
"TcpWindowSize"=dword:00040000

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\Interfaces\{4C1939E5
-155B-4B74-82C7-50F2F13B7ED1}]
"UseZeroBroadcast"=dword:00000000
"EnableDeadGWDetect"=dword:00000001
"EnableDHCP"=dword:00000000
"IPAddress"=hex(7):31,00,39,00,32,00,2e,00,31,00,36,00,38,00,2e,00,31,00,31,00,\
  2e,00,31,00,30,00,31,00,00,00,00,00
"SubnetMask"=hex(7):32,00,35,00,35,00,2e,00,32,00,35,00,35,00,2e,00,32,00,35,\
  00,35,00,2e,00,30,00,00,00,00,00
"DefaultGateway"=hex(7):00,00
"DefaultGatewayMetric"=hex(7):00,00
"NameServer"=""
"Domain"=""
"RegistrationEnabled"=dword:00000001
"RegisterAdapterName"=dword:00000000
"TCPAllowedPorts"=hex(7):30,00,00,00,00,00
"UDPAllowedPorts"=hex(7):30,00,00,00,00,00
"RawIPAllowedProtocols"=hex(7):30,00,00,00,00,00
"NTEContextList"=hex(7):30,00,78,00,30,00,30,00,30,00,30,00,30,00,30,00,30,00,\
  33,00,00,00,00,00
"DhcpClassIdBin"=hex:
"DhcpServer"="255.255.255.255"
"Lease"=dword:00000e10
"LeaseObtainedTime"=dword:47856b77
"T1"=dword:4785727f
"T2"=dword:478577c5
"LeaseTerminatesTime"=dword:47857987
"IPAutoconfigurationAddress"="0.0.0.0"
"IPAutoconfigurationMask"="255.255.0.0"
"IPAutoconfigurationSeed"=dword:00000000
"AddressType"=dword:00000000
"TcpWindowSize"=dword:00040000

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\Interfaces\{7EE74D8F
-A361-4B69-96A4-831BF20B6148}]
"UseZeroBroadcast"=dword:00000000
"EnableDeadGWDetect"=dword:00000001
"EnableDHCP"=dword:00000000
"IPAddress"=hex(7):31,00,30,00,2e,00,31,00,2e,00,32,00,2e,00,32,00,00,00,00,00
"SubnetMask"=hex(7):32,00,35,00,35,00,2e,00,32,00,35,00,35,00,2e,00,32,00,35,\
  00,35,00,2e,00,30,00,00,00,00,00
"DefaultGateway"=hex(7):00,00
"DefaultGatewayMetric"=hex(7):00,00
"NameServer"=""
"Domain"=""
"RegistrationEnabled"=dword:00000001
"RegisterAdapterName"=dword:00000000
"TCPAllowedPorts"=hex(7):30,00,00,00,00,00
"UDPAllowedPorts"=hex(7):30,00,00,00,00,00
"RawIPAllowedProtocols"=hex(7):30,00,00,00,00,00
"NTEContextList"=hex(7):30,00,78,00,30,00,30,00,30,00,30,00,30,00,30,00,30,00,\
  35,00,00,00,00,00
"DhcpClassIdBin"=hex:
"DhcpServer"="255.255.255.255"
"Lease"=dword:00000e10
"LeaseObtainedTime"=dword:47864068
"T1"=dword:47864770
"T2"=dword:47864cb6
"LeaseTerminatesTime"=dword:47864e78
"IPAutoconfigurationAddress"="0.0.0.0"
"IPAutoconfigurationMask"="255.255.0.0"
"IPAutoconfigurationSeed"=dword:00000000
"AddressType"=dword:00000000
"TcpWindowSize"=dword:00040000
```

Tpcc_software_registry.reg

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SOFTWARE\TPCC]
"dlvyLogPath"="c:\\InetPub\\wwwroot\\tpcc\\dlvy"
"dlvyQueueLen"=dword:00004e20
"dlvyThreads"=dword:00000004
"nullDB"=dword:00000000
"htmlTrace"=dword:00000000
"dbName"="tpcc"
"errorLogFile"="c:\\inetpub\\wwwroot\\tpcc\\isapi_err.log"
"htmlTraceLogFile"="c:\\inetpub\\wwwroot\\tpcc\\isapi.log"
"numUsers"=dword:00007530
"dbType"="DB2"
"dbUserName"="tpcc"
"dbPassword"="tpcc"
"isapi_trace"=dword:00000000
"numServers"=dword:00000001
"numWarehouse"=dword:000009c4
"numPools"=dword:00000001
"dbInterfacePath"="C:\\inetpub\\wwwroot\\tpcc\\tpccDB2glue.dll"
```

HTTP registry.reg

```
Windows Registry Editor Version 5.00

[[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\HTTP\Parameters]
"UriEnableCache"=dword:00000001
"UriScavangerPeriod"=dword:00002a30
"MaxConnections"=dword:000186a0
"DisableServerHeader"=dword:00000001
"MaxBufferedSendBytes"=dword:00010000
```

B.3 AIX Parameters

Bull Escala PL860R p 550 Express

```
lattr –El sys0
SW_dist_intr   false        Enable SW distribution of interrupts          True
autorestart    true         Automatically REBOOT OS after a crash         True
boottype       disk         N/A                                           False
capacity_inc   1.00         Processor capacity increment                  False
capped         true         Partition is capped                           False
conslogin      enable       System Console Login                          False
cpuguard       enable       CPU Guard                                     True
dedicated      true         Partition is dedicated                        False
ent_capacity   8.00         Entitled processor capacity                   False
frequency      1593000000   System Bus Frequency                          False
fullcore       false        Enable full CORE dump                         True
fwversion      IBM,FL320_024  Firmware version and revision levels        False
id_to_partition 0X0401809099FA1702 Partition ID                           False
id_to_system   0X0401809099FA1700 System ID                               False
iostat         false        Continuously maintain DISK I/O history        True
keylock        normal       State of system keylock at boot time          False
log_pg_dealloc true         Log predictive memory page deallocation events True
max_capacity   8.00         Maximum potential processor capacity          False
max_logname    9            Maximum login name length at boot time        True
maxbuf         20           Maximum number of pages in block I/O BUFFER CACHE True
maxmbuf        0            Maximum Kbytes of real memory allowed for MBUFS True
maxpout        0            HIGH water mark for pending write I/Os per file True
maxuproc       20000        Maximum number of PROCESSES allowed per user  True
min_capacity   1.00         Minimum potential processor capacity          False
minpout        0            LOW water mark for pending write I/Os per file True
modelname      IBM,8204-E8A Machine name                                  False
ncargs         6            ARG/ENV list size in 4K byte blocks           True
nfs4_acl_compat secure      NFS4 ACL Compatibility Mode                   True
pre430core     false        Use pre-430 style CORE dump                   True
pre520tune     disable      Pre-520 tuning compatibility mode             True
realmem        267124736    Amount of usable physical memory in Kbytes    False
rtasversion    1            Open Firmware RTAS version                    False
sed_config     select       Stack Execution Disable (SED) Mode            True
systemid       IBM,031060AAF Hardware system identifier                   False
variable_weight 0           Variable processor capacity weight            False

vmo –L

NAME                CUR   DEF   BOOT  MIN   MAX   UNIT        TYPE
   DEPENDENCIES
---------------------------------------------------------------------------
cpu_scale_memp      8     8     8     1     64                B
---------------------------------------------------------------------------
data_stagger_interval  161   161   161   0     4K-1  4KB pages   D
   lgpg_regions
---------------------------------------------------------------------------
defps               1     1     1     0     1     boolean     D
---------------------------------------------------------------------------
force_relalias_lite 0     0     0     0     1     boolean     D
---------------------------------------------------------------------------
framesets           2     2     2     1     10                B
---------------------------------------------------------------------------
htabscale           n/a   -1    -1    -4    0                 B
---------------------------------------------------------------------------
kernel_heap_psize   16M   4K    16M   4K    16M   bytes       B
---------------------------------------------------------------------------
kernel_psize        16M   0     0     0     16M   bytes       B
---------------------------------------------------------------------------
large_page_heap_size  0   0     0     0     8E-1  bytes       B
   lgpg_regions
---------------------------------------------------------------------------
lgpg_regions        3473  0     3473  0                       D
   lgpg_size
---------------------------------------------------------------------------
lgpg_size           16M   0     16M   0     16M   bytes       D
   lgpg_regions
---------------------------------------------------------------------------
low_ps_handling     1     1     1     1     2                 D
---------------------------------------------------------------------------
lru_file_repage     1     1     1     0     1     boolean     D
---------------------------------------------------------------------------
lru_poll_interval   10    10    10    0     60000 milliseconds D
---------------------------------------------------------------------------
lrubucket           128K  128K  128K  64K   4M    4KB pages   D
---------------------------------------------------------------------------
maxclient%          80    80    80    1     100   % memory    D
   maxperm%
   minperm%
---------------------------------------------------------------------------
maxfree             1088  1088  1088  8     200K  4KB pages   D
```

```
    minfree
    memory_frames
-------------------------------------------------------------------------------
maxperm              1331K      1331K                        S
-------------------------------------------------------------------------------
maxperm%             80   80   80   1    100   % memory       D
    minperm%
    maxclient%
-------------------------------------------------------------------------------
maxpin               1822K      1822K                        S
-------------------------------------------------------------------------------
maxpin%              80   80   80   1    99    % memory       D
    pinnable_frames
    memory_frames
-------------------------------------------------------------------------------
mbuf_heap_psize      16M  0    16M  0    16M   bytes          B
-------------------------------------------------------------------------------
memory_affinity      1    1    1    0    1     boolean        B
-------------------------------------------------------------------------------
memory_frames        65216K     65216K          4KB pages      S
-------------------------------------------------------------------------------
memplace_data        2    2    2    1    2                    D
    memory_affinity
-------------------------------------------------------------------------------
memplace_mapped_file 2    2    2    1    2                    D
    memory_affinity
-------------------------------------------------------------------------------
memplace_shm_anonymous 2  2    2    1    2                    D
    memory_affinity
-------------------------------------------------------------------------------
memplace_shm_named   2    2    2    1    2                    D
    memory_affinity
-------------------------------------------------------------------------------
memplace_stack       2    2    2    1    2                    D
    memory_affinity
-------------------------------------------------------------------------------
memplace_text        2    2    2    1    2                    D
    memory_affinity
-------------------------------------------------------------------------------
memplace_unmapped_file 2  2    2    1    2                    D
    memory_affinity
-------------------------------------------------------------------------------
mempools             4          4                            d
    cpu_scale_memp
-------------------------------------------------------------------------------
minfree              960  960  960  8    200K  4KB pages      D
    maxfree
    memory_frames
-------------------------------------------------------------------------------
minperm              340820     340820                       S
-------------------------------------------------------------------------------
minperm%             20   20   20   1    100   % memory       D
    maxperm%
    maxclient%
-------------------------------------------------------------------------------
nokilluid            0    0    0    0    4G-1  uid            D
-------------------------------------------------------------------------------
npskill              9K   9K   9K   1    1M-1  4KB pages      D
-------------------------------------------------------------------------------
npsrpgmax            72K  72K  72K  0    1M-1  4KB pages      D
    npsrpgmin
-------------------------------------------------------------------------------
npsrpgmin            54K  54K  54K  0    1M-1  4KB pages      D
    npsrpgmax
-------------------------------------------------------------------------------
npsscrubmax          72K  72K  72K  0    1M-1  4KB pages      D
    npsscrubmin
-------------------------------------------------------------------------------
npsscrubmin          54K  54K  54K  0    1M-1  4KB pages      D
    npsscrubmax
-------------------------------------------------------------------------------
npswarn              36K  36K  36K  0    1M-1  4KB pages      D
-------------------------------------------------------------------------------
num_spec_dataseg     0    0    0    0                        B
-------------------------------------------------------------------------------
numpsblks            1152K      1152K           4KB blocks     S
-------------------------------------------------------------------------------
page_steal_method    0    0    0    0    1     boolean        B
-------------------------------------------------------------------------------
pagecoloring         n/a  0    0    0    1     boolean        B
-------------------------------------------------------------------------------
pinnable_frames      1624K      1624K           4KB pages      S
-------------------------------------------------------------------------------
pta_balance_threshold n/a 1    1    0    99    % pta segment   D
-------------------------------------------------------------------------------
relalias_percentage  0    0    0    0    32K-1                D
-------------------------------------------------------------------------------
rpgclean             0    0    0    0    1     boolean        D
-------------------------------------------------------------------------------
rpgcontrol           2    2    2    0    3                    D
-------------------------------------------------------------------------------
scrub                0    0    0    0    1     boolean        D
-------------------------------------------------------------------------------
scrubclean           0    0    0    0    1     boolean        D
-------------------------------------------------------------------------------
soft_min_lgpgs_vmpool 0   0    0    0    90    %              D
    lgpg_regions
-------------------------------------------------------------------------------
spec_dataseg_int     512  512  512  0                        B
-------------------------------------------------------------------------------
strict_maxclient     1    1    1    0    1     boolean        D
    strict_maxperm
-------------------------------------------------------------------------------
strict_maxperm       0    0    0    0    1     boolean        D
    strict_maxclient
-------------------------------------------------------------------------------
v_pinshm             1    0    1    0    1     boolean        D
-------------------------------------------------------------------------------
vm_modlist_threshold -1   -1   -1   -2   2G-1                 D
-------------------------------------------------------------------------------
vmm_fork_policy      1    1    1    0    1     boolean        D
-------------------------------------------------------------------------------
vmm_mpsize_support   1    1    1    0    1     boolean        B
-------------------------------------------------------------------------------

n/a means parameter not supported by the current platform or kernel

Parameter types:
    S = Static: cannot be changed
    D = Dynamic: can be freely changed
    B = Bosboot: can only be changed using bosboot and reboot
    R = Reboot: can only be changed during reboot
    C = Connect: changes are only effective for future socket connections
    M = Mount: changes are only effective for future mountings
    I = Incremental: can only be incremented
    d = deprecated: deprecated and cannot be changed

Value conventions:
    K = Kilo: 2^10      G = Giga: 2^30      P = Peta: 2^50
    M = Mega: 2^20      T = Tera: 2^40      E = Exa: 2^60
2^60

schedo –L

NAME                 CUR  DEF  BOOT MIN  MAX  UNIT       TYPE
    DEPENDENCIES
-------------------------------------------------------------------------------
%usDelta             100  100  100  0    100                D
-------------------------------------------------------------------------------
affinity_lim         7    7    7    0    100  dispatches     D
-------------------------------------------------------------------------------
allowMCMmigrate      0    0    0    0    1    boolean        D
-------------------------------------------------------------------------------
big_tick_size        1    1    1    1    100  10 ms          D
-------------------------------------------------------------------------------
ded_cpu_donate_thresh 80  80   80   0    100  % busy         D
-------------------------------------------------------------------------------
fast_locks           1    0    1    0    1    boolean        B
    krlock_enable
-------------------------------------------------------------------------------
fixed_pri_global     0    0    0    0    1    boolean        D
-------------------------------------------------------------------------------
force_grq            0    0    0    0    1    boolean        D
-------------------------------------------------------------------------------
hotlocks_enable      1    0    1    0    1    boolean        D
-------------------------------------------------------------------------------
idle_migration_barrier 4  4    4    0    100  sixteenth      D
-------------------------------------------------------------------------------
krlock_confer2self   1    1    1    0    1    boolean        D
-------------------------------------------------------------------------------
krlock_conferb4alloc 1    1    1    0    1    boolean        D
-------------------------------------------------------------------------------
krlock_enable        0    1    0    0    1    boolean        D
    fast_locks
-------------------------------------------------------------------------------
krlock_spinb4alloc   1    1    1    1    2G-1                D
-------------------------------------------------------------------------------
krlock_spinb4confer  1K   1K   1K   0    2G-1                D
-------------------------------------------------------------------------------
maxspin              16K  16K  16K  1    4G-1  spins          D
-------------------------------------------------------------------------------
n_idle_loop_vlopri   100  100  100  0    976K                 D
-------------------------------------------------------------------------------
pacefork             10   10   10   10   2G-1  clock ticks    D
-------------------------------------------------------------------------------
sched_D              16   16   16   0    32                   D
-------------------------------------------------------------------------------
sched_R              16   16   16   0    32                   D
-------------------------------------------------------------------------------
search_globalrq_mload 256 256 256 0    4095M                D
-------------------------------------------------------------------------------
search_smtrunq_mload 256 256 256 0    4095M                D
-------------------------------------------------------------------------------
setnewrq_sidle_mload 384 384 384 0    4095M                D
-------------------------------------------------------------------------------
shed_primrunq_mload  64   64   64   0    4095M                D
-------------------------------------------------------------------------------
sidle_S1runq_mload   64   64   64   0    4095M                D
    sidle_S2runq_mload
-------------------------------------------------------------------------------
sidle_S2runq_mload   134  134  134  0    4095M                D
    sidle_S1runq_mload
    sidle_S3runq_mload
-------------------------------------------------------------------------------
sidle_S3runq_mload   134  134  134  0    4095M                D
    sidle_S2runq_mload
    sidle_S4runq_mload
-------------------------------------------------------------------------------
sidle_S4runq_mload   4095M 4095M 4095M 0    4095M                D
    sidle_S3runq_mload
-------------------------------------------------------------------------------
slock_spinb4confer   1K   1K   1K   0    2G-1                D
-------------------------------------------------------------------------------
smt_snooze_delay     -1   0    0    -1   97656K microsecs     D
-------------------------------------------------------------------------------
smtrunq_load_diff    4095M 2   4095M 1    4095M                D
-------------------------------------------------------------------------------
tb_balance_S0        0    0    0    0    2    ticks          D
-------------------------------------------------------------------------------
tb_balance_S1        2    2    2    0    2    ticks          D
-------------------------------------------------------------------------------
tb_threshold         100  100  100  10   1000 ticks          D
-------------------------------------------------------------------------------
timeslice            1    1    1    0    2G-1 clock ticks    D
-------------------------------------------------------------------------------
unboost_inflih       1    1    1    0    1    boolean        D
-------------------------------------------------------------------------------
v_exempt_secs        2    2    2    0    2G-1 seconds        D
-------------------------------------------------------------------------------
```

```
v_min_process          2    2    2    0    2G-1   processes       D
-------------------------------------------------------------------------------
v_repage_hi            0    0    0    0    2G-1                   D
-------------------------------------------------------------------------------
v_repage_proc          4    4    4    0    2G-1                   D
-------------------------------------------------------------------------------
v_sec_wait             1    1    1    0    2G-1   seconds         D
-------------------------------------------------------------------------------
vpm_fold_policy        1    1    1    0    3                      D
-------------------------------------------------------------------------------
vpm_xvcpus             0    0    0   -1    2G-1   processors      D
-------------------------------------------------------------------------------

n/a means parameter not supported by the current platform or kernel

Parameter types:
   S = Static: cannot be changed
   D = Dynamic: can be freely changed
   B = Bosboot: can only be changed using bosboot and reboot
   R = Reboot: can only be changed during reboot
   C = Connect: changes are only effective for future socket connections
   M = Mount: changes are only effective for future mountings
   I = Incremental: can only be incremented
   d = deprecated: deprecated and cannot be changed

Value conventions:
   K = Kilo: 2^10      G = Giga: 2^30      P = Peta: 2^50
   M = Mega: 2^20      T = Tera: 2^40      E = Exa: 2^60


ioo -L
NAME                    CUR    DEF    BOOT   MIN    MAX    UNIT          TYPE
   DEPENDENCIES
-------------------------------------------------------------------------------
j2_atimeUpdateSymlink   0      0      0      0      1      boolean       D
-------------------------------------------------------------------------------
j2_dynamicBufferPreallocation
                        16     16     16     0      256    16K slabs     D
-------------------------------------------------------------------------------
j2_inodeCacheSize       10     400    10     1      1000                 D
-------------------------------------------------------------------------------
j2_maxPageReadAhead     128    128    128    0      64K    4KB pages     D
-------------------------------------------------------------------------------
j2_maxRandomWrite       0      0      0      0      64K    4KB pages     D
-------------------------------------------------------------------------------
j2_maxUsableMaxTransfer 512    512    512    1      4K     pages         M
-------------------------------------------------------------------------------
j2_metadataCacheSize    10     400    10     1      1000                 D
-------------------------------------------------------------------------------
j2_minPageReadAhead     2      2      2      0      64K    4KB pages     D
-------------------------------------------------------------------------------
j2_nBufferPerPagerDevice 512   512    512    512    256K                 M
-------------------------------------------------------------------------------
j2_nPagesPerWriteBehindCluster
                        32     32     32     0      64K                  D
-------------------------------------------------------------------------------
j2_nRandomCluster       0      0      0      0      64K    16KB clusters D
-------------------------------------------------------------------------------
j2_nonFatalCrashesSystem 0     0      0      0      1      boolean       D
-------------------------------------------------------------------------------
j2_syncModifiedMapped   1      1      1      0      1      boolean       D
-------------------------------------------------------------------------------
j2_syncdLogSyncInterval 1      1      1      0      4K     iterations    D
-------------------------------------------------------------------------------
jfs_clread_enabled      0      0      0      0      1      boolean       D
-------------------------------------------------------------------------------
jfs_use_read_lock       1      1      1      0      1      boolean       D
-------------------------------------------------------------------------------
lvm_bufcnt              9      9      9      1      64     128KB/buffer  D
-------------------------------------------------------------------------------
maxpgahead              8      8      8      0      4K     4KB pages     D
   minpgahead
-------------------------------------------------------------------------------
maxrandwrt              0      0      0      0      512K   4KB pages     D
-------------------------------------------------------------------------------
memory_frames           65216K         65216K               4KB pages     S
-------------------------------------------------------------------------------
minpgahead              2      2      2      0      4K     4KB pages     D
   maxpgahead
-------------------------------------------------------------------------------
numclust                1      1      1      0      2G-1   16KB/cluster  D
-------------------------------------------------------------------------------
numfsbufs               196    196    196    1      2G-1                 M
-------------------------------------------------------------------------------
pd_npages               64K    64K    64K    1      512K   4KB pages     D
-------------------------------------------------------------------------------
pgahd_scale_thresh      0      0      0      0      52172K 4KB pages     D
-------------------------------------------------------------------------------
pv_min_pbuf             512    512    512    512    2G-1                 D
-------------------------------------------------------------------------------
sync_release_ilock      0      0      0      0      1      boolean       D
-------------------------------------------------------------------------------

n/a means parameter not supported by the current platform or kernel

Parameter types:
   S = Static: cannot be changed
   D = Dynamic: can be freely changed
   B = Bosboot: can only be changed using bosboot and reboot
   R = Reboot: can only be changed during reboot
   C = Connect: changes are only effective for future socket connections
   M = Mount: changes are only effective for future mountings
   I = Incremental: can only be incremented
   d = deprecated: deprecated and cannot be changed

Value conventions:
   K = Kilo: 2^10      G = Giga: 2^30      P = Peta: 2^50
   M = Mega: 2^20      T = Tera: 2^40      E = Exa: 2^60
```

Appendix - C: Database Setup Code

C.1 Database Creation Scripts

DDL/ALTTBSP_PF_0.ddl

```
connect to TPCC;
alter tablespace is_customer_001 prefetchsize 0;
alter tablespace is_customer_002 prefetchsize 0;
alter tablespace is_customer_003 prefetchsize 0;
alter tablespace is_customer_004 prefetchsize 0;
alter tablespace is_customer_005 prefetchsize 0;
alter tablespace is_customer_006 prefetchsize 0;
alter tablespace is_customer_007 prefetchsize 0;
alter tablespace is_customer_008 prefetchsize 0;
alter tablespace is_customer_009 prefetchsize 0;
alter tablespace is_customer_010 prefetchsize 0;
alter tablespace is_customer_011 prefetchsize 0;
alter tablespace is_customer_012 prefetchsize 0;
alter tablespace is_customer_013 prefetchsize 0;
alter tablespace is_customer_014 prefetchsize 0;
alter tablespace is_customer_015 prefetchsize 0;
alter tablespace is_customer_016 prefetchsize 0;
alter tablespace is_customer_017 prefetchsize 0;
alter tablespace is_customer_018 prefetchsize 0;
alter tablespace is_customer_019 prefetchsize 0;
alter tablespace is_customer_020 prefetchsize 0;
alter tablespace is_customer_021 prefetchsize 0;
alter tablespace is_customer_022 prefetchsize 0;
alter tablespace is_customer_023 prefetchsize 0;
alter tablespace is_customer_024 prefetchsize 0;
alter tablespace is_customer_025 prefetchsize 0;
alter tablespace is_customer_026 prefetchsize 0;
alter tablespace is_customer_027 prefetchsize 0;
alter tablespace is_customer_028 prefetchsize 0;
alter tablespace is_customer_029 prefetchsize 0;
alter tablespace is_customer_030 prefetchsize 0;
alter tablespace is_customer_031 prefetchsize 0;
alter tablespace is_customer_032 prefetchsize 0;
alter tablespace is_customer_033 prefetchsize 0;
alter tablespace is_customer_034 prefetchsize 0;
alter tablespace is_customer_035 prefetchsize 0;
alter tablespace is_customer_036 prefetchsize 0;
alter tablespace is_customer_037 prefetchsize 0;
alter tablespace is_customer_038 prefetchsize 0;
alter tablespace is_customer_039 prefetchsize 0;
alter tablespace is_customer_040 prefetchsize 0;
alter tablespace is_customer_041 prefetchsize 0;
alter tablespace is_customer_042 prefetchsize 0;
alter tablespace is_customer_043 prefetchsize 0;
alter tablespace is_customer_044 prefetchsize 0;
alter tablespace is_customer_045 prefetchsize 0;
alter tablespace is_customer_046 prefetchsize 0;
alter tablespace is_customer_047 prefetchsize 0;
alter tablespace is_customer_048 prefetchsize 0;
alter tablespace is_customer_049 prefetchsize 0;
alter tablespace is_customer_050 prefetchsize 0;
alter tablespace is_customer_051 prefetchsize 0;
alter tablespace is_customer_052 prefetchsize 0;
alter tablespace is_customer_053 prefetchsize 0;
alter tablespace is_customer_054 prefetchsize 0;
alter tablespace is_customer_055 prefetchsize 0;
alter tablespace is_customer_056 prefetchsize 0;
alter tablespace is_customer_057 prefetchsize 0;
alter tablespace is_customer_058 prefetchsize 0;
alter tablespace is_customer_059 prefetchsize 0;
alter tablespace is_customer_060 prefetchsize 0;
alter tablespace is_customer_061 prefetchsize 0;
alter tablespace is_customer_062 prefetchsize 0;
alter tablespace is_customer_063 prefetchsize 0;
alter tablespace is_customer_064 prefetchsize 0;
alter tablespace is_order_001 prefetchsize 0;
alter tablespace is_order_002 prefetchsize 0;
alter tablespace is_order_003 prefetchsize 0;
alter tablespace is_order_004 prefetchsize 0;
alter tablespace is_order_005 prefetchsize 0;
alter tablespace is_order_006 prefetchsize 0;
alter tablespace is_order_007 prefetchsize 0;
alter tablespace is_order_008 prefetchsize 0;
alter tablespace is_order_009 prefetchsize 0;
alter tablespace is_order_010 prefetchsize 0;
alter tablespace is_order_011 prefetchsize 0;
alter tablespace is_order_012 prefetchsize 0;
alter tablespace is_order_013 prefetchsize 0;
alter tablespace is_order_014 prefetchsize 0;
alter tablespace is_order_015 prefetchsize 0;
alter tablespace is_order_016 prefetchsize 0;
alter tablespace is_order_017 prefetchsize 0;
alter tablespace is_order_018 prefetchsize 0;
alter tablespace is_order_019 prefetchsize 0;
alter tablespace is_order_020 prefetchsize 0;
alter tablespace is_order_021 prefetchsize 0;
alter tablespace is_order_022 prefetchsize 0;
alter tablespace is_order_023 prefetchsize 0;
alter tablespace is_order_024 prefetchsize 0;
alter tablespace is_order_025 prefetchsize 0;
alter tablespace is_order_026 prefetchsize 0;
alter tablespace is_order_027 prefetchsize 0;
alter tablespace is_order_028 prefetchsize 0;
alter tablespace is_order_029 prefetchsize 0;
alter tablespace is_order_030 prefetchsize 0;
alter tablespace is_order_031 prefetchsize 0;
alter tablespace is_order_032 prefetchsize 0;
alter tablespace is_order_033 prefetchsize 0;
alter tablespace is_order_034 prefetchsize 0;
alter tablespace is_order_035 prefetchsize 0;
alter tablespace is_order_036 prefetchsize 0;
alter tablespace is_order_037 prefetchsize 0;
alter tablespace is_order_038 prefetchsize 0;
alter tablespace is_order_039 prefetchsize 0;
alter tablespace is_order_040 prefetchsize 0;
alter tablespace is_order_041 prefetchsize 0;
alter tablespace is_order_042 prefetchsize 0;
alter tablespace is_order_043 prefetchsize 0;
alter tablespace is_order_044 prefetchsize 0;
alter tablespace is_order_045 prefetchsize 0;
alter tablespace is_order_046 prefetchsize 0;
alter tablespace is_order_047 prefetchsize 0;
alter tablespace is_order_048 prefetchsize 0;
alter tablespace is_order_049 prefetchsize 0;
alter tablespace is_order_050 prefetchsize 0;
alter tablespace is_order_051 prefetchsize 0;
alter tablespace is_order_052 prefetchsize 0;
alter tablespace is_order_053 prefetchsize 0;
alter tablespace is_order_054 prefetchsize 0;
alter tablespace is_order_055 prefetchsize 0;
alter tablespace is_order_056 prefetchsize 0;
alter tablespace is_order_057 prefetchsize 0;
alter tablespace is_order_058 prefetchsize 0;
alter tablespace is_order_059 prefetchsize 0;
alter tablespace is_order_060 prefetchsize 0;
alter tablespace is_order_061 prefetchsize 0;
alter tablespace is_order_062 prefetchsize 0;
alter tablespace is_order_063 prefetchsize 0;
alter tablespace is_order_064 prefetchsize 0;
alter tablespace ts_customer_001 prefetchsize 0;
alter tablespace ts_customer_002 prefetchsize 0;
alter tablespace ts_customer_003 prefetchsize 0;
alter tablespace ts_customer_004 prefetchsize 0;
alter tablespace ts_customer_005 prefetchsize 0;
alter tablespace ts_customer_006 prefetchsize 0;
alter tablespace ts_customer_007 prefetchsize 0;
alter tablespace ts_customer_008 prefetchsize 0;
alter tablespace ts_customer_009 prefetchsize 0;
alter tablespace ts_customer_010 prefetchsize 0;
alter tablespace ts_customer_011 prefetchsize 0;
alter tablespace ts_customer_012 prefetchsize 0;
alter tablespace ts_customer_013 prefetchsize 0;
alter tablespace ts_customer_014 prefetchsize 0;
alter tablespace ts_customer_015 prefetchsize 0;
alter tablespace ts_customer_016 prefetchsize 0;
alter tablespace ts_customer_017 prefetchsize 0;
alter tablespace ts_customer_018 prefetchsize 0;
alter tablespace ts_customer_019 prefetchsize 0;
alter tablespace ts_customer_020 prefetchsize 0;
alter tablespace ts_customer_021 prefetchsize 0;
alter tablespace ts_customer_022 prefetchsize 0;
alter tablespace ts_customer_023 prefetchsize 0;
alter tablespace ts_customer_024 prefetchsize 0;
alter tablespace ts_customer_025 prefetchsize 0;
alter tablespace ts_customer_026 prefetchsize 0;
alter tablespace ts_customer_027 prefetchsize 0;
alter tablespace ts_customer_028 prefetchsize 0;
alter tablespace ts_customer_029 prefetchsize 0;
alter tablespace ts_customer_030 prefetchsize 0;
alter tablespace ts_customer_031 prefetchsize 0;
alter tablespace ts_customer_032 prefetchsize 0;
alter tablespace ts_customer_033 prefetchsize 0;
alter tablespace ts_customer_034 prefetchsize 0;
alter tablespace ts_customer_035 prefetchsize 0;
alter tablespace ts_customer_036 prefetchsize 0;
alter tablespace ts_customer_037 prefetchsize 0;
alter tablespace ts_customer_038 prefetchsize 0;
alter tablespace ts_customer_039 prefetchsize 0;
alter tablespace ts_customer_040 prefetchsize 0;
alter tablespace ts_customer_041 prefetchsize 0;
alter tablespace ts_customer_042 prefetchsize 0;
alter tablespace ts_customer_043 prefetchsize 0;
alter tablespace ts_customer_044 prefetchsize 0;
alter tablespace ts_customer_045 prefetchsize 0;
alter tablespace ts_customer_046 prefetchsize 0;
alter tablespace ts_customer_047 prefetchsize 0;
alter tablespace ts_customer_048 prefetchsize 0;
alter tablespace ts_customer_049 prefetchsize 0;
alter tablespace ts_customer_050 prefetchsize 0;
alter tablespace ts_customer_051 prefetchsize 0;
alter tablespace ts_customer_052 prefetchsize 0;
alter tablespace ts_customer_053 prefetchsize 0;
alter tablespace ts_customer_054 prefetchsize 0;
alter tablespace ts_customer_055 prefetchsize 0;
alter tablespace ts_customer_056 prefetchsize 0;
alter tablespace ts_customer_057 prefetchsize 0;
alter tablespace ts_customer_058 prefetchsize 0;
alter tablespace ts_customer_059 prefetchsize 0;
alter tablespace ts_customer_060 prefetchsize 0;
alter tablespace ts_customer_061 prefetchsize 0;
alter tablespace ts_customer_062 prefetchsize 0;
alter tablespace ts_customer_063 prefetchsize 0;
alter tablespace ts_customer_064 prefetchsize 0;
alter tablespace ts_dist_001 prefetchsize 0;
alter tablespace ts_dist_002 prefetchsize 0;
alter tablespace ts_dist_003 prefetchsize 0;
alter tablespace ts_dist_004 prefetchsize 0;
alter tablespace ts_dist_005 prefetchsize 0;
alter tablespace ts_dist_006 prefetchsize 0;
alter tablespace ts_dist_007 prefetchsize 0;
alter tablespace ts_dist_008 prefetchsize 0;
alter tablespace ts_dist_009 prefetchsize 0;
alter tablespace ts_dist_010 prefetchsize 0;
```

```
alter tablespace ts_dist_011 prefetchsize 0;
alter tablespace ts_dist_012 prefetchsize 0;
alter tablespace ts_dist_013 prefetchsize 0;
alter tablespace ts_dist_014 prefetchsize 0;
alter tablespace ts_dist_015 prefetchsize 0;
alter tablespace ts_dist_016 prefetchsize 0;
alter tablespace ts_dist_017 prefetchsize 0;
alter tablespace ts_dist_018 prefetchsize 0;
alter tablespace ts_dist_019 prefetchsize 0;
alter tablespace ts_dist_020 prefetchsize 0;
alter tablespace ts_dist_021 prefetchsize 0;
alter tablespace ts_dist_022 prefetchsize 0;
alter tablespace ts_dist_023 prefetchsize 0;
alter tablespace ts_dist_024 prefetchsize 0;
alter tablespace ts_dist_025 prefetchsize 0;
alter tablespace ts_dist_026 prefetchsize 0;
alter tablespace ts_dist_027 prefetchsize 0;
alter tablespace ts_dist_028 prefetchsize 0;
alter tablespace ts_dist_029 prefetchsize 0;
alter tablespace ts_dist_030 prefetchsize 0;
alter tablespace ts_dist_031 prefetchsize 0;
alter tablespace ts_dist_032 prefetchsize 0;
alter tablespace ts_history_001 prefetchsize 0;
alter tablespace ts_history_002 prefetchsize 0;
alter tablespace ts_history_003 prefetchsize 0;
alter tablespace ts_history_004 prefetchsize 0;
alter tablespace ts_history_005 prefetchsize 0;
alter tablespace ts_history_006 prefetchsize 0;
alter tablespace ts_history_007 prefetchsize 0;
alter tablespace ts_history_008 prefetchsize 0;
alter tablespace ts_history_009 prefetchsize 0;
alter tablespace ts_history_010 prefetchsize 0;
alter tablespace ts_history_011 prefetchsize 0;
alter tablespace ts_history_012 prefetchsize 0;
alter tablespace ts_history_013 prefetchsize 0;
alter tablespace ts_history_014 prefetchsize 0;
alter tablespace ts_history_015 prefetchsize 0;
alter tablespace ts_history_016 prefetchsize 0;
alter tablespace ts_history_017 prefetchsize 0;
alter tablespace ts_history_018 prefetchsize 0;
alter tablespace ts_history_019 prefetchsize 0;
alter tablespace ts_history_020 prefetchsize 0;
alter tablespace ts_history_021 prefetchsize 0;
alter tablespace ts_history_022 prefetchsize 0;
alter tablespace ts_history_023 prefetchsize 0;
alter tablespace ts_history_024 prefetchsize 0;
alter tablespace ts_history_025 prefetchsize 0;
alter tablespace ts_history_026 prefetchsize 0;
alter tablespace ts_history_027 prefetchsize 0;
alter tablespace ts_history_028 prefetchsize 0;
alter tablespace ts_history_029 prefetchsize 0;
alter tablespace ts_history_030 prefetchsize 0;
alter tablespace ts_history_031 prefetchsize 0;
alter tablespace ts_history_032 prefetchsize 0;
alter tablespace ts_item_001 prefetchsize 0;
alter tablespace ts_newordA_001 prefetchsize 0;
alter tablespace ts_newordA_002 prefetchsize 0;
alter tablespace ts_newordA_003 prefetchsize 0;
alter tablespace ts_newordA_004 prefetchsize 0;
alter tablespace ts_newordA_005 prefetchsize 0;
alter tablespace ts_newordA_006 prefetchsize 0;
alter tablespace ts_newordA_007 prefetchsize 0;
alter tablespace ts_newordA_008 prefetchsize 0;
alter tablespace ts_newordA_009 prefetchsize 0;
alter tablespace ts_newordA_010 prefetchsize 0;
alter tablespace ts_newordA_011 prefetchsize 0;
alter tablespace ts_newordA_012 prefetchsize 0;
alter tablespace ts_newordA_013 prefetchsize 0;
alter tablespace ts_newordA_014 prefetchsize 0;
alter tablespace ts_newordA_015 prefetchsize 0;
alter tablespace ts_newordA_016 prefetchsize 0;
alter tablespace ts_newordA_017 prefetchsize 0;
alter tablespace ts_newordA_018 prefetchsize 0;
alter tablespace ts_newordA_019 prefetchsize 0;
alter tablespace ts_newordA_020 prefetchsize 0;
alter tablespace ts_newordA_021 prefetchsize 0;
alter tablespace ts_newordA_022 prefetchsize 0;
alter tablespace ts_newordA_023 prefetchsize 0;
alter tablespace ts_newordA_024 prefetchsize 0;
alter tablespace ts_newordA_025 prefetchsize 0;
alter tablespace ts_newordA_026 prefetchsize 0;
alter tablespace ts_newordA_027 prefetchsize 0;
alter tablespace ts_newordA_028 prefetchsize 0;
alter tablespace ts_newordA_029 prefetchsize 0;
alter tablespace ts_newordA_030 prefetchsize 0;
alter tablespace ts_newordA_031 prefetchsize 0;
alter tablespace ts_newordA_032 prefetchsize 0;
alter tablespace ts_newordB_001 prefetchsize 0;
alter tablespace ts_newordB_002 prefetchsize 0;
alter tablespace ts_newordB_003 prefetchsize 0;
alter tablespace ts_newordB_004 prefetchsize 0;
alter tablespace ts_newordB_005 prefetchsize 0;
alter tablespace ts_newordB_006 prefetchsize 0;
alter tablespace ts_newordB_007 prefetchsize 0;
alter tablespace ts_newordB_008 prefetchsize 0;
alter tablespace ts_newordB_009 prefetchsize 0;
alter tablespace ts_newordB_010 prefetchsize 0;
alter tablespace ts_newordB_011 prefetchsize 0;
alter tablespace ts_newordB_012 prefetchsize 0;
alter tablespace ts_newordB_013 prefetchsize 0;
alter tablespace ts_newordB_014 prefetchsize 0;
alter tablespace ts_newordB_015 prefetchsize 0;
alter tablespace ts_newordB_016 prefetchsize 0;
alter tablespace ts_newordB_017 prefetchsize 0;
alter tablespace ts_newordB_018 prefetchsize 0;
alter tablespace ts_newordB_019 prefetchsize 0;
alter tablespace ts_newordB_020 prefetchsize 0;
alter tablespace ts_newordB_021 prefetchsize 0;
alter tablespace ts_newordB_022 prefetchsize 0;
alter tablespace ts_newordB_023 prefetchsize 0;
alter tablespace ts_newordB_024 prefetchsize 0;
alter tablespace ts_newordB_025 prefetchsize 0;
alter tablespace ts_newordB_026 prefetchsize 0;
alter tablespace ts_newordB_027 prefetchsize 0;
alter tablespace ts_newordB_028 prefetchsize 0;
alter tablespace ts_newordB_029 prefetchsize 0;
alter tablespace ts_newordB_030 prefetchsize 0;
alter tablespace ts_newordB_031 prefetchsize 0;
alter tablespace ts_newordB_032 prefetchsize 0;
alter tablespace ts_order_001 prefetchsize 0;
alter tablespace ts_order_002 prefetchsize 0;
alter tablespace ts_order_003 prefetchsize 0;
alter tablespace ts_order_004 prefetchsize 0;
alter tablespace ts_order_005 prefetchsize 0;
alter tablespace ts_order_006 prefetchsize 0;
alter tablespace ts_order_007 prefetchsize 0;
alter tablespace ts_order_008 prefetchsize 0;
alter tablespace ts_order_009 prefetchsize 0;
alter tablespace ts_order_010 prefetchsize 0;
alter tablespace ts_order_011 prefetchsize 0;
alter tablespace ts_order_012 prefetchsize 0;
alter tablespace ts_order_013 prefetchsize 0;
alter tablespace ts_order_014 prefetchsize 0;
alter tablespace ts_order_015 prefetchsize 0;
alter tablespace ts_order_016 prefetchsize 0;
alter tablespace ts_order_017 prefetchsize 0;
alter tablespace ts_order_018 prefetchsize 0;
alter tablespace ts_order_019 prefetchsize 0;
alter tablespace ts_order_020 prefetchsize 0;
alter tablespace ts_order_021 prefetchsize 0;
alter tablespace ts_order_022 prefetchsize 0;
alter tablespace ts_order_023 prefetchsize 0;
alter tablespace ts_order_024 prefetchsize 0;
alter tablespace ts_order_025 prefetchsize 0;
alter tablespace ts_order_026 prefetchsize 0;
alter tablespace ts_order_027 prefetchsize 0;
alter tablespace ts_order_028 prefetchsize 0;
alter tablespace ts_order_029 prefetchsize 0;
alter tablespace ts_order_030 prefetchsize 0;
alter tablespace ts_order_031 prefetchsize 0;
alter tablespace ts_order_032 prefetchsize 0;
alter tablespace ts_order_033 prefetchsize 0;
alter tablespace ts_order_034 prefetchsize 0;
alter tablespace ts_order_035 prefetchsize 0;
alter tablespace ts_order_036 prefetchsize 0;
alter tablespace ts_order_037 prefetchsize 0;
alter tablespace ts_order_038 prefetchsize 0;
alter tablespace ts_order_039 prefetchsize 0;
alter tablespace ts_order_040 prefetchsize 0;
alter tablespace ts_order_041 prefetchsize 0;
alter tablespace ts_order_042 prefetchsize 0;
alter tablespace ts_order_043 prefetchsize 0;
alter tablespace ts_order_044 prefetchsize 0;
alter tablespace ts_order_045 prefetchsize 0;
alter tablespace ts_order_046 prefetchsize 0;
alter tablespace ts_order_047 prefetchsize 0;
alter tablespace ts_order_048 prefetchsize 0;
alter tablespace ts_order_049 prefetchsize 0;
alter tablespace ts_order_050 prefetchsize 0;
alter tablespace ts_order_051 prefetchsize 0;
alter tablespace ts_order_052 prefetchsize 0;
alter tablespace ts_order_053 prefetchsize 0;
alter tablespace ts_order_054 prefetchsize 0;
alter tablespace ts_order_055 prefetchsize 0;
alter tablespace ts_order_056 prefetchsize 0;
alter tablespace ts_order_057 prefetchsize 0;
alter tablespace ts_order_058 prefetchsize 0;
alter tablespace ts_order_059 prefetchsize 0;
alter tablespace ts_order_060 prefetchsize 0;
alter tablespace ts_order_061 prefetchsize 0;
alter tablespace ts_order_062 prefetchsize 0;
alter tablespace ts_order_063 prefetchsize 0;
alter tablespace ts_order_064 prefetchsize 0;
alter tablespace ts_orderline_001 prefetchsize 0;
alter tablespace ts_orderline_002 prefetchsize 0;
alter tablespace ts_orderline_003 prefetchsize 0;
alter tablespace ts_orderline_004 prefetchsize 0;
alter tablespace ts_orderline_005 prefetchsize 0;
alter tablespace ts_orderline_006 prefetchsize 0;
alter tablespace ts_orderline_007 prefetchsize 0;
alter tablespace ts_orderline_008 prefetchsize 0;
alter tablespace ts_orderline_009 prefetchsize 0;
alter tablespace ts_orderline_010 prefetchsize 0;
alter tablespace ts_orderline_011 prefetchsize 0;
alter tablespace ts_orderline_012 prefetchsize 0;
alter tablespace ts_orderline_013 prefetchsize 0;
alter tablespace ts_orderline_014 prefetchsize 0;
alter tablespace ts_orderline_015 prefetchsize 0;
alter tablespace ts_orderline_016 prefetchsize 0;
alter tablespace ts_orderline_017 prefetchsize 0;
alter tablespace ts_orderline_018 prefetchsize 0;
alter tablespace ts_orderline_019 prefetchsize 0;
alter tablespace ts_orderline_020 prefetchsize 0;
alter tablespace ts_orderline_021 prefetchsize 0;
alter tablespace ts_orderline_022 prefetchsize 0;
alter tablespace ts_orderline_023 prefetchsize 0;
alter tablespace ts_orderline_024 prefetchsize 0;
alter tablespace ts_orderline_025 prefetchsize 0;
alter tablespace ts_orderline_026 prefetchsize 0;
alter tablespace ts_orderline_027 prefetchsize 0;
alter tablespace ts_orderline_028 prefetchsize 0;
alter tablespace ts_orderline_029 prefetchsize 0;
alter tablespace ts_orderline_030 prefetchsize 0;
alter tablespace ts_orderline_031 prefetchsize 0;
alter tablespace ts_orderline_032 prefetchsize 0;
alter tablespace ts_orderline_033 prefetchsize 0;
alter tablespace ts_orderline_034 prefetchsize 0;
alter tablespace ts_orderline_035 prefetchsize 0;
alter tablespace ts_orderline_036 prefetchsize 0;
```

```
alter tablespace ts_orderline_037 prefetchsize 0;
alter tablespace ts_orderline_038 prefetchsize 0;
alter tablespace ts_orderline_039 prefetchsize 0;
alter tablespace ts_orderline_040 prefetchsize 0;
alter tablespace ts_orderline_041 prefetchsize 0;
alter tablespace ts_orderline_042 prefetchsize 0;
alter tablespace ts_orderline_043 prefetchsize 0;
alter tablespace ts_orderline_044 prefetchsize 0;
alter tablespace ts_orderline_045 prefetchsize 0;
alter tablespace ts_orderline_046 prefetchsize 0;
alter tablespace ts_orderline_047 prefetchsize 0;
alter tablespace ts_orderline_048 prefetchsize 0;
alter tablespace ts_orderline_049 prefetchsize 0;
alter tablespace ts_orderline_050 prefetchsize 0;
alter tablespace ts_orderline_051 prefetchsize 0;
alter tablespace ts_orderline_052 prefetchsize 0;
alter tablespace ts_orderline_053 prefetchsize 0;
alter tablespace ts_orderline_054 prefetchsize 0;
alter tablespace ts_orderline_055 prefetchsize 0;
alter tablespace ts_orderline_056 prefetchsize 0;
alter tablespace ts_orderline_057 prefetchsize 0;
alter tablespace ts_orderline_058 prefetchsize 0;
alter tablespace ts_orderline_059 prefetchsize 0;
alter tablespace ts_orderline_060 prefetchsize 0;
alter tablespace ts_orderline_061 prefetchsize 0;
alter tablespace ts_orderline_062 prefetchsize 0;
alter tablespace ts_orderline_063 prefetchsize 0;
alter tablespace ts_orderline_064 prefetchsize 0;
alter tablespace ts_stock_001 prefetchsize 0;
alter tablespace ts_stock_002 prefetchsize 0;
alter tablespace ts_stock_003 prefetchsize 0;
alter tablespace ts_stock_004 prefetchsize 0;
alter tablespace ts_stock_005 prefetchsize 0;
alter tablespace ts_stock_006 prefetchsize 0;
alter tablespace ts_stock_007 prefetchsize 0;
alter tablespace ts_stock_008 prefetchsize 0;
alter tablespace ts_stock_009 prefetchsize 0;
alter tablespace ts_stock_010 prefetchsize 0;
alter tablespace ts_stock_011 prefetchsize 0;
alter tablespace ts_stock_012 prefetchsize 0;
alter tablespace ts_stock_013 prefetchsize 0;
alter tablespace ts_stock_014 prefetchsize 0;
alter tablespace ts_stock_015 prefetchsize 0;
alter tablespace ts_stock_016 prefetchsize 0;
alter tablespace ts_stock_017 prefetchsize 0;
alter tablespace ts_stock_018 prefetchsize 0;
alter tablespace ts_stock_019 prefetchsize 0;
alter tablespace ts_stock_020 prefetchsize 0;
alter tablespace ts_stock_021 prefetchsize 0;
alter tablespace ts_stock_022 prefetchsize 0;
alter tablespace ts_stock_023 prefetchsize 0;
alter tablespace ts_stock_024 prefetchsize 0;
alter tablespace ts_stock_025 prefetchsize 0;
alter tablespace ts_stock_026 prefetchsize 0;
alter tablespace ts_stock_027 prefetchsize 0;
alter tablespace ts_stock_028 prefetchsize 0;
alter tablespace ts_stock_029 prefetchsize 0;
alter tablespace ts_stock_030 prefetchsize 0;
alter tablespace ts_stock_031 prefetchsize 0;
alter tablespace ts_stock_032 prefetchsize 0;
alter tablespace ts_stock_033 prefetchsize 0;
alter tablespace ts_stock_034 prefetchsize 0;
alter tablespace ts_stock_035 prefetchsize 0;
alter tablespace ts_stock_036 prefetchsize 0;
alter tablespace ts_stock_037 prefetchsize 0;
alter tablespace ts_stock_038 prefetchsize 0;
alter tablespace ts_stock_039 prefetchsize 0;
alter tablespace ts_stock_040 prefetchsize 0;
alter tablespace ts_stock_041 prefetchsize 0;
alter tablespace ts_stock_042 prefetchsize 0;
alter tablespace ts_stock_043 prefetchsize 0;
alter tablespace ts_stock_044 prefetchsize 0;
alter tablespace ts_stock_045 prefetchsize 0;
alter tablespace ts_stock_046 prefetchsize 0;
alter tablespace ts_stock_047 prefetchsize 0;
alter tablespace ts_stock_048 prefetchsize 0;
alter tablespace ts_stock_049 prefetchsize 0;
alter tablespace ts_stock_050 prefetchsize 0;
alter tablespace ts_stock_051 prefetchsize 0;
alter tablespace ts_stock_052 prefetchsize 0;
alter tablespace ts_stock_053 prefetchsize 0;
alter tablespace ts_stock_054 prefetchsize 0;
alter tablespace ts_stock_055 prefetchsize 0;
alter tablespace ts_stock_056 prefetchsize 0;
alter tablespace ts_stock_057 prefetchsize 0;
alter tablespace ts_stock_058 prefetchsize 0;
alter tablespace ts_stock_059 prefetchsize 0;
alter tablespace ts_stock_060 prefetchsize 0;
alter tablespace ts_stock_061 prefetchsize 0;
alter tablespace ts_stock_062 prefetchsize 0;
alter tablespace ts_stock_063 prefetchsize 0;
alter tablespace ts_stock_064 prefetchsize 0;
alter tablespace ts_ware_001 prefetchsize 0;
alter tablespace ts_ware_002 prefetchsize 0;
alter tablespace ts_ware_003 prefetchsize 0;
alter tablespace ts_ware_004 prefetchsize 0;
alter tablespace ts_ware_005 prefetchsize 0;
alter tablespace ts_ware_006 prefetchsize 0;
alter tablespace ts_ware_007 prefetchsize 0;
alter tablespace ts_ware_008 prefetchsize 0;
alter tablespace ts_ware_009 prefetchsize 0;
alter tablespace ts_ware_010 prefetchsize 0;
alter tablespace ts_ware_011 prefetchsize 0;
alter tablespace ts_ware_012 prefetchsize 0;
alter tablespace ts_ware_013 prefetchsize 0;
alter tablespace ts_ware_014 prefetchsize 0;
alter tablespace ts_ware_015 prefetchsize 0;
alter tablespace ts_ware_016 prefetchsize 0;
alter tablespace ts_ware_017 prefetchsize 0;
alter tablespace ts_ware_018 prefetchsize 0;
alter tablespace ts_ware_019 prefetchsize 0;
alter tablespace ts_ware_020 prefetchsize 0;
alter tablespace ts_ware_021 prefetchsize 0;
alter tablespace ts_ware_022 prefetchsize 0;
alter tablespace ts_ware_023 prefetchsize 0;
alter tablespace ts_ware_024 prefetchsize 0;
alter tablespace ts_ware_025 prefetchsize 0;
alter tablespace ts_ware_026 prefetchsize 0;
alter tablespace ts_ware_027 prefetchsize 0;
alter tablespace ts_ware_028 prefetchsize 0;
alter tablespace ts_ware_029 prefetchsize 0;
alter tablespace ts_ware_030 prefetchsize 0;
alter tablespace ts_ware_031 prefetchsize 0;
alter tablespace ts_ware_032 prefetchsize 0;
connect reset;
```

DDL/ALTTBSP_PF_4096.ddl

```
connect to TPCC;
alter tablespace is_customer_001 prefetchsize 4096;
alter tablespace is_customer_002 prefetchsize 4096;
alter tablespace is_customer_003 prefetchsize 4096;
alter tablespace is_customer_004 prefetchsize 4096;
alter tablespace is_customer_005 prefetchsize 4096;
alter tablespace is_customer_006 prefetchsize 4096;
alter tablespace is_customer_007 prefetchsize 4096;
alter tablespace is_customer_008 prefetchsize 4096;
alter tablespace is_customer_009 prefetchsize 4096;
alter tablespace is_customer_010 prefetchsize 4096;
alter tablespace is_customer_011 prefetchsize 4096;
alter tablespace is_customer_012 prefetchsize 4096;
alter tablespace is_customer_013 prefetchsize 4096;
alter tablespace is_customer_014 prefetchsize 4096;
alter tablespace is_customer_015 prefetchsize 4096;
alter tablespace is_customer_016 prefetchsize 4096;
alter tablespace is_customer_017 prefetchsize 4096;
alter tablespace is_customer_018 prefetchsize 4096;
alter tablespace is_customer_019 prefetchsize 4096;
alter tablespace is_customer_020 prefetchsize 4096;
alter tablespace is_customer_021 prefetchsize 4096;
alter tablespace is_customer_022 prefetchsize 4096;
alter tablespace is_customer_023 prefetchsize 4096;
alter tablespace is_customer_024 prefetchsize 4096;
alter tablespace is_customer_025 prefetchsize 4096;
alter tablespace is_customer_026 prefetchsize 4096;
alter tablespace is_customer_027 prefetchsize 4096;
alter tablespace is_customer_028 prefetchsize 4096;
alter tablespace is_customer_029 prefetchsize 4096;
alter tablespace is_customer_030 prefetchsize 4096;
alter tablespace is_customer_031 prefetchsize 4096;
alter tablespace is_customer_032 prefetchsize 4096;
alter tablespace is_customer_033 prefetchsize 4096;
alter tablespace is_customer_034 prefetchsize 4096;
alter tablespace is_customer_035 prefetchsize 4096;
alter tablespace is_customer_036 prefetchsize 4096;
alter tablespace is_customer_037 prefetchsize 4096;
alter tablespace is_customer_038 prefetchsize 4096;
alter tablespace is_customer_039 prefetchsize 4096;
alter tablespace is_customer_040 prefetchsize 4096;
alter tablespace is_customer_041 prefetchsize 4096;
alter tablespace is_customer_042 prefetchsize 4096;
alter tablespace is_customer_043 prefetchsize 4096;
alter tablespace is_customer_044 prefetchsize 4096;
alter tablespace is_customer_045 prefetchsize 4096;
alter tablespace is_customer_046 prefetchsize 4096;
alter tablespace is_customer_047 prefetchsize 4096;
alter tablespace is_customer_048 prefetchsize 4096;
alter tablespace is_customer_049 prefetchsize 4096;
alter tablespace is_customer_050 prefetchsize 4096;
alter tablespace is_customer_051 prefetchsize 4096;
alter tablespace is_customer_052 prefetchsize 4096;
alter tablespace is_customer_053 prefetchsize 4096;
alter tablespace is_customer_054 prefetchsize 4096;
alter tablespace is_customer_055 prefetchsize 4096;
alter tablespace is_customer_056 prefetchsize 4096;
alter tablespace is_customer_057 prefetchsize 4096;
alter tablespace is_customer_058 prefetchsize 4096;
alter tablespace is_customer_059 prefetchsize 4096;
alter tablespace is_customer_060 prefetchsize 4096;
alter tablespace is_customer_061 prefetchsize 4096;
alter tablespace is_customer_062 prefetchsize 4096;
alter tablespace is_customer_063 prefetchsize 4096;
alter tablespace is_customer_064 prefetchsize 4096;
alter tablespace is_order_001 prefetchsize 4096;
alter tablespace is_order_002 prefetchsize 4096;
alter tablespace is_order_003 prefetchsize 4096;
alter tablespace is_order_004 prefetchsize 4096;
alter tablespace is_order_005 prefetchsize 4096;
alter tablespace is_order_006 prefetchsize 4096;
alter tablespace is_order_007 prefetchsize 4096;
alter tablespace is_order_008 prefetchsize 4096;
alter tablespace is_order_009 prefetchsize 4096;
alter tablespace is_order_010 prefetchsize 4096;
alter tablespace is_order_011 prefetchsize 4096;
alter tablespace is_order_012 prefetchsize 4096;
alter tablespace is_order_013 prefetchsize 4096;
alter tablespace is_order_014 prefetchsize 4096;
alter tablespace is_order_015 prefetchsize 4096;
alter tablespace is_order_016 prefetchsize 4096;
alter tablespace is_order_017 prefetchsize 4096;
alter tablespace is_order_018 prefetchsize 4096;
alter tablespace is_order_019 prefetchsize 4096;
alter tablespace is_order_020 prefetchsize 4096;
alter tablespace is_order_021 prefetchsize 4096;
alter tablespace is_order_022 prefetchsize 4096;
```

```
alter tablespace is_order_023 prefetchsize 4096;
alter tablespace is_order_024 prefetchsize 4096;
alter tablespace is_order_025 prefetchsize 4096;
alter tablespace is_order_026 prefetchsize 4096;
alter tablespace is_order_027 prefetchsize 4096;
alter tablespace is_order_028 prefetchsize 4096;
alter tablespace is_order_029 prefetchsize 4096;
alter tablespace is_order_030 prefetchsize 4096;
alter tablespace is_order_031 prefetchsize 4096;
alter tablespace is_order_032 prefetchsize 4096;
alter tablespace is_order_033 prefetchsize 4096;
alter tablespace is_order_034 prefetchsize 4096;
alter tablespace is_order_035 prefetchsize 4096;
alter tablespace is_order_036 prefetchsize 4096;
alter tablespace is_order_037 prefetchsize 4096;
alter tablespace is_order_038 prefetchsize 4096;
alter tablespace is_order_039 prefetchsize 4096;
alter tablespace is_order_040 prefetchsize 4096;
alter tablespace is_order_041 prefetchsize 4096;
alter tablespace is_order_042 prefetchsize 4096;
alter tablespace is_order_043 prefetchsize 4096;
alter tablespace is_order_044 prefetchsize 4096;
alter tablespace is_order_045 prefetchsize 4096;
alter tablespace is_order_046 prefetchsize 4096;
alter tablespace is_order_047 prefetchsize 4096;
alter tablespace is_order_048 prefetchsize 4096;
alter tablespace is_order_049 prefetchsize 4096;
alter tablespace is_order_050 prefetchsize 4096;
alter tablespace is_order_051 prefetchsize 4096;
alter tablespace is_order_052 prefetchsize 4096;
alter tablespace is_order_053 prefetchsize 4096;
alter tablespace is_order_054 prefetchsize 4096;
alter tablespace is_order_055 prefetchsize 4096;
alter tablespace is_order_056 prefetchsize 4096;
alter tablespace is_order_057 prefetchsize 4096;
alter tablespace is_order_058 prefetchsize 4096;
alter tablespace is_order_059 prefetchsize 4096;
alter tablespace is_order_060 prefetchsize 4096;
alter tablespace is_order_061 prefetchsize 4096;
alter tablespace is_order_062 prefetchsize 4096;
alter tablespace is_order_063 prefetchsize 4096;
alter tablespace is_order_064 prefetchsize 4096;
alter tablespace ts_customer_001 prefetchsize 4096;
alter tablespace ts_customer_002 prefetchsize 4096;
alter tablespace ts_customer_003 prefetchsize 4096;
alter tablespace ts_customer_004 prefetchsize 4096;
alter tablespace ts_customer_005 prefetchsize 4096;
alter tablespace ts_customer_006 prefetchsize 4096;
alter tablespace ts_customer_007 prefetchsize 4096;
alter tablespace ts_customer_008 prefetchsize 4096;
alter tablespace ts_customer_009 prefetchsize 4096;
alter tablespace ts_customer_010 prefetchsize 4096;
alter tablespace ts_customer_011 prefetchsize 4096;
alter tablespace ts_customer_012 prefetchsize 4096;
alter tablespace ts_customer_013 prefetchsize 4096;
alter tablespace ts_customer_014 prefetchsize 4096;
alter tablespace ts_customer_015 prefetchsize 4096;
alter tablespace ts_customer_016 prefetchsize 4096;
alter tablespace ts_customer_017 prefetchsize 4096;
alter tablespace ts_customer_018 prefetchsize 4096;
alter tablespace ts_customer_019 prefetchsize 4096;
alter tablespace ts_customer_020 prefetchsize 4096;
alter tablespace ts_customer_021 prefetchsize 4096;
alter tablespace ts_customer_022 prefetchsize 4096;
alter tablespace ts_customer_023 prefetchsize 4096;
alter tablespace ts_customer_024 prefetchsize 4096;
alter tablespace ts_customer_025 prefetchsize 4096;
alter tablespace ts_customer_026 prefetchsize 4096;
alter tablespace ts_customer_027 prefetchsize 4096;
alter tablespace ts_customer_028 prefetchsize 4096;
alter tablespace ts_customer_029 prefetchsize 4096;
alter tablespace ts_customer_030 prefetchsize 4096;
alter tablespace ts_customer_031 prefetchsize 4096;
alter tablespace ts_customer_032 prefetchsize 4096;
alter tablespace ts_customer_033 prefetchsize 4096;
alter tablespace ts_customer_034 prefetchsize 4096;
alter tablespace ts_customer_035 prefetchsize 4096;
alter tablespace ts_customer_036 prefetchsize 4096;
alter tablespace ts_customer_037 prefetchsize 4096;
alter tablespace ts_customer_038 prefetchsize 4096;
alter tablespace ts_customer_039 prefetchsize 4096;
alter tablespace ts_customer_040 prefetchsize 4096;
alter tablespace ts_customer_041 prefetchsize 4096;
alter tablespace ts_customer_042 prefetchsize 4096;
alter tablespace ts_customer_043 prefetchsize 4096;
alter tablespace ts_customer_044 prefetchsize 4096;
alter tablespace ts_customer_045 prefetchsize 4096;
alter tablespace ts_customer_046 prefetchsize 4096;
alter tablespace ts_customer_047 prefetchsize 4096;
alter tablespace ts_customer_048 prefetchsize 4096;
alter tablespace ts_customer_049 prefetchsize 4096;
alter tablespace ts_customer_050 prefetchsize 4096;
alter tablespace ts_customer_051 prefetchsize 4096;
alter tablespace ts_customer_052 prefetchsize 4096;
alter tablespace ts_customer_053 prefetchsize 4096;
alter tablespace ts_customer_054 prefetchsize 4096;
alter tablespace ts_customer_055 prefetchsize 4096;
alter tablespace ts_customer_056 prefetchsize 4096;
alter tablespace ts_customer_057 prefetchsize 4096;
alter tablespace ts_customer_058 prefetchsize 4096;
alter tablespace ts_customer_059 prefetchsize 4096;
alter tablespace ts_customer_060 prefetchsize 4096;
alter tablespace ts_customer_061 prefetchsize 4096;
alter tablespace ts_customer_062 prefetchsize 4096;
alter tablespace ts_customer_063 prefetchsize 4096;
alter tablespace ts_customer_064 prefetchsize 4096;
alter tablespace ts_dist_001 prefetchsize 4096;
alter tablespace ts_dist_002 prefetchsize 4096;
alter tablespace ts_dist_003 prefetchsize 4096;
alter tablespace ts_dist_004 prefetchsize 4096;
alter tablespace ts_dist_005 prefetchsize 4096;
alter tablespace ts_dist_006 prefetchsize 4096;
alter tablespace ts_dist_007 prefetchsize 4096;
alter tablespace ts_dist_008 prefetchsize 4096;
alter tablespace ts_dist_009 prefetchsize 4096;
alter tablespace ts_dist_010 prefetchsize 4096;
alter tablespace ts_dist_011 prefetchsize 4096;
alter tablespace ts_dist_012 prefetchsize 4096;
alter tablespace ts_dist_013 prefetchsize 4096;
alter tablespace ts_dist_014 prefetchsize 4096;
alter tablespace ts_dist_015 prefetchsize 4096;
alter tablespace ts_dist_016 prefetchsize 4096;
alter tablespace ts_dist_017 prefetchsize 4096;
alter tablespace ts_dist_018 prefetchsize 4096;
alter tablespace ts_dist_019 prefetchsize 4096;
alter tablespace ts_dist_020 prefetchsize 4096;
alter tablespace ts_dist_021 prefetchsize 4096;
alter tablespace ts_dist_022 prefetchsize 4096;
alter tablespace ts_dist_023 prefetchsize 4096;
alter tablespace ts_dist_024 prefetchsize 4096;
alter tablespace ts_dist_025 prefetchsize 4096;
alter tablespace ts_dist_026 prefetchsize 4096;
alter tablespace ts_dist_027 prefetchsize 4096;
alter tablespace ts_dist_028 prefetchsize 4096;
alter tablespace ts_dist_029 prefetchsize 4096;
alter tablespace ts_dist_030 prefetchsize 4096;
alter tablespace ts_dist_031 prefetchsize 4096;
alter tablespace ts_dist_032 prefetchsize 4096;
alter tablespace ts_history_001 prefetchsize 4096;
alter tablespace ts_history_002 prefetchsize 4096;
alter tablespace ts_history_003 prefetchsize 4096;
alter tablespace ts_history_004 prefetchsize 4096;
alter tablespace ts_history_005 prefetchsize 4096;
alter tablespace ts_history_006 prefetchsize 4096;
alter tablespace ts_history_007 prefetchsize 4096;
alter tablespace ts_history_008 prefetchsize 4096;
alter tablespace ts_history_009 prefetchsize 4096;
alter tablespace ts_history_010 prefetchsize 4096;
alter tablespace ts_history_011 prefetchsize 4096;
alter tablespace ts_history_012 prefetchsize 4096;
alter tablespace ts_history_013 prefetchsize 4096;
alter tablespace ts_history_014 prefetchsize 4096;
alter tablespace ts_history_015 prefetchsize 4096;
alter tablespace ts_history_016 prefetchsize 4096;
alter tablespace ts_history_017 prefetchsize 4096;
alter tablespace ts_history_018 prefetchsize 4096;
alter tablespace ts_history_019 prefetchsize 4096;
alter tablespace ts_history_020 prefetchsize 4096;
alter tablespace ts_history_021 prefetchsize 4096;
alter tablespace ts_history_022 prefetchsize 4096;
alter tablespace ts_history_023 prefetchsize 4096;
alter tablespace ts_history_024 prefetchsize 4096;
alter tablespace ts_history_025 prefetchsize 4096;
alter tablespace ts_history_026 prefetchsize 4096;
alter tablespace ts_history_027 prefetchsize 4096;
alter tablespace ts_history_028 prefetchsize 4096;
alter tablespace ts_history_029 prefetchsize 4096;
alter tablespace ts_history_030 prefetchsize 4096;
alter tablespace ts_history_031 prefetchsize 4096;
alter tablespace ts_history_032 prefetchsize 4096;
alter tablespace ts_item_001 prefetchsize 4096;
alter tablespace ts_newordA_001 prefetchsize 4096;
alter tablespace ts_newordA_002 prefetchsize 4096;
alter tablespace ts_newordA_003 prefetchsize 4096;
alter tablespace ts_newordA_004 prefetchsize 4096;
alter tablespace ts_newordA_005 prefetchsize 4096;
alter tablespace ts_newordA_006 prefetchsize 4096;
alter tablespace ts_newordA_007 prefetchsize 4096;
alter tablespace ts_newordA_008 prefetchsize 4096;
alter tablespace ts_newordA_009 prefetchsize 4096;
alter tablespace ts_newordA_010 prefetchsize 4096;
alter tablespace ts_newordA_011 prefetchsize 4096;
alter tablespace ts_newordA_012 prefetchsize 4096;
alter tablespace ts_newordA_013 prefetchsize 4096;
alter tablespace ts_newordA_014 prefetchsize 4096;
alter tablespace ts_newordA_015 prefetchsize 4096;
alter tablespace ts_newordA_016 prefetchsize 4096;
alter tablespace ts_newordA_017 prefetchsize 4096;
alter tablespace ts_newordA_018 prefetchsize 4096;
alter tablespace ts_newordA_019 prefetchsize 4096;
alter tablespace ts_newordA_020 prefetchsize 4096;
alter tablespace ts_newordA_021 prefetchsize 4096;
alter tablespace ts_newordA_022 prefetchsize 4096;
alter tablespace ts_newordA_023 prefetchsize 4096;
alter tablespace ts_newordA_024 prefetchsize 4096;
alter tablespace ts_newordA_025 prefetchsize 4096;
alter tablespace ts_newordA_026 prefetchsize 4096;
alter tablespace ts_newordA_027 prefetchsize 4096;
alter tablespace ts_newordA_028 prefetchsize 4096;
alter tablespace ts_newordA_029 prefetchsize 4096;
alter tablespace ts_newordA_030 prefetchsize 4096;
alter tablespace ts_newordA_031 prefetchsize 4096;
alter tablespace ts_newordA_032 prefetchsize 4096;
alter tablespace ts_newordB_001 prefetchsize 4096;
alter tablespace ts_newordB_002 prefetchsize 4096;
alter tablespace ts_newordB_003 prefetchsize 4096;
alter tablespace ts_newordB_004 prefetchsize 4096;
alter tablespace ts_newordB_005 prefetchsize 4096;
alter tablespace ts_newordB_006 prefetchsize 4096;
alter tablespace ts_newordB_007 prefetchsize 4096;
alter tablespace ts_newordB_008 prefetchsize 4096;
alter tablespace ts_newordB_009 prefetchsize 4096;
alter tablespace ts_newordB_010 prefetchsize 4096;
alter tablespace ts_newordB_011 prefetchsize 4096;
alter tablespace ts_newordB_012 prefetchsize 4096;
alter tablespace ts_newordB_013 prefetchsize 4096;
alter tablespace ts_newordB_014 prefetchsize 4096;
alter tablespace ts_newordB_015 prefetchsize 4096;
alter tablespace ts_newordB_016 prefetchsize 4096;
```

```
alter tablespace ts_newordB_017 prefetchsize 4096;
alter tablespace ts_newordB_018 prefetchsize 4096;
alter tablespace ts_newordB_019 prefetchsize 4096;
alter tablespace ts_newordB_020 prefetchsize 4096;
alter tablespace ts_newordB_021 prefetchsize 4096;
alter tablespace ts_newordB_022 prefetchsize 4096;
alter tablespace ts_newordB_023 prefetchsize 4096;
alter tablespace ts_newordB_024 prefetchsize 4096;
alter tablespace ts_newordB_025 prefetchsize 4096;
alter tablespace ts_newordB_026 prefetchsize 4096;
alter tablespace ts_newordB_027 prefetchsize 4096;
alter tablespace ts_newordB_028 prefetchsize 4096;
alter tablespace ts_newordB_029 prefetchsize 4096;
alter tablespace ts_newordB_030 prefetchsize 4096;
alter tablespace ts_newordB_031 prefetchsize 4096;
alter tablespace ts_newordB_032 prefetchsize 4096;
alter tablespace ts_order_001 prefetchsize 4096;
alter tablespace ts_order_002 prefetchsize 4096;
alter tablespace ts_order_003 prefetchsize 4096;
alter tablespace ts_order_004 prefetchsize 4096;
alter tablespace ts_order_005 prefetchsize 4096;
alter tablespace ts_order_006 prefetchsize 4096;
alter tablespace ts_order_007 prefetchsize 4096;
alter tablespace ts_order_008 prefetchsize 4096;
alter tablespace ts_order_009 prefetchsize 4096;
alter tablespace ts_order_010 prefetchsize 4096;
alter tablespace ts_order_011 prefetchsize 4096;
alter tablespace ts_order_012 prefetchsize 4096;
alter tablespace ts_order_013 prefetchsize 4096;
alter tablespace ts_order_014 prefetchsize 4096;
alter tablespace ts_order_015 prefetchsize 4096;
alter tablespace ts_order_016 prefetchsize 4096;
alter tablespace ts_order_017 prefetchsize 4096;
alter tablespace ts_order_018 prefetchsize 4096;
alter tablespace ts_order_019 prefetchsize 4096;
alter tablespace ts_order_020 prefetchsize 4096;
alter tablespace ts_order_021 prefetchsize 4096;
alter tablespace ts_order_022 prefetchsize 4096;
alter tablespace ts_order_023 prefetchsize 4096;
alter tablespace ts_order_024 prefetchsize 4096;
alter tablespace ts_order_025 prefetchsize 4096;
alter tablespace ts_order_026 prefetchsize 4096;
alter tablespace ts_order_027 prefetchsize 4096;
alter tablespace ts_order_028 prefetchsize 4096;
alter tablespace ts_order_029 prefetchsize 4096;
alter tablespace ts_order_030 prefetchsize 4096;
alter tablespace ts_order_031 prefetchsize 4096;
alter tablespace ts_order_032 prefetchsize 4096;
alter tablespace ts_order_033 prefetchsize 4096;
alter tablespace ts_order_034 prefetchsize 4096;
alter tablespace ts_order_035 prefetchsize 4096;
alter tablespace ts_order_036 prefetchsize 4096;
alter tablespace ts_order_037 prefetchsize 4096;
alter tablespace ts_order_038 prefetchsize 4096;
alter tablespace ts_order_039 prefetchsize 4096;
alter tablespace ts_order_040 prefetchsize 4096;
alter tablespace ts_order_041 prefetchsize 4096;
alter tablespace ts_order_042 prefetchsize 4096;
alter tablespace ts_order_043 prefetchsize 4096;
alter tablespace ts_order_044 prefetchsize 4096;
alter tablespace ts_order_045 prefetchsize 4096;
alter tablespace ts_order_046 prefetchsize 4096;
alter tablespace ts_order_047 prefetchsize 4096;
alter tablespace ts_order_048 prefetchsize 4096;
alter tablespace ts_order_049 prefetchsize 4096;
alter tablespace ts_order_050 prefetchsize 4096;
alter tablespace ts_order_051 prefetchsize 4096;
alter tablespace ts_order_052 prefetchsize 4096;
alter tablespace ts_order_053 prefetchsize 4096;
alter tablespace ts_order_054 prefetchsize 4096;
alter tablespace ts_order_055 prefetchsize 4096;
alter tablespace ts_order_056 prefetchsize 4096;
alter tablespace ts_order_057 prefetchsize 4096;
alter tablespace ts_order_058 prefetchsize 4096;
alter tablespace ts_order_059 prefetchsize 4096;
alter tablespace ts_order_060 prefetchsize 4096;
alter tablespace ts_order_061 prefetchsize 4096;
alter tablespace ts_order_062 prefetchsize 4096;
alter tablespace ts_order_063 prefetchsize 4096;
alter tablespace ts_order_064 prefetchsize 4096;
alter tablespace ts_orderline_001 prefetchsize 4096;
alter tablespace ts_orderline_002 prefetchsize 4096;
alter tablespace ts_orderline_003 prefetchsize 4096;
alter tablespace ts_orderline_004 prefetchsize 4096;
alter tablespace ts_orderline_005 prefetchsize 4096;
alter tablespace ts_orderline_006 prefetchsize 4096;
alter tablespace ts_orderline_007 prefetchsize 4096;
alter tablespace ts_orderline_008 prefetchsize 4096;
alter tablespace ts_orderline_009 prefetchsize 4096;
alter tablespace ts_orderline_010 prefetchsize 4096;
alter tablespace ts_orderline_011 prefetchsize 4096;
alter tablespace ts_orderline_012 prefetchsize 4096;
alter tablespace ts_orderline_013 prefetchsize 4096;
alter tablespace ts_orderline_014 prefetchsize 4096;
alter tablespace ts_orderline_015 prefetchsize 4096;
alter tablespace ts_orderline_016 prefetchsize 4096;
alter tablespace ts_orderline_017 prefetchsize 4096;
alter tablespace ts_orderline_018 prefetchsize 4096;
alter tablespace ts_orderline_019 prefetchsize 4096;
alter tablespace ts_orderline_020 prefetchsize 4096;
alter tablespace ts_orderline_021 prefetchsize 4096;
alter tablespace ts_orderline_022 prefetchsize 4096;
alter tablespace ts_orderline_023 prefetchsize 4096;
alter tablespace ts_orderline_024 prefetchsize 4096;
alter tablespace ts_orderline_025 prefetchsize 4096;
alter tablespace ts_orderline_026 prefetchsize 4096;
alter tablespace ts_orderline_027 prefetchsize 4096;
alter tablespace ts_orderline_028 prefetchsize 4096;
alter tablespace ts_orderline_029 prefetchsize 4096;
alter tablespace ts_orderline_030 prefetchsize 4096;
alter tablespace ts_orderline_031 prefetchsize 4096;
alter tablespace ts_orderline_032 prefetchsize 4096;
alter tablespace ts_orderline_033 prefetchsize 4096;
alter tablespace ts_orderline_034 prefetchsize 4096;
alter tablespace ts_orderline_035 prefetchsize 4096;
alter tablespace ts_orderline_036 prefetchsize 4096;
alter tablespace ts_orderline_037 prefetchsize 4096;
alter tablespace ts_orderline_038 prefetchsize 4096;
alter tablespace ts_orderline_039 prefetchsize 4096;
alter tablespace ts_orderline_040 prefetchsize 4096;
alter tablespace ts_orderline_041 prefetchsize 4096;
alter tablespace ts_orderline_042 prefetchsize 4096;
alter tablespace ts_orderline_043 prefetchsize 4096;
alter tablespace ts_orderline_044 prefetchsize 4096;
alter tablespace ts_orderline_045 prefetchsize 4096;
alter tablespace ts_orderline_046 prefetchsize 4096;
alter tablespace ts_orderline_047 prefetchsize 4096;
alter tablespace ts_orderline_048 prefetchsize 4096;
alter tablespace ts_orderline_049 prefetchsize 4096;
alter tablespace ts_orderline_050 prefetchsize 4096;
alter tablespace ts_orderline_051 prefetchsize 4096;
alter tablespace ts_orderline_052 prefetchsize 4096;
alter tablespace ts_orderline_053 prefetchsize 4096;
alter tablespace ts_orderline_054 prefetchsize 4096;
alter tablespace ts_orderline_055 prefetchsize 4096;
alter tablespace ts_orderline_056 prefetchsize 4096;
alter tablespace ts_orderline_057 prefetchsize 4096;
alter tablespace ts_orderline_058 prefetchsize 4096;
alter tablespace ts_orderline_059 prefetchsize 4096;
alter tablespace ts_orderline_060 prefetchsize 4096;
alter tablespace ts_orderline_061 prefetchsize 4096;
alter tablespace ts_orderline_062 prefetchsize 4096;
alter tablespace ts_orderline_063 prefetchsize 4096;
alter tablespace ts_orderline_064 prefetchsize 4096;
alter tablespace ts_stock_001 prefetchsize 4096;
alter tablespace ts_stock_002 prefetchsize 4096;
alter tablespace ts_stock_003 prefetchsize 4096;
alter tablespace ts_stock_004 prefetchsize 4096;
alter tablespace ts_stock_005 prefetchsize 4096;
alter tablespace ts_stock_006 prefetchsize 4096;
alter tablespace ts_stock_007 prefetchsize 4096;
alter tablespace ts_stock_008 prefetchsize 4096;
alter tablespace ts_stock_009 prefetchsize 4096;
alter tablespace ts_stock_010 prefetchsize 4096;
alter tablespace ts_stock_011 prefetchsize 4096;
alter tablespace ts_stock_012 prefetchsize 4096;
alter tablespace ts_stock_013 prefetchsize 4096;
alter tablespace ts_stock_014 prefetchsize 4096;
alter tablespace ts_stock_015 prefetchsize 4096;
alter tablespace ts_stock_016 prefetchsize 4096;
alter tablespace ts_stock_017 prefetchsize 4096;
alter tablespace ts_stock_018 prefetchsize 4096;
alter tablespace ts_stock_019 prefetchsize 4096;
alter tablespace ts_stock_020 prefetchsize 4096;
alter tablespace ts_stock_021 prefetchsize 4096;
alter tablespace ts_stock_022 prefetchsize 4096;
alter tablespace ts_stock_023 prefetchsize 4096;
alter tablespace ts_stock_024 prefetchsize 4096;
alter tablespace ts_stock_025 prefetchsize 4096;
alter tablespace ts_stock_026 prefetchsize 4096;
alter tablespace ts_stock_027 prefetchsize 4096;
alter tablespace ts_stock_028 prefetchsize 4096;
alter tablespace ts_stock_029 prefetchsize 4096;
alter tablespace ts_stock_030 prefetchsize 4096;
alter tablespace ts_stock_031 prefetchsize 4096;
alter tablespace ts_stock_032 prefetchsize 4096;
alter tablespace ts_stock_033 prefetchsize 4096;
alter tablespace ts_stock_034 prefetchsize 4096;
alter tablespace ts_stock_035 prefetchsize 4096;
alter tablespace ts_stock_036 prefetchsize 4096;
alter tablespace ts_stock_037 prefetchsize 4096;
alter tablespace ts_stock_038 prefetchsize 4096;
alter tablespace ts_stock_039 prefetchsize 4096;
alter tablespace ts_stock_040 prefetchsize 4096;
alter tablespace ts_stock_041 prefetchsize 4096;
alter tablespace ts_stock_042 prefetchsize 4096;
alter tablespace ts_stock_043 prefetchsize 4096;
alter tablespace ts_stock_044 prefetchsize 4096;
alter tablespace ts_stock_045 prefetchsize 4096;
alter tablespace ts_stock_046 prefetchsize 4096;
alter tablespace ts_stock_047 prefetchsize 4096;
alter tablespace ts_stock_048 prefetchsize 4096;
alter tablespace ts_stock_049 prefetchsize 4096;
alter tablespace ts_stock_050 prefetchsize 4096;
alter tablespace ts_stock_051 prefetchsize 4096;
alter tablespace ts_stock_052 prefetchsize 4096;
alter tablespace ts_stock_053 prefetchsize 4096;
alter tablespace ts_stock_054 prefetchsize 4096;
alter tablespace ts_stock_055 prefetchsize 4096;
alter tablespace ts_stock_056 prefetchsize 4096;
alter tablespace ts_stock_057 prefetchsize 4096;
alter tablespace ts_stock_058 prefetchsize 4096;
alter tablespace ts_stock_059 prefetchsize 4096;
alter tablespace ts_stock_060 prefetchsize 4096;
alter tablespace ts_stock_061 prefetchsize 4096;
alter tablespace ts_stock_062 prefetchsize 4096;
alter tablespace ts_stock_063 prefetchsize 4096;
alter tablespace ts_stock_064 prefetchsize 4096;
alter tablespace ts_ware_001 prefetchsize 4096;
alter tablespace ts_ware_002 prefetchsize 4096;
alter tablespace ts_ware_003 prefetchsize 4096;
alter tablespace ts_ware_004 prefetchsize 4096;
alter tablespace ts_ware_005 prefetchsize 4096;
alter tablespace ts_ware_006 prefetchsize 4096;
alter tablespace ts_ware_007 prefetchsize 4096;
alter tablespace ts_ware_008 prefetchsize 4096;
alter tablespace ts_ware_009 prefetchsize 4096;
alter tablespace ts_ware_010 prefetchsize 4096;
alter tablespace ts_ware_011 prefetchsize 4096;
```

```
alter tablespace ts_ware_012 prefetchsize 4096;
alter tablespace ts_ware_013 prefetchsize 4096;
alter tablespace ts_ware_014 prefetchsize 4096;
alter tablespace ts_ware_015 prefetchsize 4096;
alter tablespace ts_ware_016 prefetchsize 4096;
alter tablespace ts_ware_017 prefetchsize 4096;
alter tablespace ts_ware_018 prefetchsize 4096;
alter tablespace ts_ware_019 prefetchsize 4096;
alter tablespace ts_ware_020 prefetchsize 4096;
alter tablespace ts_ware_021 prefetchsize 4096;
alter tablespace ts_ware_022 prefetchsize 4096;
alter tablespace ts_ware_023 prefetchsize 4096;
alter tablespace ts_ware_024 prefetchsize 4096;
alter tablespace ts_ware_025 prefetchsize 4096;
alter tablespace ts_ware_026 prefetchsize 4096;
alter tablespace ts_ware_027 prefetchsize 4096;
alter tablespace ts_ware_028 prefetchsize 4096;
alter tablespace ts_ware_029 prefetchsize 4096;
alter tablespace ts_ware_030 prefetchsize 4096;
alter tablespace ts_ware_031 prefetchsize 4096;
alter tablespace ts_ware_032 prefetchsize 4096;
connect reset;
```

DDL/CRCONST_CUSTOMER.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER1 OFF;
ALTER TABLE CUSTOMER1 DROP CONSTRAINT CUSTOMER1CKC;
ALTER TABLE CUSTOMER1 ADD CONSTRAINT CUSTOMER1CKC CHECK (C_W_ID BETWEEN 1
AND 812);
SET INTEGRITY FOR CUSTOMER1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER2 OFF;
ALTER TABLE CUSTOMER2 DROP CONSTRAINT CUSTOMER2CKC;
ALTER TABLE CUSTOMER2 ADD CONSTRAINT CUSTOMER2CKC CHECK (C_W_ID BETWEEN 813
AND 1624);
SET INTEGRITY FOR CUSTOMER2 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER3 OFF;
ALTER TABLE CUSTOMER3 DROP CONSTRAINT CUSTOMER3CKC;
ALTER TABLE CUSTOMER3 ADD CONSTRAINT CUSTOMER3CKC CHECK (C_W_ID BETWEEN 1625
AND 2436);
SET INTEGRITY FOR CUSTOMER3 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER4 OFF;
ALTER TABLE CUSTOMER4 DROP CONSTRAINT CUSTOMER4CKC;
ALTER TABLE CUSTOMER4 ADD CONSTRAINT CUSTOMER4CKC CHECK (C_W_ID BETWEEN 2437
AND 3248);
SET INTEGRITY FOR CUSTOMER4 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER5 OFF;
ALTER TABLE CUSTOMER5 DROP CONSTRAINT CUSTOMER5CKC;
ALTER TABLE CUSTOMER5 ADD CONSTRAINT CUSTOMER5CKC CHECK (C_W_ID BETWEEN 3249
AND 4060);
SET INTEGRITY FOR CUSTOMER5 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER6 OFF;
ALTER TABLE CUSTOMER6 DROP CONSTRAINT CUSTOMER6CKC;
ALTER TABLE CUSTOMER6 ADD CONSTRAINT CUSTOMER6CKC CHECK (C_W_ID BETWEEN 4061
AND 4872);
SET INTEGRITY FOR CUSTOMER6 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER7 OFF;
ALTER TABLE CUSTOMER7 DROP CONSTRAINT CUSTOMER7CKC;
ALTER TABLE CUSTOMER7 ADD CONSTRAINT CUSTOMER7CKC CHECK (C_W_ID BETWEEN 4873
AND 5684);
SET INTEGRITY FOR CUSTOMER7 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER8 OFF;
ALTER TABLE CUSTOMER8 DROP CONSTRAINT CUSTOMER8CKC;
ALTER TABLE CUSTOMER8 ADD CONSTRAINT CUSTOMER8CKC CHECK (C_W_ID BETWEEN 5685
AND 6496);
SET INTEGRITY FOR CUSTOMER8 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER9 OFF;
ALTER TABLE CUSTOMER9 DROP CONSTRAINT CUSTOMER9CKC;
ALTER TABLE CUSTOMER9 ADD CONSTRAINT CUSTOMER9CKC CHECK (C_W_ID BETWEEN 6497
AND 7308);
SET INTEGRITY FOR CUSTOMER9 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER10 OFF;
ALTER TABLE CUSTOMER10 DROP CONSTRAINT CUSTOMER10CKC;
ALTER TABLE CUSTOMER10 ADD CONSTRAINT CUSTOMER10CKC CHECK (C_W_ID BETWEEN
7309 AND 8120);
SET INTEGRITY FOR CUSTOMER10 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER11 OFF;
ALTER TABLE CUSTOMER11 DROP CONSTRAINT CUSTOMER11CKC;
ALTER TABLE CUSTOMER11 ADD CONSTRAINT CUSTOMER11CKC CHECK (C_W_ID BETWEEN
8121 AND 8932);
SET INTEGRITY FOR CUSTOMER11 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER12 OFF;
ALTER TABLE CUSTOMER12 DROP CONSTRAINT CUSTOMER12CKC;
ALTER TABLE CUSTOMER12 ADD CONSTRAINT CUSTOMER12CKC CHECK (C_W_ID BETWEEN
8933 AND 9744);
SET INTEGRITY FOR CUSTOMER12 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER13 OFF;
ALTER TABLE CUSTOMER13 DROP CONSTRAINT CUSTOMER13CKC;
ALTER TABLE CUSTOMER13 ADD CONSTRAINT CUSTOMER13CKC CHECK (C_W_ID BETWEEN
9745 AND 10556);
SET INTEGRITY FOR CUSTOMER13 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER14 OFF;
ALTER TABLE CUSTOMER14 DROP CONSTRAINT CUSTOMER14CKC;
ALTER TABLE CUSTOMER14 ADD CONSTRAINT CUSTOMER14CKC CHECK (C_W_ID BETWEEN
10557 AND 11368);
SET INTEGRITY FOR CUSTOMER14 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER15 OFF;
ALTER TABLE CUSTOMER15 DROP CONSTRAINT CUSTOMER15CKC;
ALTER TABLE CUSTOMER15 ADD CONSTRAINT CUSTOMER15CKC CHECK (C_W_ID BETWEEN
11369 AND 12180);
SET INTEGRITY FOR CUSTOMER15 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER16 OFF;
ALTER TABLE CUSTOMER16 DROP CONSTRAINT CUSTOMER16CKC;
ALTER TABLE CUSTOMER16 ADD CONSTRAINT CUSTOMER16CKC CHECK (C_W_ID BETWEEN
12181 AND 12992);
SET INTEGRITY FOR CUSTOMER16 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER17 OFF;
ALTER TABLE CUSTOMER17 DROP CONSTRAINT CUSTOMER17CKC;
ALTER TABLE CUSTOMER17 ADD CONSTRAINT CUSTOMER17CKC CHECK (C_W_ID BETWEEN
12993 AND 13804);
SET INTEGRITY FOR CUSTOMER17 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER18 OFF;
ALTER TABLE CUSTOMER18 DROP CONSTRAINT CUSTOMER18CKC;
ALTER TABLE CUSTOMER18 ADD CONSTRAINT CUSTOMER18CKC CHECK (C_W_ID BETWEEN
13805 AND 14616);
SET INTEGRITY FOR CUSTOMER18 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER19 OFF;
ALTER TABLE CUSTOMER19 DROP CONSTRAINT CUSTOMER19CKC;
ALTER TABLE CUSTOMER19 ADD CONSTRAINT CUSTOMER19CKC CHECK (C_W_ID BETWEEN
14617 AND 15428);
SET INTEGRITY FOR CUSTOMER19 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER20 OFF;
ALTER TABLE CUSTOMER20 DROP CONSTRAINT CUSTOMER20CKC;
ALTER TABLE CUSTOMER20 ADD CONSTRAINT CUSTOMER20CKC CHECK (C_W_ID BETWEEN
15429 AND 16240);
SET INTEGRITY FOR CUSTOMER20 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER21 OFF;
ALTER TABLE CUSTOMER21 DROP CONSTRAINT CUSTOMER21CKC;
ALTER TABLE CUSTOMER21 ADD CONSTRAINT CUSTOMER21CKC CHECK (C_W_ID BETWEEN
16241 AND 17052);
SET INTEGRITY FOR CUSTOMER21 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER22 OFF;
ALTER TABLE CUSTOMER22 DROP CONSTRAINT CUSTOMER22CKC;
ALTER TABLE CUSTOMER22 ADD CONSTRAINT CUSTOMER22CKC CHECK (C_W_ID BETWEEN
17053 AND 17864);
SET INTEGRITY FOR CUSTOMER22 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER23 OFF;
ALTER TABLE CUSTOMER23 DROP CONSTRAINT CUSTOMER23CKC;
ALTER TABLE CUSTOMER23 ADD CONSTRAINT CUSTOMER23CKC CHECK (C_W_ID BETWEEN
17865 AND 18676);
SET INTEGRITY FOR CUSTOMER23 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER24 OFF;
ALTER TABLE CUSTOMER24 DROP CONSTRAINT CUSTOMER24CKC;
ALTER TABLE CUSTOMER24 ADD CONSTRAINT CUSTOMER24CKC CHECK (C_W_ID BETWEEN
18677 AND 19488);
SET INTEGRITY FOR CUSTOMER24 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER25 OFF;
ALTER TABLE CUSTOMER25 DROP CONSTRAINT CUSTOMER25CKC;
ALTER TABLE CUSTOMER25 ADD CONSTRAINT CUSTOMER25CKC CHECK (C_W_ID BETWEEN
19489 AND 20300);
SET INTEGRITY FOR CUSTOMER25 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER26 OFF;
ALTER TABLE CUSTOMER26 DROP CONSTRAINT CUSTOMER26CKC;
ALTER TABLE CUSTOMER26 ADD CONSTRAINT CUSTOMER26CKC CHECK (C_W_ID BETWEEN
20301 AND 21112);
SET INTEGRITY FOR CUSTOMER26 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER27 OFF;
ALTER TABLE CUSTOMER27 DROP CONSTRAINT CUSTOMER27CKC;
ALTER TABLE CUSTOMER27 ADD CONSTRAINT CUSTOMER27CKC CHECK (C_W_ID BETWEEN
21113 AND 21924);
SET INTEGRITY FOR CUSTOMER27 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
```

```
SET INTEGRITY FOR CUSTOMER28 OFF;
ALTER TABLE CUSTOMER28 DROP CONSTRAINT CUSTOMER28CKC;
ALTER TABLE CUSTOMER28 ADD CONSTRAINT CUSTOMER28CKC CHECK (C_W_ID BETWEEN
21925 AND 22736);
SET INTEGRITY FOR CUSTOMER28 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER29 OFF;
ALTER TABLE CUSTOMER29 DROP CONSTRAINT CUSTOMER29CKC;
ALTER TABLE CUSTOMER29 ADD CONSTRAINT CUSTOMER29CKC CHECK (C_W_ID BETWEEN
22737 AND 23548);
SET INTEGRITY FOR CUSTOMER29 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER30 OFF;
ALTER TABLE CUSTOMER30 DROP CONSTRAINT CUSTOMER30CKC;
ALTER TABLE CUSTOMER30 ADD CONSTRAINT CUSTOMER30CKC CHECK (C_W_ID BETWEEN
23549 AND 24360);
SET INTEGRITY FOR CUSTOMER30 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER31 OFF;
ALTER TABLE CUSTOMER31 DROP CONSTRAINT CUSTOMER31CKC;
ALTER TABLE CUSTOMER31 ADD CONSTRAINT CUSTOMER31CKC CHECK (C_W_ID BETWEEN
24361 AND 25172);
SET INTEGRITY FOR CUSTOMER31 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER32 OFF;
ALTER TABLE CUSTOMER32 DROP CONSTRAINT CUSTOMER32CKC;
ALTER TABLE CUSTOMER32 ADD CONSTRAINT CUSTOMER32CKC CHECK (C_W_ID BETWEEN
25173 AND 25984);
SET INTEGRITY FOR CUSTOMER32 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER33 OFF;
ALTER TABLE CUSTOMER33 DROP CONSTRAINT CUSTOMER33CKC;
ALTER TABLE CUSTOMER33 ADD CONSTRAINT CUSTOMER33CKC CHECK (C_W_ID BETWEEN
25985 AND 26796);
SET INTEGRITY FOR CUSTOMER33 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER34 OFF;
ALTER TABLE CUSTOMER34 DROP CONSTRAINT CUSTOMER34CKC;
ALTER TABLE CUSTOMER34 ADD CONSTRAINT CUSTOMER34CKC CHECK (C_W_ID BETWEEN
26797 AND 27608);
SET INTEGRITY FOR CUSTOMER34 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER35 OFF;
ALTER TABLE CUSTOMER35 DROP CONSTRAINT CUSTOMER35CKC;
ALTER TABLE CUSTOMER35 ADD CONSTRAINT CUSTOMER35CKC CHECK (C_W_ID BETWEEN
27609 AND 28420);
SET INTEGRITY FOR CUSTOMER35 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER36 OFF;
ALTER TABLE CUSTOMER36 DROP CONSTRAINT CUSTOMER36CKC;
ALTER TABLE CUSTOMER36 ADD CONSTRAINT CUSTOMER36CKC CHECK (C_W_ID BETWEEN
28421 AND 29232);
SET INTEGRITY FOR CUSTOMER36 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER37 OFF;
ALTER TABLE CUSTOMER37 DROP CONSTRAINT CUSTOMER37CKC;
ALTER TABLE CUSTOMER37 ADD CONSTRAINT CUSTOMER37CKC CHECK (C_W_ID BETWEEN
29233 AND 30044);
SET INTEGRITY FOR CUSTOMER37 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER38 OFF;
ALTER TABLE CUSTOMER38 DROP CONSTRAINT CUSTOMER38CKC;
ALTER TABLE CUSTOMER38 ADD CONSTRAINT CUSTOMER38CKC CHECK (C_W_ID BETWEEN
30045 AND 30856);
SET INTEGRITY FOR CUSTOMER38 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER39 OFF;
ALTER TABLE CUSTOMER39 DROP CONSTRAINT CUSTOMER39CKC;
ALTER TABLE CUSTOMER39 ADD CONSTRAINT CUSTOMER39CKC CHECK (C_W_ID BETWEEN
30857 AND 31668);
SET INTEGRITY FOR CUSTOMER39 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER40 OFF;
ALTER TABLE CUSTOMER40 DROP CONSTRAINT CUSTOMER40CKC;
ALTER TABLE CUSTOMER40 ADD CONSTRAINT CUSTOMER40CKC CHECK (C_W_ID BETWEEN
31669 AND 32480);
SET INTEGRITY FOR CUSTOMER40 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER41 OFF;
ALTER TABLE CUSTOMER41 DROP CONSTRAINT CUSTOMER41CKC;
ALTER TABLE CUSTOMER41 ADD CONSTRAINT CUSTOMER41CKC CHECK (C_W_ID BETWEEN
32481 AND 33292);
SET INTEGRITY FOR CUSTOMER41 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER42 OFF;
ALTER TABLE CUSTOMER42 DROP CONSTRAINT CUSTOMER42CKC;
ALTER TABLE CUSTOMER42 ADD CONSTRAINT CUSTOMER42CKC CHECK (C_W_ID BETWEEN
33293 AND 34104);
SET INTEGRITY FOR CUSTOMER42 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER43 OFF;
ALTER TABLE CUSTOMER43 DROP CONSTRAINT CUSTOMER43CKC;
ALTER TABLE CUSTOMER43 ADD CONSTRAINT CUSTOMER43CKC CHECK (C_W_ID BETWEEN
34105 AND 34916);
SET INTEGRITY FOR CUSTOMER43 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER44 OFF;
ALTER TABLE CUSTOMER44 DROP CONSTRAINT CUSTOMER44CKC;
ALTER TABLE CUSTOMER44 ADD CONSTRAINT CUSTOMER44CKC CHECK (C_W_ID BETWEEN
34917 AND 35728);
SET INTEGRITY FOR CUSTOMER44 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER45 OFF;
ALTER TABLE CUSTOMER45 DROP CONSTRAINT CUSTOMER45CKC;
ALTER TABLE CUSTOMER45 ADD CONSTRAINT CUSTOMER45CKC CHECK (C_W_ID BETWEEN
35729 AND 36540);
SET INTEGRITY FOR CUSTOMER45 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER46 OFF;
ALTER TABLE CUSTOMER46 DROP CONSTRAINT CUSTOMER46CKC;
ALTER TABLE CUSTOMER46 ADD CONSTRAINT CUSTOMER46CKC CHECK (C_W_ID BETWEEN
36541 AND 37352);
SET INTEGRITY FOR CUSTOMER46 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER47 OFF;
ALTER TABLE CUSTOMER47 DROP CONSTRAINT CUSTOMER47CKC;
ALTER TABLE CUSTOMER47 ADD CONSTRAINT CUSTOMER47CKC CHECK (C_W_ID BETWEEN
37353 AND 38164);
SET INTEGRITY FOR CUSTOMER47 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER48 OFF;
ALTER TABLE CUSTOMER48 DROP CONSTRAINT CUSTOMER48CKC;
ALTER TABLE CUSTOMER48 ADD CONSTRAINT CUSTOMER48CKC CHECK (C_W_ID BETWEEN
38165 AND 38976);
SET INTEGRITY FOR CUSTOMER48 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER49 OFF;
ALTER TABLE CUSTOMER49 DROP CONSTRAINT CUSTOMER49CKC;
ALTER TABLE CUSTOMER49 ADD CONSTRAINT CUSTOMER49CKC CHECK (C_W_ID BETWEEN
38977 AND 39788);
SET INTEGRITY FOR CUSTOMER49 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER50 OFF;
ALTER TABLE CUSTOMER50 DROP CONSTRAINT CUSTOMER50CKC;
ALTER TABLE CUSTOMER50 ADD CONSTRAINT CUSTOMER50CKC CHECK (C_W_ID BETWEEN
39789 AND 40600);
SET INTEGRITY FOR CUSTOMER50 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER51 OFF;
ALTER TABLE CUSTOMER51 DROP CONSTRAINT CUSTOMER51CKC;
ALTER TABLE CUSTOMER51 ADD CONSTRAINT CUSTOMER51CKC CHECK (C_W_ID BETWEEN
40601 AND 41412);
SET INTEGRITY FOR CUSTOMER51 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER52 OFF;
ALTER TABLE CUSTOMER52 DROP CONSTRAINT CUSTOMER52CKC;
ALTER TABLE CUSTOMER52 ADD CONSTRAINT CUSTOMER52CKC CHECK (C_W_ID BETWEEN
41413 AND 42224);
SET INTEGRITY FOR CUSTOMER52 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER53 OFF;
ALTER TABLE CUSTOMER53 DROP CONSTRAINT CUSTOMER53CKC;
ALTER TABLE CUSTOMER53 ADD CONSTRAINT CUSTOMER53CKC CHECK (C_W_ID BETWEEN
42225 AND 43036);
SET INTEGRITY FOR CUSTOMER53 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER54 OFF;
ALTER TABLE CUSTOMER54 DROP CONSTRAINT CUSTOMER54CKC;
ALTER TABLE CUSTOMER54 ADD CONSTRAINT CUSTOMER54CKC CHECK (C_W_ID BETWEEN
43037 AND 43848);
SET INTEGRITY FOR CUSTOMER54 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER55 OFF;
ALTER TABLE CUSTOMER55 DROP CONSTRAINT CUSTOMER55CKC;
ALTER TABLE CUSTOMER55 ADD CONSTRAINT CUSTOMER55CKC CHECK (C_W_ID BETWEEN
43849 AND 44660);
SET INTEGRITY FOR CUSTOMER55 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER56 OFF;
ALTER TABLE CUSTOMER56 DROP CONSTRAINT CUSTOMER56CKC;
ALTER TABLE CUSTOMER56 ADD CONSTRAINT CUSTOMER56CKC CHECK (C_W_ID BETWEEN
44661 AND 45472);
SET INTEGRITY FOR CUSTOMER56 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER57 OFF;
ALTER TABLE CUSTOMER57 DROP CONSTRAINT CUSTOMER57CKC;
ALTER TABLE CUSTOMER57 ADD CONSTRAINT CUSTOMER57CKC CHECK (C_W_ID BETWEEN
45473 AND 46284);
SET INTEGRITY FOR CUSTOMER57 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER58 OFF;
ALTER TABLE CUSTOMER58 DROP CONSTRAINT CUSTOMER58CKC;
ALTER TABLE CUSTOMER58 ADD CONSTRAINT CUSTOMER58CKC CHECK (C_W_ID BETWEEN
46285 AND 47096);
SET INTEGRITY FOR CUSTOMER58 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER59 OFF;
```

```
ALTER TABLE CUSTOMER59 DROP CONSTRAINT CUSTOMER59CKC;
ALTER TABLE CUSTOMER59 ADD CONSTRAINT CUSTOMER59CKC CHECK (C_W_ID BETWEEN 47097 AND 47908);
SET INTEGRITY FOR CUSTOMER59 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER60 OFF;
ALTER TABLE CUSTOMER60 DROP CONSTRAINT CUSTOMER60CKC;
ALTER TABLE CUSTOMER60 ADD CONSTRAINT CUSTOMER60CKC CHECK (C_W_ID BETWEEN 47909 AND 48720);
SET INTEGRITY FOR CUSTOMER60 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER61 OFF;
ALTER TABLE CUSTOMER61 DROP CONSTRAINT CUSTOMER61CKC;
ALTER TABLE CUSTOMER61 ADD CONSTRAINT CUSTOMER61CKC CHECK (C_W_ID BETWEEN 48721 AND 49532);
SET INTEGRITY FOR CUSTOMER61 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER62 OFF;
ALTER TABLE CUSTOMER62 DROP CONSTRAINT CUSTOMER62CKC;
ALTER TABLE CUSTOMER62 ADD CONSTRAINT CUSTOMER62CKC CHECK (C_W_ID BETWEEN 49533 AND 50344);
SET INTEGRITY FOR CUSTOMER62 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER63 OFF;
ALTER TABLE CUSTOMER63 DROP CONSTRAINT CUSTOMER63CKC;
ALTER TABLE CUSTOMER63 ADD CONSTRAINT CUSTOMER63CKC CHECK (C_W_ID BETWEEN 50345 AND 51156);
SET INTEGRITY FOR CUSTOMER63 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER64 OFF;
ALTER TABLE CUSTOMER64 DROP CONSTRAINT CUSTOMER64CKC;
ALTER TABLE CUSTOMER64 ADD CONSTRAINT CUSTOMER64CKC CHECK (C_W_ID >= 51157);
SET INTEGRITY FOR CUSTOMER64 ALL IMMEDIATE UNCHECKED;
connect reset;
```

DDL/CRCONST_DISTRICT.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT1 OFF;
ALTER TABLE DISTRICT1 DROP CONSTRAINT DISTRICT1CKC;
ALTER TABLE DISTRICT1 ADD CONSTRAINT DISTRICT1CKC CHECK (D_W_ID BETWEEN 1 AND 1624);
SET INTEGRITY FOR DISTRICT1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT2 OFF;
ALTER TABLE DISTRICT2 DROP CONSTRAINT DISTRICT2CKC;
ALTER TABLE DISTRICT2 ADD CONSTRAINT DISTRICT2CKC CHECK (D_W_ID BETWEEN 1625 AND 3248);
SET INTEGRITY FOR DISTRICT2 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT3 OFF;
ALTER TABLE DISTRICT3 DROP CONSTRAINT DISTRICT3CKC;
ALTER TABLE DISTRICT3 ADD CONSTRAINT DISTRICT3CKC CHECK (D_W_ID BETWEEN 3249 AND 4872);
SET INTEGRITY FOR DISTRICT3 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT4 OFF;
ALTER TABLE DISTRICT4 DROP CONSTRAINT DISTRICT4CKC;
ALTER TABLE DISTRICT4 ADD CONSTRAINT DISTRICT4CKC CHECK (D_W_ID BETWEEN 4873 AND 6496);
SET INTEGRITY FOR DISTRICT4 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT5 OFF;
ALTER TABLE DISTRICT5 DROP CONSTRAINT DISTRICT5CKC;
ALTER TABLE DISTRICT5 ADD CONSTRAINT DISTRICT5CKC CHECK (D_W_ID BETWEEN 6497 AND 8120);
SET INTEGRITY FOR DISTRICT5 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT6 OFF;
ALTER TABLE DISTRICT6 DROP CONSTRAINT DISTRICT6CKC;
ALTER TABLE DISTRICT6 ADD CONSTRAINT DISTRICT6CKC CHECK (D_W_ID BETWEEN 8121 AND 9744);
SET INTEGRITY FOR DISTRICT6 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT7 OFF;
ALTER TABLE DISTRICT7 DROP CONSTRAINT DISTRICT7CKC;
ALTER TABLE DISTRICT7 ADD CONSTRAINT DISTRICT7CKC CHECK (D_W_ID BETWEEN 9745 AND 11368);
SET INTEGRITY FOR DISTRICT7 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT8 OFF;
ALTER TABLE DISTRICT8 DROP CONSTRAINT DISTRICT8CKC;
ALTER TABLE DISTRICT8 ADD CONSTRAINT DISTRICT8CKC CHECK (D_W_ID BETWEEN 11369 AND 12992);
SET INTEGRITY FOR DISTRICT8 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT9 OFF;
ALTER TABLE DISTRICT9 DROP CONSTRAINT DISTRICT9CKC;
ALTER TABLE DISTRICT9 ADD CONSTRAINT DISTRICT9CKC CHECK (D_W_ID BETWEEN 12993 AND 14616);
SET INTEGRITY FOR DISTRICT9 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT10 OFF;
ALTER TABLE DISTRICT10 DROP CONSTRAINT DISTRICT10CKC;
ALTER TABLE DISTRICT10 ADD CONSTRAINT DISTRICT10CKC CHECK (D_W_ID BETWEEN 14617 AND 16240);
SET INTEGRITY FOR DISTRICT10 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT11 OFF;
ALTER TABLE DISTRICT11 DROP CONSTRAINT DISTRICT11CKC;
ALTER TABLE DISTRICT11 ADD CONSTRAINT DISTRICT11CKC CHECK (D_W_ID BETWEEN 16241 AND 17864);
SET INTEGRITY FOR DISTRICT11 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT12 OFF;
ALTER TABLE DISTRICT12 DROP CONSTRAINT DISTRICT12CKC;
ALTER TABLE DISTRICT12 ADD CONSTRAINT DISTRICT12CKC CHECK (D_W_ID BETWEEN 17865 AND 19488);
SET INTEGRITY FOR DISTRICT12 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT13 OFF;
ALTER TABLE DISTRICT13 DROP CONSTRAINT DISTRICT13CKC;
ALTER TABLE DISTRICT13 ADD CONSTRAINT DISTRICT13CKC CHECK (D_W_ID BETWEEN 19489 AND 21112);
SET INTEGRITY FOR DISTRICT13 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT14 OFF;
ALTER TABLE DISTRICT14 DROP CONSTRAINT DISTRICT14CKC;
ALTER TABLE DISTRICT14 ADD CONSTRAINT DISTRICT14CKC CHECK (D_W_ID BETWEEN 21113 AND 22736);
SET INTEGRITY FOR DISTRICT14 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT15 OFF;
ALTER TABLE DISTRICT15 DROP CONSTRAINT DISTRICT15CKC;
ALTER TABLE DISTRICT15 ADD CONSTRAINT DISTRICT15CKC CHECK (D_W_ID BETWEEN 22737 AND 24360);
SET INTEGRITY FOR DISTRICT15 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT16 OFF;
ALTER TABLE DISTRICT16 DROP CONSTRAINT DISTRICT16CKC;
ALTER TABLE DISTRICT16 ADD CONSTRAINT DISTRICT16CKC CHECK (D_W_ID BETWEEN 24361 AND 25984);
SET INTEGRITY FOR DISTRICT16 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT17 OFF;
ALTER TABLE DISTRICT17 DROP CONSTRAINT DISTRICT17CKC;
ALTER TABLE DISTRICT17 ADD CONSTRAINT DISTRICT17CKC CHECK (D_W_ID BETWEEN 25985 AND 27608);
SET INTEGRITY FOR DISTRICT17 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT18 OFF;
ALTER TABLE DISTRICT18 DROP CONSTRAINT DISTRICT18CKC;
ALTER TABLE DISTRICT18 ADD CONSTRAINT DISTRICT18CKC CHECK (D_W_ID BETWEEN 27609 AND 29232);
SET INTEGRITY FOR DISTRICT18 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT19 OFF;
ALTER TABLE DISTRICT19 DROP CONSTRAINT DISTRICT19CKC;
ALTER TABLE DISTRICT19 ADD CONSTRAINT DISTRICT19CKC CHECK (D_W_ID BETWEEN 29233 AND 30856);
SET INTEGRITY FOR DISTRICT19 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT20 OFF;
ALTER TABLE DISTRICT20 DROP CONSTRAINT DISTRICT20CKC;
ALTER TABLE DISTRICT20 ADD CONSTRAINT DISTRICT20CKC CHECK (D_W_ID BETWEEN 30857 AND 32480);
SET INTEGRITY FOR DISTRICT20 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT21 OFF;
ALTER TABLE DISTRICT21 DROP CONSTRAINT DISTRICT21CKC;
ALTER TABLE DISTRICT21 ADD CONSTRAINT DISTRICT21CKC CHECK (D_W_ID BETWEEN 32481 AND 34104);
SET INTEGRITY FOR DISTRICT21 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT22 OFF;
ALTER TABLE DISTRICT22 DROP CONSTRAINT DISTRICT22CKC;
ALTER TABLE DISTRICT22 ADD CONSTRAINT DISTRICT22CKC CHECK (D_W_ID BETWEEN 34105 AND 35728);
SET INTEGRITY FOR DISTRICT22 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT23 OFF;
ALTER TABLE DISTRICT23 DROP CONSTRAINT DISTRICT23CKC;
ALTER TABLE DISTRICT23 ADD CONSTRAINT DISTRICT23CKC CHECK (D_W_ID BETWEEN 35729 AND 37352);
SET INTEGRITY FOR DISTRICT23 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT24 OFF;
ALTER TABLE DISTRICT24 DROP CONSTRAINT DISTRICT24CKC;
ALTER TABLE DISTRICT24 ADD CONSTRAINT DISTRICT24CKC CHECK (D_W_ID BETWEEN 37353 AND 38976);
SET INTEGRITY FOR DISTRICT24 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT25 OFF;
ALTER TABLE DISTRICT25 DROP CONSTRAINT DISTRICT25CKC;
ALTER TABLE DISTRICT25 ADD CONSTRAINT DISTRICT25CKC CHECK (D_W_ID BETWEEN 38977 AND 40600);
```

```
SET INTEGRITY FOR DISTRICT25 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT26 OFF;
ALTER TABLE DISTRICT26 DROP CONSTRAINT DISTRICT26CKC;
ALTER TABLE DISTRICT26 ADD CONSTRAINT DISTRICT26CKC CHECK (D_W_ID BETWEEN 40601
AND 42224);
SET INTEGRITY FOR DISTRICT26 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT27 OFF;
ALTER TABLE DISTRICT27 DROP CONSTRAINT DISTRICT27CKC;
ALTER TABLE DISTRICT27 ADD CONSTRAINT DISTRICT27CKC CHECK (D_W_ID BETWEEN 42225
AND 43848);
SET INTEGRITY FOR DISTRICT27 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT28 OFF;
ALTER TABLE DISTRICT28 DROP CONSTRAINT DISTRICT28CKC;
ALTER TABLE DISTRICT28 ADD CONSTRAINT DISTRICT28CKC CHECK (D_W_ID BETWEEN 43849
AND 45472);
SET INTEGRITY FOR DISTRICT28 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT29 OFF;
ALTER TABLE DISTRICT29 DROP CONSTRAINT DISTRICT29CKC;
ALTER TABLE DISTRICT29 ADD CONSTRAINT DISTRICT29CKC CHECK (D_W_ID BETWEEN 45473
AND 47096);
SET INTEGRITY FOR DISTRICT29 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT30 OFF;
ALTER TABLE DISTRICT30 DROP CONSTRAINT DISTRICT30CKC;
ALTER TABLE DISTRICT30 ADD CONSTRAINT DISTRICT30CKC CHECK (D_W_ID BETWEEN 47097
AND 48720);
SET INTEGRITY FOR DISTRICT30 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT31 OFF;
ALTER TABLE DISTRICT31 DROP CONSTRAINT DISTRICT31CKC;
ALTER TABLE DISTRICT31 ADD CONSTRAINT DISTRICT31CKC CHECK (D_W_ID BETWEEN 48721
AND 50344);
SET INTEGRITY FOR DISTRICT31 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT32 OFF;
ALTER TABLE DISTRICT32 DROP CONSTRAINT DISTRICT32CKC;
ALTER TABLE DISTRICT32 ADD CONSTRAINT DISTRICT32CKC CHECK (D_W_ID >= 50345);
SET INTEGRITY FOR DISTRICT32 ALL IMMEDIATE UNCHECKED;
connect reset;
```

DDL/CRCONST_HISTORY.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY1 OFF;
ALTER TABLE HISTORY1 DROP CONSTRAINT HISTORY1CKC;
ALTER TABLE HISTORY1 ADD CONSTRAINT HISTORY1CKC CHECK (H_W_ID BETWEEN 1 AND
1624);
SET INTEGRITY FOR HISTORY1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY2 OFF;
ALTER TABLE HISTORY2 DROP CONSTRAINT HISTORY2CKC;
ALTER TABLE HISTORY2 ADD CONSTRAINT HISTORY2CKC CHECK (H_W_ID BETWEEN 1625 AND
3248);
SET INTEGRITY FOR HISTORY2 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY3 OFF;
ALTER TABLE HISTORY3 DROP CONSTRAINT HISTORY3CKC;
ALTER TABLE HISTORY3 ADD CONSTRAINT HISTORY3CKC CHECK (H_W_ID BETWEEN 3249 AND
4872);
SET INTEGRITY FOR HISTORY3 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY4 OFF;
ALTER TABLE HISTORY4 DROP CONSTRAINT HISTORY4CKC;
ALTER TABLE HISTORY4 ADD CONSTRAINT HISTORY4CKC CHECK (H_W_ID BETWEEN 4873 AND
6496);
SET INTEGRITY FOR HISTORY4 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY5 OFF;
ALTER TABLE HISTORY5 DROP CONSTRAINT HISTORY5CKC;
ALTER TABLE HISTORY5 ADD CONSTRAINT HISTORY5CKC CHECK (H_W_ID BETWEEN 6497 AND
8120);
SET INTEGRITY FOR HISTORY5 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY6 OFF;
ALTER TABLE HISTORY6 DROP CONSTRAINT HISTORY6CKC;
ALTER TABLE HISTORY6 ADD CONSTRAINT HISTORY6CKC CHECK (H_W_ID BETWEEN 8121 AND
9744);
SET INTEGRITY FOR HISTORY6 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY7 OFF;
ALTER TABLE HISTORY7 DROP CONSTRAINT HISTORY7CKC;
ALTER TABLE HISTORY7 ADD CONSTRAINT HISTORY7CKC CHECK (H_W_ID BETWEEN 9745 AND
11368);
SET INTEGRITY FOR HISTORY7 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY8 OFF;
ALTER TABLE HISTORY8 DROP CONSTRAINT HISTORY8CKC;
ALTER TABLE HISTORY8 ADD CONSTRAINT HISTORY8CKC CHECK (H_W_ID BETWEEN 11369
AND 12992);
SET INTEGRITY FOR HISTORY8 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY9 OFF;
ALTER TABLE HISTORY9 DROP CONSTRAINT HISTORY9CKC;
ALTER TABLE HISTORY9 ADD CONSTRAINT HISTORY9CKC CHECK (H_W_ID BETWEEN 12993
AND 14616);
SET INTEGRITY FOR HISTORY9 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY10 OFF;
ALTER TABLE HISTORY10 DROP CONSTRAINT HISTORY10CKC;
ALTER TABLE HISTORY10 ADD CONSTRAINT HISTORY10CKC CHECK (H_W_ID BETWEEN 14617
AND 16240);
SET INTEGRITY FOR HISTORY10 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY11 OFF;
ALTER TABLE HISTORY11 DROP CONSTRAINT HISTORY11CKC;
ALTER TABLE HISTORY11 ADD CONSTRAINT HISTORY11CKC CHECK (H_W_ID BETWEEN 16241
AND 17864);
SET INTEGRITY FOR HISTORY11 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY12 OFF;
ALTER TABLE HISTORY12 DROP CONSTRAINT HISTORY12CKC;
ALTER TABLE HISTORY12 ADD CONSTRAINT HISTORY12CKC CHECK (H_W_ID BETWEEN 17865
AND 19488);
SET INTEGRITY FOR HISTORY12 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY13 OFF;
ALTER TABLE HISTORY13 DROP CONSTRAINT HISTORY13CKC;
ALTER TABLE HISTORY13 ADD CONSTRAINT HISTORY13CKC CHECK (H_W_ID BETWEEN 19489
AND 21112);
SET INTEGRITY FOR HISTORY13 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY14 OFF;
ALTER TABLE HISTORY14 DROP CONSTRAINT HISTORY14CKC;
ALTER TABLE HISTORY14 ADD CONSTRAINT HISTORY14CKC CHECK (H_W_ID BETWEEN 21113
AND 22736);
SET INTEGRITY FOR HISTORY14 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY15 OFF;
ALTER TABLE HISTORY15 DROP CONSTRAINT HISTORY15CKC;
ALTER TABLE HISTORY15 ADD CONSTRAINT HISTORY15CKC CHECK (H_W_ID BETWEEN 22737
AND 24360);
SET INTEGRITY FOR HISTORY15 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY16 OFF;
ALTER TABLE HISTORY16 DROP CONSTRAINT HISTORY16CKC;
ALTER TABLE HISTORY16 ADD CONSTRAINT HISTORY16CKC CHECK (H_W_ID BETWEEN 24361
AND 25984);
SET INTEGRITY FOR HISTORY16 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY17 OFF;
ALTER TABLE HISTORY17 DROP CONSTRAINT HISTORY17CKC;
ALTER TABLE HISTORY17 ADD CONSTRAINT HISTORY17CKC CHECK (H_W_ID BETWEEN 25985
AND 27608);
SET INTEGRITY FOR HISTORY17 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY18 OFF;
ALTER TABLE HISTORY18 DROP CONSTRAINT HISTORY18CKC;
ALTER TABLE HISTORY18 ADD CONSTRAINT HISTORY18CKC CHECK (H_W_ID BETWEEN 27609
AND 29232);
SET INTEGRITY FOR HISTORY18 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY19 OFF;
ALTER TABLE HISTORY19 DROP CONSTRAINT HISTORY19CKC;
ALTER TABLE HISTORY19 ADD CONSTRAINT HISTORY19CKC CHECK (H_W_ID BETWEEN 29233
AND 30856);
SET INTEGRITY FOR HISTORY19 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY20 OFF;
ALTER TABLE HISTORY20 DROP CONSTRAINT HISTORY20CKC;
ALTER TABLE HISTORY20 ADD CONSTRAINT HISTORY20CKC CHECK (H_W_ID BETWEEN 30857
AND 32480);
SET INTEGRITY FOR HISTORY20 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY21 OFF;
ALTER TABLE HISTORY21 DROP CONSTRAINT HISTORY21CKC;
ALTER TABLE HISTORY21 ADD CONSTRAINT HISTORY21CKC CHECK (H_W_ID BETWEEN 32481
AND 34104);
SET INTEGRITY FOR HISTORY21 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY22 OFF;
ALTER TABLE HISTORY22 DROP CONSTRAINT HISTORY22CKC;
ALTER TABLE HISTORY22 ADD CONSTRAINT HISTORY22CKC CHECK (H_W_ID BETWEEN 34105
AND 35728);
SET INTEGRITY FOR HISTORY22 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY23 OFF;
ALTER TABLE HISTORY23 DROP CONSTRAINT HISTORY23CKC;
ALTER TABLE HISTORY23 ADD CONSTRAINT HISTORY23CKC CHECK (H_W_ID BETWEEN 35729
AND 37352);
SET INTEGRITY FOR HISTORY23 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
```

```
SET INTEGRITY FOR HISTORY24 OFF;
ALTER TABLE HISTORY24 DROP CONSTRAINT HISTORY24CKC;
ALTER TABLE HISTORY24 ADD CONSTRAINT HISTORY24CKC CHECK (H_W_ID BETWEEN 37353
AND 38976);
SET INTEGRITY FOR HISTORY24 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY25 OFF;
ALTER TABLE HISTORY25 DROP CONSTRAINT HISTORY25CKC;
ALTER TABLE HISTORY25 ADD CONSTRAINT HISTORY25CKC CHECK (H_W_ID BETWEEN 38977
AND 40600);
SET INTEGRITY FOR HISTORY25 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY26 OFF;
ALTER TABLE HISTORY26 DROP CONSTRAINT HISTORY26CKC;
ALTER TABLE HISTORY26 ADD CONSTRAINT HISTORY26CKC CHECK (H_W_ID BETWEEN 40601
AND 42224);
SET INTEGRITY FOR HISTORY26 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY27 OFF;
ALTER TABLE HISTORY27 DROP CONSTRAINT HISTORY27CKC;
ALTER TABLE HISTORY27 ADD CONSTRAINT HISTORY27CKC CHECK (H_W_ID BETWEEN 42225
AND 43848);
SET INTEGRITY FOR HISTORY27 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY28 OFF;
ALTER TABLE HISTORY28 DROP CONSTRAINT HISTORY28CKC;
ALTER TABLE HISTORY28 ADD CONSTRAINT HISTORY28CKC CHECK (H_W_ID BETWEEN 43849
AND 45472);
SET INTEGRITY FOR HISTORY28 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY29 OFF;
ALTER TABLE HISTORY29 DROP CONSTRAINT HISTORY29CKC;
ALTER TABLE HISTORY29 ADD CONSTRAINT HISTORY29CKC CHECK (H_W_ID BETWEEN 45473
AND 47096);
SET INTEGRITY FOR HISTORY29 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY30 OFF;
ALTER TABLE HISTORY30 DROP CONSTRAINT HISTORY30CKC;
ALTER TABLE HISTORY30 ADD CONSTRAINT HISTORY30CKC CHECK (H_W_ID BETWEEN 47097
AND 48720);
SET INTEGRITY FOR HISTORY30 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY31 OFF;
ALTER TABLE HISTORY31 DROP CONSTRAINT HISTORY31CKC;
ALTER TABLE HISTORY31 ADD CONSTRAINT HISTORY31CKC CHECK (H_W_ID BETWEEN 48721
AND 50344);
SET INTEGRITY FOR HISTORY31 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY32 OFF;
ALTER TABLE HISTORY32 DROP CONSTRAINT HISTORY32CKC;
ALTER TABLE HISTORY32 ADD CONSTRAINT HISTORY32CKC CHECK (H_W_ID >= 50345);
SET INTEGRITY FOR HISTORY32 ALL IMMEDIATE UNCHECKED;
connect reset;
```

DDL/CRCONST_NEW_ORDERA.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA1 OFF;
ALTER TABLE NEW_ORDERA1 DROP CONSTRAINT NEW_ORDERA1CKC;
ALTER TABLE NEW_ORDERA1 ADD CONSTRAINT NEW_ORDERA1CKC CHECK ((NO_W_ID
BETWEEN 1 AND 1624) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA2 OFF;
ALTER TABLE NEW_ORDERA2 DROP CONSTRAINT NEW_ORDERA2CKC;
ALTER TABLE NEW_ORDERA2 ADD CONSTRAINT NEW_ORDERA2CKC CHECK ((NO_W_ID
BETWEEN 1625 AND 3248) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA2 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA3 OFF;
ALTER TABLE NEW_ORDERA3 DROP CONSTRAINT NEW_ORDERA3CKC;
ALTER TABLE NEW_ORDERA3 ADD CONSTRAINT NEW_ORDERA3CKC CHECK ((NO_W_ID
BETWEEN 3249 AND 4872) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA3 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA4 OFF;
ALTER TABLE NEW_ORDERA4 DROP CONSTRAINT NEW_ORDERA4CKC;
ALTER TABLE NEW_ORDERA4 ADD CONSTRAINT NEW_ORDERA4CKC CHECK ((NO_W_ID
BETWEEN 4873 AND 6496) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA4 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA5 OFF;
ALTER TABLE NEW_ORDERA5 DROP CONSTRAINT NEW_ORDERA5CKC;
ALTER TABLE NEW_ORDERA5 ADD CONSTRAINT NEW_ORDERA5CKC CHECK ((NO_W_ID
BETWEEN 6497 AND 8120) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA5 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA6 OFF;
ALTER TABLE NEW_ORDERA6 DROP CONSTRAINT NEW_ORDERA6CKC;
ALTER TABLE NEW_ORDERA6 ADD CONSTRAINT NEW_ORDERA6CKC CHECK ((NO_W_ID
BETWEEN 8121 AND 9744) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA6 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA7 OFF;
ALTER TABLE NEW_ORDERA7 DROP CONSTRAINT NEW_ORDERA7CKC;
ALTER TABLE NEW_ORDERA7 ADD CONSTRAINT NEW_ORDERA7CKC CHECK ((NO_W_ID
BETWEEN 9745 AND 11368) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA7 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA8 OFF;
ALTER TABLE NEW_ORDERA8 DROP CONSTRAINT NEW_ORDERA8CKC;
ALTER TABLE NEW_ORDERA8 ADD CONSTRAINT NEW_ORDERA8CKC CHECK ((NO_W_ID
BETWEEN 11369 AND 12992) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA8 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA9 OFF;
ALTER TABLE NEW_ORDERA9 DROP CONSTRAINT NEW_ORDERA9CKC;
ALTER TABLE NEW_ORDERA9 ADD CONSTRAINT NEW_ORDERA9CKC CHECK ((NO_W_ID
BETWEEN 12993 AND 14616) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA9 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA10 OFF;
ALTER TABLE NEW_ORDERA10 DROP CONSTRAINT NEW_ORDERA10CKC;
ALTER TABLE NEW_ORDERA10 ADD CONSTRAINT NEW_ORDERA10CKC CHECK ((NO_W_ID
BETWEEN 14617 AND 16240) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA10 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA11 OFF;
ALTER TABLE NEW_ORDERA11 DROP CONSTRAINT NEW_ORDERA11CKC;
ALTER TABLE NEW_ORDERA11 ADD CONSTRAINT NEW_ORDERA11CKC CHECK ((NO_W_ID
BETWEEN 16241 AND 17864) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA11 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA12 OFF;
ALTER TABLE NEW_ORDERA12 DROP CONSTRAINT NEW_ORDERA12CKC;
ALTER TABLE NEW_ORDERA12 ADD CONSTRAINT NEW_ORDERA12CKC CHECK ((NO_W_ID
BETWEEN 17865 AND 19488) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA12 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA13 OFF;
ALTER TABLE NEW_ORDERA13 DROP CONSTRAINT NEW_ORDERA13CKC;
ALTER TABLE NEW_ORDERA13 ADD CONSTRAINT NEW_ORDERA13CKC CHECK ((NO_W_ID
BETWEEN 19489 AND 21112) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA13 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA14 OFF;
ALTER TABLE NEW_ORDERA14 DROP CONSTRAINT NEW_ORDERA14CKC;
ALTER TABLE NEW_ORDERA14 ADD CONSTRAINT NEW_ORDERA14CKC CHECK ((NO_W_ID
BETWEEN 21113 AND 22736) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA14 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA15 OFF;
ALTER TABLE NEW_ORDERA15 DROP CONSTRAINT NEW_ORDERA15CKC;
ALTER TABLE NEW_ORDERA15 ADD CONSTRAINT NEW_ORDERA15CKC CHECK ((NO_W_ID
BETWEEN 22737 AND 24360) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA15 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA16 OFF;
ALTER TABLE NEW_ORDERA16 DROP CONSTRAINT NEW_ORDERA16CKC;
ALTER TABLE NEW_ORDERA16 ADD CONSTRAINT NEW_ORDERA16CKC CHECK ((NO_W_ID
BETWEEN 24361 AND 25984) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA16 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA17 OFF;
ALTER TABLE NEW_ORDERA17 DROP CONSTRAINT NEW_ORDERA17CKC;
ALTER TABLE NEW_ORDERA17 ADD CONSTRAINT NEW_ORDERA17CKC CHECK ((NO_W_ID
BETWEEN 25985 AND 27608) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA17 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA18 OFF;
ALTER TABLE NEW_ORDERA18 DROP CONSTRAINT NEW_ORDERA18CKC;
ALTER TABLE NEW_ORDERA18 ADD CONSTRAINT NEW_ORDERA18CKC CHECK ((NO_W_ID
BETWEEN 27609 AND 29232) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA18 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA19 OFF;
ALTER TABLE NEW_ORDERA19 DROP CONSTRAINT NEW_ORDERA19CKC;
ALTER TABLE NEW_ORDERA19 ADD CONSTRAINT NEW_ORDERA19CKC CHECK ((NO_W_ID
BETWEEN 29233 AND 30856) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA19 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA20 OFF;
ALTER TABLE NEW_ORDERA20 DROP CONSTRAINT NEW_ORDERA20CKC;
ALTER TABLE NEW_ORDERA20 ADD CONSTRAINT NEW_ORDERA20CKC CHECK ((NO_W_ID
BETWEEN 30857 AND 32480) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA20 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA21 OFF;
ALTER TABLE NEW_ORDERA21 DROP CONSTRAINT NEW_ORDERA21CKC;
ALTER TABLE NEW_ORDERA21 ADD CONSTRAINT NEW_ORDERA21CKC CHECK ((NO_W_ID
BETWEEN 32481 AND 34104) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA21 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA22 OFF;
ALTER TABLE NEW_ORDERA22 DROP CONSTRAINT NEW_ORDERA22CKC;
```

```
ALTER TABLE NEW_ORDERA22 ADD CONSTRAINT NEW_ORDERA22CKC CHECK ((NO_W_ID
BETWEEN 34105 AND 35728) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA22 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA23 OFF;
ALTER TABLE NEW_ORDERA23 DROP CONSTRAINT NEW_ORDERA23CKC;
ALTER TABLE NEW_ORDERA23 ADD CONSTRAINT NEW_ORDERA23CKC CHECK ((NO_W_ID
BETWEEN 35729 AND 37352) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA23 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA24 OFF;
ALTER TABLE NEW_ORDERA24 DROP CONSTRAINT NEW_ORDERA24CKC;
ALTER TABLE NEW_ORDERA24 ADD CONSTRAINT NEW_ORDERA24CKC CHECK ((NO_W_ID
BETWEEN 37353 AND 38976) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA24 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA25 OFF;
ALTER TABLE NEW_ORDERA25 DROP CONSTRAINT NEW_ORDERA25CKC;
ALTER TABLE NEW_ORDERA25 ADD CONSTRAINT NEW_ORDERA25CKC CHECK ((NO_W_ID
BETWEEN 38977 AND 40600) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA25 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA26 OFF;
ALTER TABLE NEW_ORDERA26 DROP CONSTRAINT NEW_ORDERA26CKC;
ALTER TABLE NEW_ORDERA26 ADD CONSTRAINT NEW_ORDERA26CKC CHECK ((NO_W_ID
BETWEEN 40601 AND 42224) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA26 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA27 OFF;
ALTER TABLE NEW_ORDERA27 DROP CONSTRAINT NEW_ORDERA27CKC;
ALTER TABLE NEW_ORDERA27 ADD CONSTRAINT NEW_ORDERA27CKC CHECK ((NO_W_ID
BETWEEN 42225 AND 43848) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA27 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA28 OFF;
ALTER TABLE NEW_ORDERA28 DROP CONSTRAINT NEW_ORDERA28CKC;
ALTER TABLE NEW_ORDERA28 ADD CONSTRAINT NEW_ORDERA28CKC CHECK ((NO_W_ID
BETWEEN 43849 AND 45472) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA28 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA29 OFF;
ALTER TABLE NEW_ORDERA29 DROP CONSTRAINT NEW_ORDERA29CKC;
ALTER TABLE NEW_ORDERA29 ADD CONSTRAINT NEW_ORDERA29CKC CHECK ((NO_W_ID
BETWEEN 45473 AND 47096) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA29 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA30 OFF;
ALTER TABLE NEW_ORDERA30 DROP CONSTRAINT NEW_ORDERA30CKC;
ALTER TABLE NEW_ORDERA30 ADD CONSTRAINT NEW_ORDERA30CKC CHECK ((NO_W_ID
BETWEEN 47097 AND 48720) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA30 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA31 OFF;
ALTER TABLE NEW_ORDERA31 DROP CONSTRAINT NEW_ORDERA31CKC;
ALTER TABLE NEW_ORDERA31 ADD CONSTRAINT NEW_ORDERA31CKC CHECK ((NO_W_ID
BETWEEN 48721 AND 50344) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA31 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA32 OFF;
ALTER TABLE NEW_ORDERA32 DROP CONSTRAINT NEW_ORDERA32CKC;
ALTER TABLE NEW_ORDERA32 ADD CONSTRAINT NEW_ORDERA32CKC CHECK ((NO_W_ID >=
50345) AND (NO_O_ID <= 3695));
SET INTEGRITY FOR NEW_ORDERA32 ALL IMMEDIATE UNCHECKED;
connect reset;
```

DDL/CRCONST_NEW_ORDERB.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB1 OFF;
ALTER TABLE NEW_ORDERB1 DROP CONSTRAINT NEW_ORDERB1CKC;
ALTER TABLE NEW_ORDERB1 ADD CONSTRAINT NEW_ORDERB1CKC CHECK ((NO_W_ID
BETWEEN 1 AND 1624) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB2 OFF;
ALTER TABLE NEW_ORDERB2 DROP CONSTRAINT NEW_ORDERB2CKC;
ALTER TABLE NEW_ORDERB2 ADD CONSTRAINT NEW_ORDERB2CKC CHECK ((NO_W_ID
BETWEEN 1625 AND 3248) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB2 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB3 OFF;
ALTER TABLE NEW_ORDERB3 DROP CONSTRAINT NEW_ORDERB3CKC;
ALTER TABLE NEW_ORDERB3 ADD CONSTRAINT NEW_ORDERB3CKC CHECK ((NO_W_ID
BETWEEN 3249 AND 4872) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB3 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB4 OFF;
ALTER TABLE NEW_ORDERB4 DROP CONSTRAINT NEW_ORDERB4CKC;
ALTER TABLE NEW_ORDERB4 ADD CONSTRAINT NEW_ORDERB4CKC CHECK ((NO_W_ID
BETWEEN 4873 AND 6496) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB4 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB5 OFF;
ALTER TABLE NEW_ORDERB5 DROP CONSTRAINT NEW_ORDERB5CKC;
ALTER TABLE NEW_ORDERB5 ADD CONSTRAINT NEW_ORDERB5CKC CHECK ((NO_W_ID
BETWEEN 6497 AND 8120) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB5 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB6 OFF;
ALTER TABLE NEW_ORDERB6 DROP CONSTRAINT NEW_ORDERB6CKC;
ALTER TABLE NEW_ORDERB6 ADD CONSTRAINT NEW_ORDERB6CKC CHECK ((NO_W_ID
BETWEEN 8121 AND 9744) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB6 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB7 OFF;
ALTER TABLE NEW_ORDERB7 DROP CONSTRAINT NEW_ORDERB7CKC;
ALTER TABLE NEW_ORDERB7 ADD CONSTRAINT NEW_ORDERB7CKC CHECK ((NO_W_ID
BETWEEN 9745 AND 11368) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB7 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB8 OFF;
ALTER TABLE NEW_ORDERB8 DROP CONSTRAINT NEW_ORDERB8CKC;
ALTER TABLE NEW_ORDERB8 ADD CONSTRAINT NEW_ORDERB8CKC CHECK ((NO_W_ID
BETWEEN 11369 AND 12992) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB8 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB9 OFF;
ALTER TABLE NEW_ORDERB9 DROP CONSTRAINT NEW_ORDERB9CKC;
ALTER TABLE NEW_ORDERB9 ADD CONSTRAINT NEW_ORDERB9CKC CHECK ((NO_W_ID
BETWEEN 12993 AND 14616) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB9 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB10 OFF;
ALTER TABLE NEW_ORDERB10 DROP CONSTRAINT NEW_ORDERB10CKC;
ALTER TABLE NEW_ORDERB10 ADD CONSTRAINT NEW_ORDERB10CKC CHECK ((NO_W_ID
BETWEEN 14617 AND 16240) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB10 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB11 OFF;
ALTER TABLE NEW_ORDERB11 DROP CONSTRAINT NEW_ORDERB11CKC;
ALTER TABLE NEW_ORDERB11 ADD CONSTRAINT NEW_ORDERB11CKC CHECK ((NO_W_ID
BETWEEN 16241 AND 17864) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB11 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB12 OFF;
ALTER TABLE NEW_ORDERB12 DROP CONSTRAINT NEW_ORDERB12CKC;
ALTER TABLE NEW_ORDERB12 ADD CONSTRAINT NEW_ORDERB12CKC CHECK ((NO_W_ID
BETWEEN 17865 AND 19488) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB12 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB13 OFF;
ALTER TABLE NEW_ORDERB13 DROP CONSTRAINT NEW_ORDERB13CKC;
ALTER TABLE NEW_ORDERB13 ADD CONSTRAINT NEW_ORDERB13CKC CHECK ((NO_W_ID
BETWEEN 19489 AND 21112) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB13 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB14 OFF;
ALTER TABLE NEW_ORDERB14 DROP CONSTRAINT NEW_ORDERB14CKC;
ALTER TABLE NEW_ORDERB14 ADD CONSTRAINT NEW_ORDERB14CKC CHECK ((NO_W_ID
BETWEEN 21113 AND 22736) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB14 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB15 OFF;
ALTER TABLE NEW_ORDERB15 DROP CONSTRAINT NEW_ORDERB15CKC;
ALTER TABLE NEW_ORDERB15 ADD CONSTRAINT NEW_ORDERB15CKC CHECK ((NO_W_ID
BETWEEN 22737 AND 24360) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB15 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB16 OFF;
ALTER TABLE NEW_ORDERB16 DROP CONSTRAINT NEW_ORDERB16CKC;
ALTER TABLE NEW_ORDERB16 ADD CONSTRAINT NEW_ORDERB16CKC CHECK ((NO_W_ID
BETWEEN 24361 AND 25984) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB16 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB17 OFF;
ALTER TABLE NEW_ORDERB17 DROP CONSTRAINT NEW_ORDERB17CKC;
ALTER TABLE NEW_ORDERB17 ADD CONSTRAINT NEW_ORDERB17CKC CHECK ((NO_W_ID
BETWEEN 25985 AND 27608) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB17 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB18 OFF;
ALTER TABLE NEW_ORDERB18 DROP CONSTRAINT NEW_ORDERB18CKC;
ALTER TABLE NEW_ORDERB18 ADD CONSTRAINT NEW_ORDERB18CKC CHECK ((NO_W_ID
BETWEEN 27609 AND 29232) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB18 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB19 OFF;
ALTER TABLE NEW_ORDERB19 DROP CONSTRAINT NEW_ORDERB19CKC;
ALTER TABLE NEW_ORDERB19 ADD CONSTRAINT NEW_ORDERB19CKC CHECK ((NO_W_ID
BETWEEN 29233 AND 30856) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB19 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB20 OFF;
ALTER TABLE NEW_ORDERB20 DROP CONSTRAINT NEW_ORDERB20CKC;
ALTER TABLE NEW_ORDERB20 ADD CONSTRAINT NEW_ORDERB20CKC CHECK ((NO_W_ID
BETWEEN 30857 AND 32480) AND (NO_O_ID >= 3696));
```

```
SET INTEGRITY FOR NEW_ORDERB20 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB21 OFF;
ALTER TABLE NEW_ORDERB21 DROP CONSTRAINT NEW_ORDERB21CKC;
ALTER TABLE NEW_ORDERB21 ADD CONSTRAINT NEW_ORDERB21CKC CHECK ((NO_W_ID
BETWEEN 32481 AND 34104) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB21 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB22 OFF;
ALTER TABLE NEW_ORDERB22 DROP CONSTRAINT NEW_ORDERB22CKC;
ALTER TABLE NEW_ORDERB22 ADD CONSTRAINT NEW_ORDERB22CKC CHECK ((NO_W_ID
BETWEEN 34105 AND 35728) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB22 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB23 OFF;
ALTER TABLE NEW_ORDERB23 DROP CONSTRAINT NEW_ORDERB23CKC;
ALTER TABLE NEW_ORDERB23 ADD CONSTRAINT NEW_ORDERB23CKC CHECK ((NO_W_ID
BETWEEN 35729 AND 37352) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB23 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB24 OFF;
ALTER TABLE NEW_ORDERB24 DROP CONSTRAINT NEW_ORDERB24CKC;
ALTER TABLE NEW_ORDERB24 ADD CONSTRAINT NEW_ORDERB24CKC CHECK ((NO_W_ID
BETWEEN 37353 AND 38976) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB24 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB25 OFF;
ALTER TABLE NEW_ORDERB25 DROP CONSTRAINT NEW_ORDERB25CKC;
ALTER TABLE NEW_ORDERB25 ADD CONSTRAINT NEW_ORDERB25CKC CHECK ((NO_W_ID
BETWEEN 38977 AND 40600) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB25 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB26 OFF;
ALTER TABLE NEW_ORDERB26 DROP CONSTRAINT NEW_ORDERB26CKC;
ALTER TABLE NEW_ORDERB26 ADD CONSTRAINT NEW_ORDERB26CKC CHECK ((NO_W_ID
BETWEEN 40601 AND 42224) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB26 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB27 OFF;
ALTER TABLE NEW_ORDERB27 DROP CONSTRAINT NEW_ORDERB27CKC;
ALTER TABLE NEW_ORDERB27 ADD CONSTRAINT NEW_ORDERB27CKC CHECK ((NO_W_ID
BETWEEN 42225 AND 43848) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB27 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB28 OFF;
ALTER TABLE NEW_ORDERB28 DROP CONSTRAINT NEW_ORDERB28CKC;
ALTER TABLE NEW_ORDERB28 ADD CONSTRAINT NEW_ORDERB28CKC CHECK ((NO_W_ID
BETWEEN 43849 AND 45472) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB28 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB29 OFF;
ALTER TABLE NEW_ORDERB29 DROP CONSTRAINT NEW_ORDERB29CKC;
ALTER TABLE NEW_ORDERB29 ADD CONSTRAINT NEW_ORDERB29CKC CHECK ((NO_W_ID
BETWEEN 45473 AND 47096) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB29 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB30 OFF;
ALTER TABLE NEW_ORDERB30 DROP CONSTRAINT NEW_ORDERB30CKC;
ALTER TABLE NEW_ORDERB30 ADD CONSTRAINT NEW_ORDERB30CKC CHECK ((NO_W_ID
BETWEEN 47097 AND 48720) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB30 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB31 OFF;
ALTER TABLE NEW_ORDERB31 DROP CONSTRAINT NEW_ORDERB31CKC;
ALTER TABLE NEW_ORDERB31 ADD CONSTRAINT NEW_ORDERB31CKC CHECK ((NO_W_ID
BETWEEN 48721 AND 50344) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB31 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB32 OFF;
ALTER TABLE NEW_ORDERB32 DROP CONSTRAINT NEW_ORDERB32CKC;
ALTER TABLE NEW_ORDERB32 ADD CONSTRAINT NEW_ORDERB32CKC CHECK ((NO_W_ID >=
50345) AND (NO_O_ID >= 3696));
SET INTEGRITY FOR NEW_ORDERB32 ALL IMMEDIATE UNCHECKED;
connect reset;
```

DDL/CRCONST_ORDERS.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS1 OFF;
ALTER TABLE ORDERS1 DROP CONSTRAINT ORDERS1CKC;
ALTER TABLE ORDERS1 ADD CONSTRAINT ORDERS1CKC CHECK (O_W_ID BETWEEN 1 AND
812);
SET INTEGRITY FOR ORDERS1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS2 OFF;
ALTER TABLE ORDERS2 DROP CONSTRAINT ORDERS2CKC;
ALTER TABLE ORDERS2 ADD CONSTRAINT ORDERS2CKC CHECK (O_W_ID BETWEEN 813 AND
1624);
SET INTEGRITY FOR ORDERS2 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS3 OFF;
ALTER TABLE ORDERS3 DROP CONSTRAINT ORDERS3CKC;
ALTER TABLE ORDERS3 ADD CONSTRAINT ORDERS3CKC CHECK (O_W_ID BETWEEN 1625 AND
2436);
SET INTEGRITY FOR ORDERS3 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS4 OFF;
ALTER TABLE ORDERS4 DROP CONSTRAINT ORDERS4CKC;
ALTER TABLE ORDERS4 ADD CONSTRAINT ORDERS4CKC CHECK (O_W_ID BETWEEN 2437 AND
3248);
SET INTEGRITY FOR ORDERS4 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS5 OFF;
ALTER TABLE ORDERS5 DROP CONSTRAINT ORDERS5CKC;
ALTER TABLE ORDERS5 ADD CONSTRAINT ORDERS5CKC CHECK (O_W_ID BETWEEN 3249 AND
4060);
SET INTEGRITY FOR ORDERS5 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS6 OFF;
ALTER TABLE ORDERS6 DROP CONSTRAINT ORDERS6CKC;
ALTER TABLE ORDERS6 ADD CONSTRAINT ORDERS6CKC CHECK (O_W_ID BETWEEN 4061 AND
4872);
SET INTEGRITY FOR ORDERS6 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS7 OFF;
ALTER TABLE ORDERS7 DROP CONSTRAINT ORDERS7CKC;
ALTER TABLE ORDERS7 ADD CONSTRAINT ORDERS7CKC CHECK (O_W_ID BETWEEN 4873 AND
5684);
SET INTEGRITY FOR ORDERS7 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS8 OFF;
ALTER TABLE ORDERS8 DROP CONSTRAINT ORDERS8CKC;
ALTER TABLE ORDERS8 ADD CONSTRAINT ORDERS8CKC CHECK (O_W_ID BETWEEN 5685 AND
6496);
SET INTEGRITY FOR ORDERS8 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS9 OFF;
ALTER TABLE ORDERS9 DROP CONSTRAINT ORDERS9CKC;
ALTER TABLE ORDERS9 ADD CONSTRAINT ORDERS9CKC CHECK (O_W_ID BETWEEN 6497 AND
7308);
SET INTEGRITY FOR ORDERS9 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS10 OFF;
ALTER TABLE ORDERS10 DROP CONSTRAINT ORDERS10CKC;
ALTER TABLE ORDERS10 ADD CONSTRAINT ORDERS10CKC CHECK (O_W_ID BETWEEN 7309
AND 8120);
SET INTEGRITY FOR ORDERS10 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS11 OFF;
ALTER TABLE ORDERS11 DROP CONSTRAINT ORDERS11CKC;
ALTER TABLE ORDERS11 ADD CONSTRAINT ORDERS11CKC CHECK (O_W_ID BETWEEN 8121
AND 8932);
SET INTEGRITY FOR ORDERS11 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS12 OFF;
ALTER TABLE ORDERS12 DROP CONSTRAINT ORDERS12CKC;
ALTER TABLE ORDERS12 ADD CONSTRAINT ORDERS12CKC CHECK (O_W_ID BETWEEN 8933
AND 9744);
SET INTEGRITY FOR ORDERS12 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS13 OFF;
ALTER TABLE ORDERS13 DROP CONSTRAINT ORDERS13CKC;
ALTER TABLE ORDERS13 ADD CONSTRAINT ORDERS13CKC CHECK (O_W_ID BETWEEN 9745
AND 10556);
SET INTEGRITY FOR ORDERS13 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS14 OFF;
ALTER TABLE ORDERS14 DROP CONSTRAINT ORDERS14CKC;
ALTER TABLE ORDERS14 ADD CONSTRAINT ORDERS14CKC CHECK (O_W_ID BETWEEN 10557
AND 11368);
SET INTEGRITY FOR ORDERS14 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS15 OFF;
ALTER TABLE ORDERS15 DROP CONSTRAINT ORDERS15CKC;
ALTER TABLE ORDERS15 ADD CONSTRAINT ORDERS15CKC CHECK (O_W_ID BETWEEN 11369
AND 12180);
SET INTEGRITY FOR ORDERS15 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS16 OFF;
ALTER TABLE ORDERS16 DROP CONSTRAINT ORDERS16CKC;
ALTER TABLE ORDERS16 ADD CONSTRAINT ORDERS16CKC CHECK (O_W_ID BETWEEN 12181
AND 12992);
SET INTEGRITY FOR ORDERS16 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS17 OFF;
ALTER TABLE ORDERS17 DROP CONSTRAINT ORDERS17CKC;
ALTER TABLE ORDERS17 ADD CONSTRAINT ORDERS17CKC CHECK (O_W_ID BETWEEN 12993
AND 13804);
SET INTEGRITY FOR ORDERS17 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS18 OFF;
ALTER TABLE ORDERS18 DROP CONSTRAINT ORDERS18CKC;
ALTER TABLE ORDERS18 ADD CONSTRAINT ORDERS18CKC CHECK (O_W_ID BETWEEN 13805
AND 14616);
SET INTEGRITY FOR ORDERS18 ALL IMMEDIATE UNCHECKED;
```

```
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS19 OFF;
ALTER TABLE ORDERS19 DROP CONSTRAINT ORDERS19CKC;
ALTER TABLE ORDERS19 ADD CONSTRAINT ORDERS19CKC CHECK (O_W_ID BETWEEN 14617
AND 15428);
SET INTEGRITY FOR ORDERS19 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS20 OFF;
ALTER TABLE ORDERS20 DROP CONSTRAINT ORDERS20CKC;
ALTER TABLE ORDERS20 ADD CONSTRAINT ORDERS20CKC CHECK (O_W_ID BETWEEN 15429
AND 16240);
SET INTEGRITY FOR ORDERS20 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS21 OFF;
ALTER TABLE ORDERS21 DROP CONSTRAINT ORDERS21CKC;
ALTER TABLE ORDERS21 ADD CONSTRAINT ORDERS21CKC CHECK (O_W_ID BETWEEN 16241
AND 17052);
SET INTEGRITY FOR ORDERS21 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS22 OFF;
ALTER TABLE ORDERS22 DROP CONSTRAINT ORDERS22CKC;
ALTER TABLE ORDERS22 ADD CONSTRAINT ORDERS22CKC CHECK (O_W_ID BETWEEN 17053
AND 17864);
SET INTEGRITY FOR ORDERS22 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS23 OFF;
ALTER TABLE ORDERS23 DROP CONSTRAINT ORDERS23CKC;
ALTER TABLE ORDERS23 ADD CONSTRAINT ORDERS23CKC CHECK (O_W_ID BETWEEN 17865
AND 18676);
SET INTEGRITY FOR ORDERS23 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS24 OFF;
ALTER TABLE ORDERS24 DROP CONSTRAINT ORDERS24CKC;
ALTER TABLE ORDERS24 ADD CONSTRAINT ORDERS24CKC CHECK (O_W_ID BETWEEN 18677
AND 19488);
SET INTEGRITY FOR ORDERS24 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS25 OFF;
ALTER TABLE ORDERS25 DROP CONSTRAINT ORDERS25CKC;
ALTER TABLE ORDERS25 ADD CONSTRAINT ORDERS25CKC CHECK (O_W_ID BETWEEN 19489
AND 20300);
SET INTEGRITY FOR ORDERS25 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS26 OFF;
ALTER TABLE ORDERS26 DROP CONSTRAINT ORDERS26CKC;
ALTER TABLE ORDERS26 ADD CONSTRAINT ORDERS26CKC CHECK (O_W_ID BETWEEN 20301
AND 21112);
SET INTEGRITY FOR ORDERS26 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS27 OFF;
ALTER TABLE ORDERS27 DROP CONSTRAINT ORDERS27CKC;
ALTER TABLE ORDERS27 ADD CONSTRAINT ORDERS27CKC CHECK (O_W_ID BETWEEN 21113
AND 21924);
SET INTEGRITY FOR ORDERS27 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS28 OFF;
ALTER TABLE ORDERS28 DROP CONSTRAINT ORDERS28CKC;
ALTER TABLE ORDERS28 ADD CONSTRAINT ORDERS28CKC CHECK (O_W_ID BETWEEN 21925
AND 22736);
SET INTEGRITY FOR ORDERS28 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS29 OFF;
ALTER TABLE ORDERS29 DROP CONSTRAINT ORDERS29CKC;
ALTER TABLE ORDERS29 ADD CONSTRAINT ORDERS29CKC CHECK (O_W_ID BETWEEN 22737
AND 23548);
SET INTEGRITY FOR ORDERS29 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS30 OFF;
ALTER TABLE ORDERS30 DROP CONSTRAINT ORDERS30CKC;
ALTER TABLE ORDERS30 ADD CONSTRAINT ORDERS30CKC CHECK (O_W_ID BETWEEN 23549
AND 24360);
SET INTEGRITY FOR ORDERS30 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS31 OFF;
ALTER TABLE ORDERS31 DROP CONSTRAINT ORDERS31CKC;
ALTER TABLE ORDERS31 ADD CONSTRAINT ORDERS31CKC CHECK (O_W_ID BETWEEN 24361
AND 25172);
SET INTEGRITY FOR ORDERS31 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS32 OFF;
ALTER TABLE ORDERS32 DROP CONSTRAINT ORDERS32CKC;
ALTER TABLE ORDERS32 ADD CONSTRAINT ORDERS32CKC CHECK (O_W_ID BETWEEN 25173
AND 25984);
SET INTEGRITY FOR ORDERS32 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS33 OFF;
ALTER TABLE ORDERS33 DROP CONSTRAINT ORDERS33CKC;
ALTER TABLE ORDERS33 ADD CONSTRAINT ORDERS33CKC CHECK (O_W_ID BETWEEN 25985
AND 26796);
SET INTEGRITY FOR ORDERS33 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS34 OFF;
ALTER TABLE ORDERS34 DROP CONSTRAINT ORDERS34CKC;
ALTER TABLE ORDERS34 ADD CONSTRAINT ORDERS34CKC CHECK (O_W_ID BETWEEN 26797
AND 27608);
SET INTEGRITY FOR ORDERS34 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS35 OFF;
ALTER TABLE ORDERS35 DROP CONSTRAINT ORDERS35CKC;
ALTER TABLE ORDERS35 ADD CONSTRAINT ORDERS35CKC CHECK (O_W_ID BETWEEN 27609
AND 28420);
SET INTEGRITY FOR ORDERS35 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS36 OFF;
ALTER TABLE ORDERS36 DROP CONSTRAINT ORDERS36CKC;
ALTER TABLE ORDERS36 ADD CONSTRAINT ORDERS36CKC CHECK (O_W_ID BETWEEN 28421
AND 29232);
SET INTEGRITY FOR ORDERS36 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS37 OFF;
ALTER TABLE ORDERS37 DROP CONSTRAINT ORDERS37CKC;
ALTER TABLE ORDERS37 ADD CONSTRAINT ORDERS37CKC CHECK (O_W_ID BETWEEN 29233
AND 30044);
SET INTEGRITY FOR ORDERS37 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS38 OFF;
ALTER TABLE ORDERS38 DROP CONSTRAINT ORDERS38CKC;
ALTER TABLE ORDERS38 ADD CONSTRAINT ORDERS38CKC CHECK (O_W_ID BETWEEN 30045
AND 30856);
SET INTEGRITY FOR ORDERS38 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS39 OFF;
ALTER TABLE ORDERS39 DROP CONSTRAINT ORDERS39CKC;
ALTER TABLE ORDERS39 ADD CONSTRAINT ORDERS39CKC CHECK (O_W_ID BETWEEN 30857
AND 31668);
SET INTEGRITY FOR ORDERS39 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS40 OFF;
ALTER TABLE ORDERS40 DROP CONSTRAINT ORDERS40CKC;
ALTER TABLE ORDERS40 ADD CONSTRAINT ORDERS40CKC CHECK (O_W_ID BETWEEN 31669
AND 32480);
SET INTEGRITY FOR ORDERS40 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS41 OFF;
ALTER TABLE ORDERS41 DROP CONSTRAINT ORDERS41CKC;
ALTER TABLE ORDERS41 ADD CONSTRAINT ORDERS41CKC CHECK (O_W_ID BETWEEN 32481
AND 33292);
SET INTEGRITY FOR ORDERS41 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS42 OFF;
ALTER TABLE ORDERS42 DROP CONSTRAINT ORDERS42CKC;
ALTER TABLE ORDERS42 ADD CONSTRAINT ORDERS42CKC CHECK (O_W_ID BETWEEN 33293
AND 34104);
SET INTEGRITY FOR ORDERS42 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS43 OFF;
ALTER TABLE ORDERS43 DROP CONSTRAINT ORDERS43CKC;
ALTER TABLE ORDERS43 ADD CONSTRAINT ORDERS43CKC CHECK (O_W_ID BETWEEN 34105
AND 34916);
SET INTEGRITY FOR ORDERS43 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS44 OFF;
ALTER TABLE ORDERS44 DROP CONSTRAINT ORDERS44CKC;
ALTER TABLE ORDERS44 ADD CONSTRAINT ORDERS44CKC CHECK (O_W_ID BETWEEN 34917
AND 35728);
SET INTEGRITY FOR ORDERS44 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS45 OFF;
ALTER TABLE ORDERS45 DROP CONSTRAINT ORDERS45CKC;
ALTER TABLE ORDERS45 ADD CONSTRAINT ORDERS45CKC CHECK (O_W_ID BETWEEN 35729
AND 36540);
SET INTEGRITY FOR ORDERS45 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS46 OFF;
ALTER TABLE ORDERS46 DROP CONSTRAINT ORDERS46CKC;
ALTER TABLE ORDERS46 ADD CONSTRAINT ORDERS46CKC CHECK (O_W_ID BETWEEN 36541
AND 37352);
SET INTEGRITY FOR ORDERS46 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS47 OFF;
ALTER TABLE ORDERS47 DROP CONSTRAINT ORDERS47CKC;
ALTER TABLE ORDERS47 ADD CONSTRAINT ORDERS47CKC CHECK (O_W_ID BETWEEN 37353
AND 38164);
SET INTEGRITY FOR ORDERS47 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS48 OFF;
ALTER TABLE ORDERS48 DROP CONSTRAINT ORDERS48CKC;
ALTER TABLE ORDERS48 ADD CONSTRAINT ORDERS48CKC CHECK (O_W_ID BETWEEN 38165
AND 38976);
SET INTEGRITY FOR ORDERS48 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS49 OFF;
ALTER TABLE ORDERS49 DROP CONSTRAINT ORDERS49CKC;
ALTER TABLE ORDERS49 ADD CONSTRAINT ORDERS49CKC CHECK (O_W_ID BETWEEN 38977
AND 39788);
SET INTEGRITY FOR ORDERS49 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
```

```
SET INTEGRITY FOR ORDERS50 OFF;
ALTER TABLE ORDERS50 DROP CONSTRAINT ORDERS50CKC;
ALTER TABLE ORDERS50 ADD CONSTRAINT ORDERS50CKC CHECK (O_W_ID BETWEEN 39789
AND 40600);
SET INTEGRITY FOR ORDERS50 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS51 OFF;
ALTER TABLE ORDERS51 DROP CONSTRAINT ORDERS51CKC;
ALTER TABLE ORDERS51 ADD CONSTRAINT ORDERS51CKC CHECK (O_W_ID BETWEEN 40601
AND 41412);
SET INTEGRITY FOR ORDERS51 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS52 OFF;
ALTER TABLE ORDERS52 DROP CONSTRAINT ORDERS52CKC;
ALTER TABLE ORDERS52 ADD CONSTRAINT ORDERS52CKC CHECK (O_W_ID BETWEEN 41413
AND 42224);
SET INTEGRITY FOR ORDERS52 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS53 OFF;
ALTER TABLE ORDERS53 DROP CONSTRAINT ORDERS53CKC;
ALTER TABLE ORDERS53 ADD CONSTRAINT ORDERS53CKC CHECK (O_W_ID BETWEEN 42225
AND 43036);
SET INTEGRITY FOR ORDERS53 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS54 OFF;
ALTER TABLE ORDERS54 DROP CONSTRAINT ORDERS54CKC;
ALTER TABLE ORDERS54 ADD CONSTRAINT ORDERS54CKC CHECK (O_W_ID BETWEEN 43037
AND 43848);
SET INTEGRITY FOR ORDERS54 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS55 OFF;
ALTER TABLE ORDERS55 DROP CONSTRAINT ORDERS55CKC;
ALTER TABLE ORDERS55 ADD CONSTRAINT ORDERS55CKC CHECK (O_W_ID BETWEEN 43849
AND 44660);
SET INTEGRITY FOR ORDERS55 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS56 OFF;
ALTER TABLE ORDERS56 DROP CONSTRAINT ORDERS56CKC;
ALTER TABLE ORDERS56 ADD CONSTRAINT ORDERS56CKC CHECK (O_W_ID BETWEEN 44661
AND 45472);
SET INTEGRITY FOR ORDERS56 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS57 OFF;
ALTER TABLE ORDERS57 DROP CONSTRAINT ORDERS57CKC;
ALTER TABLE ORDERS57 ADD CONSTRAINT ORDERS57CKC CHECK (O_W_ID BETWEEN 45473
AND 46284);
SET INTEGRITY FOR ORDERS57 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS58 OFF;
ALTER TABLE ORDERS58 DROP CONSTRAINT ORDERS58CKC;
ALTER TABLE ORDERS58 ADD CONSTRAINT ORDERS58CKC CHECK (O_W_ID BETWEEN 46285
AND 47096);
SET INTEGRITY FOR ORDERS58 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS59 OFF;
ALTER TABLE ORDERS59 DROP CONSTRAINT ORDERS59CKC;
ALTER TABLE ORDERS59 ADD CONSTRAINT ORDERS59CKC CHECK (O_W_ID BETWEEN 47097
AND 47908);
SET INTEGRITY FOR ORDERS59 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS60 OFF;
ALTER TABLE ORDERS60 DROP CONSTRAINT ORDERS60CKC;
ALTER TABLE ORDERS60 ADD CONSTRAINT ORDERS60CKC CHECK (O_W_ID BETWEEN 47909
AND 48720);
SET INTEGRITY FOR ORDERS60 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS61 OFF;
ALTER TABLE ORDERS61 DROP CONSTRAINT ORDERS61CKC;
ALTER TABLE ORDERS61 ADD CONSTRAINT ORDERS61CKC CHECK (O_W_ID BETWEEN 48721
AND 49532);
SET INTEGRITY FOR ORDERS61 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS62 OFF;
ALTER TABLE ORDERS62 DROP CONSTRAINT ORDERS62CKC;
ALTER TABLE ORDERS62 ADD CONSTRAINT ORDERS62CKC CHECK (O_W_ID BETWEEN 49533
AND 50344);
SET INTEGRITY FOR ORDERS62 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS63 OFF;
ALTER TABLE ORDERS63 DROP CONSTRAINT ORDERS63CKC;
ALTER TABLE ORDERS63 ADD CONSTRAINT ORDERS63CKC CHECK (O_W_ID BETWEEN 50345
AND 51156);
SET INTEGRITY FOR ORDERS63 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS64 OFF;
ALTER TABLE ORDERS64 DROP CONSTRAINT ORDERS64CKC;
ALTER TABLE ORDERS64 ADD CONSTRAINT ORDERS64CKC CHECK (O_W_ID >= 51157);
SET INTEGRITY FOR ORDERS64 ALL IMMEDIATE UNCHECKED;
connect reset;
```

DDL/CRCONST_ORDER_LINE.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE1 OFF;
ALTER TABLE ORDER_LINE1 DROP CONSTRAINT ORDER_LINE1CKC;
ALTER TABLE ORDER_LINE1 ADD CONSTRAINT ORDER_LINE1CKC CHECK (OL_W_ID BETWEEN 1
AND 812);
SET INTEGRITY FOR ORDER_LINE1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE2 OFF;
ALTER TABLE ORDER_LINE2 DROP CONSTRAINT ORDER_LINE2CKC;
ALTER TABLE ORDER_LINE2 ADD CONSTRAINT ORDER_LINE2CKC CHECK (OL_W_ID BETWEEN
813 AND 1624);
SET INTEGRITY FOR ORDER_LINE2 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE3 OFF;
ALTER TABLE ORDER_LINE3 DROP CONSTRAINT ORDER_LINE3CKC;
ALTER TABLE ORDER_LINE3 ADD CONSTRAINT ORDER_LINE3CKC CHECK (OL_W_ID BETWEEN
1625 AND 2436);
SET INTEGRITY FOR ORDER_LINE3 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE4 OFF;
ALTER TABLE ORDER_LINE4 DROP CONSTRAINT ORDER_LINE4CKC;
ALTER TABLE ORDER_LINE4 ADD CONSTRAINT ORDER_LINE4CKC CHECK (OL_W_ID BETWEEN
2437 AND 3248);
SET INTEGRITY FOR ORDER_LINE4 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE5 OFF;
ALTER TABLE ORDER_LINE5 DROP CONSTRAINT ORDER_LINE5CKC;
ALTER TABLE ORDER_LINE5 ADD CONSTRAINT ORDER_LINE5CKC CHECK (OL_W_ID BETWEEN
3249 AND 4060);
SET INTEGRITY FOR ORDER_LINE5 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE6 OFF;
ALTER TABLE ORDER_LINE6 DROP CONSTRAINT ORDER_LINE6CKC;
ALTER TABLE ORDER_LINE6 ADD CONSTRAINT ORDER_LINE6CKC CHECK (OL_W_ID BETWEEN
4061 AND 4872);
SET INTEGRITY FOR ORDER_LINE6 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE7 OFF;
ALTER TABLE ORDER_LINE7 DROP CONSTRAINT ORDER_LINE7CKC;
ALTER TABLE ORDER_LINE7 ADD CONSTRAINT ORDER_LINE7CKC CHECK (OL_W_ID BETWEEN
4873 AND 5684);
SET INTEGRITY FOR ORDER_LINE7 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE8 OFF;
ALTER TABLE ORDER_LINE8 DROP CONSTRAINT ORDER_LINE8CKC;
ALTER TABLE ORDER_LINE8 ADD CONSTRAINT ORDER_LINE8CKC CHECK (OL_W_ID BETWEEN
5685 AND 6496);
SET INTEGRITY FOR ORDER_LINE8 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE9 OFF;
ALTER TABLE ORDER_LINE9 DROP CONSTRAINT ORDER_LINE9CKC;
ALTER TABLE ORDER_LINE9 ADD CONSTRAINT ORDER_LINE9CKC CHECK (OL_W_ID BETWEEN
6497 AND 7308);
SET INTEGRITY FOR ORDER_LINE9 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE10 OFF;
ALTER TABLE ORDER_LINE10 DROP CONSTRAINT ORDER_LINE10CKC;
ALTER TABLE ORDER_LINE10 ADD CONSTRAINT ORDER_LINE10CKC CHECK (OL_W_ID
BETWEEN 7309 AND 8120);
SET INTEGRITY FOR ORDER_LINE10 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE11 OFF;
ALTER TABLE ORDER_LINE11 DROP CONSTRAINT ORDER_LINE11CKC;
ALTER TABLE ORDER_LINE11 ADD CONSTRAINT ORDER_LINE11CKC CHECK (OL_W_ID
BETWEEN 8121 AND 8932);
SET INTEGRITY FOR ORDER_LINE11 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE12 OFF;
ALTER TABLE ORDER_LINE12 DROP CONSTRAINT ORDER_LINE12CKC;
ALTER TABLE ORDER_LINE12 ADD CONSTRAINT ORDER_LINE12CKC CHECK (OL_W_ID
BETWEEN 8933 AND 9744);
SET INTEGRITY FOR ORDER_LINE12 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE13 OFF;
ALTER TABLE ORDER_LINE13 DROP CONSTRAINT ORDER_LINE13CKC;
ALTER TABLE ORDER_LINE13 ADD CONSTRAINT ORDER_LINE13CKC CHECK (OL_W_ID
BETWEEN 9745 AND 10556);
SET INTEGRITY FOR ORDER_LINE13 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE14 OFF;
ALTER TABLE ORDER_LINE14 DROP CONSTRAINT ORDER_LINE14CKC;
ALTER TABLE ORDER_LINE14 ADD CONSTRAINT ORDER_LINE14CKC CHECK (OL_W_ID
BETWEEN 10557 AND 11368);
SET INTEGRITY FOR ORDER_LINE14 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE15 OFF;
ALTER TABLE ORDER_LINE15 DROP CONSTRAINT ORDER_LINE15CKC;
ALTER TABLE ORDER_LINE15 ADD CONSTRAINT ORDER_LINE15CKC CHECK (OL_W_ID
BETWEEN 11369 AND 12180);
SET INTEGRITY FOR ORDER_LINE15 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE16 OFF;
ALTER TABLE ORDER_LINE16 DROP CONSTRAINT ORDER_LINE16CKC;
```

```
ALTER TABLE ORDER_LINE16 ADD CONSTRAINT ORDER_LINE16CKC CHECK (OL_W_ID
BETWEEN 12181 AND 12992);
SET INTEGRITY FOR ORDER_LINE16 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE17 OFF;
ALTER TABLE ORDER_LINE17 DROP CONSTRAINT ORDER_LINE17CKC;
ALTER TABLE ORDER_LINE17 ADD CONSTRAINT ORDER_LINE17CKC CHECK (OL_W_ID
BETWEEN 12993 AND 13804);
SET INTEGRITY FOR ORDER_LINE17 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE18 OFF;
ALTER TABLE ORDER_LINE18 DROP CONSTRAINT ORDER_LINE18CKC;
ALTER TABLE ORDER_LINE18 ADD CONSTRAINT ORDER_LINE18CKC CHECK (OL_W_ID
BETWEEN 13805 AND 14616);
SET INTEGRITY FOR ORDER_LINE18 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE19 OFF;
ALTER TABLE ORDER_LINE19 DROP CONSTRAINT ORDER_LINE19CKC;
ALTER TABLE ORDER_LINE19 ADD CONSTRAINT ORDER_LINE19CKC CHECK (OL_W_ID
BETWEEN 14617 AND 15428);
SET INTEGRITY FOR ORDER_LINE19 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE20 OFF;
ALTER TABLE ORDER_LINE20 DROP CONSTRAINT ORDER_LINE20CKC;
ALTER TABLE ORDER_LINE20 ADD CONSTRAINT ORDER_LINE20CKC CHECK (OL_W_ID
BETWEEN 15429 AND 16240);
SET INTEGRITY FOR ORDER_LINE20 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE21 OFF;
ALTER TABLE ORDER_LINE21 DROP CONSTRAINT ORDER_LINE21CKC;
ALTER TABLE ORDER_LINE21 ADD CONSTRAINT ORDER_LINE21CKC CHECK (OL_W_ID
BETWEEN 16241 AND 17052);
SET INTEGRITY FOR ORDER_LINE21 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE22 OFF;
ALTER TABLE ORDER_LINE22 DROP CONSTRAINT ORDER_LINE22CKC;
ALTER TABLE ORDER_LINE22 ADD CONSTRAINT ORDER_LINE22CKC CHECK (OL_W_ID
BETWEEN 17053 AND 17864);
SET INTEGRITY FOR ORDER_LINE22 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE23 OFF;
ALTER TABLE ORDER_LINE23 DROP CONSTRAINT ORDER_LINE23CKC;
ALTER TABLE ORDER_LINE23 ADD CONSTRAINT ORDER_LINE23CKC CHECK (OL_W_ID
BETWEEN 17865 AND 18676);
SET INTEGRITY FOR ORDER_LINE23 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE24 OFF;
ALTER TABLE ORDER_LINE24 DROP CONSTRAINT ORDER_LINE24CKC;
ALTER TABLE ORDER_LINE24 ADD CONSTRAINT ORDER_LINE24CKC CHECK (OL_W_ID
BETWEEN 18677 AND 19488);
SET INTEGRITY FOR ORDER_LINE24 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE25 OFF;
ALTER TABLE ORDER_LINE25 DROP CONSTRAINT ORDER_LINE25CKC;
ALTER TABLE ORDER_LINE25 ADD CONSTRAINT ORDER_LINE25CKC CHECK (OL_W_ID
BETWEEN 19489 AND 20300);
SET INTEGRITY FOR ORDER_LINE25 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE26 OFF;
ALTER TABLE ORDER_LINE26 DROP CONSTRAINT ORDER_LINE26CKC;
ALTER TABLE ORDER_LINE26 ADD CONSTRAINT ORDER_LINE26CKC CHECK (OL_W_ID
BETWEEN 20301 AND 21112);
SET INTEGRITY FOR ORDER_LINE26 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE27 OFF;
ALTER TABLE ORDER_LINE27 DROP CONSTRAINT ORDER_LINE27CKC;
ALTER TABLE ORDER_LINE27 ADD CONSTRAINT ORDER_LINE27CKC CHECK (OL_W_ID
BETWEEN 21113 AND 21924);
SET INTEGRITY FOR ORDER_LINE27 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE28 OFF;
ALTER TABLE ORDER_LINE28 DROP CONSTRAINT ORDER_LINE28CKC;
ALTER TABLE ORDER_LINE28 ADD CONSTRAINT ORDER_LINE28CKC CHECK (OL_W_ID
BETWEEN 21925 AND 22736);
SET INTEGRITY FOR ORDER_LINE28 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE29 OFF;
ALTER TABLE ORDER_LINE29 DROP CONSTRAINT ORDER_LINE29CKC;
ALTER TABLE ORDER_LINE29 ADD CONSTRAINT ORDER_LINE29CKC CHECK (OL_W_ID
BETWEEN 22737 AND 23548);
SET INTEGRITY FOR ORDER_LINE29 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE30 OFF;
ALTER TABLE ORDER_LINE30 DROP CONSTRAINT ORDER_LINE30CKC;
ALTER TABLE ORDER_LINE30 ADD CONSTRAINT ORDER_LINE30CKC CHECK (OL_W_ID
BETWEEN 23549 AND 24360);
SET INTEGRITY FOR ORDER_LINE30 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE31 OFF;
ALTER TABLE ORDER_LINE31 DROP CONSTRAINT ORDER_LINE31CKC;
ALTER TABLE ORDER_LINE31 ADD CONSTRAINT ORDER_LINE31CKC CHECK (OL_W_ID
BETWEEN 24361 AND 25172);
SET INTEGRITY FOR ORDER_LINE31 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE32 OFF;
ALTER TABLE ORDER_LINE32 DROP CONSTRAINT ORDER_LINE32CKC;
ALTER TABLE ORDER_LINE32 ADD CONSTRAINT ORDER_LINE32CKC CHECK (OL_W_ID
BETWEEN 25173 AND 25984);
SET INTEGRITY FOR ORDER_LINE32 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE33 OFF;
ALTER TABLE ORDER_LINE33 DROP CONSTRAINT ORDER_LINE33CKC;
ALTER TABLE ORDER_LINE33 ADD CONSTRAINT ORDER_LINE33CKC CHECK (OL_W_ID
BETWEEN 25985 AND 26796);
SET INTEGRITY FOR ORDER_LINE33 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE34 OFF;
ALTER TABLE ORDER_LINE34 DROP CONSTRAINT ORDER_LINE34CKC;
ALTER TABLE ORDER_LINE34 ADD CONSTRAINT ORDER_LINE34CKC CHECK (OL_W_ID
BETWEEN 26797 AND 27608);
SET INTEGRITY FOR ORDER_LINE34 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE35 OFF;
ALTER TABLE ORDER_LINE35 DROP CONSTRAINT ORDER_LINE35CKC;
ALTER TABLE ORDER_LINE35 ADD CONSTRAINT ORDER_LINE35CKC CHECK (OL_W_ID
BETWEEN 27609 AND 28420);
SET INTEGRITY FOR ORDER_LINE35 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE36 OFF;
ALTER TABLE ORDER_LINE36 DROP CONSTRAINT ORDER_LINE36CKC;
ALTER TABLE ORDER_LINE36 ADD CONSTRAINT ORDER_LINE36CKC CHECK (OL_W_ID
BETWEEN 28421 AND 29232);
SET INTEGRITY FOR ORDER_LINE36 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE37 OFF;
ALTER TABLE ORDER_LINE37 DROP CONSTRAINT ORDER_LINE37CKC;
ALTER TABLE ORDER_LINE37 ADD CONSTRAINT ORDER_LINE37CKC CHECK (OL_W_ID
BETWEEN 29233 AND 30044);
SET INTEGRITY FOR ORDER_LINE37 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE38 OFF;
ALTER TABLE ORDER_LINE38 DROP CONSTRAINT ORDER_LINE38CKC;
ALTER TABLE ORDER_LINE38 ADD CONSTRAINT ORDER_LINE38CKC CHECK (OL_W_ID
BETWEEN 30045 AND 30856);
SET INTEGRITY FOR ORDER_LINE38 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE39 OFF;
ALTER TABLE ORDER_LINE39 DROP CONSTRAINT ORDER_LINE39CKC;
ALTER TABLE ORDER_LINE39 ADD CONSTRAINT ORDER_LINE39CKC CHECK (OL_W_ID
BETWEEN 30857 AND 31668);
SET INTEGRITY FOR ORDER_LINE39 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE40 OFF;
ALTER TABLE ORDER_LINE40 DROP CONSTRAINT ORDER_LINE40CKC;
ALTER TABLE ORDER_LINE40 ADD CONSTRAINT ORDER_LINE40CKC CHECK (OL_W_ID
BETWEEN 31669 AND 32480);
SET INTEGRITY FOR ORDER_LINE40 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE41 OFF;
ALTER TABLE ORDER_LINE41 DROP CONSTRAINT ORDER_LINE41CKC;
ALTER TABLE ORDER_LINE41 ADD CONSTRAINT ORDER_LINE41CKC CHECK (OL_W_ID
BETWEEN 32481 AND 33292);
SET INTEGRITY FOR ORDER_LINE41 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE42 OFF;
ALTER TABLE ORDER_LINE42 DROP CONSTRAINT ORDER_LINE42CKC;
ALTER TABLE ORDER_LINE42 ADD CONSTRAINT ORDER_LINE42CKC CHECK (OL_W_ID
BETWEEN 33293 AND 34104);
SET INTEGRITY FOR ORDER_LINE42 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE43 OFF;
ALTER TABLE ORDER_LINE43 DROP CONSTRAINT ORDER_LINE43CKC;
ALTER TABLE ORDER_LINE43 ADD CONSTRAINT ORDER_LINE43CKC CHECK (OL_W_ID
BETWEEN 34105 AND 34916);
SET INTEGRITY FOR ORDER_LINE43 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE44 OFF;
ALTER TABLE ORDER_LINE44 DROP CONSTRAINT ORDER_LINE44CKC;
ALTER TABLE ORDER_LINE44 ADD CONSTRAINT ORDER_LINE44CKC CHECK (OL_W_ID
BETWEEN 34917 AND 35728);
SET INTEGRITY FOR ORDER_LINE44 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE45 OFF;
ALTER TABLE ORDER_LINE45 DROP CONSTRAINT ORDER_LINE45CKC;
ALTER TABLE ORDER_LINE45 ADD CONSTRAINT ORDER_LINE45CKC CHECK (OL_W_ID
BETWEEN 35729 AND 36540);
SET INTEGRITY FOR ORDER_LINE45 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE46 OFF;
ALTER TABLE ORDER_LINE46 DROP CONSTRAINT ORDER_LINE46CKC;
ALTER TABLE ORDER_LINE46 ADD CONSTRAINT ORDER_LINE46CKC CHECK (OL_W_ID
BETWEEN 36541 AND 37352);
SET INTEGRITY FOR ORDER_LINE46 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE47 OFF;
ALTER TABLE ORDER_LINE47 DROP CONSTRAINT ORDER_LINE47CKC;
ALTER TABLE ORDER_LINE47 ADD CONSTRAINT ORDER_LINE47CKC CHECK (OL_W_ID
BETWEEN 37353 AND 38164);
```

```
SET INTEGRITY FOR ORDER_LINE47 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE48 OFF;
ALTER TABLE ORDER_LINE48 DROP CONSTRAINT ORDER_LINE48CKC;
ALTER TABLE ORDER_LINE48 ADD CONSTRAINT ORDER_LINE48CKC CHECK (OL_W_ID
BETWEEN 38165 AND 38976);
SET INTEGRITY FOR ORDER_LINE48 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE49 OFF;
ALTER TABLE ORDER_LINE49 DROP CONSTRAINT ORDER_LINE49CKC;
ALTER TABLE ORDER_LINE49 ADD CONSTRAINT ORDER_LINE49CKC CHECK (OL_W_ID
BETWEEN 38977 AND 39788);
SET INTEGRITY FOR ORDER_LINE49 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE50 OFF;
ALTER TABLE ORDER_LINE50 DROP CONSTRAINT ORDER_LINE50CKC;
ALTER TABLE ORDER_LINE50 ADD CONSTRAINT ORDER_LINE50CKC CHECK (OL_W_ID
BETWEEN 39789 AND 40600);
SET INTEGRITY FOR ORDER_LINE50 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE51 OFF;
ALTER TABLE ORDER_LINE51 DROP CONSTRAINT ORDER_LINE51CKC;
ALTER TABLE ORDER_LINE51 ADD CONSTRAINT ORDER_LINE51CKC CHECK (OL_W_ID
BETWEEN 40601 AND 41412);
SET INTEGRITY FOR ORDER_LINE51 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE52 OFF;
ALTER TABLE ORDER_LINE52 DROP CONSTRAINT ORDER_LINE52CKC;
ALTER TABLE ORDER_LINE52 ADD CONSTRAINT ORDER_LINE52CKC CHECK (OL_W_ID
BETWEEN 41413 AND 42224);
SET INTEGRITY FOR ORDER_LINE52 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE53 OFF;
ALTER TABLE ORDER_LINE53 DROP CONSTRAINT ORDER_LINE53CKC;
ALTER TABLE ORDER_LINE53 ADD CONSTRAINT ORDER_LINE53CKC CHECK (OL_W_ID
BETWEEN 42225 AND 43036);
SET INTEGRITY FOR ORDER_LINE53 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE54 OFF;
ALTER TABLE ORDER_LINE54 DROP CONSTRAINT ORDER_LINE54CKC;
ALTER TABLE ORDER_LINE54 ADD CONSTRAINT ORDER_LINE54CKC CHECK (OL_W_ID
BETWEEN 43037 AND 43848);
SET INTEGRITY FOR ORDER_LINE54 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE55 OFF;
ALTER TABLE ORDER_LINE55 DROP CONSTRAINT ORDER_LINE55CKC;
ALTER TABLE ORDER_LINE55 ADD CONSTRAINT ORDER_LINE55CKC CHECK (OL_W_ID
BETWEEN 43849 AND 44660);
SET INTEGRITY FOR ORDER_LINE55 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE56 OFF;
ALTER TABLE ORDER_LINE56 DROP CONSTRAINT ORDER_LINE56CKC;
ALTER TABLE ORDER_LINE56 ADD CONSTRAINT ORDER_LINE56CKC CHECK (OL_W_ID
BETWEEN 44661 AND 45472);
SET INTEGRITY FOR ORDER_LINE56 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE57 OFF;
ALTER TABLE ORDER_LINE57 DROP CONSTRAINT ORDER_LINE57CKC;
ALTER TABLE ORDER_LINE57 ADD CONSTRAINT ORDER_LINE57CKC CHECK (OL_W_ID
BETWEEN 45473 AND 46284);
SET INTEGRITY FOR ORDER_LINE57 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE58 OFF;
ALTER TABLE ORDER_LINE58 DROP CONSTRAINT ORDER_LINE58CKC;
ALTER TABLE ORDER_LINE58 ADD CONSTRAINT ORDER_LINE58CKC CHECK (OL_W_ID
BETWEEN 46285 AND 47096);
SET INTEGRITY FOR ORDER_LINE58 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE59 OFF;
ALTER TABLE ORDER_LINE59 DROP CONSTRAINT ORDER_LINE59CKC;
ALTER TABLE ORDER_LINE59 ADD CONSTRAINT ORDER_LINE59CKC CHECK (OL_W_ID
BETWEEN 47097 AND 47908);
SET INTEGRITY FOR ORDER_LINE59 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE60 OFF;
ALTER TABLE ORDER_LINE60 DROP CONSTRAINT ORDER_LINE60CKC;
ALTER TABLE ORDER_LINE60 ADD CONSTRAINT ORDER_LINE60CKC CHECK (OL_W_ID
BETWEEN 47909 AND 48720);
SET INTEGRITY FOR ORDER_LINE60 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE61 OFF;
ALTER TABLE ORDER_LINE61 DROP CONSTRAINT ORDER_LINE61CKC;
ALTER TABLE ORDER_LINE61 ADD CONSTRAINT ORDER_LINE61CKC CHECK (OL_W_ID
BETWEEN 48721 AND 49532);
SET INTEGRITY FOR ORDER_LINE61 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE62 OFF;
ALTER TABLE ORDER_LINE62 DROP CONSTRAINT ORDER_LINE62CKC;
ALTER TABLE ORDER_LINE62 ADD CONSTRAINT ORDER_LINE62CKC CHECK (OL_W_ID
BETWEEN 49533 AND 50344);
SET INTEGRITY FOR ORDER_LINE62 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE63 OFF;
ALTER TABLE ORDER_LINE63 DROP CONSTRAINT ORDER_LINE63CKC;
ALTER TABLE ORDER_LINE63 ADD CONSTRAINT ORDER_LINE63CKC CHECK (OL_W_ID
BETWEEN 50345 AND 51156);
SET INTEGRITY FOR ORDER_LINE63 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE64 OFF;
ALTER TABLE ORDER_LINE64 DROP CONSTRAINT ORDER_LINE64CKC;
ALTER TABLE ORDER_LINE64 ADD CONSTRAINT ORDER_LINE64CKC CHECK (OL_W_ID >=
51157);
SET INTEGRITY FOR ORDER_LINE64 ALL IMMEDIATE UNCHECKED;
connect reset;
```

DDL/CRCONST_STOCK.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK1 OFF;
ALTER TABLE STOCK1 DROP CONSTRAINT STOCK1CKC;
ALTER TABLE STOCK1 ADD CONSTRAINT STOCK1CKC CHECK (S_W_ID BETWEEN 1 AND 812);
SET INTEGRITY FOR STOCK1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK2 OFF;
ALTER TABLE STOCK2 DROP CONSTRAINT STOCK2CKC;
ALTER TABLE STOCK2 ADD CONSTRAINT STOCK2CKC CHECK (S_W_ID BETWEEN 813 AND 1624);
SET INTEGRITY FOR STOCK2 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK3 OFF;
ALTER TABLE STOCK3 DROP CONSTRAINT STOCK3CKC;
ALTER TABLE STOCK3 ADD CONSTRAINT STOCK3CKC CHECK (S_W_ID BETWEEN 1625 AND
2436);
SET INTEGRITY FOR STOCK3 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK4 OFF;
ALTER TABLE STOCK4 DROP CONSTRAINT STOCK4CKC;
ALTER TABLE STOCK4 ADD CONSTRAINT STOCK4CKC CHECK (S_W_ID BETWEEN 2437 AND
3248);
SET INTEGRITY FOR STOCK4 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK5 OFF;
ALTER TABLE STOCK5 DROP CONSTRAINT STOCK5CKC;
ALTER TABLE STOCK5 ADD CONSTRAINT STOCK5CKC CHECK (S_W_ID BETWEEN 3249 AND
4060);
SET INTEGRITY FOR STOCK5 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK6 OFF;
ALTER TABLE STOCK6 DROP CONSTRAINT STOCK6CKC;
ALTER TABLE STOCK6 ADD CONSTRAINT STOCK6CKC CHECK (S_W_ID BETWEEN 4061 AND
4872);
SET INTEGRITY FOR STOCK6 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK7 OFF;
ALTER TABLE STOCK7 DROP CONSTRAINT STOCK7CKC;
ALTER TABLE STOCK7 ADD CONSTRAINT STOCK7CKC CHECK (S_W_ID BETWEEN 4873 AND
5684);
SET INTEGRITY FOR STOCK7 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK8 OFF;
ALTER TABLE STOCK8 DROP CONSTRAINT STOCK8CKC;
ALTER TABLE STOCK8 ADD CONSTRAINT STOCK8CKC CHECK (S_W_ID BETWEEN 5685 AND
6496);
SET INTEGRITY FOR STOCK8 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK9 OFF;
ALTER TABLE STOCK9 DROP CONSTRAINT STOCK9CKC;
ALTER TABLE STOCK9 ADD CONSTRAINT STOCK9CKC CHECK (S_W_ID BETWEEN 6497 AND
7308);
SET INTEGRITY FOR STOCK9 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK10 OFF;
ALTER TABLE STOCK10 DROP CONSTRAINT STOCK10CKC;
ALTER TABLE STOCK10 ADD CONSTRAINT STOCK10CKC CHECK (S_W_ID BETWEEN 7309 AND
8120);
SET INTEGRITY FOR STOCK10 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK11 OFF;
ALTER TABLE STOCK11 DROP CONSTRAINT STOCK11CKC;
ALTER TABLE STOCK11 ADD CONSTRAINT STOCK11CKC CHECK (S_W_ID BETWEEN 8121 AND
8932);
SET INTEGRITY FOR STOCK11 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK12 OFF;
ALTER TABLE STOCK12 DROP CONSTRAINT STOCK12CKC;
ALTER TABLE STOCK12 ADD CONSTRAINT STOCK12CKC CHECK (S_W_ID BETWEEN 8933 AND
9744);
SET INTEGRITY FOR STOCK12 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK13 OFF;
ALTER TABLE STOCK13 DROP CONSTRAINT STOCK13CKC;
ALTER TABLE STOCK13 ADD CONSTRAINT STOCK13CKC CHECK (S_W_ID BETWEEN 9745 AND
10556);
SET INTEGRITY FOR STOCK13 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK14 OFF;
```

```
ALTER TABLE STOCK14 DROP CONSTRAINT STOCK14CKC;
ALTER TABLE STOCK14 ADD CONSTRAINT STOCK14CKC CHECK (S_W_ID BETWEEN 10557 AND 11368);
SET INTEGRITY FOR STOCK14 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK15 OFF;
ALTER TABLE STOCK15 DROP CONSTRAINT STOCK15CKC;
ALTER TABLE STOCK15 ADD CONSTRAINT STOCK15CKC CHECK (S_W_ID BETWEEN 11369 AND 12180);
SET INTEGRITY FOR STOCK15 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK16 OFF;
ALTER TABLE STOCK16 DROP CONSTRAINT STOCK16CKC;
ALTER TABLE STOCK16 ADD CONSTRAINT STOCK16CKC CHECK (S_W_ID BETWEEN 12181 AND 12992);
SET INTEGRITY FOR STOCK16 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK17 OFF;
ALTER TABLE STOCK17 DROP CONSTRAINT STOCK17CKC;
ALTER TABLE STOCK17 ADD CONSTRAINT STOCK17CKC CHECK (S_W_ID BETWEEN 12993 AND 13804);
SET INTEGRITY FOR STOCK17 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK18 OFF;
ALTER TABLE STOCK18 DROP CONSTRAINT STOCK18CKC;
ALTER TABLE STOCK18 ADD CONSTRAINT STOCK18CKC CHECK (S_W_ID BETWEEN 13805 AND 14616);
SET INTEGRITY FOR STOCK18 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK19 OFF;
ALTER TABLE STOCK19 DROP CONSTRAINT STOCK19CKC;
ALTER TABLE STOCK19 ADD CONSTRAINT STOCK19CKC CHECK (S_W_ID BETWEEN 14617 AND 15428);
SET INTEGRITY FOR STOCK19 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK20 OFF;
ALTER TABLE STOCK20 DROP CONSTRAINT STOCK20CKC;
ALTER TABLE STOCK20 ADD CONSTRAINT STOCK20CKC CHECK (S_W_ID BETWEEN 15429 AND 16240);
SET INTEGRITY FOR STOCK20 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK21 OFF;
ALTER TABLE STOCK21 DROP CONSTRAINT STOCK21CKC;
ALTER TABLE STOCK21 ADD CONSTRAINT STOCK21CKC CHECK (S_W_ID BETWEEN 16241 AND 17052);
SET INTEGRITY FOR STOCK21 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK22 OFF;
ALTER TABLE STOCK22 DROP CONSTRAINT STOCK22CKC;
ALTER TABLE STOCK22 ADD CONSTRAINT STOCK22CKC CHECK (S_W_ID BETWEEN 17053 AND 17864);
SET INTEGRITY FOR STOCK22 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK23 OFF;
ALTER TABLE STOCK23 DROP CONSTRAINT STOCK23CKC;
ALTER TABLE STOCK23 ADD CONSTRAINT STOCK23CKC CHECK (S_W_ID BETWEEN 17865 AND 18676);
SET INTEGRITY FOR STOCK23 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK24 OFF;
ALTER TABLE STOCK24 DROP CONSTRAINT STOCK24CKC;
ALTER TABLE STOCK24 ADD CONSTRAINT STOCK24CKC CHECK (S_W_ID BETWEEN 18677 AND 19488);
SET INTEGRITY FOR STOCK24 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK25 OFF;
ALTER TABLE STOCK25 DROP CONSTRAINT STOCK25CKC;
ALTER TABLE STOCK25 ADD CONSTRAINT STOCK25CKC CHECK (S_W_ID BETWEEN 19489 AND 20300);
SET INTEGRITY FOR STOCK25 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK26 OFF;
ALTER TABLE STOCK26 DROP CONSTRAINT STOCK26CKC;
ALTER TABLE STOCK26 ADD CONSTRAINT STOCK26CKC CHECK (S_W_ID BETWEEN 20301 AND 21112);
SET INTEGRITY FOR STOCK26 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK27 OFF;
ALTER TABLE STOCK27 DROP CONSTRAINT STOCK27CKC;
ALTER TABLE STOCK27 ADD CONSTRAINT STOCK27CKC CHECK (S_W_ID BETWEEN 21113 AND 21924);
SET INTEGRITY FOR STOCK27 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK28 OFF;
ALTER TABLE STOCK28 DROP CONSTRAINT STOCK28CKC;
ALTER TABLE STOCK28 ADD CONSTRAINT STOCK28CKC CHECK (S_W_ID BETWEEN 21925 AND 22736);
SET INTEGRITY FOR STOCK28 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK29 OFF;
ALTER TABLE STOCK29 DROP CONSTRAINT STOCK29CKC;
ALTER TABLE STOCK29 ADD CONSTRAINT STOCK29CKC CHECK (S_W_ID BETWEEN 22737 AND 23548);
SET INTEGRITY FOR STOCK29 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK30 OFF;
ALTER TABLE STOCK30 DROP CONSTRAINT STOCK30CKC;
ALTER TABLE STOCK30 ADD CONSTRAINT STOCK30CKC CHECK (S_W_ID BETWEEN 23549 AND 24360);
SET INTEGRITY FOR STOCK30 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK31 OFF;
ALTER TABLE STOCK31 DROP CONSTRAINT STOCK31CKC;
ALTER TABLE STOCK31 ADD CONSTRAINT STOCK31CKC CHECK (S_W_ID BETWEEN 24361 AND 25172);
SET INTEGRITY FOR STOCK31 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK32 OFF;
ALTER TABLE STOCK32 DROP CONSTRAINT STOCK32CKC;
ALTER TABLE STOCK32 ADD CONSTRAINT STOCK32CKC CHECK (S_W_ID BETWEEN 25173 AND 25984);
SET INTEGRITY FOR STOCK32 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK33 OFF;
ALTER TABLE STOCK33 DROP CONSTRAINT STOCK33CKC;
ALTER TABLE STOCK33 ADD CONSTRAINT STOCK33CKC CHECK (S_W_ID BETWEEN 25985 AND 26796);
SET INTEGRITY FOR STOCK33 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK34 OFF;
ALTER TABLE STOCK34 DROP CONSTRAINT STOCK34CKC;
ALTER TABLE STOCK34 ADD CONSTRAINT STOCK34CKC CHECK (S_W_ID BETWEEN 26797 AND 27608);
SET INTEGRITY FOR STOCK34 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK35 OFF;
ALTER TABLE STOCK35 DROP CONSTRAINT STOCK35CKC;
ALTER TABLE STOCK35 ADD CONSTRAINT STOCK35CKC CHECK (S_W_ID BETWEEN 27609 AND 28420);
SET INTEGRITY FOR STOCK35 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK36 OFF;
ALTER TABLE STOCK36 DROP CONSTRAINT STOCK36CKC;
ALTER TABLE STOCK36 ADD CONSTRAINT STOCK36CKC CHECK (S_W_ID BETWEEN 28421 AND 29232);
SET INTEGRITY FOR STOCK36 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK37 OFF;
ALTER TABLE STOCK37 DROP CONSTRAINT STOCK37CKC;
ALTER TABLE STOCK37 ADD CONSTRAINT STOCK37CKC CHECK (S_W_ID BETWEEN 29233 AND 30044);
SET INTEGRITY FOR STOCK37 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK38 OFF;
ALTER TABLE STOCK38 DROP CONSTRAINT STOCK38CKC;
ALTER TABLE STOCK38 ADD CONSTRAINT STOCK38CKC CHECK (S_W_ID BETWEEN 30045 AND 30856);
SET INTEGRITY FOR STOCK38 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK39 OFF;
ALTER TABLE STOCK39 DROP CONSTRAINT STOCK39CKC;
ALTER TABLE STOCK39 ADD CONSTRAINT STOCK39CKC CHECK (S_W_ID BETWEEN 30857 AND 31668);
SET INTEGRITY FOR STOCK39 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK40 OFF;
ALTER TABLE STOCK40 DROP CONSTRAINT STOCK40CKC;
ALTER TABLE STOCK40 ADD CONSTRAINT STOCK40CKC CHECK (S_W_ID BETWEEN 31669 AND 32480);
SET INTEGRITY FOR STOCK40 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK41 OFF;
ALTER TABLE STOCK41 DROP CONSTRAINT STOCK41CKC;
ALTER TABLE STOCK41 ADD CONSTRAINT STOCK41CKC CHECK (S_W_ID BETWEEN 32481 AND 33292);
SET INTEGRITY FOR STOCK41 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK42 OFF;
ALTER TABLE STOCK42 DROP CONSTRAINT STOCK42CKC;
ALTER TABLE STOCK42 ADD CONSTRAINT STOCK42CKC CHECK (S_W_ID BETWEEN 33293 AND 34104);
SET INTEGRITY FOR STOCK42 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK43 OFF;
ALTER TABLE STOCK43 DROP CONSTRAINT STOCK43CKC;
ALTER TABLE STOCK43 ADD CONSTRAINT STOCK43CKC CHECK (S_W_ID BETWEEN 34105 AND 34916);
SET INTEGRITY FOR STOCK43 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK44 OFF;
ALTER TABLE STOCK44 DROP CONSTRAINT STOCK44CKC;
ALTER TABLE STOCK44 ADD CONSTRAINT STOCK44CKC CHECK (S_W_ID BETWEEN 34917 AND 35728);
SET INTEGRITY FOR STOCK44 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK45 OFF;
ALTER TABLE STOCK45 DROP CONSTRAINT STOCK45CKC;
```

```
ALTER TABLE STOCK45 ADD CONSTRAINT STOCK45CKC CHECK (S_W_ID BETWEEN 35729 AND
36540);
SET INTEGRITY FOR STOCK45 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK46 OFF;
ALTER TABLE STOCK46 DROP CONSTRAINT STOCK46CKC;
ALTER TABLE STOCK46 ADD CONSTRAINT STOCK46CKC CHECK (S_W_ID BETWEEN 36541 AND
37352);
SET INTEGRITY FOR STOCK46 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK47 OFF;
ALTER TABLE STOCK47 DROP CONSTRAINT STOCK47CKC;
ALTER TABLE STOCK47 ADD CONSTRAINT STOCK47CKC CHECK (S_W_ID BETWEEN 37353 AND
38164);
SET INTEGRITY FOR STOCK47 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK48 OFF;
ALTER TABLE STOCK48 DROP CONSTRAINT STOCK48CKC;
ALTER TABLE STOCK48 ADD CONSTRAINT STOCK48CKC CHECK (S_W_ID BETWEEN 38165 AND
38976);
SET INTEGRITY FOR STOCK48 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK49 OFF;
ALTER TABLE STOCK49 DROP CONSTRAINT STOCK49CKC;
ALTER TABLE STOCK49 ADD CONSTRAINT STOCK49CKC CHECK (S_W_ID BETWEEN 38977 AND
39788);
SET INTEGRITY FOR STOCK49 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK50 OFF;
ALTER TABLE STOCK50 DROP CONSTRAINT STOCK50CKC;
ALTER TABLE STOCK50 ADD CONSTRAINT STOCK50CKC CHECK (S_W_ID BETWEEN 39789 AND
40600);
SET INTEGRITY FOR STOCK50 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK51 OFF;
ALTER TABLE STOCK51 DROP CONSTRAINT STOCK51CKC;
ALTER TABLE STOCK51 ADD CONSTRAINT STOCK51CKC CHECK (S_W_ID BETWEEN 40601 AND
41412);
SET INTEGRITY FOR STOCK51 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK52 OFF;
ALTER TABLE STOCK52 DROP CONSTRAINT STOCK52CKC;
ALTER TABLE STOCK52 ADD CONSTRAINT STOCK52CKC CHECK (S_W_ID BETWEEN 41413 AND
42224);
SET INTEGRITY FOR STOCK52 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK53 OFF;
ALTER TABLE STOCK53 DROP CONSTRAINT STOCK53CKC;
ALTER TABLE STOCK53 ADD CONSTRAINT STOCK53CKC CHECK (S_W_ID BETWEEN 42225 AND
43036);
SET INTEGRITY FOR STOCK53 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK54 OFF;
ALTER TABLE STOCK54 DROP CONSTRAINT STOCK54CKC;
ALTER TABLE STOCK54 ADD CONSTRAINT STOCK54CKC CHECK (S_W_ID BETWEEN 43037 AND
43848);
SET INTEGRITY FOR STOCK54 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK55 OFF;
ALTER TABLE STOCK55 DROP CONSTRAINT STOCK55CKC;
ALTER TABLE STOCK55 ADD CONSTRAINT STOCK55CKC CHECK (S_W_ID BETWEEN 43849 AND
44660);
SET INTEGRITY FOR STOCK55 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK56 OFF;
ALTER TABLE STOCK56 DROP CONSTRAINT STOCK56CKC;
ALTER TABLE STOCK56 ADD CONSTRAINT STOCK56CKC CHECK (S_W_ID BETWEEN 44661 AND
45472);
SET INTEGRITY FOR STOCK56 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK57 OFF;
ALTER TABLE STOCK57 DROP CONSTRAINT STOCK57CKC;
ALTER TABLE STOCK57 ADD CONSTRAINT STOCK57CKC CHECK (S_W_ID BETWEEN 45473 AND
46284);
SET INTEGRITY FOR STOCK57 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK58 OFF;
ALTER TABLE STOCK58 DROP CONSTRAINT STOCK58CKC;
ALTER TABLE STOCK58 ADD CONSTRAINT STOCK58CKC CHECK (S_W_ID BETWEEN 46285 AND
47096);
SET INTEGRITY FOR STOCK58 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK59 OFF;
ALTER TABLE STOCK59 DROP CONSTRAINT STOCK59CKC;
ALTER TABLE STOCK59 ADD CONSTRAINT STOCK59CKC CHECK (S_W_ID BETWEEN 47097 AND
47908);
SET INTEGRITY FOR STOCK59 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK60 OFF;
ALTER TABLE STOCK60 DROP CONSTRAINT STOCK60CKC;
ALTER TABLE STOCK60 ADD CONSTRAINT STOCK60CKC CHECK (S_W_ID BETWEEN 47909 AND
48720);
SET INTEGRITY FOR STOCK60 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK61 OFF;
ALTER TABLE STOCK61 DROP CONSTRAINT STOCK61CKC;
ALTER TABLE STOCK61 ADD CONSTRAINT STOCK61CKC CHECK (S_W_ID BETWEEN 48721 AND
49532);
SET INTEGRITY FOR STOCK61 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK62 OFF;
ALTER TABLE STOCK62 DROP CONSTRAINT STOCK62CKC;
ALTER TABLE STOCK62 ADD CONSTRAINT STOCK62CKC CHECK (S_W_ID BETWEEN 49533 AND
50344);
SET INTEGRITY FOR STOCK62 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK63 OFF;
ALTER TABLE STOCK63 DROP CONSTRAINT STOCK63CKC;
ALTER TABLE STOCK63 ADD CONSTRAINT STOCK63CKC CHECK (S_W_ID BETWEEN 50345 AND
51156);
SET INTEGRITY FOR STOCK63 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK64 OFF;
ALTER TABLE STOCK64 DROP CONSTRAINT STOCK64CKC;
ALTER TABLE STOCK64 ADD CONSTRAINT STOCK64CKC CHECK (S_W_ID >= 51157);
SET INTEGRITY FOR STOCK64 ALL IMMEDIATE UNCHECKED;
connect reset;
```

DDL/CRCONST_WAREHOUSE.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE1 OFF;
ALTER TABLE WAREHOUSE1 DROP CONSTRAINT WAREHOUSE1CKC;
ALTER TABLE WAREHOUSE1 ADD CONSTRAINT WAREHOUSE1CKC CHECK (W_ID BETWEEN 1
AND 1624);
SET INTEGRITY FOR WAREHOUSE1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE2 OFF;
ALTER TABLE WAREHOUSE2 DROP CONSTRAINT WAREHOUSE2CKC;
ALTER TABLE WAREHOUSE2 ADD CONSTRAINT WAREHOUSE2CKC CHECK (W_ID BETWEEN
1625 AND 3248);
SET INTEGRITY FOR WAREHOUSE2 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE3 OFF;
ALTER TABLE WAREHOUSE3 DROP CONSTRAINT WAREHOUSE3CKC;
ALTER TABLE WAREHOUSE3 ADD CONSTRAINT WAREHOUSE3CKC CHECK (W_ID BETWEEN
3249 AND 4872);
SET INTEGRITY FOR WAREHOUSE3 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE4 OFF;
ALTER TABLE WAREHOUSE4 DROP CONSTRAINT WAREHOUSE4CKC;
ALTER TABLE WAREHOUSE4 ADD CONSTRAINT WAREHOUSE4CKC CHECK (W_ID BETWEEN
4873 AND 6496);
SET INTEGRITY FOR WAREHOUSE4 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE5 OFF;
ALTER TABLE WAREHOUSE5 DROP CONSTRAINT WAREHOUSE5CKC;
ALTER TABLE WAREHOUSE5 ADD CONSTRAINT WAREHOUSE5CKC CHECK (W_ID BETWEEN
6497 AND 8120);
SET INTEGRITY FOR WAREHOUSE5 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE6 OFF;
ALTER TABLE WAREHOUSE6 DROP CONSTRAINT WAREHOUSE6CKC;
ALTER TABLE WAREHOUSE6 ADD CONSTRAINT WAREHOUSE6CKC CHECK (W_ID BETWEEN
8121 AND 9744);
SET INTEGRITY FOR WAREHOUSE6 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE7 OFF;
ALTER TABLE WAREHOUSE7 DROP CONSTRAINT WAREHOUSE7CKC;
ALTER TABLE WAREHOUSE7 ADD CONSTRAINT WAREHOUSE7CKC CHECK (W_ID BETWEEN
9745 AND 11368);
SET INTEGRITY FOR WAREHOUSE7 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE8 OFF;
ALTER TABLE WAREHOUSE8 DROP CONSTRAINT WAREHOUSE8CKC;
ALTER TABLE WAREHOUSE8 ADD CONSTRAINT WAREHOUSE8CKC CHECK (W_ID BETWEEN
11369 AND 12992);
SET INTEGRITY FOR WAREHOUSE8 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE9 OFF;
ALTER TABLE WAREHOUSE9 DROP CONSTRAINT WAREHOUSE9CKC;
ALTER TABLE WAREHOUSE9 ADD CONSTRAINT WAREHOUSE9CKC CHECK (W_ID BETWEEN
12993 AND 14616);
SET INTEGRITY FOR WAREHOUSE9 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE10 OFF;
ALTER TABLE WAREHOUSE10 DROP CONSTRAINT WAREHOUSE10CKC;
ALTER TABLE WAREHOUSE10 ADD CONSTRAINT WAREHOUSE10CKC CHECK (W_ID BETWEEN
14617 AND 16240);
SET INTEGRITY FOR WAREHOUSE10 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE11 OFF;
ALTER TABLE WAREHOUSE11 DROP CONSTRAINT WAREHOUSE11CKC;
ALTER TABLE WAREHOUSE11 ADD CONSTRAINT WAREHOUSE11CKC CHECK (W_ID BETWEEN
16241 AND 17864);
SET INTEGRITY FOR WAREHOUSE11 ALL IMMEDIATE UNCHECKED;
```

```
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE12 OFF;
ALTER TABLE WAREHOUSE12 DROP CONSTRAINT WAREHOUSE12CKC;
ALTER TABLE WAREHOUSE12 ADD CONSTRAINT WAREHOUSE12CKC CHECK (W_ID BETWEEN
17865 AND 19488);
SET INTEGRITY FOR WAREHOUSE12 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE13 OFF;
ALTER TABLE WAREHOUSE13 DROP CONSTRAINT WAREHOUSE13CKC;
ALTER TABLE WAREHOUSE13 ADD CONSTRAINT WAREHOUSE13CKC CHECK (W_ID BETWEEN
19489 AND 21112);
SET INTEGRITY FOR WAREHOUSE13 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE14 OFF;
ALTER TABLE WAREHOUSE14 DROP CONSTRAINT WAREHOUSE14CKC;
ALTER TABLE WAREHOUSE14 ADD CONSTRAINT WAREHOUSE14CKC CHECK (W_ID BETWEEN
21113 AND 22736);
SET INTEGRITY FOR WAREHOUSE14 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE15 OFF;
ALTER TABLE WAREHOUSE15 DROP CONSTRAINT WAREHOUSE15CKC;
ALTER TABLE WAREHOUSE15 ADD CONSTRAINT WAREHOUSE15CKC CHECK (W_ID BETWEEN
22737 AND 24360);
SET INTEGRITY FOR WAREHOUSE15 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE16 OFF;
ALTER TABLE WAREHOUSE16 DROP CONSTRAINT WAREHOUSE16CKC;
ALTER TABLE WAREHOUSE16 ADD CONSTRAINT WAREHOUSE16CKC CHECK (W_ID BETWEEN
24361 AND 25984);
SET INTEGRITY FOR WAREHOUSE16 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE17 OFF;
ALTER TABLE WAREHOUSE17 DROP CONSTRAINT WAREHOUSE17CKC;
ALTER TABLE WAREHOUSE17 ADD CONSTRAINT WAREHOUSE17CKC CHECK (W_ID BETWEEN
25985 AND 27608);
SET INTEGRITY FOR WAREHOUSE17 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE18 OFF;
ALTER TABLE WAREHOUSE18 DROP CONSTRAINT WAREHOUSE18CKC;
ALTER TABLE WAREHOUSE18 ADD CONSTRAINT WAREHOUSE18CKC CHECK (W_ID BETWEEN
27609 AND 29232);
SET INTEGRITY FOR WAREHOUSE18 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE19 OFF;
ALTER TABLE WAREHOUSE19 DROP CONSTRAINT WAREHOUSE19CKC;
ALTER TABLE WAREHOUSE19 ADD CONSTRAINT WAREHOUSE19CKC CHECK (W_ID BETWEEN
29233 AND 30856);
SET INTEGRITY FOR WAREHOUSE19 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE20 OFF;
ALTER TABLE WAREHOUSE20 DROP CONSTRAINT WAREHOUSE20CKC;
ALTER TABLE WAREHOUSE20 ADD CONSTRAINT WAREHOUSE20CKC CHECK (W_ID BETWEEN
30857 AND 32480);
SET INTEGRITY FOR WAREHOUSE20 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE21 OFF;
ALTER TABLE WAREHOUSE21 DROP CONSTRAINT WAREHOUSE21CKC;
ALTER TABLE WAREHOUSE21 ADD CONSTRAINT WAREHOUSE21CKC CHECK (W_ID BETWEEN
32481 AND 34104);
SET INTEGRITY FOR WAREHOUSE21 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE22 OFF;
ALTER TABLE WAREHOUSE22 DROP CONSTRAINT WAREHOUSE22CKC;
ALTER TABLE WAREHOUSE22 ADD CONSTRAINT WAREHOUSE22CKC CHECK (W_ID BETWEEN
34105 AND 35728);
SET INTEGRITY FOR WAREHOUSE22 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE23 OFF;
ALTER TABLE WAREHOUSE23 DROP CONSTRAINT WAREHOUSE23CKC;
ALTER TABLE WAREHOUSE23 ADD CONSTRAINT WAREHOUSE23CKC CHECK (W_ID BETWEEN
35729 AND 37352);
SET INTEGRITY FOR WAREHOUSE23 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE24 OFF;
ALTER TABLE WAREHOUSE24 DROP CONSTRAINT WAREHOUSE24CKC;
ALTER TABLE WAREHOUSE24 ADD CONSTRAINT WAREHOUSE24CKC CHECK (W_ID BETWEEN
37353 AND 38976);
SET INTEGRITY FOR WAREHOUSE24 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE25 OFF;
ALTER TABLE WAREHOUSE25 DROP CONSTRAINT WAREHOUSE25CKC;
ALTER TABLE WAREHOUSE25 ADD CONSTRAINT WAREHOUSE25CKC CHECK (W_ID BETWEEN
38977 AND 40600);
SET INTEGRITY FOR WAREHOUSE25 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE26 OFF;
ALTER TABLE WAREHOUSE26 DROP CONSTRAINT WAREHOUSE26CKC;
ALTER TABLE WAREHOUSE26 ADD CONSTRAINT WAREHOUSE26CKC CHECK (W_ID BETWEEN
40601 AND 42224);
SET INTEGRITY FOR WAREHOUSE26 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE27 OFF;
ALTER TABLE WAREHOUSE27 DROP CONSTRAINT WAREHOUSE27CKC;
ALTER TABLE WAREHOUSE27 ADD CONSTRAINT WAREHOUSE27CKC CHECK (W_ID BETWEEN
42225 AND 43848);
SET INTEGRITY FOR WAREHOUSE27 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE28 OFF;
ALTER TABLE WAREHOUSE28 DROP CONSTRAINT WAREHOUSE28CKC;
ALTER TABLE WAREHOUSE28 ADD CONSTRAINT WAREHOUSE28CKC CHECK (W_ID BETWEEN
43849 AND 45472);
SET INTEGRITY FOR WAREHOUSE28 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE29 OFF;
ALTER TABLE WAREHOUSE29 DROP CONSTRAINT WAREHOUSE29CKC;
ALTER TABLE WAREHOUSE29 ADD CONSTRAINT WAREHOUSE29CKC CHECK (W_ID BETWEEN
45473 AND 47096);
SET INTEGRITY FOR WAREHOUSE29 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE30 OFF;
ALTER TABLE WAREHOUSE30 DROP CONSTRAINT WAREHOUSE30CKC;
ALTER TABLE WAREHOUSE30 ADD CONSTRAINT WAREHOUSE30CKC CHECK (W_ID BETWEEN
47097 AND 48720);
SET INTEGRITY FOR WAREHOUSE30 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE31 OFF;
ALTER TABLE WAREHOUSE31 DROP CONSTRAINT WAREHOUSE31CKC;
ALTER TABLE WAREHOUSE31 ADD CONSTRAINT WAREHOUSE31CKC CHECK (W_ID BETWEEN
48721 AND 50344);
SET INTEGRITY FOR WAREHOUSE31 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE32 OFF;
ALTER TABLE WAREHOUSE32 DROP CONSTRAINT WAREHOUSE32CKC;
ALTER TABLE WAREHOUSE32 ADD CONSTRAINT WAREHOUSE32CKC CHECK (W_ID >= 50345);
SET INTEGRITY FOR WAREHOUSE32 ALL IMMEDIATE UNCHECKED;
connect reset;
```

DDL/CRIDX_CUST_IDXB.ddl

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB1;
CREATE INDEX CUST_IDXB1
        ON CUSTOMER1(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB2;
CREATE INDEX CUST_IDXB2
        ON CUSTOMER2(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB3;
CREATE INDEX CUST_IDXB3
        ON CUSTOMER3(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB4;
CREATE INDEX CUST_IDXB4
        ON CUSTOMER4(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB5;
CREATE INDEX CUST_IDXB5
        ON CUSTOMER5(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB6;
CREATE INDEX CUST_IDXB6
        ON CUSTOMER6(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB7;
CREATE INDEX CUST_IDXB7
        ON CUSTOMER7(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB8;
CREATE INDEX CUST_IDXB8
        ON CUSTOMER8(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB9;
CREATE INDEX CUST_IDXB9
        ON CUSTOMER9(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB10;
CREATE INDEX CUST_IDXB10
        ON CUSTOMER10(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB11;
CREATE INDEX CUST_IDXB11
        ON CUSTOMER11(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB12;
CREATE INDEX CUST_IDXB12
        ON CUSTOMER12(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB13;
CREATE INDEX CUST_IDXB13
        ON CUSTOMER13(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
```

```
connect to TPCC in share mode;
DROP INDEX CUST_IDXB14;
CREATE INDEX CUST_IDXB14
        ON CUSTOMER14(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB15;
CREATE INDEX CUST_IDXB15
        ON CUSTOMER15(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB16;
CREATE INDEX CUST_IDXB16
        ON CUSTOMER16(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB17;
CREATE INDEX CUST_IDXB17
        ON CUSTOMER17(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB18;
CREATE INDEX CUST_IDXB18
        ON CUSTOMER18(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB19;
CREATE INDEX CUST_IDXB19
        ON CUSTOMER19(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB20;
CREATE INDEX CUST_IDXB20
        ON CUSTOMER20(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB21;
CREATE INDEX CUST_IDXB21
        ON CUSTOMER21(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB22;
CREATE INDEX CUST_IDXB22
        ON CUSTOMER22(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB23;
CREATE INDEX CUST_IDXB23
        ON CUSTOMER23(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB24;
CREATE INDEX CUST_IDXB24
        ON CUSTOMER24(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB25;
CREATE INDEX CUST_IDXB25
        ON CUSTOMER25(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB26;
CREATE INDEX CUST_IDXB26
        ON CUSTOMER26(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB27;
CREATE INDEX CUST_IDXB27
        ON CUSTOMER27(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB28;
CREATE INDEX CUST_IDXB28
        ON CUSTOMER28(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB29;
CREATE INDEX CUST_IDXB29
        ON CUSTOMER29(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB30;
CREATE INDEX CUST_IDXB30
        ON CUSTOMER30(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB31;
CREATE INDEX CUST_IDXB31
        ON CUSTOMER31(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB32;
CREATE INDEX CUST_IDXB32
        ON CUSTOMER32(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB33;
CREATE INDEX CUST_IDXB33
        ON CUSTOMER33(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB34;
CREATE INDEX CUST_IDXB34
        ON CUSTOMER34(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB35;
CREATE INDEX CUST_IDXB35
        ON CUSTOMER35(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB36;
CREATE INDEX CUST_IDXB36
        ON CUSTOMER36(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB37;
CREATE INDEX CUST_IDXB37
        ON CUSTOMER37(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB38;
CREATE INDEX CUST_IDXB38
        ON CUSTOMER38(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB39;
CREATE INDEX CUST_IDXB39
        ON CUSTOMER39(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB40;
CREATE INDEX CUST_IDXB40
        ON CUSTOMER40(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB41;
CREATE INDEX CUST_IDXB41
        ON CUSTOMER41(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB42;
CREATE INDEX CUST_IDXB42
        ON CUSTOMER42(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB43;
CREATE INDEX CUST_IDXB43
        ON CUSTOMER43(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB44;
CREATE INDEX CUST_IDXB44
        ON CUSTOMER44(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB45;
CREATE INDEX CUST_IDXB45
        ON CUSTOMER45(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB46;
CREATE INDEX CUST_IDXB46
        ON CUSTOMER46(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB47;
CREATE INDEX CUST_IDXB47
        ON CUSTOMER47(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB48;
CREATE INDEX CUST_IDXB48
        ON CUSTOMER48(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB49;
CREATE INDEX CUST_IDXB49
        ON CUSTOMER49(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB50;
CREATE INDEX CUST_IDXB50
        ON CUSTOMER50(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB51;
CREATE INDEX CUST_IDXB51
        ON CUSTOMER51(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB52;
CREATE INDEX CUST_IDXB52
        ON CUSTOMER52(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB53;
CREATE INDEX CUST_IDXB53
        ON CUSTOMER53(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB54;
CREATE INDEX CUST_IDXB54
        ON CUSTOMER54(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB55;
CREATE INDEX CUST_IDXB55
        ON CUSTOMER55(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB56;
CREATE INDEX CUST_IDXB56
        ON CUSTOMER56(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB57;
CREATE INDEX CUST_IDXB57
        ON CUSTOMER57(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
```

```
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB58;
CREATE INDEX CUST_IDXB58
        ON CUSTOMER58(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB59;
CREATE INDEX CUST_IDXB59
        ON CUSTOMER59(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB60;
CREATE INDEX CUST_IDXB60
        ON CUSTOMER60(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB61;
CREATE INDEX CUST_IDXB61
        ON CUSTOMER61(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB62;
CREATE INDEX CUST_IDXB62
        ON CUSTOMER62(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB63;
CREATE INDEX CUST_IDXB63
        ON CUSTOMER63(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB64;
CREATE INDEX CUST_IDXB64
        ON CUSTOMER64(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
```

DDL/CRIDX_ORDR_IDXB.ddl

```
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB1;
CREATE INDEX ORDR_IDXB1
        ON ORDERS1(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB2;
CREATE INDEX ORDR_IDXB2
        ON ORDERS2(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB3;
CREATE INDEX ORDR_IDXB3
        ON ORDERS3(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB4;
CREATE INDEX ORDR_IDXB4
        ON ORDERS4(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB5;
CREATE INDEX ORDR_IDXB5
        ON ORDERS5(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB6;
CREATE INDEX ORDR_IDXB6
        ON ORDERS6(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB7;
CREATE INDEX ORDR_IDXB7
        ON ORDERS7(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB8;
CREATE INDEX ORDR_IDXB8
        ON ORDERS8(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB9;
CREATE INDEX ORDR_IDXB9
        ON ORDERS9(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB10;
CREATE INDEX ORDR_IDXB10
        ON ORDERS10(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB11;
CREATE INDEX ORDR_IDXB11
        ON ORDERS11(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB12;
CREATE INDEX ORDR_IDXB12
        ON ORDERS12(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB13;
CREATE INDEX ORDR_IDXB13
        ON ORDERS13(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB14;
CREATE INDEX ORDR_IDXB14
        ON ORDERS14(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB15;
CREATE INDEX ORDR_IDXB15
        ON ORDERS15(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB16;
CREATE INDEX ORDR_IDXB16
        ON ORDERS16(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB17;
CREATE INDEX ORDR_IDXB17
        ON ORDERS17(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB18;
CREATE INDEX ORDR_IDXB18
        ON ORDERS18(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB19;
CREATE INDEX ORDR_IDXB19
        ON ORDERS19(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB20;
CREATE INDEX ORDR_IDXB20
        ON ORDERS20(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB21;
CREATE INDEX ORDR_IDXB21
        ON ORDERS21(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB22;
CREATE INDEX ORDR_IDXB22
        ON ORDERS22(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB23;
CREATE INDEX ORDR_IDXB23
        ON ORDERS23(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB24;
CREATE INDEX ORDR_IDXB24
        ON ORDERS24(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB25;
CREATE INDEX ORDR_IDXB25
        ON ORDERS25(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB26;
CREATE INDEX ORDR_IDXB26
        ON ORDERS26(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB27;
CREATE INDEX ORDR_IDXB27
        ON ORDERS27(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB28;
CREATE INDEX ORDR_IDXB28
        ON ORDERS28(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB29;
CREATE INDEX ORDR_IDXB29
        ON ORDERS29(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB30;
CREATE INDEX ORDR_IDXB30
        ON ORDERS30(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB31;
CREATE INDEX ORDR_IDXB31
        ON ORDERS31(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB32;
CREATE INDEX ORDR_IDXB32
        ON ORDERS32(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB33;
CREATE INDEX ORDR_IDXB33
        ON ORDERS33(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB34;
CREATE INDEX ORDR_IDXB34
        ON ORDERS34(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB35;
CREATE INDEX ORDR_IDXB35
        ON ORDERS35(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
```

```
DROP INDEX ORDR_IDXB36;
CREATE INDEX ORDR_IDXB36
        ON ORDERS36(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB37;
CREATE INDEX ORDR_IDXB37
        ON ORDERS37(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB38;
CREATE INDEX ORDR_IDXB38
        ON ORDERS38(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB39;
CREATE INDEX ORDR_IDXB39
        ON ORDERS39(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB40;
CREATE INDEX ORDR_IDXB40
        ON ORDERS40(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB41;
CREATE INDEX ORDR_IDXB41
        ON ORDERS41(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB42;
CREATE INDEX ORDR_IDXB42
        ON ORDERS42(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB43;
CREATE INDEX ORDR_IDXB43
        ON ORDERS43(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB44;
CREATE INDEX ORDR_IDXB44
        ON ORDERS44(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB45;
CREATE INDEX ORDR_IDXB45
        ON ORDERS45(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB46;
CREATE INDEX ORDR_IDXB46
        ON ORDERS46(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB47;
CREATE INDEX ORDR_IDXB47
        ON ORDERS47(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB48;
CREATE INDEX ORDR_IDXB48
        ON ORDERS48(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB49;
CREATE INDEX ORDR_IDXB49
        ON ORDERS49(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB50;
CREATE INDEX ORDR_IDXB50
        ON ORDERS50(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB51;
CREATE INDEX ORDR_IDXB51
        ON ORDERS51(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB52;
CREATE INDEX ORDR_IDXB52
        ON ORDERS52(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB53;
CREATE INDEX ORDR_IDXB53
        ON ORDERS53(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB54;
CREATE INDEX ORDR_IDXB54
        ON ORDERS54(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB55;
CREATE INDEX ORDR_IDXB55
        ON ORDERS55(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB56;
CREATE INDEX ORDR_IDXB56
        ON ORDERS56(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB57;
CREATE INDEX ORDR_IDXB57
        ON ORDERS57(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB58;
CREATE INDEX ORDR_IDXB58
        ON ORDERS58(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB59;
CREATE INDEX ORDR_IDXB59
        ON ORDERS59(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB60;
CREATE INDEX ORDR_IDXB60
        ON ORDERS60(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB61;
CREATE INDEX ORDR_IDXB61
        ON ORDERS61(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB62;
CREATE INDEX ORDR_IDXB62
        ON ORDERS62(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB63;
CREATE INDEX ORDR_IDXB63
        ON ORDERS63(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB64;
CREATE INDEX ORDR_IDXB64
        ON ORDERS64(O_C_ID, O_W_ID, O_D_ID, O_ID DESC) PCTFREE 20;
connect reset;
```

DDL/CRTB_CUSTOMER.ddl

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER1;
CREATE TABLE CUSTOMER1
        (
        C_ID          INTEGER      NOT NULL,
        C_STATE       CHAR(2)      NOT NULL,
        C_ZIP         CHAR(9)      NOT NULL,
        C_PHONE       CHAR(16)     NOT NULL,
        C_SINCE       TIMESTAMP    NOT NULL,
        C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
        C_MIDDLE      CHAR(2)      NOT NULL,
        C_CREDIT      CHAR(2)      NOT NULL,
        C_DISCOUNT    REAL         NOT NULL,
        C_DATA        VARCHAR(500) NOT NULL,
        C_LAST        VARCHAR(16)  NOT NULL,
        C_FIRST       VARCHAR(16)  NOT NULL,
        C_STREET_1    VARCHAR(20)  NOT NULL,
        C_STREET_2    VARCHAR(20)  NOT NULL,
        C_CITY        VARCHAR(20)  NOT NULL,
        C_D_ID        SMALLINT     NOT NULL,
        C_W_ID        INTEGER      NOT NULL,
        C_DELIVERY_CNT INTEGER     NOT NULL,
        C_BALANCE     DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_001
        INDEX IN is_customer_001
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 1 ENDING AT 812,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER2;
CREATE TABLE CUSTOMER2
        (
        C_ID          INTEGER      NOT NULL,
        C_STATE       CHAR(2)      NOT NULL,
        C_ZIP         CHAR(9)      NOT NULL,
        C_PHONE       CHAR(16)     NOT NULL,
        C_SINCE       TIMESTAMP    NOT NULL,
        C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
        C_MIDDLE      CHAR(2)      NOT NULL,
        C_CREDIT      CHAR(2)      NOT NULL,
        C_DISCOUNT    REAL         NOT NULL,
        C_DATA        VARCHAR(500) NOT NULL,
        C_LAST        VARCHAR(16)  NOT NULL,
        C_FIRST       VARCHAR(16)  NOT NULL,
        C_STREET_1    VARCHAR(20)  NOT NULL,
        C_STREET_2    VARCHAR(20)  NOT NULL,
        C_CITY        VARCHAR(20)  NOT NULL,
        C_D_ID        SMALLINT     NOT NULL,
        C_W_ID        INTEGER      NOT NULL,
        C_DELIVERY_CNT INTEGER     NOT NULL,
        C_BALANCE     DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_002
        INDEX IN is_customer_002
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 813 ENDING AT 1624,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
```

```
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER3;
CREATE TABLE CUSTOMER3
        (
         C_ID          INTEGER      NOT NULL,
         C_STATE       CHAR(2)      NOT NULL,
         C_ZIP         CHAR(9)      NOT NULL,
         C_PHONE       CHAR(16)     NOT NULL,
         C_SINCE       TIMESTAMP    NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)      NOT NULL,
         C_CREDIT      CHAR(2)      NOT NULL,
         C_DISCOUNT    REAL         NOT NULL,
         C_DATA        VARCHAR(500) NOT NULL,
         C_LAST        VARCHAR(16)  NOT NULL,
         C_FIRST       VARCHAR(16)  NOT NULL,
         C_STREET_1    VARCHAR(20)  NOT NULL,
         C_STREET_2    VARCHAR(20)  NOT NULL,
         C_CITY        VARCHAR(20)  NOT NULL,
         C_D_ID        SMALLINT     NOT NULL,
         C_W_ID        INTEGER      NOT NULL,
         C_DELIVERY_CNT INTEGER     NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_003
        INDEX IN is_customer_003
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 1625 ENDING AT 2436,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER4;
CREATE TABLE CUSTOMER4
        (
         C_ID          INTEGER      NOT NULL,
         C_STATE       CHAR(2)      NOT NULL,
         C_ZIP         CHAR(9)      NOT NULL,
         C_PHONE       CHAR(16)     NOT NULL,
         C_SINCE       TIMESTAMP    NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)      NOT NULL,
         C_CREDIT      CHAR(2)      NOT NULL,
         C_DISCOUNT    REAL         NOT NULL,
         C_DATA        VARCHAR(500) NOT NULL,
         C_LAST        VARCHAR(16)  NOT NULL,
         C_FIRST       VARCHAR(16)  NOT NULL,
         C_STREET_1    VARCHAR(20)  NOT NULL,
         C_STREET_2    VARCHAR(20)  NOT NULL,
         C_CITY        VARCHAR(20)  NOT NULL,
         C_D_ID        SMALLINT     NOT NULL,
         C_W_ID        INTEGER      NOT NULL,
         C_DELIVERY_CNT INTEGER     NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_004
        INDEX IN is_customer_004
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 2437 ENDING AT 3248,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER5;
CREATE TABLE CUSTOMER5
        (
         C_ID          INTEGER      NOT NULL,
         C_STATE       CHAR(2)      NOT NULL,
         C_ZIP         CHAR(9)      NOT NULL,
         C_PHONE       CHAR(16)     NOT NULL,
         C_SINCE       TIMESTAMP    NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)      NOT NULL,
         C_CREDIT      CHAR(2)      NOT NULL,
         C_DISCOUNT    REAL         NOT NULL,
         C_DATA        VARCHAR(500) NOT NULL,
         C_LAST        VARCHAR(16)  NOT NULL,
         C_FIRST       VARCHAR(16)  NOT NULL,
         C_STREET_1    VARCHAR(20)  NOT NULL,
         C_STREET_2    VARCHAR(20)  NOT NULL,
         C_CITY        VARCHAR(20)  NOT NULL,
         C_D_ID        SMALLINT     NOT NULL,
         C_W_ID        INTEGER      NOT NULL,
         C_DELIVERY_CNT INTEGER     NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_005
        INDEX IN is_customer_005
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 3249 ENDING AT 4060,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER6;
CREATE TABLE CUSTOMER6
        (
         C_ID          INTEGER      NOT NULL,
         C_STATE       CHAR(2)      NOT NULL,
         C_ZIP         CHAR(9)      NOT NULL,
         C_PHONE       CHAR(16)     NOT NULL,
         C_SINCE       TIMESTAMP    NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)      NOT NULL,
         C_CREDIT      CHAR(2)      NOT NULL,
         C_DISCOUNT    REAL         NOT NULL,
         C_DATA        VARCHAR(500) NOT NULL,
         C_LAST        VARCHAR(16)  NOT NULL,
         C_FIRST       VARCHAR(16)  NOT NULL,
         C_STREET_1    VARCHAR(20)  NOT NULL,
         C_STREET_2    VARCHAR(20)  NOT NULL,
         C_CITY        VARCHAR(20)  NOT NULL,
         C_D_ID        SMALLINT     NOT NULL,
         C_W_ID        INTEGER      NOT NULL,
         C_DELIVERY_CNT INTEGER     NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_006
        INDEX IN is_customer_006
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 4061 ENDING AT 4872,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER7;
CREATE TABLE CUSTOMER7
        (
         C_ID          INTEGER      NOT NULL,
         C_STATE       CHAR(2)      NOT NULL,
         C_ZIP         CHAR(9)      NOT NULL,
         C_PHONE       CHAR(16)     NOT NULL,
         C_SINCE       TIMESTAMP    NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)      NOT NULL,
         C_CREDIT      CHAR(2)      NOT NULL,
         C_DISCOUNT    REAL         NOT NULL,
         C_DATA        VARCHAR(500) NOT NULL,
         C_LAST        VARCHAR(16)  NOT NULL,
         C_FIRST       VARCHAR(16)  NOT NULL,
         C_STREET_1    VARCHAR(20)  NOT NULL,
         C_STREET_2    VARCHAR(20)  NOT NULL,
         C_CITY        VARCHAR(20)  NOT NULL,
         C_D_ID        SMALLINT     NOT NULL,
         C_W_ID        INTEGER      NOT NULL,
         C_DELIVERY_CNT INTEGER     NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_007
        INDEX IN is_customer_007
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 4873 ENDING AT 5684,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER8;
CREATE TABLE CUSTOMER8
        (
         C_ID          INTEGER      NOT NULL,
         C_STATE       CHAR(2)      NOT NULL,
         C_ZIP         CHAR(9)      NOT NULL,
         C_PHONE       CHAR(16)     NOT NULL,
         C_SINCE       TIMESTAMP    NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)      NOT NULL,
         C_CREDIT      CHAR(2)      NOT NULL,
         C_DISCOUNT    REAL         NOT NULL,
         C_DATA        VARCHAR(500) NOT NULL,
         C_LAST        VARCHAR(16)  NOT NULL,
         C_FIRST       VARCHAR(16)  NOT NULL,
         C_STREET_1    VARCHAR(20)  NOT NULL,
         C_STREET_2    VARCHAR(20)  NOT NULL,
         C_CITY        VARCHAR(20)  NOT NULL,
         C_D_ID        SMALLINT     NOT NULL,
         C_W_ID        INTEGER      NOT NULL,
         C_DELIVERY_CNT INTEGER     NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_008
        INDEX IN is_customer_008
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 5685 ENDING AT 6496,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER9;
CREATE TABLE CUSTOMER9
        (
         C_ID          INTEGER      NOT NULL,
         C_STATE       CHAR(2)      NOT NULL,
         C_ZIP         CHAR(9)      NOT NULL,
         C_PHONE       CHAR(16)     NOT NULL,
```

```
C_SINCE        TIMESTAMP    NOT NULL,
C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
C_MIDDLE       CHAR(2)      NOT NULL,
C_CREDIT       CHAR(2)      NOT NULL,
C_DISCOUNT     REAL         NOT NULL,
C_DATA         VARCHAR(500) NOT NULL,
C_LAST         VARCHAR(16)  NOT NULL,
C_FIRST        VARCHAR(16)  NOT NULL,
C_STREET_1     VARCHAR(20)  NOT NULL,
C_STREET_2     VARCHAR(20)  NOT NULL,
C_CITY         VARCHAR(20)  NOT NULL,
C_D_ID         SMALLINT     NOT NULL,
C_W_ID         INTEGER      NOT NULL,
C_DELIVERY_CNT INTEGER      NOT NULL,
C_BALANCE      DECIMAL(12,2) NOT NULL,
C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
C_PAYMENT_CNT  INTEGER      NOT NULL
)
IN ts_customer_009
INDEX IN is_customer_009
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 6497 ENDING AT 7308,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER10;
CREATE TABLE CUSTOMER10
(
C_ID           INTEGER      NOT NULL,
C_STATE        CHAR(2)      NOT NULL,
C_ZIP          CHAR(9)      NOT NULL,
C_PHONE        CHAR(16)     NOT NULL,
C_SINCE        TIMESTAMP    NOT NULL,
C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
C_MIDDLE       CHAR(2)      NOT NULL,
C_CREDIT       CHAR(2)      NOT NULL,
C_DISCOUNT     REAL         NOT NULL,
C_DATA         VARCHAR(500) NOT NULL,
C_LAST         VARCHAR(16)  NOT NULL,
C_FIRST        VARCHAR(16)  NOT NULL,
C_STREET_1     VARCHAR(20)  NOT NULL,
C_STREET_2     VARCHAR(20)  NOT NULL,
C_CITY         VARCHAR(20)  NOT NULL,
C_D_ID         SMALLINT     NOT NULL,
C_W_ID         INTEGER      NOT NULL,
C_DELIVERY_CNT INTEGER      NOT NULL,
C_BALANCE      DECIMAL(12,2) NOT NULL,
C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
C_PAYMENT_CNT  INTEGER      NOT NULL
)
IN ts_customer_010
INDEX IN is_customer_010
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 7309 ENDING AT 8120,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER11;
CREATE TABLE CUSTOMER11
(
C_ID           INTEGER      NOT NULL,
C_STATE        CHAR(2)      NOT NULL,
C_ZIP          CHAR(9)      NOT NULL,
C_PHONE        CHAR(16)     NOT NULL,
C_SINCE        TIMESTAMP    NOT NULL,
C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
C_MIDDLE       CHAR(2)      NOT NULL,
C_CREDIT       CHAR(2)      NOT NULL,
C_DISCOUNT     REAL         NOT NULL,
C_DATA         VARCHAR(500) NOT NULL,
C_LAST         VARCHAR(16)  NOT NULL,
C_FIRST        VARCHAR(16)  NOT NULL,
C_STREET_1     VARCHAR(20)  NOT NULL,
C_STREET_2     VARCHAR(20)  NOT NULL,
C_CITY         VARCHAR(20)  NOT NULL,
C_D_ID         SMALLINT     NOT NULL,
C_W_ID         INTEGER      NOT NULL,
C_DELIVERY_CNT INTEGER      NOT NULL,
C_BALANCE      DECIMAL(12,2) NOT NULL,
C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
C_PAYMENT_CNT  INTEGER      NOT NULL
)
IN ts_customer_011
INDEX IN is_customer_011
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 8121 ENDING AT 8932,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER12;
CREATE TABLE CUSTOMER12
(
C_ID           INTEGER      NOT NULL,
C_STATE        CHAR(2)      NOT NULL,
C_ZIP          CHAR(9)      NOT NULL,
C_PHONE        CHAR(16)     NOT NULL,
C_SINCE        TIMESTAMP    NOT NULL,
C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
C_MIDDLE       CHAR(2)      NOT NULL,
C_CREDIT       CHAR(2)      NOT NULL,
C_DISCOUNT     REAL         NOT NULL,
C_DATA         VARCHAR(500) NOT NULL,
C_LAST         VARCHAR(16)  NOT NULL,
C_FIRST        VARCHAR(16)  NOT NULL,
C_STREET_1     VARCHAR(20)  NOT NULL,
C_STREET_2     VARCHAR(20)  NOT NULL,
C_CITY         VARCHAR(20)  NOT NULL,
C_D_ID         SMALLINT     NOT NULL,
C_W_ID         INTEGER      NOT NULL,
C_DELIVERY_CNT INTEGER      NOT NULL,
C_BALANCE      DECIMAL(12,2) NOT NULL,
C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
C_PAYMENT_CNT  INTEGER      NOT NULL
)
IN ts_customer_012
INDEX IN is_customer_012
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 8933 ENDING AT 9744,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER13;
CREATE TABLE CUSTOMER13
(
C_ID           INTEGER      NOT NULL,
C_STATE        CHAR(2)      NOT NULL,
C_ZIP          CHAR(9)      NOT NULL,
C_PHONE        CHAR(16)     NOT NULL,
C_SINCE        TIMESTAMP    NOT NULL,
C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
C_MIDDLE       CHAR(2)      NOT NULL,
C_CREDIT       CHAR(2)      NOT NULL,
C_DISCOUNT     REAL         NOT NULL,
C_DATA         VARCHAR(500) NOT NULL,
C_LAST         VARCHAR(16)  NOT NULL,
C_FIRST        VARCHAR(16)  NOT NULL,
C_STREET_1     VARCHAR(20)  NOT NULL,
C_STREET_2     VARCHAR(20)  NOT NULL,
C_CITY         VARCHAR(20)  NOT NULL,
C_D_ID         SMALLINT     NOT NULL,
C_W_ID         INTEGER      NOT NULL,
C_DELIVERY_CNT INTEGER      NOT NULL,
C_BALANCE      DECIMAL(12,2) NOT NULL,
C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
C_PAYMENT_CNT  INTEGER      NOT NULL
)
IN ts_customer_013
INDEX IN is_customer_013
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 9745 ENDING AT 10556,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER14;
CREATE TABLE CUSTOMER14
(
C_ID           INTEGER      NOT NULL,
C_STATE        CHAR(2)      NOT NULL,
C_ZIP          CHAR(9)      NOT NULL,
C_PHONE        CHAR(16)     NOT NULL,
C_SINCE        TIMESTAMP    NOT NULL,
C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
C_MIDDLE       CHAR(2)      NOT NULL,
C_CREDIT       CHAR(2)      NOT NULL,
C_DISCOUNT     REAL         NOT NULL,
C_DATA         VARCHAR(500) NOT NULL,
C_LAST         VARCHAR(16)  NOT NULL,
C_FIRST        VARCHAR(16)  NOT NULL,
C_STREET_1     VARCHAR(20)  NOT NULL,
C_STREET_2     VARCHAR(20)  NOT NULL,
C_CITY         VARCHAR(20)  NOT NULL,
C_D_ID         SMALLINT     NOT NULL,
C_W_ID         INTEGER      NOT NULL,
C_DELIVERY_CNT INTEGER      NOT NULL,
C_BALANCE      DECIMAL(12,2) NOT NULL,
C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
C_PAYMENT_CNT  INTEGER      NOT NULL
)
IN ts_customer_014
INDEX IN is_customer_014
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 10557 ENDING AT 11368,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER15;
CREATE TABLE CUSTOMER15
(
C_ID           INTEGER      NOT NULL,
C_STATE        CHAR(2)      NOT NULL,
C_ZIP          CHAR(9)      NOT NULL,
C_PHONE        CHAR(16)     NOT NULL,
C_SINCE        TIMESTAMP    NOT NULL,
C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
C_MIDDLE       CHAR(2)      NOT NULL,
C_CREDIT       CHAR(2)      NOT NULL,
C_DISCOUNT     REAL         NOT NULL,
C_DATA         VARCHAR(500) NOT NULL,
C_LAST         VARCHAR(16)  NOT NULL,
C_FIRST        VARCHAR(16)  NOT NULL,
C_STREET_1     VARCHAR(20)  NOT NULL,
```

```
        C_STREET_2     VARCHAR(20)  NOT NULL,
        C_CITY         VARCHAR(20)  NOT NULL,
        C_D_ID         SMALLINT     NOT NULL,
        C_W_ID         INTEGER      NOT NULL,
        C_DELIVERY_CNT INTEGER      NOT NULL,
        C_BALANCE      DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT  INTEGER      NOT NULL
        )
        IN ts_customer_015
        INDEX IN is_customer_015
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 11369 ENDING AT 12180,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER16;
CREATE TABLE CUSTOMER16
        (
        C_ID           INTEGER      NOT NULL,
        C_STATE        CHAR(2)      NOT NULL,
        C_ZIP          CHAR(9)      NOT NULL,
        C_PHONE        CHAR(16)     NOT NULL,
        C_SINCE        TIMESTAMP    NOT NULL,
        C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
        C_MIDDLE       CHAR(2)      NOT NULL,
        C_CREDIT       CHAR(2)      NOT NULL,
        C_DISCOUNT     REAL         NOT NULL,
        C_DATA         VARCHAR(500) NOT NULL,
        C_LAST         VARCHAR(16)  NOT NULL,
        C_FIRST        VARCHAR(16)  NOT NULL,
        C_STREET_1     VARCHAR(20)  NOT NULL,
        C_STREET_2     VARCHAR(20)  NOT NULL,
        C_CITY         VARCHAR(20)  NOT NULL,
        C_D_ID         SMALLINT     NOT NULL,
        C_W_ID         INTEGER      NOT NULL,
        C_DELIVERY_CNT INTEGER      NOT NULL,
        C_BALANCE      DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT  INTEGER      NOT NULL
        )
        IN ts_customer_016
        INDEX IN is_customer_016
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 12181 ENDING AT 12992,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER17;
CREATE TABLE CUSTOMER17
        (
        C_ID           INTEGER      NOT NULL,
        C_STATE        CHAR(2)      NOT NULL,
        C_ZIP          CHAR(9)      NOT NULL,
        C_PHONE        CHAR(16)     NOT NULL,
        C_SINCE        TIMESTAMP    NOT NULL,
        C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
        C_MIDDLE       CHAR(2)      NOT NULL,
        C_CREDIT       CHAR(2)      NOT NULL,
        C_DISCOUNT     REAL         NOT NULL,
        C_DATA         VARCHAR(500) NOT NULL,
        C_LAST         VARCHAR(16)  NOT NULL,
        C_FIRST        VARCHAR(16)  NOT NULL,
        C_STREET_1     VARCHAR(20)  NOT NULL,
        C_STREET_2     VARCHAR(20)  NOT NULL,
        C_CITY         VARCHAR(20)  NOT NULL,
        C_D_ID         SMALLINT     NOT NULL,
        C_W_ID         INTEGER      NOT NULL,
        C_DELIVERY_CNT INTEGER      NOT NULL,
        C_BALANCE      DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT  INTEGER      NOT NULL
        )
        IN ts_customer_017
        INDEX IN is_customer_017
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 12993 ENDING AT 13804,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER18;
CREATE TABLE CUSTOMER18
        (
        C_ID           INTEGER      NOT NULL,
        C_STATE        CHAR(2)      NOT NULL,
        C_ZIP          CHAR(9)      NOT NULL,
        C_PHONE        CHAR(16)     NOT NULL,
        C_SINCE        TIMESTAMP    NOT NULL,
        C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
        C_MIDDLE       CHAR(2)      NOT NULL,
        C_CREDIT       CHAR(2)      NOT NULL,
        C_DISCOUNT     REAL         NOT NULL,
        C_DATA         VARCHAR(500) NOT NULL,
        C_LAST         VARCHAR(16)  NOT NULL,
        C_FIRST        VARCHAR(16)  NOT NULL,
        C_STREET_1     VARCHAR(20)  NOT NULL,
        C_STREET_2     VARCHAR(20)  NOT NULL,
        C_CITY         VARCHAR(20)  NOT NULL,
        C_D_ID         SMALLINT     NOT NULL,
        C_W_ID         INTEGER      NOT NULL,
        C_DELIVERY_CNT INTEGER      NOT NULL,
        C_BALANCE      DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT  INTEGER      NOT NULL
        )
        IN ts_customer_018
        INDEX IN is_customer_018
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 13805 ENDING AT 14616,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER19;
CREATE TABLE CUSTOMER19
        (
        C_ID           INTEGER      NOT NULL,
        C_STATE        CHAR(2)      NOT NULL,
        C_ZIP          CHAR(9)      NOT NULL,
        C_PHONE        CHAR(16)     NOT NULL,
        C_SINCE        TIMESTAMP    NOT NULL,
        C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
        C_MIDDLE       CHAR(2)      NOT NULL,
        C_CREDIT       CHAR(2)      NOT NULL,
        C_DISCOUNT     REAL         NOT NULL,
        C_DATA         VARCHAR(500) NOT NULL,
        C_LAST         VARCHAR(16)  NOT NULL,
        C_FIRST        VARCHAR(16)  NOT NULL,
        C_STREET_1     VARCHAR(20)  NOT NULL,
        C_STREET_2     VARCHAR(20)  NOT NULL,
        C_CITY         VARCHAR(20)  NOT NULL,
        C_D_ID         SMALLINT     NOT NULL,
        C_W_ID         INTEGER      NOT NULL,
        C_DELIVERY_CNT INTEGER      NOT NULL,
        C_BALANCE      DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT  INTEGER      NOT NULL
        )
        IN ts_customer_019
        INDEX IN is_customer_019
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 14617 ENDING AT 15428,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER20;
CREATE TABLE CUSTOMER20
        (
        C_ID           INTEGER      NOT NULL,
        C_STATE        CHAR(2)      NOT NULL,
        C_ZIP          CHAR(9)      NOT NULL,
        C_PHONE        CHAR(16)     NOT NULL,
        C_SINCE        TIMESTAMP    NOT NULL,
        C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
        C_MIDDLE       CHAR(2)      NOT NULL,
        C_CREDIT       CHAR(2)      NOT NULL,
        C_DISCOUNT     REAL         NOT NULL,
        C_DATA         VARCHAR(500) NOT NULL,
        C_LAST         VARCHAR(16)  NOT NULL,
        C_FIRST        VARCHAR(16)  NOT NULL,
        C_STREET_1     VARCHAR(20)  NOT NULL,
        C_STREET_2     VARCHAR(20)  NOT NULL,
        C_CITY         VARCHAR(20)  NOT NULL,
        C_D_ID         SMALLINT     NOT NULL,
        C_W_ID         INTEGER      NOT NULL,
        C_DELIVERY_CNT INTEGER      NOT NULL,
        C_BALANCE      DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT  INTEGER      NOT NULL
        )
        IN ts_customer_020
        INDEX IN is_customer_020
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 15429 ENDING AT 16240,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER21;
CREATE TABLE CUSTOMER21
        (
        C_ID           INTEGER      NOT NULL,
        C_STATE        CHAR(2)      NOT NULL,
        C_ZIP          CHAR(9)      NOT NULL,
        C_PHONE        CHAR(16)     NOT NULL,
        C_SINCE        TIMESTAMP    NOT NULL,
        C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
        C_MIDDLE       CHAR(2)      NOT NULL,
        C_CREDIT       CHAR(2)      NOT NULL,
        C_DISCOUNT     REAL         NOT NULL,
        C_DATA         VARCHAR(500) NOT NULL,
        C_LAST         VARCHAR(16)  NOT NULL,
        C_FIRST        VARCHAR(16)  NOT NULL,
        C_STREET_1     VARCHAR(20)  NOT NULL,
        C_STREET_2     VARCHAR(20)  NOT NULL,
        C_CITY         VARCHAR(20)  NOT NULL,
        C_D_ID         SMALLINT     NOT NULL,
        C_W_ID         INTEGER      NOT NULL,
        C_DELIVERY_CNT INTEGER      NOT NULL,
        C_BALANCE      DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT  INTEGER      NOT NULL
        )
```

```
        IN ts_customer_021
        INDEX IN is_customer_021
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 16241 ENDING AT 17052,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER22;
CREATE TABLE CUSTOMER22
        (
         C_ID          INTEGER      NOT NULL,
         C_STATE       CHAR(2)      NOT NULL,
         C_ZIP         CHAR(9)      NOT NULL,
         C_PHONE       CHAR(16)     NOT NULL,
         C_SINCE       TIMESTAMP    NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)      NOT NULL,
         C_CREDIT      CHAR(2)      NOT NULL,
         C_DISCOUNT    REAL         NOT NULL,
         C_DATA        VARCHAR(500) NOT NULL,
         C_LAST        VARCHAR(16)  NOT NULL,
         C_FIRST       VARCHAR(16)  NOT NULL,
         C_STREET_1    VARCHAR(20)  NOT NULL,
         C_STREET_2    VARCHAR(20)  NOT NULL,
         C_CITY        VARCHAR(20)  NOT NULL,
         C_D_ID        SMALLINT     NOT NULL,
         C_W_ID        INTEGER      NOT NULL,
         C_DELIVERY_CNT INTEGER     NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_022
        INDEX IN is_customer_022
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 17053 ENDING AT 17864,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER23;
CREATE TABLE CUSTOMER23
        (
         C_ID          INTEGER      NOT NULL,
         C_STATE       CHAR(2)      NOT NULL,
         C_ZIP         CHAR(9)      NOT NULL,
         C_PHONE       CHAR(16)     NOT NULL,
         C_SINCE       TIMESTAMP    NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)      NOT NULL,
         C_CREDIT      CHAR(2)      NOT NULL,
         C_DISCOUNT    REAL         NOT NULL,
         C_DATA        VARCHAR(500) NOT NULL,
         C_LAST        VARCHAR(16)  NOT NULL,
         C_FIRST       VARCHAR(16)  NOT NULL,
         C_STREET_1    VARCHAR(20)  NOT NULL,
         C_STREET_2    VARCHAR(20)  NOT NULL,
         C_CITY        VARCHAR(20)  NOT NULL,
         C_D_ID        SMALLINT     NOT NULL,
         C_W_ID        INTEGER      NOT NULL,
         C_DELIVERY_CNT INTEGER     NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_023
        INDEX IN is_customer_023
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 17865 ENDING AT 18676,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER24;
CREATE TABLE CUSTOMER24
        (
         C_ID          INTEGER      NOT NULL,
         C_STATE       CHAR(2)      NOT NULL,
         C_ZIP         CHAR(9)      NOT NULL,
         C_PHONE       CHAR(16)     NOT NULL,
         C_SINCE       TIMESTAMP    NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)      NOT NULL,
         C_CREDIT      CHAR(2)      NOT NULL,
         C_DISCOUNT    REAL         NOT NULL,
         C_DATA        VARCHAR(500) NOT NULL,
         C_LAST        VARCHAR(16)  NOT NULL,
         C_FIRST       VARCHAR(16)  NOT NULL,
         C_STREET_1    VARCHAR(20)  NOT NULL,
         C_STREET_2    VARCHAR(20)  NOT NULL,
         C_CITY        VARCHAR(20)  NOT NULL,
         C_D_ID        SMALLINT     NOT NULL,
         C_W_ID        INTEGER      NOT NULL,
         C_DELIVERY_CNT INTEGER     NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_024
        INDEX IN is_customer_024
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 18677 ENDING AT 19488,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER25;
CREATE TABLE CUSTOMER25
        (
         C_ID          INTEGER      NOT NULL,
         C_STATE       CHAR(2)      NOT NULL,
         C_ZIP         CHAR(9)      NOT NULL,
         C_PHONE       CHAR(16)     NOT NULL,
         C_SINCE       TIMESTAMP    NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)      NOT NULL,
         C_CREDIT      CHAR(2)      NOT NULL,
         C_DISCOUNT    REAL         NOT NULL,
         C_DATA        VARCHAR(500) NOT NULL,
         C_LAST        VARCHAR(16)  NOT NULL,
         C_FIRST       VARCHAR(16)  NOT NULL,
         C_STREET_1    VARCHAR(20)  NOT NULL,
         C_STREET_2    VARCHAR(20)  NOT NULL,
         C_CITY        VARCHAR(20)  NOT NULL,
         C_D_ID        SMALLINT     NOT NULL,
         C_W_ID        INTEGER      NOT NULL,
         C_DELIVERY_CNT INTEGER     NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_025
        INDEX IN is_customer_025
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 19489 ENDING AT 20300,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER26;
CREATE TABLE CUSTOMER26
        (
         C_ID          INTEGER      NOT NULL,
         C_STATE       CHAR(2)      NOT NULL,
         C_ZIP         CHAR(9)      NOT NULL,
         C_PHONE       CHAR(16)     NOT NULL,
         C_SINCE       TIMESTAMP    NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)      NOT NULL,
         C_CREDIT      CHAR(2)      NOT NULL,
         C_DISCOUNT    REAL         NOT NULL,
         C_DATA        VARCHAR(500) NOT NULL,
         C_LAST        VARCHAR(16)  NOT NULL,
         C_FIRST       VARCHAR(16)  NOT NULL,
         C_STREET_1    VARCHAR(20)  NOT NULL,
         C_STREET_2    VARCHAR(20)  NOT NULL,
         C_CITY        VARCHAR(20)  NOT NULL,
         C_D_ID        SMALLINT     NOT NULL,
         C_W_ID        INTEGER      NOT NULL,
         C_DELIVERY_CNT INTEGER     NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_026
        INDEX IN is_customer_026
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 20301 ENDING AT 21112,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER27;
CREATE TABLE CUSTOMER27
        (
         C_ID          INTEGER      NOT NULL,
         C_STATE       CHAR(2)      NOT NULL,
         C_ZIP         CHAR(9)      NOT NULL,
         C_PHONE       CHAR(16)     NOT NULL,
         C_SINCE       TIMESTAMP    NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)      NOT NULL,
         C_CREDIT      CHAR(2)      NOT NULL,
         C_DISCOUNT    REAL         NOT NULL,
         C_DATA        VARCHAR(500) NOT NULL,
         C_LAST        VARCHAR(16)  NOT NULL,
         C_FIRST       VARCHAR(16)  NOT NULL,
         C_STREET_1    VARCHAR(20)  NOT NULL,
         C_STREET_2    VARCHAR(20)  NOT NULL,
         C_CITY        VARCHAR(20)  NOT NULL,
         C_D_ID        SMALLINT     NOT NULL,
         C_W_ID        INTEGER      NOT NULL,
         C_DELIVERY_CNT INTEGER     NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_027
        INDEX IN is_customer_027
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 21113 ENDING AT 21924,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

```
connect to TPCC in share mode;
DROP TABLE CUSTOMER28;
CREATE TABLE CUSTOMER28
        (
         C_ID          INTEGER      NOT NULL,
         C_STATE       CHAR(2)      NOT NULL,
         C_ZIP         CHAR(9)      NOT NULL,
         C_PHONE       CHAR(16)     NOT NULL,
         C_SINCE       TIMESTAMP    NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)      NOT NULL,
         C_CREDIT      CHAR(2)      NOT NULL,
         C_DISCOUNT    REAL         NOT NULL,
         C_DATA        VARCHAR(500) NOT NULL,
         C_LAST        VARCHAR(16)  NOT NULL,
         C_FIRST       VARCHAR(16)  NOT NULL,
         C_STREET_1    VARCHAR(20)  NOT NULL,
         C_STREET_2    VARCHAR(20)  NOT NULL,
         C_CITY        VARCHAR(20)  NOT NULL,
         C_D_ID        SMALLINT     NOT NULL,
         C_W_ID        INTEGER      NOT NULL,
         C_DELIVERY_CNT INTEGER     NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_028
        INDEX IN is_customer_028
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 21925 ENDING AT 22736,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER29;
CREATE TABLE CUSTOMER29
        (
         C_ID          INTEGER      NOT NULL,
         C_STATE       CHAR(2)      NOT NULL,
         C_ZIP         CHAR(9)      NOT NULL,
         C_PHONE       CHAR(16)     NOT NULL,
         C_SINCE       TIMESTAMP    NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)      NOT NULL,
         C_CREDIT      CHAR(2)      NOT NULL,
         C_DISCOUNT    REAL         NOT NULL,
         C_DATA        VARCHAR(500) NOT NULL,
         C_LAST        VARCHAR(16)  NOT NULL,
         C_FIRST       VARCHAR(16)  NOT NULL,
         C_STREET_1    VARCHAR(20)  NOT NULL,
         C_STREET_2    VARCHAR(20)  NOT NULL,
         C_CITY        VARCHAR(20)  NOT NULL,
         C_D_ID        SMALLINT     NOT NULL,
         C_W_ID        INTEGER      NOT NULL,
         C_DELIVERY_CNT INTEGER     NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_029
        INDEX IN is_customer_029
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 22737 ENDING AT 23548,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER30;
CREATE TABLE CUSTOMER30
        (
         C_ID          INTEGER      NOT NULL,
         C_STATE       CHAR(2)      NOT NULL,
         C_ZIP         CHAR(9)      NOT NULL,
         C_PHONE       CHAR(16)     NOT NULL,
         C_SINCE       TIMESTAMP    NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)      NOT NULL,
         C_CREDIT      CHAR(2)      NOT NULL,
         C_DISCOUNT    REAL         NOT NULL,
         C_DATA        VARCHAR(500) NOT NULL,
         C_LAST        VARCHAR(16)  NOT NULL,
         C_FIRST       VARCHAR(16)  NOT NULL,
         C_STREET_1    VARCHAR(20)  NOT NULL,
         C_STREET_2    VARCHAR(20)  NOT NULL,
         C_CITY        VARCHAR(20)  NOT NULL,
         C_D_ID        SMALLINT     NOT NULL,
         C_W_ID        INTEGER      NOT NULL,
         C_DELIVERY_CNT INTEGER     NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_030
        INDEX IN is_customer_030
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 23549 ENDING AT 24360,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER31;
CREATE TABLE CUSTOMER31
        (
         C_ID          INTEGER      NOT NULL,
         C_STATE       CHAR(2)      NOT NULL,
         C_ZIP         CHAR(9)      NOT NULL,
         C_PHONE       CHAR(16)     NOT NULL,
         C_SINCE       TIMESTAMP    NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)      NOT NULL,
         C_CREDIT      CHAR(2)      NOT NULL,
         C_DISCOUNT    REAL         NOT NULL,
         C_DATA        VARCHAR(500) NOT NULL,
         C_LAST        VARCHAR(16)  NOT NULL,
         C_FIRST       VARCHAR(16)  NOT NULL,
         C_STREET_1    VARCHAR(20)  NOT NULL,
         C_STREET_2    VARCHAR(20)  NOT NULL,
         C_CITY        VARCHAR(20)  NOT NULL,
         C_D_ID        SMALLINT     NOT NULL,
         C_W_ID        INTEGER      NOT NULL,
         C_DELIVERY_CNT INTEGER     NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_031
        INDEX IN is_customer_031
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 24361 ENDING AT 25172,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER32;
CREATE TABLE CUSTOMER32
        (
         C_ID          INTEGER      NOT NULL,
         C_STATE       CHAR(2)      NOT NULL,
         C_ZIP         CHAR(9)      NOT NULL,
         C_PHONE       CHAR(16)     NOT NULL,
         C_SINCE       TIMESTAMP    NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)      NOT NULL,
         C_CREDIT      CHAR(2)      NOT NULL,
         C_DISCOUNT    REAL         NOT NULL,
         C_DATA        VARCHAR(500) NOT NULL,
         C_LAST        VARCHAR(16)  NOT NULL,
         C_FIRST       VARCHAR(16)  NOT NULL,
         C_STREET_1    VARCHAR(20)  NOT NULL,
         C_STREET_2    VARCHAR(20)  NOT NULL,
         C_CITY        VARCHAR(20)  NOT NULL,
         C_D_ID        SMALLINT     NOT NULL,
         C_W_ID        INTEGER      NOT NULL,
         C_DELIVERY_CNT INTEGER     NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_032
        INDEX IN is_customer_032
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 25173 ENDING AT 25984,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER33;
CREATE TABLE CUSTOMER33
        (
         C_ID          INTEGER      NOT NULL,
         C_STATE       CHAR(2)      NOT NULL,
         C_ZIP         CHAR(9)      NOT NULL,
         C_PHONE       CHAR(16)     NOT NULL,
         C_SINCE       TIMESTAMP    NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)      NOT NULL,
         C_CREDIT      CHAR(2)      NOT NULL,
         C_DISCOUNT    REAL         NOT NULL,
         C_DATA        VARCHAR(500) NOT NULL,
         C_LAST        VARCHAR(16)  NOT NULL,
         C_FIRST       VARCHAR(16)  NOT NULL,
         C_STREET_1    VARCHAR(20)  NOT NULL,
         C_STREET_2    VARCHAR(20)  NOT NULL,
         C_CITY        VARCHAR(20)  NOT NULL,
         C_D_ID        SMALLINT     NOT NULL,
         C_W_ID        INTEGER      NOT NULL,
         C_DELIVERY_CNT INTEGER     NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_033
        INDEX IN is_customer_033
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 25985 ENDING AT 26796,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER34;
CREATE TABLE CUSTOMER34
        (
         C_ID          INTEGER      NOT NULL,
         C_STATE       CHAR(2)      NOT NULL,
         C_ZIP         CHAR(9)      NOT NULL,
         C_PHONE       CHAR(16)     NOT NULL,
         C_SINCE       TIMESTAMP    NOT NULL,
```

```
C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
C_MIDDLE       CHAR(2)       NOT NULL,
C_CREDIT       CHAR(2)       NOT NULL,
C_DISCOUNT     REAL          NOT NULL,
C_DATA         VARCHAR(500)  NOT NULL,
C_LAST         VARCHAR(16)   NOT NULL,
C_FIRST        VARCHAR(16)   NOT NULL,
C_STREET_1     VARCHAR(20)   NOT NULL,
C_STREET_2     VARCHAR(20)   NOT NULL,
C_CITY         VARCHAR(20)   NOT NULL,
C_D_ID         SMALLINT      NOT NULL,
C_W_ID         INTEGER       NOT NULL,
C_DELIVERY_CNT INTEGER       NOT NULL,
C_BALANCE      DECIMAL(12,2) NOT NULL,
C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
C_PAYMENT_CNT  INTEGER       NOT NULL
)
IN ts_customer_034
INDEX IN is_customer_034
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 26797 ENDING AT 27608,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER35;
CREATE TABLE CUSTOMER35
(
C_ID           INTEGER       NOT NULL,
C_STATE        CHAR(2)       NOT NULL,
C_ZIP          CHAR(9)       NOT NULL,
C_PHONE        CHAR(16)      NOT NULL,
C_SINCE        TIMESTAMP     NOT NULL,
C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
C_MIDDLE       CHAR(2)       NOT NULL,
C_CREDIT       CHAR(2)       NOT NULL,
C_DISCOUNT     REAL          NOT NULL,
C_DATA         VARCHAR(500)  NOT NULL,
C_LAST         VARCHAR(16)   NOT NULL,
C_FIRST        VARCHAR(16)   NOT NULL,
C_STREET_1     VARCHAR(20)   NOT NULL,
C_STREET_2     VARCHAR(20)   NOT NULL,
C_CITY         VARCHAR(20)   NOT NULL,
C_D_ID         SMALLINT      NOT NULL,
C_W_ID         INTEGER       NOT NULL,
C_DELIVERY_CNT INTEGER       NOT NULL,
C_BALANCE      DECIMAL(12,2) NOT NULL,
C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
C_PAYMENT_CNT  INTEGER       NOT NULL
)
IN ts_customer_035
INDEX IN is_customer_035
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 27609 ENDING AT 28420,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER36;
CREATE TABLE CUSTOMER36
(
C_ID           INTEGER       NOT NULL,
C_STATE        CHAR(2)       NOT NULL,
C_ZIP          CHAR(9)       NOT NULL,
C_PHONE        CHAR(16)      NOT NULL,
C_SINCE        TIMESTAMP     NOT NULL,
C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
C_MIDDLE       CHAR(2)       NOT NULL,
C_CREDIT       CHAR(2)       NOT NULL,
C_DISCOUNT     REAL          NOT NULL,
C_DATA         VARCHAR(500)  NOT NULL,
C_LAST         VARCHAR(16)   NOT NULL,
C_FIRST        VARCHAR(16)   NOT NULL,
C_STREET_1     VARCHAR(20)   NOT NULL,
C_STREET_2     VARCHAR(20)   NOT NULL,
C_CITY         VARCHAR(20)   NOT NULL,
C_D_ID         SMALLINT      NOT NULL,
C_W_ID         INTEGER       NOT NULL,
C_DELIVERY_CNT INTEGER       NOT NULL,
C_BALANCE      DECIMAL(12,2) NOT NULL,
C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
C_PAYMENT_CNT  INTEGER       NOT NULL
)
IN ts_customer_036
INDEX IN is_customer_036
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 28421 ENDING AT 29232,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER37;
CREATE TABLE CUSTOMER37
(
C_ID           INTEGER       NOT NULL,
C_STATE        CHAR(2)       NOT NULL,
C_ZIP          CHAR(9)       NOT NULL,
C_PHONE        CHAR(16)      NOT NULL,
C_SINCE        TIMESTAMP     NOT NULL,
C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
C_MIDDLE       CHAR(2)       NOT NULL,
C_CREDIT       CHAR(2)       NOT NULL,
C_DISCOUNT     REAL          NOT NULL,
C_DATA         VARCHAR(500)  NOT NULL,
C_LAST         VARCHAR(16)   NOT NULL,
C_FIRST        VARCHAR(16)   NOT NULL,
C_STREET_1     VARCHAR(20)   NOT NULL,
C_STREET_2     VARCHAR(20)   NOT NULL,
C_CITY         VARCHAR(20)   NOT NULL,
C_D_ID         SMALLINT      NOT NULL,
C_W_ID         INTEGER       NOT NULL,
C_DELIVERY_CNT INTEGER       NOT NULL,
C_BALANCE      DECIMAL(12,2) NOT NULL,
C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
C_PAYMENT_CNT  INTEGER       NOT NULL
)
IN ts_customer_037
INDEX IN is_customer_037
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 29233 ENDING AT 30044,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER38;
CREATE TABLE CUSTOMER38
(
C_ID           INTEGER       NOT NULL,
C_STATE        CHAR(2)       NOT NULL,
C_ZIP          CHAR(9)       NOT NULL,
C_PHONE        CHAR(16)      NOT NULL,
C_SINCE        TIMESTAMP     NOT NULL,
C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
C_MIDDLE       CHAR(2)       NOT NULL,
C_CREDIT       CHAR(2)       NOT NULL,
C_DISCOUNT     REAL          NOT NULL,
C_DATA         VARCHAR(500)  NOT NULL,
C_LAST         VARCHAR(16)   NOT NULL,
C_FIRST        VARCHAR(16)   NOT NULL,
C_STREET_1     VARCHAR(20)   NOT NULL,
C_STREET_2     VARCHAR(20)   NOT NULL,
C_CITY         VARCHAR(20)   NOT NULL,
C_D_ID         SMALLINT      NOT NULL,
C_W_ID         INTEGER       NOT NULL,
C_DELIVERY_CNT INTEGER       NOT NULL,
C_BALANCE      DECIMAL(12,2) NOT NULL,
C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
C_PAYMENT_CNT  INTEGER       NOT NULL
)
IN ts_customer_038
INDEX IN is_customer_038
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 30045 ENDING AT 30856,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER39;
CREATE TABLE CUSTOMER39
(
C_ID           INTEGER       NOT NULL,
C_STATE        CHAR(2)       NOT NULL,
C_ZIP          CHAR(9)       NOT NULL,
C_PHONE        CHAR(16)      NOT NULL,
C_SINCE        TIMESTAMP     NOT NULL,
C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
C_MIDDLE       CHAR(2)       NOT NULL,
C_CREDIT       CHAR(2)       NOT NULL,
C_DISCOUNT     REAL          NOT NULL,
C_DATA         VARCHAR(500)  NOT NULL,
C_LAST         VARCHAR(16)   NOT NULL,
C_FIRST        VARCHAR(16)   NOT NULL,
C_STREET_1     VARCHAR(20)   NOT NULL,
C_STREET_2     VARCHAR(20)   NOT NULL,
C_CITY         VARCHAR(20)   NOT NULL,
C_D_ID         SMALLINT      NOT NULL,
C_W_ID         INTEGER       NOT NULL,
C_DELIVERY_CNT INTEGER       NOT NULL,
C_BALANCE      DECIMAL(12,2) NOT NULL,
C_YTD_PAYMENT  DECIMAL(12,2) NOT NULL,
C_PAYMENT_CNT  INTEGER       NOT NULL
)
IN ts_customer_039
INDEX IN is_customer_039
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 30857 ENDING AT 31668,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER40;
CREATE TABLE CUSTOMER40
(
C_ID           INTEGER       NOT NULL,
C_STATE        CHAR(2)       NOT NULL,
C_ZIP          CHAR(9)       NOT NULL,
C_PHONE        CHAR(16)      NOT NULL,
C_SINCE        TIMESTAMP     NOT NULL,
C_CREDIT_LIM   DECIMAL(12,2) NOT NULL,
C_MIDDLE       CHAR(2)       NOT NULL,
C_CREDIT       CHAR(2)       NOT NULL,
C_DISCOUNT     REAL          NOT NULL,
C_DATA         VARCHAR(500)  NOT NULL,
C_LAST         VARCHAR(16)   NOT NULL,
C_FIRST        VARCHAR(16)   NOT NULL,
C_STREET_1     VARCHAR(20)   NOT NULL,
C_STREET_2     VARCHAR(20)   NOT NULL,
```

```
 C_CITY        VARCHAR(20)  NOT NULL,
 C_D_ID        SMALLINT     NOT NULL,
 C_W_ID        INTEGER      NOT NULL,
 C_DELIVERY_CNT  INTEGER      NOT NULL,
 C_BALANCE       DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT   INTEGER      NOT NULL
)
IN ts_customer_040
INDEX IN is_customer_040
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 31669 ENDING AT 32480,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER41;
CREATE TABLE CUSTOMER41
(
 C_ID          INTEGER      NOT NULL,
 C_STATE       CHAR(2)      NOT NULL,
 C_ZIP         CHAR(9)      NOT NULL,
 C_PHONE       CHAR(16)     NOT NULL,
 C_SINCE       TIMESTAMP    NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)      NOT NULL,
 C_CREDIT      CHAR(2)      NOT NULL,
 C_DISCOUNT    REAL         NOT NULL,
 C_DATA        VARCHAR(500) NOT NULL,
 C_LAST        VARCHAR(16)  NOT NULL,
 C_FIRST       VARCHAR(16)  NOT NULL,
 C_STREET_1    VARCHAR(20)  NOT NULL,
 C_STREET_2    VARCHAR(20)  NOT NULL,
 C_CITY        VARCHAR(20)  NOT NULL,
 C_D_ID        SMALLINT     NOT NULL,
 C_W_ID        INTEGER      NOT NULL,
 C_DELIVERY_CNT  INTEGER      NOT NULL,
 C_BALANCE       DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT   INTEGER      NOT NULL
)
IN ts_customer_041
INDEX IN is_customer_041
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 32481 ENDING AT 33292,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER42;
CREATE TABLE CUSTOMER42
(
 C_ID          INTEGER      NOT NULL,
 C_STATE       CHAR(2)      NOT NULL,
 C_ZIP         CHAR(9)      NOT NULL,
 C_PHONE       CHAR(16)     NOT NULL,
 C_SINCE       TIMESTAMP    NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)      NOT NULL,
 C_CREDIT      CHAR(2)      NOT NULL,
 C_DISCOUNT    REAL         NOT NULL,
 C_DATA        VARCHAR(500) NOT NULL,
 C_LAST        VARCHAR(16)  NOT NULL,
 C_FIRST       VARCHAR(16)  NOT NULL,
 C_STREET_1    VARCHAR(20)  NOT NULL,
 C_STREET_2    VARCHAR(20)  NOT NULL,
 C_CITY        VARCHAR(20)  NOT NULL,
 C_D_ID        SMALLINT     NOT NULL,
 C_W_ID        INTEGER      NOT NULL,
 C_DELIVERY_CNT  INTEGER      NOT NULL,
 C_BALANCE       DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT   INTEGER      NOT NULL
)
IN ts_customer_042
INDEX IN is_customer_042
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 33293 ENDING AT 34104,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER43;
CREATE TABLE CUSTOMER43
(
 C_ID          INTEGER      NOT NULL,
 C_STATE       CHAR(2)      NOT NULL,
 C_ZIP         CHAR(9)      NOT NULL,
 C_PHONE       CHAR(16)     NOT NULL,
 C_SINCE       TIMESTAMP    NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)      NOT NULL,
 C_CREDIT      CHAR(2)      NOT NULL,
 C_DISCOUNT    REAL         NOT NULL,
 C_DATA        VARCHAR(500) NOT NULL,
 C_LAST        VARCHAR(16)  NOT NULL,
 C_FIRST       VARCHAR(16)  NOT NULL,
 C_STREET_1    VARCHAR(20)  NOT NULL,
 C_STREET_2    VARCHAR(20)  NOT NULL,
 C_CITY        VARCHAR(20)  NOT NULL,
 C_D_ID        SMALLINT     NOT NULL,
 C_W_ID        INTEGER      NOT NULL,
 C_DELIVERY_CNT  INTEGER      NOT NULL,
 C_BALANCE       DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT   INTEGER      NOT NULL
)
IN ts_customer_043
INDEX IN is_customer_043
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 34105 ENDING AT 34916,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER44;
CREATE TABLE CUSTOMER44
(
 C_ID          INTEGER      NOT NULL,
 C_STATE       CHAR(2)      NOT NULL,
 C_ZIP         CHAR(9)      NOT NULL,
 C_PHONE       CHAR(16)     NOT NULL,
 C_SINCE       TIMESTAMP    NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)      NOT NULL,
 C_CREDIT      CHAR(2)      NOT NULL,
 C_DISCOUNT    REAL         NOT NULL,
 C_DATA        VARCHAR(500) NOT NULL,
 C_LAST        VARCHAR(16)  NOT NULL,
 C_FIRST       VARCHAR(16)  NOT NULL,
 C_STREET_1    VARCHAR(20)  NOT NULL,
 C_STREET_2    VARCHAR(20)  NOT NULL,
 C_CITY        VARCHAR(20)  NOT NULL,
 C_D_ID        SMALLINT     NOT NULL,
 C_W_ID        INTEGER      NOT NULL,
 C_DELIVERY_CNT  INTEGER      NOT NULL,
 C_BALANCE       DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT   INTEGER      NOT NULL
)
IN ts_customer_044
INDEX IN is_customer_044
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 34917 ENDING AT 35728,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER45;
CREATE TABLE CUSTOMER45
(
 C_ID          INTEGER      NOT NULL,
 C_STATE       CHAR(2)      NOT NULL,
 C_ZIP         CHAR(9)      NOT NULL,
 C_PHONE       CHAR(16)     NOT NULL,
 C_SINCE       TIMESTAMP    NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)      NOT NULL,
 C_CREDIT      CHAR(2)      NOT NULL,
 C_DISCOUNT    REAL         NOT NULL,
 C_DATA        VARCHAR(500) NOT NULL,
 C_LAST        VARCHAR(16)  NOT NULL,
 C_FIRST       VARCHAR(16)  NOT NULL,
 C_STREET_1    VARCHAR(20)  NOT NULL,
 C_STREET_2    VARCHAR(20)  NOT NULL,
 C_CITY        VARCHAR(20)  NOT NULL,
 C_D_ID        SMALLINT     NOT NULL,
 C_W_ID        INTEGER      NOT NULL,
 C_DELIVERY_CNT  INTEGER      NOT NULL,
 C_BALANCE       DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT   INTEGER      NOT NULL
)
IN ts_customer_045
INDEX IN is_customer_045
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 35729 ENDING AT 36540,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER46;
CREATE TABLE CUSTOMER46
(
 C_ID          INTEGER      NOT NULL,
 C_STATE       CHAR(2)      NOT NULL,
 C_ZIP         CHAR(9)      NOT NULL,
 C_PHONE       CHAR(16)     NOT NULL,
 C_SINCE       TIMESTAMP    NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)      NOT NULL,
 C_CREDIT      CHAR(2)      NOT NULL,
 C_DISCOUNT    REAL         NOT NULL,
 C_DATA        VARCHAR(500) NOT NULL,
 C_LAST        VARCHAR(16)  NOT NULL,
 C_FIRST       VARCHAR(16)  NOT NULL,
 C_STREET_1    VARCHAR(20)  NOT NULL,
 C_STREET_2    VARCHAR(20)  NOT NULL,
 C_CITY        VARCHAR(20)  NOT NULL,
 C_D_ID        SMALLINT     NOT NULL,
 C_W_ID        INTEGER      NOT NULL,
 C_DELIVERY_CNT  INTEGER      NOT NULL,
 C_BALANCE       DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT   INTEGER      NOT NULL
)
IN ts_customer_046
```

```
INDEX IN is_customer_046
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 36541 ENDING AT 37352,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER47;
CREATE TABLE CUSTOMER47
(
 C_ID          INTEGER      NOT NULL,
 C_STATE       CHAR(2)      NOT NULL,
 C_ZIP         CHAR(9)      NOT NULL,
 C_PHONE       CHAR(16)     NOT NULL,
 C_SINCE       TIMESTAMP    NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)      NOT NULL,
 C_CREDIT      CHAR(2)      NOT NULL,
 C_DISCOUNT    REAL         NOT NULL,
 C_DATA        VARCHAR(500) NOT NULL,
 C_LAST        VARCHAR(16)  NOT NULL,
 C_FIRST       VARCHAR(16)  NOT NULL,
 C_STREET_1    VARCHAR(20)  NOT NULL,
 C_STREET_2    VARCHAR(20)  NOT NULL,
 C_CITY        VARCHAR(20)  NOT NULL,
 C_D_ID        SMALLINT     NOT NULL,
 C_W_ID        INTEGER      NOT NULL,
 C_DELIVERY_CNT INTEGER     NOT NULL,
 C_BALANCE     DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT INTEGER      NOT NULL
)
IN ts_customer_047
INDEX IN is_customer_047
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 37353 ENDING AT 38164,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER48;
CREATE TABLE CUSTOMER48
(
 C_ID          INTEGER      NOT NULL,
 C_STATE       CHAR(2)      NOT NULL,
 C_ZIP         CHAR(9)      NOT NULL,
 C_PHONE       CHAR(16)     NOT NULL,
 C_SINCE       TIMESTAMP    NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)      NOT NULL,
 C_CREDIT      CHAR(2)      NOT NULL,
 C_DISCOUNT    REAL         NOT NULL,
 C_DATA        VARCHAR(500) NOT NULL,
 C_LAST        VARCHAR(16)  NOT NULL,
 C_FIRST       VARCHAR(16)  NOT NULL,
 C_STREET_1    VARCHAR(20)  NOT NULL,
 C_STREET_2    VARCHAR(20)  NOT NULL,
 C_CITY        VARCHAR(20)  NOT NULL,
 C_D_ID        SMALLINT     NOT NULL,
 C_W_ID        INTEGER      NOT NULL,
 C_DELIVERY_CNT INTEGER     NOT NULL,
 C_BALANCE     DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT INTEGER      NOT NULL
)
IN ts_customer_048
INDEX IN is_customer_048
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 38165 ENDING AT 38976,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER49;
CREATE TABLE CUSTOMER49
(
 C_ID          INTEGER      NOT NULL,
 C_STATE       CHAR(2)      NOT NULL,
 C_ZIP         CHAR(9)      NOT NULL,
 C_PHONE       CHAR(16)     NOT NULL,
 C_SINCE       TIMESTAMP    NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)      NOT NULL,
 C_CREDIT      CHAR(2)      NOT NULL,
 C_DISCOUNT    REAL         NOT NULL,
 C_DATA        VARCHAR(500) NOT NULL,
 C_LAST        VARCHAR(16)  NOT NULL,
 C_FIRST       VARCHAR(16)  NOT NULL,
 C_STREET_1    VARCHAR(20)  NOT NULL,
 C_STREET_2    VARCHAR(20)  NOT NULL,
 C_CITY        VARCHAR(20)  NOT NULL,
 C_D_ID        SMALLINT     NOT NULL,
 C_W_ID        INTEGER      NOT NULL,
 C_DELIVERY_CNT INTEGER     NOT NULL,
 C_BALANCE     DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT INTEGER      NOT NULL
)
IN ts_customer_049
INDEX IN is_customer_049
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 38977 ENDING AT 39788,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER50;
CREATE TABLE CUSTOMER50
(
 C_ID          INTEGER      NOT NULL,
 C_STATE       CHAR(2)      NOT NULL,
 C_ZIP         CHAR(9)      NOT NULL,
 C_PHONE       CHAR(16)     NOT NULL,
 C_SINCE       TIMESTAMP    NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)      NOT NULL,
 C_CREDIT      CHAR(2)      NOT NULL,
 C_DISCOUNT    REAL         NOT NULL,
 C_DATA        VARCHAR(500) NOT NULL,
 C_LAST        VARCHAR(16)  NOT NULL,
 C_FIRST       VARCHAR(16)  NOT NULL,
 C_STREET_1    VARCHAR(20)  NOT NULL,
 C_STREET_2    VARCHAR(20)  NOT NULL,
 C_CITY        VARCHAR(20)  NOT NULL,
 C_D_ID        SMALLINT     NOT NULL,
 C_W_ID        INTEGER      NOT NULL,
 C_DELIVERY_CNT INTEGER     NOT NULL,
 C_BALANCE     DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT INTEGER      NOT NULL
)
IN ts_customer_050
INDEX IN is_customer_050
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 39789 ENDING AT 40600,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER51;
CREATE TABLE CUSTOMER51
(
 C_ID          INTEGER      NOT NULL,
 C_STATE       CHAR(2)      NOT NULL,
 C_ZIP         CHAR(9)      NOT NULL,
 C_PHONE       CHAR(16)     NOT NULL,
 C_SINCE       TIMESTAMP    NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)      NOT NULL,
 C_CREDIT      CHAR(2)      NOT NULL,
 C_DISCOUNT    REAL         NOT NULL,
 C_DATA        VARCHAR(500) NOT NULL,
 C_LAST        VARCHAR(16)  NOT NULL,
 C_FIRST       VARCHAR(16)  NOT NULL,
 C_STREET_1    VARCHAR(20)  NOT NULL,
 C_STREET_2    VARCHAR(20)  NOT NULL,
 C_CITY        VARCHAR(20)  NOT NULL,
 C_D_ID        SMALLINT     NOT NULL,
 C_W_ID        INTEGER      NOT NULL,
 C_DELIVERY_CNT INTEGER     NOT NULL,
 C_BALANCE     DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT INTEGER      NOT NULL
)
IN ts_customer_051
INDEX IN is_customer_051
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 40601 ENDING AT 41412,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER52;
CREATE TABLE CUSTOMER52
(
 C_ID          INTEGER      NOT NULL,
 C_STATE       CHAR(2)      NOT NULL,
 C_ZIP         CHAR(9)      NOT NULL,
 C_PHONE       CHAR(16)     NOT NULL,
 C_SINCE       TIMESTAMP    NOT NULL,
 C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
 C_MIDDLE      CHAR(2)      NOT NULL,
 C_CREDIT      CHAR(2)      NOT NULL,
 C_DISCOUNT    REAL         NOT NULL,
 C_DATA        VARCHAR(500) NOT NULL,
 C_LAST        VARCHAR(16)  NOT NULL,
 C_FIRST       VARCHAR(16)  NOT NULL,
 C_STREET_1    VARCHAR(20)  NOT NULL,
 C_STREET_2    VARCHAR(20)  NOT NULL,
 C_CITY        VARCHAR(20)  NOT NULL,
 C_D_ID        SMALLINT     NOT NULL,
 C_W_ID        INTEGER      NOT NULL,
 C_DELIVERY_CNT INTEGER     NOT NULL,
 C_BALANCE     DECIMAL(12,2) NOT NULL,
 C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
 C_PAYMENT_CNT INTEGER      NOT NULL
)
IN ts_customer_052
INDEX IN is_customer_052
ORGANIZE BY KEY SEQUENCE (
 C_ID STARTING FROM 1 ENDING AT 3000,
 C_W_ID STARTING FROM 41413 ENDING AT 42224,
 C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
```

```
DROP TABLE CUSTOMER53;
CREATE TABLE CUSTOMER53
        (
         C_ID          INTEGER      NOT NULL,
         C_STATE       CHAR(2)      NOT NULL,
         C_ZIP         CHAR(9)      NOT NULL,
         C_PHONE       CHAR(16)     NOT NULL,
         C_SINCE       TIMESTAMP    NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)      NOT NULL,
         C_CREDIT      CHAR(2)      NOT NULL,
         C_DISCOUNT    REAL         NOT NULL,
         C_DATA        VARCHAR(500) NOT NULL,
         C_LAST        VARCHAR(16)  NOT NULL,
         C_FIRST       VARCHAR(16)  NOT NULL,
         C_STREET_1    VARCHAR(20)  NOT NULL,
         C_STREET_2    VARCHAR(20)  NOT NULL,
         C_CITY        VARCHAR(20)  NOT NULL,
         C_D_ID        SMALLINT     NOT NULL,
         C_W_ID        INTEGER      NOT NULL,
         C_DELIVERY_CNT INTEGER     NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_053
        INDEX IN is_customer_053
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 42225 ENDING AT 43036,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER54;
CREATE TABLE CUSTOMER54
        (
         C_ID          INTEGER      NOT NULL,
         C_STATE       CHAR(2)      NOT NULL,
         C_ZIP         CHAR(9)      NOT NULL,
         C_PHONE       CHAR(16)     NOT NULL,
         C_SINCE       TIMESTAMP    NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)      NOT NULL,
         C_CREDIT      CHAR(2)      NOT NULL,
         C_DISCOUNT    REAL         NOT NULL,
         C_DATA        VARCHAR(500) NOT NULL,
         C_LAST        VARCHAR(16)  NOT NULL,
         C_FIRST       VARCHAR(16)  NOT NULL,
         C_STREET_1    VARCHAR(20)  NOT NULL,
         C_STREET_2    VARCHAR(20)  NOT NULL,
         C_CITY        VARCHAR(20)  NOT NULL,
         C_D_ID        SMALLINT     NOT NULL,
         C_W_ID        INTEGER      NOT NULL,
         C_DELIVERY_CNT INTEGER     NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_054
        INDEX IN is_customer_054
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 43037 ENDING AT 43848,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER55;
CREATE TABLE CUSTOMER55
        (
         C_ID          INTEGER      NOT NULL,
         C_STATE       CHAR(2)      NOT NULL,
         C_ZIP         CHAR(9)      NOT NULL,
         C_PHONE       CHAR(16)     NOT NULL,
         C_SINCE       TIMESTAMP    NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)      NOT NULL,
         C_CREDIT      CHAR(2)      NOT NULL,
         C_DISCOUNT    REAL         NOT NULL,
         C_DATA        VARCHAR(500) NOT NULL,
         C_LAST        VARCHAR(16)  NOT NULL,
         C_FIRST       VARCHAR(16)  NOT NULL,
         C_STREET_1    VARCHAR(20)  NOT NULL,
         C_STREET_2    VARCHAR(20)  NOT NULL,
         C_CITY        VARCHAR(20)  NOT NULL,
         C_D_ID        SMALLINT     NOT NULL,
         C_W_ID        INTEGER      NOT NULL,
         C_DELIVERY_CNT INTEGER     NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_055
        INDEX IN is_customer_055
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 43849 ENDING AT 44660,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER56;
CREATE TABLE CUSTOMER56
        (
         C_ID          INTEGER      NOT NULL,
         C_STATE       CHAR(2)      NOT NULL,
         C_ZIP         CHAR(9)      NOT NULL,
         C_PHONE       CHAR(16)     NOT NULL,
         C_SINCE       TIMESTAMP    NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)      NOT NULL,
         C_CREDIT      CHAR(2)      NOT NULL,
         C_DISCOUNT    REAL         NOT NULL,
         C_DATA        VARCHAR(500) NOT NULL,
         C_LAST        VARCHAR(16)  NOT NULL,
         C_FIRST       VARCHAR(16)  NOT NULL,
         C_STREET_1    VARCHAR(20)  NOT NULL,
         C_STREET_2    VARCHAR(20)  NOT NULL,
         C_CITY        VARCHAR(20)  NOT NULL,
         C_D_ID        SMALLINT     NOT NULL,
         C_W_ID        INTEGER      NOT NULL,
         C_DELIVERY_CNT INTEGER     NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_056
        INDEX IN is_customer_056
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 44661 ENDING AT 45472,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER57;
CREATE TABLE CUSTOMER57
        (
         C_ID          INTEGER      NOT NULL,
         C_STATE       CHAR(2)      NOT NULL,
         C_ZIP         CHAR(9)      NOT NULL,
         C_PHONE       CHAR(16)     NOT NULL,
         C_SINCE       TIMESTAMP    NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)      NOT NULL,
         C_CREDIT      CHAR(2)      NOT NULL,
         C_DISCOUNT    REAL         NOT NULL,
         C_DATA        VARCHAR(500) NOT NULL,
         C_LAST        VARCHAR(16)  NOT NULL,
         C_FIRST       VARCHAR(16)  NOT NULL,
         C_STREET_1    VARCHAR(20)  NOT NULL,
         C_STREET_2    VARCHAR(20)  NOT NULL,
         C_CITY        VARCHAR(20)  NOT NULL,
         C_D_ID        SMALLINT     NOT NULL,
         C_W_ID        INTEGER      NOT NULL,
         C_DELIVERY_CNT INTEGER     NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_057
        INDEX IN is_customer_057
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 45473 ENDING AT 46284,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER58;
CREATE TABLE CUSTOMER58
        (
         C_ID          INTEGER      NOT NULL,
         C_STATE       CHAR(2)      NOT NULL,
         C_ZIP         CHAR(9)      NOT NULL,
         C_PHONE       CHAR(16)     NOT NULL,
         C_SINCE       TIMESTAMP    NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
         C_MIDDLE      CHAR(2)      NOT NULL,
         C_CREDIT      CHAR(2)      NOT NULL,
         C_DISCOUNT    REAL         NOT NULL,
         C_DATA        VARCHAR(500) NOT NULL,
         C_LAST        VARCHAR(16)  NOT NULL,
         C_FIRST       VARCHAR(16)  NOT NULL,
         C_STREET_1    VARCHAR(20)  NOT NULL,
         C_STREET_2    VARCHAR(20)  NOT NULL,
         C_CITY        VARCHAR(20)  NOT NULL,
         C_D_ID        SMALLINT     NOT NULL,
         C_W_ID        INTEGER      NOT NULL,
         C_DELIVERY_CNT INTEGER     NOT NULL,
         C_BALANCE     DECIMAL(12,2) NOT NULL,
         C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
         C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_058
        INDEX IN is_customer_058
        ORGANIZE BY KEY SEQUENCE (
         C_ID STARTING FROM 1 ENDING AT 3000,
         C_W_ID STARTING FROM 46285 ENDING AT 47096,
         C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER59;
CREATE TABLE CUSTOMER59
        (
         C_ID          INTEGER      NOT NULL,
         C_STATE       CHAR(2)      NOT NULL,
         C_ZIP         CHAR(9)      NOT NULL,
         C_PHONE       CHAR(16)     NOT NULL,
         C_SINCE       TIMESTAMP    NOT NULL,
         C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
```

```
        C_MIDDLE      CHAR(2)      NOT NULL,
        C_CREDIT      CHAR(2)      NOT NULL,
        C_DISCOUNT    REAL         NOT NULL,
        C_DATA        VARCHAR(500) NOT NULL,
        C_LAST        VARCHAR(16)  NOT NULL,
        C_FIRST       VARCHAR(16)  NOT NULL,
        C_STREET_1    VARCHAR(20)  NOT NULL,
        C_STREET_2    VARCHAR(20)  NOT NULL,
        C_CITY        VARCHAR(20)  NOT NULL,
        C_D_ID        SMALLINT     NOT NULL,
        C_W_ID        INTEGER      NOT NULL,
        C_DELIVERY_CNT INTEGER     NOT NULL,
        C_BALANCE     DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_059
        INDEX IN is_customer_059
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 47097 ENDING AT 47908,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER60;
CREATE TABLE CUSTOMER60
        (
        C_ID          INTEGER      NOT NULL,
        C_STATE       CHAR(2)      NOT NULL,
        C_ZIP         CHAR(9)      NOT NULL,
        C_PHONE       CHAR(16)     NOT NULL,
        C_SINCE       TIMESTAMP    NOT NULL,
        C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
        C_MIDDLE      CHAR(2)      NOT NULL,
        C_CREDIT      CHAR(2)      NOT NULL,
        C_DISCOUNT    REAL         NOT NULL,
        C_DATA        VARCHAR(500) NOT NULL,
        C_LAST        VARCHAR(16)  NOT NULL,
        C_FIRST       VARCHAR(16)  NOT NULL,
        C_STREET_1    VARCHAR(20)  NOT NULL,
        C_STREET_2    VARCHAR(20)  NOT NULL,
        C_CITY        VARCHAR(20)  NOT NULL,
        C_D_ID        SMALLINT     NOT NULL,
        C_W_ID        INTEGER      NOT NULL,
        C_DELIVERY_CNT INTEGER     NOT NULL,
        C_BALANCE     DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_060
        INDEX IN is_customer_060
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 47909 ENDING AT 48720,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER61;
CREATE TABLE CUSTOMER61
        (
        C_ID          INTEGER      NOT NULL,
        C_STATE       CHAR(2)      NOT NULL,
        C_ZIP         CHAR(9)      NOT NULL,
        C_PHONE       CHAR(16)     NOT NULL,
        C_SINCE       TIMESTAMP    NOT NULL,
        C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
        C_MIDDLE      CHAR(2)      NOT NULL,
        C_CREDIT      CHAR(2)      NOT NULL,
        C_DISCOUNT    REAL         NOT NULL,
        C_DATA        VARCHAR(500) NOT NULL,
        C_LAST        VARCHAR(16)  NOT NULL,
        C_FIRST       VARCHAR(16)  NOT NULL,
        C_STREET_1    VARCHAR(20)  NOT NULL,
        C_STREET_2    VARCHAR(20)  NOT NULL,
        C_CITY        VARCHAR(20)  NOT NULL,
        C_D_ID        SMALLINT     NOT NULL,
        C_W_ID        INTEGER      NOT NULL,
        C_DELIVERY_CNT INTEGER     NOT NULL,
        C_BALANCE     DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_061
        INDEX IN is_customer_061
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 48721 ENDING AT 49532,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER62;
CREATE TABLE CUSTOMER62
        (
        C_ID          INTEGER      NOT NULL,
        C_STATE       CHAR(2)      NOT NULL,
        C_ZIP         CHAR(9)      NOT NULL,
        C_PHONE       CHAR(16)     NOT NULL,
        C_SINCE       TIMESTAMP    NOT NULL,
        C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
        C_MIDDLE      CHAR(2)      NOT NULL,
        C_CREDIT      CHAR(2)      NOT NULL,
        C_DISCOUNT    REAL         NOT NULL,
        C_DATA        VARCHAR(500) NOT NULL,
        C_LAST        VARCHAR(16)  NOT NULL,
        C_FIRST       VARCHAR(16)  NOT NULL,
        C_STREET_1    VARCHAR(20)  NOT NULL,
        C_STREET_2    VARCHAR(20)  NOT NULL,
        C_CITY        VARCHAR(20)  NOT NULL,
        C_D_ID        SMALLINT     NOT NULL,
        C_W_ID        INTEGER      NOT NULL,
        C_DELIVERY_CNT INTEGER     NOT NULL,
        C_BALANCE     DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_062
        INDEX IN is_customer_062
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 49533 ENDING AT 50344,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER63;
CREATE TABLE CUSTOMER63
        (
        C_ID          INTEGER      NOT NULL,
        C_STATE       CHAR(2)      NOT NULL,
        C_ZIP         CHAR(9)      NOT NULL,
        C_PHONE       CHAR(16)     NOT NULL,
        C_SINCE       TIMESTAMP    NOT NULL,
        C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
        C_MIDDLE      CHAR(2)      NOT NULL,
        C_CREDIT      CHAR(2)      NOT NULL,
        C_DISCOUNT    REAL         NOT NULL,
        C_DATA        VARCHAR(500) NOT NULL,
        C_LAST        VARCHAR(16)  NOT NULL,
        C_FIRST       VARCHAR(16)  NOT NULL,
        C_STREET_1    VARCHAR(20)  NOT NULL,
        C_STREET_2    VARCHAR(20)  NOT NULL,
        C_CITY        VARCHAR(20)  NOT NULL,
        C_D_ID        SMALLINT     NOT NULL,
        C_W_ID        INTEGER      NOT NULL,
        C_DELIVERY_CNT INTEGER     NOT NULL,
        C_BALANCE     DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_063
        INDEX IN is_customer_063
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 50345 ENDING AT 51156,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER64;
CREATE TABLE CUSTOMER64
        (
        C_ID          INTEGER      NOT NULL,
        C_STATE       CHAR(2)      NOT NULL,
        C_ZIP         CHAR(9)      NOT NULL,
        C_PHONE       CHAR(16)     NOT NULL,
        C_SINCE       TIMESTAMP    NOT NULL,
        C_CREDIT_LIM  DECIMAL(12,2) NOT NULL,
        C_MIDDLE      CHAR(2)      NOT NULL,
        C_CREDIT      CHAR(2)      NOT NULL,
        C_DISCOUNT    REAL         NOT NULL,
        C_DATA        VARCHAR(500) NOT NULL,
        C_LAST        VARCHAR(16)  NOT NULL,
        C_FIRST       VARCHAR(16)  NOT NULL,
        C_STREET_1    VARCHAR(20)  NOT NULL,
        C_STREET_2    VARCHAR(20)  NOT NULL,
        C_CITY        VARCHAR(20)  NOT NULL,
        C_D_ID        SMALLINT     NOT NULL,
        C_W_ID        INTEGER      NOT NULL,
        C_DELIVERY_CNT INTEGER     NOT NULL,
        C_BALANCE     DECIMAL(12,2) NOT NULL,
        C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
        C_PAYMENT_CNT INTEGER      NOT NULL
        )
        IN ts_customer_064
        INDEX IN is_customer_064
        ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 51157 ENDING AT 51968,
        C_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

DDL/CRTB_DISTRICT.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT1;
CREATE TABLE DISTRICT1
        (
        D_NEXT_O_ID INTEGER      NOT NULL,
        D_TAX       REAL         NOT NULL,
        D_YTD       DECIMAL(12,2) NOT NULL,
        D_NAME      CHAR(10)     NOT NULL,
        D_STREET_1  CHAR(20)     NOT NULL,
        D_STREET_2  CHAR(20)     NOT NULL,
        D_CITY      CHAR(20)     NOT NULL,
        D_STATE     CHAR(2)      NOT NULL,
```

```
        D_ZIP      CHAR(9)       NOT NULL,
        D_ID       SMALLINT      NOT NULL,
        D_W_ID     INTEGER       NOT NULL
        )
        IN ts_dist_001
        INDEX IN ts_dist_001
        ORGANIZE BY KEY SEQUENCE (
        D_ID STARTING FROM 1 ENDING AT 10,
        D_W_ID STARTING FROM 1 ENDING AT 1624
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT2;
CREATE TABLE DISTRICT2
        (
        D_NEXT_O_ID INTEGER       NOT NULL,
        D_TAX      REAL          NOT NULL,
        D_YTD      DECIMAL(12,2)  NOT NULL,
        D_NAME     CHAR(10)      NOT NULL,
        D_STREET_1  CHAR(20)      NOT NULL,
        D_STREET_2  CHAR(20)      NOT NULL,
        D_CITY     CHAR(20)      NOT NULL,
        D_STATE    CHAR(2)       NOT NULL,
        D_ZIP      CHAR(9)       NOT NULL,
        D_ID       SMALLINT      NOT NULL,
        D_W_ID     INTEGER       NOT NULL
        )
        IN ts_dist_002
        INDEX IN ts_dist_002
        ORGANIZE BY KEY SEQUENCE (
        D_ID STARTING FROM 1 ENDING AT 10,
        D_W_ID STARTING FROM 1625 ENDING AT 3248
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT3;
CREATE TABLE DISTRICT3
        (
        D_NEXT_O_ID INTEGER       NOT NULL,
        D_TAX      REAL          NOT NULL,
        D_YTD      DECIMAL(12,2)  NOT NULL,
        D_NAME     CHAR(10)      NOT NULL,
        D_STREET_1  CHAR(20)      NOT NULL,
        D_STREET_2  CHAR(20)      NOT NULL,
        D_CITY     CHAR(20)      NOT NULL,
        D_STATE    CHAR(2)       NOT NULL,
        D_ZIP      CHAR(9)       NOT NULL,
        D_ID       SMALLINT      NOT NULL,
        D_W_ID     INTEGER       NOT NULL
        )
        IN ts_dist_003
        INDEX IN ts_dist_003
        ORGANIZE BY KEY SEQUENCE (
        D_ID STARTING FROM 1 ENDING AT 10,
        D_W_ID STARTING FROM 3249 ENDING AT 4872
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT4;
CREATE TABLE DISTRICT4
        (
        D_NEXT_O_ID INTEGER       NOT NULL,
        D_TAX      REAL          NOT NULL,
        D_YTD      DECIMAL(12,2)  NOT NULL,
        D_NAME     CHAR(10)      NOT NULL,
        D_STREET_1  CHAR(20)      NOT NULL,
        D_STREET_2  CHAR(20)      NOT NULL,
        D_CITY     CHAR(20)      NOT NULL,
        D_STATE    CHAR(2)       NOT NULL,
        D_ZIP      CHAR(9)       NOT NULL,
        D_ID       SMALLINT      NOT NULL,
        D_W_ID     INTEGER       NOT NULL
        )
        IN ts_dist_004
        INDEX IN ts_dist_004
        ORGANIZE BY KEY SEQUENCE (
        D_ID STARTING FROM 1 ENDING AT 10,
        D_W_ID STARTING FROM 4873 ENDING AT 6496
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT5;
CREATE TABLE DISTRICT5
        (
        D_NEXT_O_ID INTEGER       NOT NULL,
        D_TAX      REAL          NOT NULL,
        D_YTD      DECIMAL(12,2)  NOT NULL,
        D_NAME     CHAR(10)      NOT NULL,
        D_STREET_1  CHAR(20)      NOT NULL,
        D_STREET_2  CHAR(20)      NOT NULL,
        D_CITY     CHAR(20)      NOT NULL,
        D_STATE    CHAR(2)       NOT NULL,
        D_ZIP      CHAR(9)       NOT NULL,
        D_ID       SMALLINT      NOT NULL,
        D_W_ID     INTEGER       NOT NULL
        )
        IN ts_dist_005
        INDEX IN ts_dist_005
        ORGANIZE BY KEY SEQUENCE (
        D_ID STARTING FROM 1 ENDING AT 10,
        D_W_ID STARTING FROM 6497 ENDING AT 8120
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT6;
CREATE TABLE DISTRICT6
        (
        D_NEXT_O_ID INTEGER       NOT NULL,
        D_TAX      REAL          NOT NULL,
        D_YTD      DECIMAL(12,2)  NOT NULL,
        D_NAME     CHAR(10)      NOT NULL,
        D_STREET_1  CHAR(20)      NOT NULL,
        D_STREET_2  CHAR(20)      NOT NULL,
        D_CITY     CHAR(20)      NOT NULL,
        D_STATE    CHAR(2)       NOT NULL,
        D_ZIP      CHAR(9)       NOT NULL,
        D_ID       SMALLINT      NOT NULL,
        D_W_ID     INTEGER       NOT NULL
        )
        IN ts_dist_006
        INDEX IN ts_dist_006
        ORGANIZE BY KEY SEQUENCE (
        D_ID STARTING FROM 1 ENDING AT 10,
        D_W_ID STARTING FROM 8121 ENDING AT 9744
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT7;
CREATE TABLE DISTRICT7
        (
        D_NEXT_O_ID INTEGER       NOT NULL,
        D_TAX      REAL          NOT NULL,
        D_YTD      DECIMAL(12,2)  NOT NULL,
        D_NAME     CHAR(10)      NOT NULL,
        D_STREET_1  CHAR(20)      NOT NULL,
        D_STREET_2  CHAR(20)      NOT NULL,
        D_CITY     CHAR(20)      NOT NULL,
        D_STATE    CHAR(2)       NOT NULL,
        D_ZIP      CHAR(9)       NOT NULL,
        D_ID       SMALLINT      NOT NULL,
        D_W_ID     INTEGER       NOT NULL
        )
        IN ts_dist_007
        INDEX IN ts_dist_007
        ORGANIZE BY KEY SEQUENCE (
        D_ID STARTING FROM 1 ENDING AT 10,
        D_W_ID STARTING FROM 9745 ENDING AT 11368
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT8;
CREATE TABLE DISTRICT8
        (
        D_NEXT_O_ID INTEGER       NOT NULL,
        D_TAX      REAL          NOT NULL,
        D_YTD      DECIMAL(12,2)  NOT NULL,
        D_NAME     CHAR(10)      NOT NULL,
        D_STREET_1  CHAR(20)      NOT NULL,
        D_STREET_2  CHAR(20)      NOT NULL,
        D_CITY     CHAR(20)      NOT NULL,
        D_STATE    CHAR(2)       NOT NULL,
        D_ZIP      CHAR(9)       NOT NULL,
        D_ID       SMALLINT      NOT NULL,
        D_W_ID     INTEGER       NOT NULL
        )
        IN ts_dist_008
        INDEX IN ts_dist_008
        ORGANIZE BY KEY SEQUENCE (
        D_ID STARTING FROM 1 ENDING AT 10,
        D_W_ID STARTING FROM 11369 ENDING AT 12992
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT9;
CREATE TABLE DISTRICT9
        (
        D_NEXT_O_ID INTEGER       NOT NULL,
        D_TAX      REAL          NOT NULL,
        D_YTD      DECIMAL(12,2)  NOT NULL,
        D_NAME     CHAR(10)      NOT NULL,
        D_STREET_1  CHAR(20)      NOT NULL,
        D_STREET_2  CHAR(20)      NOT NULL,
        D_CITY     CHAR(20)      NOT NULL,
        D_STATE    CHAR(2)       NOT NULL,
        D_ZIP      CHAR(9)       NOT NULL,
        D_ID       SMALLINT      NOT NULL,
        D_W_ID     INTEGER       NOT NULL
        )
        IN ts_dist_009
        INDEX IN ts_dist_009
        ORGANIZE BY KEY SEQUENCE (
        D_ID STARTING FROM 1 ENDING AT 10,
        D_W_ID STARTING FROM 12993 ENDING AT 14616
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT10;
CREATE TABLE DISTRICT10
        (
        D_NEXT_O_ID INTEGER       NOT NULL,
        D_TAX      REAL          NOT NULL,
        D_YTD      DECIMAL(12,2)  NOT NULL,
        D_NAME     CHAR(10)      NOT NULL,
        D_STREET_1  CHAR(20)      NOT NULL,
        D_STREET_2  CHAR(20)      NOT NULL,
        D_CITY     CHAR(20)      NOT NULL,
        D_STATE    CHAR(2)       NOT NULL,
        D_ZIP      CHAR(9)       NOT NULL,
        D_ID       SMALLINT      NOT NULL,
        D_W_ID     INTEGER       NOT NULL
```

```
)
IN ts_dist_010
INDEX IN ts_dist_010
ORGANIZE BY KEY SEQUENCE (
 D_ID STARTING FROM 1 ENDING AT 10,
 D_W_ID STARTING FROM 14617 ENDING AT 16240
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT11;
CREATE TABLE DISTRICT11
(
 D_NEXT_O_ID INTEGER       NOT NULL,
 D_TAX      REAL          NOT NULL,
 D_YTD      DECIMAL(12,2)  NOT NULL,
 D_NAME     CHAR(10)       NOT NULL,
 D_STREET_1  CHAR(20)       NOT NULL,
 D_STREET_2  CHAR(20)       NOT NULL,
 D_CITY     CHAR(20)       NOT NULL,
 D_STATE    CHAR(2)        NOT NULL,
 D_ZIP      CHAR(9)        NOT NULL,
 D_ID       SMALLINT       NOT NULL,
 D_W_ID     INTEGER        NOT NULL
)
IN ts_dist_011
INDEX IN ts_dist_011
ORGANIZE BY KEY SEQUENCE (
 D_ID STARTING FROM 1 ENDING AT 10,
 D_W_ID STARTING FROM 16241 ENDING AT 17864
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT12;
CREATE TABLE DISTRICT12
(
 D_NEXT_O_ID INTEGER       NOT NULL,
 D_TAX      REAL          NOT NULL,
 D_YTD      DECIMAL(12,2)  NOT NULL,
 D_NAME     CHAR(10)       NOT NULL,
 D_STREET_1  CHAR(20)       NOT NULL,
 D_STREET_2  CHAR(20)       NOT NULL,
 D_CITY     CHAR(20)       NOT NULL,
 D_STATE    CHAR(2)        NOT NULL,
 D_ZIP      CHAR(9)        NOT NULL,
 D_ID       SMALLINT       NOT NULL,
 D_W_ID     INTEGER        NOT NULL
)
IN ts_dist_012
INDEX IN ts_dist_012
ORGANIZE BY KEY SEQUENCE (
 D_ID STARTING FROM 1 ENDING AT 10,
 D_W_ID STARTING FROM 17865 ENDING AT 19488
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT13;
CREATE TABLE DISTRICT13
(
 D_NEXT_O_ID INTEGER       NOT NULL,
 D_TAX      REAL          NOT NULL,
 D_YTD      DECIMAL(12,2)  NOT NULL,
 D_NAME     CHAR(10)       NOT NULL,
 D_STREET_1  CHAR(20)       NOT NULL,
 D_STREET_2  CHAR(20)       NOT NULL,
 D_CITY     CHAR(20)       NOT NULL,
 D_STATE    CHAR(2)        NOT NULL,
 D_ZIP      CHAR(9)        NOT NULL,
 D_ID       SMALLINT       NOT NULL,
 D_W_ID     INTEGER        NOT NULL
)
IN ts_dist_013
INDEX IN ts_dist_013
ORGANIZE BY KEY SEQUENCE (
 D_ID STARTING FROM 1 ENDING AT 10,
 D_W_ID STARTING FROM 19489 ENDING AT 21112
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT14;
CREATE TABLE DISTRICT14
(
 D_NEXT_O_ID INTEGER       NOT NULL,
 D_TAX      REAL          NOT NULL,
 D_YTD      DECIMAL(12,2)  NOT NULL,
 D_NAME     CHAR(10)       NOT NULL,
 D_STREET_1  CHAR(20)       NOT NULL,
 D_STREET_2  CHAR(20)       NOT NULL,
 D_CITY     CHAR(20)       NOT NULL,
 D_STATE    CHAR(2)        NOT NULL,
 D_ZIP      CHAR(9)        NOT NULL,
 D_ID       SMALLINT       NOT NULL,
 D_W_ID     INTEGER        NOT NULL
)
IN ts_dist_014
INDEX IN ts_dist_014
ORGANIZE BY KEY SEQUENCE (
 D_ID STARTING FROM 1 ENDING AT 10,
 D_W_ID STARTING FROM 21113 ENDING AT 22736
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT15;
CREATE TABLE DISTRICT15
(
 D_NEXT_O_ID INTEGER       NOT NULL,
 D_TAX      REAL          NOT NULL,
 D_YTD      DECIMAL(12,2)  NOT NULL,
 D_NAME     CHAR(10)       NOT NULL,
 D_STREET_1  CHAR(20)       NOT NULL,
 D_STREET_2  CHAR(20)       NOT NULL,
 D_CITY     CHAR(20)       NOT NULL,
 D_STATE    CHAR(2)        NOT NULL,
 D_ZIP      CHAR(9)        NOT NULL,
 D_ID       SMALLINT       NOT NULL,
 D_W_ID     INTEGER        NOT NULL
)
IN ts_dist_015
INDEX IN ts_dist_015
ORGANIZE BY KEY SEQUENCE (
 D_ID STARTING FROM 1 ENDING AT 10,
 D_W_ID STARTING FROM 22737 ENDING AT 24360
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT16;
CREATE TABLE DISTRICT16
(
 D_NEXT_O_ID INTEGER       NOT NULL,
 D_TAX      REAL          NOT NULL,
 D_YTD      DECIMAL(12,2)  NOT NULL,
 D_NAME     CHAR(10)       NOT NULL,
 D_STREET_1  CHAR(20)       NOT NULL,
 D_STREET_2  CHAR(20)       NOT NULL,
 D_CITY     CHAR(20)       NOT NULL,
 D_STATE    CHAR(2)        NOT NULL,
 D_ZIP      CHAR(9)        NOT NULL,
 D_ID       SMALLINT       NOT NULL,
 D_W_ID     INTEGER        NOT NULL
)
IN ts_dist_016
INDEX IN ts_dist_016
ORGANIZE BY KEY SEQUENCE (
 D_ID STARTING FROM 1 ENDING AT 10,
 D_W_ID STARTING FROM 24361 ENDING AT 25984
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT17;
CREATE TABLE DISTRICT17
(
 D_NEXT_O_ID INTEGER       NOT NULL,
 D_TAX      REAL          NOT NULL,
 D_YTD      DECIMAL(12,2)  NOT NULL,
 D_NAME     CHAR(10)       NOT NULL,
 D_STREET_1  CHAR(20)       NOT NULL,
 D_STREET_2  CHAR(20)       NOT NULL,
 D_CITY     CHAR(20)       NOT NULL,
 D_STATE    CHAR(2)        NOT NULL,
 D_ZIP      CHAR(9)        NOT NULL,
 D_ID       SMALLINT       NOT NULL,
 D_W_ID     INTEGER        NOT NULL
)
IN ts_dist_017
INDEX IN ts_dist_017
ORGANIZE BY KEY SEQUENCE (
 D_ID STARTING FROM 1 ENDING AT 10,
 D_W_ID STARTING FROM 25985 ENDING AT 27608
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT18;
CREATE TABLE DISTRICT18
(
 D_NEXT_O_ID INTEGER       NOT NULL,
 D_TAX      REAL          NOT NULL,
 D_YTD      DECIMAL(12,2)  NOT NULL,
 D_NAME     CHAR(10)       NOT NULL,
 D_STREET_1  CHAR(20)       NOT NULL,
 D_STREET_2  CHAR(20)       NOT NULL,
 D_CITY     CHAR(20)       NOT NULL,
 D_STATE    CHAR(2)        NOT NULL,
 D_ZIP      CHAR(9)        NOT NULL,
 D_ID       SMALLINT       NOT NULL,
 D_W_ID     INTEGER        NOT NULL
)
IN ts_dist_018
INDEX IN ts_dist_018
ORGANIZE BY KEY SEQUENCE (
 D_ID STARTING FROM 1 ENDING AT 10,
 D_W_ID STARTING FROM 27609 ENDING AT 29232
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT19;
CREATE TABLE DISTRICT19
(
 D_NEXT_O_ID INTEGER       NOT NULL,
 D_TAX      REAL          NOT NULL,
 D_YTD      DECIMAL(12,2)  NOT NULL,
 D_NAME     CHAR(10)       NOT NULL,
 D_STREET_1  CHAR(20)       NOT NULL,
 D_STREET_2  CHAR(20)       NOT NULL,
 D_CITY     CHAR(20)       NOT NULL,
 D_STATE    CHAR(2)        NOT NULL,
 D_ZIP      CHAR(9)        NOT NULL,
 D_ID       SMALLINT       NOT NULL,
 D_W_ID     INTEGER        NOT NULL
)
IN ts_dist_019
INDEX IN ts_dist_019
```

```
ORGANIZE BY KEY SEQUENCE (
 D_ID STARTING FROM 1 ENDING AT 10,
 D_W_ID STARTING FROM 29233 ENDING AT 30856
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT20;
CREATE TABLE DISTRICT20
(
 D_NEXT_O_ID INTEGER        NOT NULL,
 D_TAX       REAL           NOT NULL,
 D_YTD       DECIMAL(12,2)  NOT NULL,
 D_NAME      CHAR(10)       NOT NULL,
 D_STREET_1  CHAR(20)       NOT NULL,
 D_STREET_2  CHAR(20)       NOT NULL,
 D_CITY      CHAR(20)       NOT NULL,
 D_STATE     CHAR(2)        NOT NULL,
 D_ZIP       CHAR(9)        NOT NULL,
 D_ID        SMALLINT       NOT NULL,
 D_W_ID      INTEGER        NOT NULL
)
IN ts_dist_020
INDEX IN ts_dist_020
ORGANIZE BY KEY SEQUENCE (
 D_ID STARTING FROM 1 ENDING AT 10,
 D_W_ID STARTING FROM 30857 ENDING AT 32480
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT21;
CREATE TABLE DISTRICT21
(
 D_NEXT_O_ID INTEGER        NOT NULL,
 D_TAX       REAL           NOT NULL,
 D_YTD       DECIMAL(12,2)  NOT NULL,
 D_NAME      CHAR(10)       NOT NULL,
 D_STREET_1  CHAR(20)       NOT NULL,
 D_STREET_2  CHAR(20)       NOT NULL,
 D_CITY      CHAR(20)       NOT NULL,
 D_STATE     CHAR(2)        NOT NULL,
 D_ZIP       CHAR(9)        NOT NULL,
 D_ID        SMALLINT       NOT NULL,
 D_W_ID      INTEGER        NOT NULL
)
IN ts_dist_021
INDEX IN ts_dist_021
ORGANIZE BY KEY SEQUENCE (
 D_ID STARTING FROM 1 ENDING AT 10,
 D_W_ID STARTING FROM 32481 ENDING AT 34104
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT22;
CREATE TABLE DISTRICT22
(
 D_NEXT_O_ID INTEGER        NOT NULL,
 D_TAX       REAL           NOT NULL,
 D_YTD       DECIMAL(12,2)  NOT NULL,
 D_NAME      CHAR(10)       NOT NULL,
 D_STREET_1  CHAR(20)       NOT NULL,
 D_STREET_2  CHAR(20)       NOT NULL,
 D_CITY      CHAR(20)       NOT NULL,
 D_STATE     CHAR(2)        NOT NULL,
 D_ZIP       CHAR(9)        NOT NULL,
 D_ID        SMALLINT       NOT NULL,
 D_W_ID      INTEGER        NOT NULL
)
IN ts_dist_022
INDEX IN ts_dist_022
ORGANIZE BY KEY SEQUENCE (
 D_ID STARTING FROM 1 ENDING AT 10,
 D_W_ID STARTING FROM 34105 ENDING AT 35728
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT23;
CREATE TABLE DISTRICT23
(
 D_NEXT_O_ID INTEGER        NOT NULL,
 D_TAX       REAL           NOT NULL,
 D_YTD       DECIMAL(12,2)  NOT NULL,
 D_NAME      CHAR(10)       NOT NULL,
 D_STREET_1  CHAR(20)       NOT NULL,
 D_STREET_2  CHAR(20)       NOT NULL,
 D_CITY      CHAR(20)       NOT NULL,
 D_STATE     CHAR(2)        NOT NULL,
 D_ZIP       CHAR(9)        NOT NULL,
 D_ID        SMALLINT       NOT NULL,
 D_W_ID      INTEGER        NOT NULL
)
IN ts_dist_023
INDEX IN ts_dist_023
ORGANIZE BY KEY SEQUENCE (
 D_ID STARTING FROM 1 ENDING AT 10,
 D_W_ID STARTING FROM 35729 ENDING AT 37352
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT24;
CREATE TABLE DISTRICT24
(
 D_NEXT_O_ID INTEGER        NOT NULL,
 D_TAX       REAL           NOT NULL,
 D_YTD       DECIMAL(12,2)  NOT NULL,
 D_NAME      CHAR(10)       NOT NULL,
 D_STREET_1  CHAR(20)       NOT NULL,
 D_STREET_2  CHAR(20)       NOT NULL,
 D_CITY      CHAR(20)       NOT NULL,
 D_STATE     CHAR(2)        NOT NULL,
 D_ZIP       CHAR(9)        NOT NULL,
 D_ID        SMALLINT       NOT NULL,
 D_W_ID      INTEGER        NOT NULL
)
IN ts_dist_024
INDEX IN ts_dist_024
ORGANIZE BY KEY SEQUENCE (
 D_ID STARTING FROM 1 ENDING AT 10,
 D_W_ID STARTING FROM 37353 ENDING AT 38976
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT25;
CREATE TABLE DISTRICT25
(
 D_NEXT_O_ID INTEGER        NOT NULL,
 D_TAX       REAL           NOT NULL,
 D_YTD       DECIMAL(12,2)  NOT NULL,
 D_NAME      CHAR(10)       NOT NULL,
 D_STREET_1  CHAR(20)       NOT NULL,
 D_STREET_2  CHAR(20)       NOT NULL,
 D_CITY      CHAR(20)       NOT NULL,
 D_STATE     CHAR(2)        NOT NULL,
 D_ZIP       CHAR(9)        NOT NULL,
 D_ID        SMALLINT       NOT NULL,
 D_W_ID      INTEGER        NOT NULL
)
IN ts_dist_025
INDEX IN ts_dist_025
ORGANIZE BY KEY SEQUENCE (
 D_ID STARTING FROM 1 ENDING AT 10,
 D_W_ID STARTING FROM 38977 ENDING AT 40600
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT26;
CREATE TABLE DISTRICT26
(
 D_NEXT_O_ID INTEGER        NOT NULL,
 D_TAX       REAL           NOT NULL,
 D_YTD       DECIMAL(12,2)  NOT NULL,
 D_NAME      CHAR(10)       NOT NULL,
 D_STREET_1  CHAR(20)       NOT NULL,
 D_STREET_2  CHAR(20)       NOT NULL,
 D_CITY      CHAR(20)       NOT NULL,
 D_STATE     CHAR(2)        NOT NULL,
 D_ZIP       CHAR(9)        NOT NULL,
 D_ID        SMALLINT       NOT NULL,
 D_W_ID      INTEGER        NOT NULL
)
IN ts_dist_026
INDEX IN ts_dist_026
ORGANIZE BY KEY SEQUENCE (
 D_ID STARTING FROM 1 ENDING AT 10,
 D_W_ID STARTING FROM 40601 ENDING AT 42224
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT27;
CREATE TABLE DISTRICT27
(
 D_NEXT_O_ID INTEGER        NOT NULL,
 D_TAX       REAL           NOT NULL,
 D_YTD       DECIMAL(12,2)  NOT NULL,
 D_NAME      CHAR(10)       NOT NULL,
 D_STREET_1  CHAR(20)       NOT NULL,
 D_STREET_2  CHAR(20)       NOT NULL,
 D_CITY      CHAR(20)       NOT NULL,
 D_STATE     CHAR(2)        NOT NULL,
 D_ZIP       CHAR(9)        NOT NULL,
 D_ID        SMALLINT       NOT NULL,
 D_W_ID      INTEGER        NOT NULL
)
IN ts_dist_027
INDEX IN ts_dist_027
ORGANIZE BY KEY SEQUENCE (
 D_ID STARTING FROM 1 ENDING AT 10,
 D_W_ID STARTING FROM 42225 ENDING AT 43848
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT28;
CREATE TABLE DISTRICT28
(
 D_NEXT_O_ID INTEGER        NOT NULL,
 D_TAX       REAL           NOT NULL,
 D_YTD       DECIMAL(12,2)  NOT NULL,
 D_NAME      CHAR(10)       NOT NULL,
 D_STREET_1  CHAR(20)       NOT NULL,
 D_STREET_2  CHAR(20)       NOT NULL,
 D_CITY      CHAR(20)       NOT NULL,
 D_STATE     CHAR(2)        NOT NULL,
 D_ZIP       CHAR(9)        NOT NULL,
 D_ID        SMALLINT       NOT NULL,
 D_W_ID      INTEGER        NOT NULL
)
IN ts_dist_028
INDEX IN ts_dist_028
ORGANIZE BY KEY SEQUENCE (
 D_ID STARTING FROM 1 ENDING AT 10,
 D_W_ID STARTING FROM 43849 ENDING AT 45472
```

```
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT29;
CREATE TABLE DISTRICT29
        (
         D_NEXT_O_ID INTEGER        NOT NULL,
         D_TAX       REAL           NOT NULL,
         D_YTD       DECIMAL(12,2)  NOT NULL,
         D_NAME      CHAR(10)       NOT NULL,
         D_STREET_1  CHAR(20)       NOT NULL,
         D_STREET_2  CHAR(20)       NOT NULL,
         D_CITY      CHAR(20)       NOT NULL,
         D_STATE     CHAR(2)        NOT NULL,
         D_ZIP       CHAR(9)        NOT NULL,
         D_ID        SMALLINT       NOT NULL,
         D_W_ID      INTEGER        NOT NULL
        )
        IN ts_dist_029
        INDEX IN ts_dist_029
        ORGANIZE BY KEY SEQUENCE (
         D_ID STARTING FROM 1 ENDING AT 10,
         D_W_ID STARTING FROM 45473 ENDING AT 47096
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT30;
CREATE TABLE DISTRICT30
        (
         D_NEXT_O_ID INTEGER        NOT NULL,
         D_TAX       REAL           NOT NULL,
         D_YTD       DECIMAL(12,2)  NOT NULL,
         D_NAME      CHAR(10)       NOT NULL,
         D_STREET_1  CHAR(20)       NOT NULL,
         D_STREET_2  CHAR(20)       NOT NULL,
         D_CITY      CHAR(20)       NOT NULL,
         D_STATE     CHAR(2)        NOT NULL,
         D_ZIP       CHAR(9)        NOT NULL,
         D_ID        SMALLINT       NOT NULL,
         D_W_ID      INTEGER        NOT NULL
        )
        IN ts_dist_030
        INDEX IN ts_dist_030
        ORGANIZE BY KEY SEQUENCE (
         D_ID STARTING FROM 1 ENDING AT 10,
         D_W_ID STARTING FROM 47097 ENDING AT 48720
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT31;
CREATE TABLE DISTRICT31
        (
         D_NEXT_O_ID INTEGER        NOT NULL,
         D_TAX       REAL           NOT NULL,
         D_YTD       DECIMAL(12,2)  NOT NULL,
         D_NAME      CHAR(10)       NOT NULL,
         D_STREET_1  CHAR(20)       NOT NULL,
         D_STREET_2  CHAR(20)       NOT NULL,
         D_CITY      CHAR(20)       NOT NULL,
         D_STATE     CHAR(2)        NOT NULL,
         D_ZIP       CHAR(9)        NOT NULL,
         D_ID        SMALLINT       NOT NULL,
         D_W_ID      INTEGER        NOT NULL
        )
        IN ts_dist_031
        INDEX IN ts_dist_031
        ORGANIZE BY KEY SEQUENCE (
         D_ID STARTING FROM 1 ENDING AT 10,
         D_W_ID STARTING FROM 48721 ENDING AT 50344
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT32;
CREATE TABLE DISTRICT32
        (
         D_NEXT_O_ID INTEGER        NOT NULL,
         D_TAX       REAL           NOT NULL,
         D_YTD       DECIMAL(12,2)  NOT NULL,
         D_NAME      CHAR(10)       NOT NULL,
         D_STREET_1  CHAR(20)       NOT NULL,
         D_STREET_2  CHAR(20)       NOT NULL,
         D_CITY      CHAR(20)       NOT NULL,
         D_STATE     CHAR(2)        NOT NULL,
         D_ZIP       CHAR(9)        NOT NULL,
         D_ID        SMALLINT       NOT NULL,
         D_W_ID      INTEGER        NOT NULL
        )
        IN ts_dist_032
        INDEX IN ts_dist_032
        ORGANIZE BY KEY SEQUENCE (
         D_ID STARTING FROM 1 ENDING AT 10,
         D_W_ID STARTING FROM 50345 ENDING AT 51968
        )
        ALLOW OVERFLOW;
connect reset;
```

DDL/CRTB_HISTORY.ddl

```
connect to TPCC in share mode;
DROP TABLE HISTORY1;
CREATE TABLE HISTORY1
        (
         H_C_ID      INTEGER     NOT NULL,
         H_C_D_ID    SMALLINT    NOT NULL,
         H_C_W_ID    INTEGER     NOT NULL,
         H_D_ID      SMALLINT    NOT NULL,
         H_W_ID      INTEGER     NOT NULL,
         H_DATE      TIMESTAMP   NOT NULL,
         H_AMOUNT    DECIMAL(6,2) NOT NULL,
         H_DATA      CHAR(24)    NOT NULL
        )
        IN ts_history_001
        INDEX IN ts_history_001;
ALTER TABLE HISTORY1 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY2;
CREATE TABLE HISTORY2
        (
         H_C_ID      INTEGER     NOT NULL,
         H_C_D_ID    SMALLINT    NOT NULL,
         H_C_W_ID    INTEGER     NOT NULL,
         H_D_ID      SMALLINT    NOT NULL,
         H_W_ID      INTEGER     NOT NULL,
         H_DATE      TIMESTAMP   NOT NULL,
         H_AMOUNT    DECIMAL(6,2) NOT NULL,
         H_DATA      CHAR(24)    NOT NULL
        )
        IN ts_history_002
        INDEX IN ts_history_002;
ALTER TABLE HISTORY2 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY3;
CREATE TABLE HISTORY3
        (
         H_C_ID      INTEGER     NOT NULL,
         H_C_D_ID    SMALLINT    NOT NULL,
         H_C_W_ID    INTEGER     NOT NULL,
         H_D_ID      SMALLINT    NOT NULL,
         H_W_ID      INTEGER     NOT NULL,
         H_DATE      TIMESTAMP   NOT NULL,
         H_AMOUNT    DECIMAL(6,2) NOT NULL,
         H_DATA      CHAR(24)    NOT NULL
        )
        IN ts_history_003
        INDEX IN ts_history_003;
ALTER TABLE HISTORY3 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY4;
CREATE TABLE HISTORY4
        (
         H_C_ID      INTEGER     NOT NULL,
         H_C_D_ID    SMALLINT    NOT NULL,
         H_C_W_ID    INTEGER     NOT NULL,
         H_D_ID      SMALLINT    NOT NULL,
         H_W_ID      INTEGER     NOT NULL,
         H_DATE      TIMESTAMP   NOT NULL,
         H_AMOUNT    DECIMAL(6,2) NOT NULL,
         H_DATA      CHAR(24)    NOT NULL
        )
        IN ts_history_004
        INDEX IN ts_history_004;
ALTER TABLE HISTORY4 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY5;
CREATE TABLE HISTORY5
        (
         H_C_ID      INTEGER     NOT NULL,
         H_C_D_ID    SMALLINT    NOT NULL,
         H_C_W_ID    INTEGER     NOT NULL,
         H_D_ID      SMALLINT    NOT NULL,
         H_W_ID      INTEGER     NOT NULL,
         H_DATE      TIMESTAMP   NOT NULL,
         H_AMOUNT    DECIMAL(6,2) NOT NULL,
         H_DATA      CHAR(24)    NOT NULL
        )
        IN ts_history_005
        INDEX IN ts_history_005;
ALTER TABLE HISTORY5 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY6;
CREATE TABLE HISTORY6
        (
         H_C_ID      INTEGER     NOT NULL,
         H_C_D_ID    SMALLINT    NOT NULL,
         H_C_W_ID    INTEGER     NOT NULL,
         H_D_ID      SMALLINT    NOT NULL,
         H_W_ID      INTEGER     NOT NULL,
         H_DATE      TIMESTAMP   NOT NULL,
         H_AMOUNT    DECIMAL(6,2) NOT NULL,
         H_DATA      CHAR(24)    NOT NULL
        )
        IN ts_history_006
        INDEX IN ts_history_006;
ALTER TABLE HISTORY6 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY7;
CREATE TABLE HISTORY7
        (
         H_C_ID      INTEGER     NOT NULL,
         H_C_D_ID    SMALLINT    NOT NULL,
         H_C_W_ID    INTEGER     NOT NULL,
         H_D_ID      SMALLINT    NOT NULL,
         H_W_ID      INTEGER     NOT NULL,
         H_DATE      TIMESTAMP   NOT NULL,
         H_AMOUNT    DECIMAL(6,2) NOT NULL,
```

```
        H_DATA        CHAR(24)    NOT NULL
        )
        IN ts_history_007
        INDEX IN ts_history_007;
ALTER TABLE HISTORY7 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY8;
CREATE TABLE HISTORY8
        (
        H_C_ID        INTEGER     NOT NULL,
        H_C_D_ID      SMALLINT    NOT NULL,
        H_C_W_ID      INTEGER     NOT NULL,
        H_D_ID        SMALLINT    NOT NULL,
        H_W_ID        INTEGER     NOT NULL,
        H_DATE        TIMESTAMP   NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)    NOT NULL
        )
        IN ts_history_008
        INDEX IN ts_history_008;
ALTER TABLE HISTORY8 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY9;
CREATE TABLE HISTORY9
        (
        H_C_ID        INTEGER     NOT NULL,
        H_C_D_ID      SMALLINT    NOT NULL,
        H_C_W_ID      INTEGER     NOT NULL,
        H_D_ID        SMALLINT    NOT NULL,
        H_W_ID        INTEGER     NOT NULL,
        H_DATE        TIMESTAMP   NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)    NOT NULL
        )
        IN ts_history_009
        INDEX IN ts_history_009;
ALTER TABLE HISTORY9 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY10;
CREATE TABLE HISTORY10
        (
        H_C_ID        INTEGER     NOT NULL,
        H_C_D_ID      SMALLINT    NOT NULL,
        H_C_W_ID      INTEGER     NOT NULL,
        H_D_ID        SMALLINT    NOT NULL,
        H_W_ID        INTEGER     NOT NULL,
        H_DATE        TIMESTAMP   NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)    NOT NULL
        )
        IN ts_history_010
        INDEX IN ts_history_010;
ALTER TABLE HISTORY10 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY11;
CREATE TABLE HISTORY11
        (
        H_C_ID        INTEGER     NOT NULL,
        H_C_D_ID      SMALLINT    NOT NULL,
        H_C_W_ID      INTEGER     NOT NULL,
        H_D_ID        SMALLINT    NOT NULL,
        H_W_ID        INTEGER     NOT NULL,
        H_DATE        TIMESTAMP   NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)    NOT NULL
        )
        IN ts_history_011
        INDEX IN ts_history_011;
ALTER TABLE HISTORY11 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY12;
CREATE TABLE HISTORY12
        (
        H_C_ID        INTEGER     NOT NULL,
        H_C_D_ID      SMALLINT    NOT NULL,
        H_C_W_ID      INTEGER     NOT NULL,
        H_D_ID        SMALLINT    NOT NULL,
        H_W_ID        INTEGER     NOT NULL,
        H_DATE        TIMESTAMP   NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)    NOT NULL
        )
        IN ts_history_012
        INDEX IN ts_history_012;
ALTER TABLE HISTORY12 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY13;
CREATE TABLE HISTORY13
        (
        H_C_ID        INTEGER     NOT NULL,
        H_C_D_ID      SMALLINT    NOT NULL,
        H_C_W_ID      INTEGER     NOT NULL,
        H_D_ID        SMALLINT    NOT NULL,
        H_W_ID        INTEGER     NOT NULL,
        H_DATE        TIMESTAMP   NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)    NOT NULL
        )
        IN ts_history_013
        INDEX IN ts_history_013;
ALTER TABLE HISTORY13 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY14;
CREATE TABLE HISTORY14
        (
        H_C_ID        INTEGER     NOT NULL,
        H_C_D_ID      SMALLINT    NOT NULL,
        H_C_W_ID      INTEGER     NOT NULL,
        H_D_ID        SMALLINT    NOT NULL,
        H_W_ID        INTEGER     NOT NULL,
        H_DATE        TIMESTAMP   NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)    NOT NULL
        )
        IN ts_history_014
        INDEX IN ts_history_014;
ALTER TABLE HISTORY14 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY15;
CREATE TABLE HISTORY15
        (
        H_C_ID        INTEGER     NOT NULL,
        H_C_D_ID      SMALLINT    NOT NULL,
        H_C_W_ID      INTEGER     NOT NULL,
        H_D_ID        SMALLINT    NOT NULL,
        H_W_ID        INTEGER     NOT NULL,
        H_DATE        TIMESTAMP   NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)    NOT NULL
        )
        IN ts_history_015
        INDEX IN ts_history_015;
ALTER TABLE HISTORY15 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY16;
CREATE TABLE HISTORY16
        (
        H_C_ID        INTEGER     NOT NULL,
        H_C_D_ID      SMALLINT    NOT NULL,
        H_C_W_ID      INTEGER     NOT NULL,
        H_D_ID        SMALLINT    NOT NULL,
        H_W_ID        INTEGER     NOT NULL,
        H_DATE        TIMESTAMP   NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)    NOT NULL
        )
        IN ts_history_016
        INDEX IN ts_history_016;
ALTER TABLE HISTORY16 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY17;
CREATE TABLE HISTORY17
        (
        H_C_ID        INTEGER     NOT NULL,
        H_C_D_ID      SMALLINT    NOT NULL,
        H_C_W_ID      INTEGER     NOT NULL,
        H_D_ID        SMALLINT    NOT NULL,
        H_W_ID        INTEGER     NOT NULL,
        H_DATE        TIMESTAMP   NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)    NOT NULL
        )
        IN ts_history_017
        INDEX IN ts_history_017;
ALTER TABLE HISTORY17 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY18;
CREATE TABLE HISTORY18
        (
        H_C_ID        INTEGER     NOT NULL,
        H_C_D_ID      SMALLINT    NOT NULL,
        H_C_W_ID      INTEGER     NOT NULL,
        H_D_ID        SMALLINT    NOT NULL,
        H_W_ID        INTEGER     NOT NULL,
        H_DATE        TIMESTAMP   NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)    NOT NULL
        )
        IN ts_history_018
        INDEX IN ts_history_018;
ALTER TABLE HISTORY18 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY19;
CREATE TABLE HISTORY19
        (
        H_C_ID        INTEGER     NOT NULL,
        H_C_D_ID      SMALLINT    NOT NULL,
        H_C_W_ID      INTEGER     NOT NULL,
        H_D_ID        SMALLINT    NOT NULL,
        H_W_ID        INTEGER     NOT NULL,
        H_DATE        TIMESTAMP   NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)    NOT NULL
        )
        IN ts_history_019
        INDEX IN ts_history_019;
ALTER TABLE HISTORY19 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY20;
CREATE TABLE HISTORY20
        (
        H_C_ID        INTEGER     NOT NULL,
        H_C_D_ID      SMALLINT    NOT NULL,
        H_C_W_ID      INTEGER     NOT NULL,
        H_D_ID        SMALLINT    NOT NULL,
        H_W_ID        INTEGER     NOT NULL,
```

```
        H_DATE        TIMESTAMP    NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_history_020
        INDEX IN ts_history_020;
ALTER TABLE HISTORY20 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY21;
CREATE TABLE HISTORY21
        (
        H_C_ID        INTEGER      NOT NULL,
        H_C_D_ID      SMALLINT     NOT NULL,
        H_C_W_ID      INTEGER      NOT NULL,
        H_D_ID        SMALLINT     NOT NULL,
        H_W_ID        INTEGER      NOT NULL,
        H_DATE        TIMESTAMP    NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_history_021
        INDEX IN ts_history_021;
ALTER TABLE HISTORY21 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY22;
CREATE TABLE HISTORY22
        (
        H_C_ID        INTEGER      NOT NULL,
        H_C_D_ID      SMALLINT     NOT NULL,
        H_C_W_ID      INTEGER      NOT NULL,
        H_D_ID        SMALLINT     NOT NULL,
        H_W_ID        INTEGER      NOT NULL,
        H_DATE        TIMESTAMP    NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_history_022
        INDEX IN ts_history_022;
ALTER TABLE HISTORY22 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY23;
CREATE TABLE HISTORY23
        (
        H_C_ID        INTEGER      NOT NULL,
        H_C_D_ID      SMALLINT     NOT NULL,
        H_C_W_ID      INTEGER      NOT NULL,
        H_D_ID        SMALLINT     NOT NULL,
        H_W_ID        INTEGER      NOT NULL,
        H_DATE        TIMESTAMP    NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_history_023
        INDEX IN ts_history_023;
ALTER TABLE HISTORY23 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY24;
CREATE TABLE HISTORY24
        (
        H_C_ID        INTEGER      NOT NULL,
        H_C_D_ID      SMALLINT     NOT NULL,
        H_C_W_ID      INTEGER      NOT NULL,
        H_D_ID        SMALLINT     NOT NULL,
        H_W_ID        INTEGER      NOT NULL,
        H_DATE        TIMESTAMP    NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_history_024
        INDEX IN ts_history_024;
ALTER TABLE HISTORY24 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY25;
CREATE TABLE HISTORY25
        (
        H_C_ID        INTEGER      NOT NULL,
        H_C_D_ID      SMALLINT     NOT NULL,
        H_C_W_ID      INTEGER      NOT NULL,
        H_D_ID        SMALLINT     NOT NULL,
        H_W_ID        INTEGER      NOT NULL,
        H_DATE        TIMESTAMP    NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_history_025
        INDEX IN ts_history_025;
ALTER TABLE HISTORY25 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY26;
CREATE TABLE HISTORY26
        (
        H_C_ID        INTEGER      NOT NULL,
        H_C_D_ID      SMALLINT     NOT NULL,
        H_C_W_ID      INTEGER      NOT NULL,
        H_D_ID        SMALLINT     NOT NULL,
        H_W_ID        INTEGER      NOT NULL,
        H_DATE        TIMESTAMP    NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_history_026
        INDEX IN ts_history_026;
ALTER TABLE HISTORY26 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY27;
CREATE TABLE HISTORY27
        (
        H_C_ID        INTEGER      NOT NULL,
        H_C_D_ID      SMALLINT     NOT NULL,
        H_C_W_ID      INTEGER      NOT NULL,
        H_D_ID        SMALLINT     NOT NULL,
        H_W_ID        INTEGER      NOT NULL,
        H_DATE        TIMESTAMP    NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_history_027
        INDEX IN ts_history_027;
ALTER TABLE HISTORY27 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY28;
CREATE TABLE HISTORY28
        (
        H_C_ID        INTEGER      NOT NULL,
        H_C_D_ID      SMALLINT     NOT NULL,
        H_C_W_ID      INTEGER      NOT NULL,
        H_D_ID        SMALLINT     NOT NULL,
        H_W_ID        INTEGER      NOT NULL,
        H_DATE        TIMESTAMP    NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_history_028
        INDEX IN ts_history_028;
ALTER TABLE HISTORY28 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY29;
CREATE TABLE HISTORY29
        (
        H_C_ID        INTEGER      NOT NULL,
        H_C_D_ID      SMALLINT     NOT NULL,
        H_C_W_ID      INTEGER      NOT NULL,
        H_D_ID        SMALLINT     NOT NULL,
        H_W_ID        INTEGER      NOT NULL,
        H_DATE        TIMESTAMP    NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_history_029
        INDEX IN ts_history_029;
ALTER TABLE HISTORY29 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY30;
CREATE TABLE HISTORY30
        (
        H_C_ID        INTEGER      NOT NULL,
        H_C_D_ID      SMALLINT     NOT NULL,
        H_C_W_ID      INTEGER      NOT NULL,
        H_D_ID        SMALLINT     NOT NULL,
        H_W_ID        INTEGER      NOT NULL,
        H_DATE        TIMESTAMP    NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_history_030
        INDEX IN ts_history_030;
ALTER TABLE HISTORY30 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY31;
CREATE TABLE HISTORY31
        (
        H_C_ID        INTEGER      NOT NULL,
        H_C_D_ID      SMALLINT     NOT NULL,
        H_C_W_ID      INTEGER      NOT NULL,
        H_D_ID        SMALLINT     NOT NULL,
        H_W_ID        INTEGER      NOT NULL,
        H_DATE        TIMESTAMP    NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_history_031
        INDEX IN ts_history_031;
ALTER TABLE HISTORY31 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY32;
CREATE TABLE HISTORY32
        (
        H_C_ID        INTEGER      NOT NULL,
        H_C_D_ID      SMALLINT     NOT NULL,
        H_C_W_ID      INTEGER      NOT NULL,
        H_D_ID        SMALLINT     NOT NULL,
        H_W_ID        INTEGER      NOT NULL,
        H_DATE        TIMESTAMP    NOT NULL,
        H_AMOUNT      DECIMAL(6,2) NOT NULL,
        H_DATA        CHAR(24)     NOT NULL
        )
        IN ts_history_032
        INDEX IN ts_history_032;
ALTER TABLE HISTORY32 APPEND ON;
connect reset;
```

DDL/CRTB_ITEM.ddl

```
connect to TPCC in share mode;
DROP TABLE ITEM;
CREATE TABLE ITEM
        (
         I_NAME        CHAR(24)     NOT NULL,
         I_PRICE       DECIMAL(5,2) NOT NULL,
         I_DATA        VARCHAR(50)  NOT NULL,
         I_IM_ID       INTEGER      NOT NULL,
         I_ID          INTEGER      NOT NULL
        )
        IN ts_item_001
        INDEX IN ts_item_001
        ORGANIZE BY KEY SEQUENCE (
         I_ID STARTING FROM 1 ENDING AT 100000
        )
        ALLOW OVERFLOW;
ALTER TABLE ITEM LOCKSIZE TABLE;
connect reset;
```

DDL/CRTB_NEW_ORDERA.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA1;
CREATE TABLE NEW_ORDERA1
        (
         NO_O_ID       INTEGER    NOT NULL,
         NO_D_ID       SMALLINT   NOT NULL,
         NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordA_001
        INDEX IN ts_newordA_001
        ORGANIZE BY KEY SEQUENCE (
         NO_W_ID STARTING FROM 1 ENDING AT 1624,
         NO_D_ID STARTING FROM 1 ENDING AT 10,
         NO_O_ID STARTING FROM 1900 ENDING AT 3695
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA2;
CREATE TABLE NEW_ORDERA2
        (
         NO_O_ID       INTEGER    NOT NULL,
         NO_D_ID       SMALLINT   NOT NULL,
         NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordA_002
        INDEX IN ts_newordA_002
        ORGANIZE BY KEY SEQUENCE (
         NO_W_ID STARTING FROM 1625 ENDING AT 3248,
         NO_D_ID STARTING FROM 1 ENDING AT 10,
         NO_O_ID STARTING FROM 1900 ENDING AT 3695
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA3;
CREATE TABLE NEW_ORDERA3
        (
         NO_O_ID       INTEGER    NOT NULL,
         NO_D_ID       SMALLINT   NOT NULL,
         NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordA_003
        INDEX IN ts_newordA_003
        ORGANIZE BY KEY SEQUENCE (
         NO_W_ID STARTING FROM 3249 ENDING AT 4872,
         NO_D_ID STARTING FROM 1 ENDING AT 10,
         NO_O_ID STARTING FROM 1900 ENDING AT 3695
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA4;
CREATE TABLE NEW_ORDERA4
        (
         NO_O_ID       INTEGER    NOT NULL,
         NO_D_ID       SMALLINT   NOT NULL,
         NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordA_004
        INDEX IN ts_newordA_004
        ORGANIZE BY KEY SEQUENCE (
         NO_W_ID STARTING FROM 4873 ENDING AT 6496,
         NO_D_ID STARTING FROM 1 ENDING AT 10,
         NO_O_ID STARTING FROM 1900 ENDING AT 3695
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA5;
CREATE TABLE NEW_ORDERA5
        (
         NO_O_ID       INTEGER    NOT NULL,
         NO_D_ID       SMALLINT   NOT NULL,
         NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordA_005
        INDEX IN ts_newordA_005
        ORGANIZE BY KEY SEQUENCE (
         NO_W_ID STARTING FROM 6497 ENDING AT 8120,
         NO_D_ID STARTING FROM 1 ENDING AT 10,
         NO_O_ID STARTING FROM 1900 ENDING AT 3695
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA6;
CREATE TABLE NEW_ORDERA6
        (
         NO_O_ID       INTEGER    NOT NULL,
         NO_D_ID       SMALLINT   NOT NULL,
         NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordA_006
        INDEX IN ts_newordA_006
        ORGANIZE BY KEY SEQUENCE (
         NO_W_ID STARTING FROM 8121 ENDING AT 9744,
         NO_D_ID STARTING FROM 1 ENDING AT 10,
         NO_O_ID STARTING FROM 1900 ENDING AT 3695
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA7;
CREATE TABLE NEW_ORDERA7
        (
         NO_O_ID       INTEGER    NOT NULL,
         NO_D_ID       SMALLINT   NOT NULL,
         NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordA_007
        INDEX IN ts_newordA_007
        ORGANIZE BY KEY SEQUENCE (
         NO_W_ID STARTING FROM 9745 ENDING AT 11368,
         NO_D_ID STARTING FROM 1 ENDING AT 10,
         NO_O_ID STARTING FROM 1900 ENDING AT 3695
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA8;
CREATE TABLE NEW_ORDERA8
        (
         NO_O_ID       INTEGER    NOT NULL,
         NO_D_ID       SMALLINT   NOT NULL,
         NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordA_008
        INDEX IN ts_newordA_008
        ORGANIZE BY KEY SEQUENCE (
         NO_W_ID STARTING FROM 11369 ENDING AT 12992,
         NO_D_ID STARTING FROM 1 ENDING AT 10,
         NO_O_ID STARTING FROM 1900 ENDING AT 3695
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA9;
CREATE TABLE NEW_ORDERA9
        (
         NO_O_ID       INTEGER    NOT NULL,
         NO_D_ID       SMALLINT   NOT NULL,
         NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordA_009
        INDEX IN ts_newordA_009
        ORGANIZE BY KEY SEQUENCE (
         NO_W_ID STARTING FROM 12993 ENDING AT 14616,
         NO_D_ID STARTING FROM 1 ENDING AT 10,
         NO_O_ID STARTING FROM 1900 ENDING AT 3695
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA10;
CREATE TABLE NEW_ORDERA10
        (
         NO_O_ID       INTEGER    NOT NULL,
         NO_D_ID       SMALLINT   NOT NULL,
         NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordA_010
        INDEX IN ts_newordA_010
        ORGANIZE BY KEY SEQUENCE (
         NO_W_ID STARTING FROM 14617 ENDING AT 16240,
         NO_D_ID STARTING FROM 1 ENDING AT 10,
         NO_O_ID STARTING FROM 1900 ENDING AT 3695
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA11;
CREATE TABLE NEW_ORDERA11
        (
         NO_O_ID       INTEGER    NOT NULL,
         NO_D_ID       SMALLINT   NOT NULL,
         NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordA_011
        INDEX IN ts_newordA_011
        ORGANIZE BY KEY SEQUENCE (
         NO_W_ID STARTING FROM 16241 ENDING AT 17864,
         NO_D_ID STARTING FROM 1 ENDING AT 10,
         NO_O_ID STARTING FROM 1900 ENDING AT 3695
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA12;
CREATE TABLE NEW_ORDERA12
        (
         NO_O_ID       INTEGER    NOT NULL,
         NO_D_ID       SMALLINT   NOT NULL,
         NO_W_ID       INTEGER    NOT NULL
```

```
)
IN ts_newordA_012
INDEX IN ts_newordA_012
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 17865 ENDING AT 19488,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3695
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA13;
CREATE TABLE NEW_ORDERA13
(
NO_O_ID      INTEGER    NOT NULL,
NO_D_ID      SMALLINT   NOT NULL,
NO_W_ID      INTEGER    NOT NULL
)
IN ts_newordA_013
INDEX IN ts_newordA_013
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 19489 ENDING AT 21112,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3695
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA14;
CREATE TABLE NEW_ORDERA14
(
NO_O_ID      INTEGER    NOT NULL,
NO_D_ID      SMALLINT   NOT NULL,
NO_W_ID      INTEGER    NOT NULL
)
IN ts_newordA_014
INDEX IN ts_newordA_014
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 21113 ENDING AT 22736,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3695
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA15;
CREATE TABLE NEW_ORDERA15
(
NO_O_ID      INTEGER    NOT NULL,
NO_D_ID      SMALLINT   NOT NULL,
NO_W_ID      INTEGER    NOT NULL
)
IN ts_newordA_015
INDEX IN ts_newordA_015
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 22737 ENDING AT 24360,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3695
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA16;
CREATE TABLE NEW_ORDERA16
(
NO_O_ID      INTEGER    NOT NULL,
NO_D_ID      SMALLINT   NOT NULL,
NO_W_ID      INTEGER    NOT NULL
)
IN ts_newordA_016
INDEX IN ts_newordA_016
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 24361 ENDING AT 25984,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3695
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA17;
CREATE TABLE NEW_ORDERA17
(
NO_O_ID      INTEGER    NOT NULL,
NO_D_ID      SMALLINT   NOT NULL,
NO_W_ID      INTEGER    NOT NULL
)
IN ts_newordA_017
INDEX IN ts_newordA_017
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 25985 ENDING AT 27608,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3695
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA18;
CREATE TABLE NEW_ORDERA18
(
NO_O_ID      INTEGER    NOT NULL,
NO_D_ID      SMALLINT   NOT NULL,
NO_W_ID      INTEGER    NOT NULL
)
IN ts_newordA_018
INDEX IN ts_newordA_018
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 27609 ENDING AT 29232,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3695
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA19;
CREATE TABLE NEW_ORDERA19
(
NO_O_ID      INTEGER    NOT NULL,
NO_D_ID      SMALLINT   NOT NULL,
NO_W_ID      INTEGER    NOT NULL
)
IN ts_newordA_019
INDEX IN ts_newordA_019
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 29233 ENDING AT 30856,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3695
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA20;
CREATE TABLE NEW_ORDERA20
(
NO_O_ID      INTEGER    NOT NULL,
NO_D_ID      SMALLINT   NOT NULL,
NO_W_ID      INTEGER    NOT NULL
)
IN ts_newordA_020
INDEX IN ts_newordA_020
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 30857 ENDING AT 32480,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3695
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA21;
CREATE TABLE NEW_ORDERA21
(
NO_O_ID      INTEGER    NOT NULL,
NO_D_ID      SMALLINT   NOT NULL,
NO_W_ID      INTEGER    NOT NULL
)
IN ts_newordA_021
INDEX IN ts_newordA_021
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 32481 ENDING AT 34104,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3695
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA22;
CREATE TABLE NEW_ORDERA22
(
NO_O_ID      INTEGER    NOT NULL,
NO_D_ID      SMALLINT   NOT NULL,
NO_W_ID      INTEGER    NOT NULL
)
IN ts_newordA_022
INDEX IN ts_newordA_022
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 34105 ENDING AT 35728,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3695
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA23;
CREATE TABLE NEW_ORDERA23
(
NO_O_ID      INTEGER    NOT NULL,
NO_D_ID      SMALLINT   NOT NULL,
NO_W_ID      INTEGER    NOT NULL
)
IN ts_newordA_023
INDEX IN ts_newordA_023
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 35729 ENDING AT 37352,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3695
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA24;
CREATE TABLE NEW_ORDERA24
(
NO_O_ID      INTEGER    NOT NULL,
NO_D_ID      SMALLINT   NOT NULL,
NO_W_ID      INTEGER    NOT NULL
)
IN ts_newordA_024
INDEX IN ts_newordA_024
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 37353 ENDING AT 38976,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3695
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA25;
CREATE TABLE NEW_ORDERA25
(
NO_O_ID      INTEGER    NOT NULL,
```

```
        NO_D_ID       SMALLINT   NOT NULL,
        NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordA_025
        INDEX IN ts_newordA_025
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 38977 ENDING AT 40600,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 1900 ENDING AT 3695
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA26;
CREATE TABLE NEW_ORDERA26
        (
        NO_O_ID       INTEGER    NOT NULL,
        NO_D_ID       SMALLINT   NOT NULL,
        NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordA_026
        INDEX IN ts_newordA_026
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 40601 ENDING AT 42224,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 1900 ENDING AT 3695
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA27;
CREATE TABLE NEW_ORDERA27
        (
        NO_O_ID       INTEGER    NOT NULL,
        NO_D_ID       SMALLINT   NOT NULL,
        NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordA_027
        INDEX IN ts_newordA_027
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 42225 ENDING AT 43848,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 1900 ENDING AT 3695
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA28;
CREATE TABLE NEW_ORDERA28
        (
        NO_O_ID       INTEGER    NOT NULL,
        NO_D_ID       SMALLINT   NOT NULL,
        NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordA_028
        INDEX IN ts_newordA_028
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 43849 ENDING AT 45472,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 1900 ENDING AT 3695
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA29;
CREATE TABLE NEW_ORDERA29
        (
        NO_O_ID       INTEGER    NOT NULL,
        NO_D_ID       SMALLINT   NOT NULL,
        NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordA_029
        INDEX IN ts_newordA_029
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 45473 ENDING AT 47096,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 1900 ENDING AT 3695
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA30;
CREATE TABLE NEW_ORDERA30
        (
        NO_O_ID       INTEGER    NOT NULL,
        NO_D_ID       SMALLINT   NOT NULL,
        NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordA_030
        INDEX IN ts_newordA_030
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 47097 ENDING AT 48720,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 1900 ENDING AT 3695
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA31;
CREATE TABLE NEW_ORDERA31
        (
        NO_O_ID       INTEGER    NOT NULL,
        NO_D_ID       SMALLINT   NOT NULL,
        NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordA_031
        INDEX IN ts_newordA_031
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 48721 ENDING AT 50344,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 1900 ENDING AT 3695
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA32;
CREATE TABLE NEW_ORDERA32
        (
        NO_O_ID       INTEGER    NOT NULL,
        NO_D_ID       SMALLINT   NOT NULL,
        NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordA_032
        INDEX IN ts_newordA_032
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 50345 ENDING AT 51968,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 1900 ENDING AT 3695
        )
        ALLOW OVERFLOW;
connect reset;
```

DDL/CRTB_NEW_ORDERB.ddl

```
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB1;
CREATE TABLE NEW_ORDERB1
        (
        NO_O_ID       INTEGER    NOT NULL,
        NO_D_ID       SMALLINT   NOT NULL,
        NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordB_001
        INDEX IN ts_newordB_001
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 1 ENDING AT 1624,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 3696 ENDING AT 5491
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB2;
CREATE TABLE NEW_ORDERB2
        (
        NO_O_ID       INTEGER    NOT NULL,
        NO_D_ID       SMALLINT   NOT NULL,
        NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordB_002
        INDEX IN ts_newordB_002
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 1625 ENDING AT 3248,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 3696 ENDING AT 5491
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB3;
CREATE TABLE NEW_ORDERB3
        (
        NO_O_ID       INTEGER    NOT NULL,
        NO_D_ID       SMALLINT   NOT NULL,
        NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordB_003
        INDEX IN ts_newordB_003
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 3249 ENDING AT 4872,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 3696 ENDING AT 5491
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB4;
CREATE TABLE NEW_ORDERB4
        (
        NO_O_ID       INTEGER    NOT NULL,
        NO_D_ID       SMALLINT   NOT NULL,
        NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordB_004
        INDEX IN ts_newordB_004
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 4873 ENDING AT 6496,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 3696 ENDING AT 5491
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB5;
CREATE TABLE NEW_ORDERB5
        (
        NO_O_ID       INTEGER    NOT NULL,
        NO_D_ID       SMALLINT   NOT NULL,
        NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordB_005
        INDEX IN ts_newordB_005
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 6497 ENDING AT 8120,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
```

```
 NO_O_ID STARTING FROM 3696 ENDING AT 5491
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB6;
CREATE TABLE NEW_ORDERB6
(
 NO_O_ID      INTEGER    NOT NULL,
 NO_D_ID      SMALLINT   NOT NULL,
 NO_W_ID      INTEGER    NOT NULL
)
IN ts_newordB_006
INDEX IN ts_newordB_006
ORGANIZE BY KEY SEQUENCE (
 NO_W_ID STARTING FROM 8121 ENDING AT 9744,
 NO_D_ID STARTING FROM 1 ENDING AT 10,
 NO_O_ID STARTING FROM 3696 ENDING AT 5491
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB7;
CREATE TABLE NEW_ORDERB7
(
 NO_O_ID      INTEGER    NOT NULL,
 NO_D_ID      SMALLINT   NOT NULL,
 NO_W_ID      INTEGER    NOT NULL
)
IN ts_newordB_007
INDEX IN ts_newordB_007
ORGANIZE BY KEY SEQUENCE (
 NO_W_ID STARTING FROM 9745 ENDING AT 11368,
 NO_D_ID STARTING FROM 1 ENDING AT 10,
 NO_O_ID STARTING FROM 3696 ENDING AT 5491
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB8;
CREATE TABLE NEW_ORDERB8
(
 NO_O_ID      INTEGER    NOT NULL,
 NO_D_ID      SMALLINT   NOT NULL,
 NO_W_ID      INTEGER    NOT NULL
)
IN ts_newordB_008
INDEX IN ts_newordB_008
ORGANIZE BY KEY SEQUENCE (
 NO_W_ID STARTING FROM 11369 ENDING AT 12992,
 NO_D_ID STARTING FROM 1 ENDING AT 10,
 NO_O_ID STARTING FROM 3696 ENDING AT 5491
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB9;
CREATE TABLE NEW_ORDERB9
(
 NO_O_ID      INTEGER    NOT NULL,
 NO_D_ID      SMALLINT   NOT NULL,
 NO_W_ID      INTEGER    NOT NULL
)
IN ts_newordB_009
INDEX IN ts_newordB_009
ORGANIZE BY KEY SEQUENCE (
 NO_W_ID STARTING FROM 12993 ENDING AT 14616,
 NO_D_ID STARTING FROM 1 ENDING AT 10,
 NO_O_ID STARTING FROM 3696 ENDING AT 5491
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB10;
CREATE TABLE NEW_ORDERB10
(
 NO_O_ID      INTEGER    NOT NULL,
 NO_D_ID      SMALLINT   NOT NULL,
 NO_W_ID      INTEGER    NOT NULL
)
IN ts_newordB_010
INDEX IN ts_newordB_010
ORGANIZE BY KEY SEQUENCE (
 NO_W_ID STARTING FROM 14617 ENDING AT 16240,
 NO_D_ID STARTING FROM 1 ENDING AT 10,
 NO_O_ID STARTING FROM 3696 ENDING AT 5491
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB11;
CREATE TABLE NEW_ORDERB11
(
 NO_O_ID      INTEGER    NOT NULL,
 NO_D_ID      SMALLINT   NOT NULL,
 NO_W_ID      INTEGER    NOT NULL
)
IN ts_newordB_011
INDEX IN ts_newordB_011
ORGANIZE BY KEY SEQUENCE (
 NO_W_ID STARTING FROM 16241 ENDING AT 17864,
 NO_D_ID STARTING FROM 1 ENDING AT 10,
 NO_O_ID STARTING FROM 3696 ENDING AT 5491
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB12;
CREATE TABLE NEW_ORDERB12
(
 NO_O_ID      INTEGER    NOT NULL,
 NO_D_ID      SMALLINT   NOT NULL,
 NO_W_ID      INTEGER    NOT NULL
)
IN ts_newordB_012
INDEX IN ts_newordB_012
ORGANIZE BY KEY SEQUENCE (
 NO_W_ID STARTING FROM 17865 ENDING AT 19488,
 NO_D_ID STARTING FROM 1 ENDING AT 10,
 NO_O_ID STARTING FROM 3696 ENDING AT 5491
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB13;
CREATE TABLE NEW_ORDERB13
(
 NO_O_ID      INTEGER    NOT NULL,
 NO_D_ID      SMALLINT   NOT NULL,
 NO_W_ID      INTEGER    NOT NULL
)
IN ts_newordB_013
INDEX IN ts_newordB_013
ORGANIZE BY KEY SEQUENCE (
 NO_W_ID STARTING FROM 19489 ENDING AT 21112,
 NO_D_ID STARTING FROM 1 ENDING AT 10,
 NO_O_ID STARTING FROM 3696 ENDING AT 5491
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB14;
CREATE TABLE NEW_ORDERB14
(
 NO_O_ID      INTEGER    NOT NULL,
 NO_D_ID      SMALLINT   NOT NULL,
 NO_W_ID      INTEGER    NOT NULL
)
IN ts_newordB_014
INDEX IN ts_newordB_014
ORGANIZE BY KEY SEQUENCE (
 NO_W_ID STARTING FROM 21113 ENDING AT 22736,
 NO_D_ID STARTING FROM 1 ENDING AT 10,
 NO_O_ID STARTING FROM 3696 ENDING AT 5491
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB15;
CREATE TABLE NEW_ORDERB15
(
 NO_O_ID      INTEGER    NOT NULL,
 NO_D_ID      SMALLINT   NOT NULL,
 NO_W_ID      INTEGER    NOT NULL
)
IN ts_newordB_015
INDEX IN ts_newordB_015
ORGANIZE BY KEY SEQUENCE (
 NO_W_ID STARTING FROM 22737 ENDING AT 24360,
 NO_D_ID STARTING FROM 1 ENDING AT 10,
 NO_O_ID STARTING FROM 3696 ENDING AT 5491
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB16;
CREATE TABLE NEW_ORDERB16
(
 NO_O_ID      INTEGER    NOT NULL,
 NO_D_ID      SMALLINT   NOT NULL,
 NO_W_ID      INTEGER    NOT NULL
)
IN ts_newordB_016
INDEX IN ts_newordB_016
ORGANIZE BY KEY SEQUENCE (
 NO_W_ID STARTING FROM 24361 ENDING AT 25984,
 NO_D_ID STARTING FROM 1 ENDING AT 10,
 NO_O_ID STARTING FROM 3696 ENDING AT 5491
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB17;
CREATE TABLE NEW_ORDERB17
(
 NO_O_ID      INTEGER    NOT NULL,
 NO_D_ID      SMALLINT   NOT NULL,
 NO_W_ID      INTEGER    NOT NULL
)
IN ts_newordB_017
INDEX IN ts_newordB_017
ORGANIZE BY KEY SEQUENCE (
 NO_W_ID STARTING FROM 25985 ENDING AT 27608,
 NO_D_ID STARTING FROM 1 ENDING AT 10,
 NO_O_ID STARTING FROM 3696 ENDING AT 5491
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB18;
CREATE TABLE NEW_ORDERB18
(
 NO_O_ID      INTEGER    NOT NULL,
 NO_D_ID      SMALLINT   NOT NULL,
 NO_W_ID      INTEGER    NOT NULL
)
IN ts_newordB_018
INDEX IN ts_newordB_018
ORGANIZE BY KEY SEQUENCE (
```

```
        NO_W_ID STARTING FROM 27609 ENDING AT 29232,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 3696 ENDING AT 5491
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB19;
CREATE TABLE NEW_ORDERB19
        (
        NO_O_ID       INTEGER    NOT NULL,
        NO_D_ID       SMALLINT   NOT NULL,
        NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordB_019
        INDEX IN ts_newordB_019
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 29233 ENDING AT 30856,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 3696 ENDING AT 5491
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB20;
CREATE TABLE NEW_ORDERB20
        (
        NO_O_ID       INTEGER    NOT NULL,
        NO_D_ID       SMALLINT   NOT NULL,
        NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordB_020
        INDEX IN ts_newordB_020
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 30857 ENDING AT 32480,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 3696 ENDING AT 5491
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB21;
CREATE TABLE NEW_ORDERB21
        (
        NO_O_ID       INTEGER    NOT NULL,
        NO_D_ID       SMALLINT   NOT NULL,
        NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordB_021
        INDEX IN ts_newordB_021
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 32481 ENDING AT 34104,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 3696 ENDING AT 5491
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB22;
CREATE TABLE NEW_ORDERB22
        (
        NO_O_ID       INTEGER    NOT NULL,
        NO_D_ID       SMALLINT   NOT NULL,
        NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordB_022
        INDEX IN ts_newordB_022
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 34105 ENDING AT 35728,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 3696 ENDING AT 5491
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB23;
CREATE TABLE NEW_ORDERB23
        (
        NO_O_ID       INTEGER    NOT NULL,
        NO_D_ID       SMALLINT   NOT NULL,
        NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordB_023
        INDEX IN ts_newordB_023
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 35729 ENDING AT 37352,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 3696 ENDING AT 5491
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB24;
CREATE TABLE NEW_ORDERB24
        (
        NO_O_ID       INTEGER    NOT NULL,
        NO_D_ID       SMALLINT   NOT NULL,
        NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordB_024
        INDEX IN ts_newordB_024
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 37353 ENDING AT 38976,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 3696 ENDING AT 5491
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB25;
CREATE TABLE NEW_ORDERB25
        (
        NO_O_ID       INTEGER    NOT NULL,
        NO_D_ID       SMALLINT   NOT NULL,
        NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordB_025
        INDEX IN ts_newordB_025
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 38977 ENDING AT 40600,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 3696 ENDING AT 5491
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB26;
CREATE TABLE NEW_ORDERB26
        (
        NO_O_ID       INTEGER    NOT NULL,
        NO_D_ID       SMALLINT   NOT NULL,
        NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordB_026
        INDEX IN ts_newordB_026
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 40601 ENDING AT 42224,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 3696 ENDING AT 5491
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB27;
CREATE TABLE NEW_ORDERB27
        (
        NO_O_ID       INTEGER    NOT NULL,
        NO_D_ID       SMALLINT   NOT NULL,
        NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordB_027
        INDEX IN ts_newordB_027
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 42225 ENDING AT 43848,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 3696 ENDING AT 5491
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB28;
CREATE TABLE NEW_ORDERB28
        (
        NO_O_ID       INTEGER    NOT NULL,
        NO_D_ID       SMALLINT   NOT NULL,
        NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordB_028
        INDEX IN ts_newordB_028
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 43849 ENDING AT 45472,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 3696 ENDING AT 5491
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB29;
CREATE TABLE NEW_ORDERB29
        (
        NO_O_ID       INTEGER    NOT NULL,
        NO_D_ID       SMALLINT   NOT NULL,
        NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordB_029
        INDEX IN ts_newordB_029
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 45473 ENDING AT 47096,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 3696 ENDING AT 5491
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB30;
CREATE TABLE NEW_ORDERB30
        (
        NO_O_ID       INTEGER    NOT NULL,
        NO_D_ID       SMALLINT   NOT NULL,
        NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordB_030
        INDEX IN ts_newordB_030
        ORGANIZE BY KEY SEQUENCE (
        NO_W_ID STARTING FROM 47097 ENDING AT 48720,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 3696 ENDING AT 5491
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB31;
CREATE TABLE NEW_ORDERB31
        (
        NO_O_ID       INTEGER    NOT NULL,
        NO_D_ID       SMALLINT   NOT NULL,
        NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordB_031
```

```
        INDEX IN ts_newordB_031
        ORGANIZE BY KEY SEQUENCE (
         NO_W_ID STARTING FROM 48721 ENDING AT 50344,
         NO_D_ID STARTING FROM 1 ENDING AT 10,
         NO_O_ID STARTING FROM 3696 ENDING AT 5491
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB32;
CREATE TABLE NEW_ORDERB32
        (
         NO_O_ID       INTEGER    NOT NULL,
         NO_D_ID       SMALLINT   NOT NULL,
         NO_W_ID       INTEGER    NOT NULL
        )
        IN ts_newordB_032
        INDEX IN ts_newordB_032
        ORGANIZE BY KEY SEQUENCE (
         NO_W_ID STARTING FROM 50345 ENDING AT 51968,
         NO_D_ID STARTING FROM 1 ENDING AT 10,
         NO_O_ID STARTING FROM 3696 ENDING AT 5491
        )
        ALLOW OVERFLOW;
connect reset;
```

DDL/CRTB_ORDERS.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDERS1;
CREATE TABLE ORDERS1
        (
         O_C_ID        INTEGER    NOT NULL,
         O_ENTRY_D     TIMESTAMP  NOT NULL,
         O_CARRIER_ID  SMALLINT   NOT NULL,
         O_OL_CNT      SMALLINT   NOT NULL,
         O_ALL_LOCAL   SMALLINT   NOT NULL,
         O_ID          INTEGER    NOT NULL,
         O_W_ID        INTEGER    NOT NULL,
         O_D_ID        SMALLINT   NOT NULL
        )
        IN ts_order_001
        INDEX IN is_order_001
        ORGANIZE BY KEY SEQUENCE (
         O_ID STARTING FROM 1 ENDING AT 3695,
         O_W_ID STARTING FROM 1 ENDING AT 812,
         O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS2;
CREATE TABLE ORDERS2
        (
         O_C_ID        INTEGER    NOT NULL,
         O_ENTRY_D     TIMESTAMP  NOT NULL,
         O_CARRIER_ID  SMALLINT   NOT NULL,
         O_OL_CNT      SMALLINT   NOT NULL,
         O_ALL_LOCAL   SMALLINT   NOT NULL,
         O_ID          INTEGER    NOT NULL,
         O_W_ID        INTEGER    NOT NULL,
         O_D_ID        SMALLINT   NOT NULL
        )
        IN ts_order_002
        INDEX IN is_order_002
        ORGANIZE BY KEY SEQUENCE (
         O_ID STARTING FROM 1 ENDING AT 3695,
         O_W_ID STARTING FROM 813 ENDING AT 1624,
         O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS3;
CREATE TABLE ORDERS3
        (
         O_C_ID        INTEGER    NOT NULL,
         O_ENTRY_D     TIMESTAMP  NOT NULL,
         O_CARRIER_ID  SMALLINT   NOT NULL,
         O_OL_CNT      SMALLINT   NOT NULL,
         O_ALL_LOCAL   SMALLINT   NOT NULL,
         O_ID          INTEGER    NOT NULL,
         O_W_ID        INTEGER    NOT NULL,
         O_D_ID        SMALLINT   NOT NULL
        )
        IN ts_order_003
        INDEX IN is_order_003
        ORGANIZE BY KEY SEQUENCE (
         O_ID STARTING FROM 1 ENDING AT 3695,
         O_W_ID STARTING FROM 1625 ENDING AT 2436,
         O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS4;
CREATE TABLE ORDERS4
        (
         O_C_ID        INTEGER    NOT NULL,
         O_ENTRY_D     TIMESTAMP  NOT NULL,
         O_CARRIER_ID  SMALLINT   NOT NULL,
         O_OL_CNT      SMALLINT   NOT NULL,
         O_ALL_LOCAL   SMALLINT   NOT NULL,
         O_ID          INTEGER    NOT NULL,
         O_W_ID        INTEGER    NOT NULL,
         O_D_ID        SMALLINT   NOT NULL
        )
        IN ts_order_004
        INDEX IN is_order_004
        ORGANIZE BY KEY SEQUENCE (
         O_ID STARTING FROM 1 ENDING AT 3695,
         O_W_ID STARTING FROM 2437 ENDING AT 3248,
         O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS5;
CREATE TABLE ORDERS5
        (
         O_C_ID        INTEGER    NOT NULL,
         O_ENTRY_D     TIMESTAMP  NOT NULL,
         O_CARRIER_ID  SMALLINT   NOT NULL,
         O_OL_CNT      SMALLINT   NOT NULL,
         O_ALL_LOCAL   SMALLINT   NOT NULL,
         O_ID          INTEGER    NOT NULL,
         O_W_ID        INTEGER    NOT NULL,
         O_D_ID        SMALLINT   NOT NULL
        )
        IN ts_order_005
        INDEX IN is_order_005
        ORGANIZE BY KEY SEQUENCE (
         O_ID STARTING FROM 1 ENDING AT 3695,
         O_W_ID STARTING FROM 3249 ENDING AT 4060,
         O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS6;
CREATE TABLE ORDERS6
        (
         O_C_ID        INTEGER    NOT NULL,
         O_ENTRY_D     TIMESTAMP  NOT NULL,
         O_CARRIER_ID  SMALLINT   NOT NULL,
         O_OL_CNT      SMALLINT   NOT NULL,
         O_ALL_LOCAL   SMALLINT   NOT NULL,
         O_ID          INTEGER    NOT NULL,
         O_W_ID        INTEGER    NOT NULL,
         O_D_ID        SMALLINT   NOT NULL
        )
        IN ts_order_006
        INDEX IN is_order_006
        ORGANIZE BY KEY SEQUENCE (
         O_ID STARTING FROM 1 ENDING AT 3695,
         O_W_ID STARTING FROM 4061 ENDING AT 4872,
         O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS7;
CREATE TABLE ORDERS7
        (
         O_C_ID        INTEGER    NOT NULL,
         O_ENTRY_D     TIMESTAMP  NOT NULL,
         O_CARRIER_ID  SMALLINT   NOT NULL,
         O_OL_CNT      SMALLINT   NOT NULL,
         O_ALL_LOCAL   SMALLINT   NOT NULL,
         O_ID          INTEGER    NOT NULL,
         O_W_ID        INTEGER    NOT NULL,
         O_D_ID        SMALLINT   NOT NULL
        )
        IN ts_order_007
        INDEX IN is_order_007
        ORGANIZE BY KEY SEQUENCE (
         O_ID STARTING FROM 1 ENDING AT 3695,
         O_W_ID STARTING FROM 4873 ENDING AT 5684,
         O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS8;
CREATE TABLE ORDERS8
        (
         O_C_ID        INTEGER    NOT NULL,
         O_ENTRY_D     TIMESTAMP  NOT NULL,
         O_CARRIER_ID  SMALLINT   NOT NULL,
         O_OL_CNT      SMALLINT   NOT NULL,
         O_ALL_LOCAL   SMALLINT   NOT NULL,
         O_ID          INTEGER    NOT NULL,
         O_W_ID        INTEGER    NOT NULL,
         O_D_ID        SMALLINT   NOT NULL
        )
        IN ts_order_008
        INDEX IN is_order_008
        ORGANIZE BY KEY SEQUENCE (
         O_ID STARTING FROM 1 ENDING AT 3695,
         O_W_ID STARTING FROM 5685 ENDING AT 6496,
         O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS9;
CREATE TABLE ORDERS9
        (
         O_C_ID        INTEGER    NOT NULL,
         O_ENTRY_D     TIMESTAMP  NOT NULL,
         O_CARRIER_ID  SMALLINT   NOT NULL,
         O_OL_CNT      SMALLINT   NOT NULL,
         O_ALL_LOCAL   SMALLINT   NOT NULL,
         O_ID          INTEGER    NOT NULL,
         O_W_ID        INTEGER    NOT NULL,
```

```
        O_D_ID        SMALLINT    NOT NULL
        )
        IN ts_order_009
        INDEX IN is_order_009
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 6497 ENDING AT 7308,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS10;
CREATE TABLE ORDERS10
        (
        O_C_ID        INTEGER    NOT NULL,
        O_ENTRY_D     TIMESTAMP  NOT NULL,
        O_CARRIER_ID  SMALLINT   NOT NULL,
        O_OL_CNT      SMALLINT   NOT NULL,
        O_ALL_LOCAL   SMALLINT   NOT NULL,
        O_ID          INTEGER    NOT NULL,
        O_W_ID        INTEGER    NOT NULL,
        O_D_ID        SMALLINT   NOT NULL
        )
        IN ts_order_010
        INDEX IN is_order_010
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 7309 ENDING AT 8120,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS11;
CREATE TABLE ORDERS11
        (
        O_C_ID        INTEGER    NOT NULL,
        O_ENTRY_D     TIMESTAMP  NOT NULL,
        O_CARRIER_ID  SMALLINT   NOT NULL,
        O_OL_CNT      SMALLINT   NOT NULL,
        O_ALL_LOCAL   SMALLINT   NOT NULL,
        O_ID          INTEGER    NOT NULL,
        O_W_ID        INTEGER    NOT NULL,
        O_D_ID        SMALLINT   NOT NULL
        )
        IN ts_order_011
        INDEX IN is_order_011
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 8121 ENDING AT 8932,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS12;
CREATE TABLE ORDERS12
        (
        O_C_ID        INTEGER    NOT NULL,
        O_ENTRY_D     TIMESTAMP  NOT NULL,
        O_CARRIER_ID  SMALLINT   NOT NULL,
        O_OL_CNT      SMALLINT   NOT NULL,
        O_ALL_LOCAL   SMALLINT   NOT NULL,
        O_ID          INTEGER    NOT NULL,
        O_W_ID        INTEGER    NOT NULL,
        O_D_ID        SMALLINT   NOT NULL
        )
        IN ts_order_012
        INDEX IN is_order_012
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 8933 ENDING AT 9744,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS13;
CREATE TABLE ORDERS13
        (
        O_C_ID        INTEGER    NOT NULL,
        O_ENTRY_D     TIMESTAMP  NOT NULL,
        O_CARRIER_ID  SMALLINT   NOT NULL,
        O_OL_CNT      SMALLINT   NOT NULL,
        O_ALL_LOCAL   SMALLINT   NOT NULL,
        O_ID          INTEGER    NOT NULL,
        O_W_ID        INTEGER    NOT NULL,
        O_D_ID        SMALLINT   NOT NULL
        )
        IN ts_order_013
        INDEX IN is_order_013
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 9745 ENDING AT 10556,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS14;
CREATE TABLE ORDERS14
        (
        O_C_ID        INTEGER    NOT NULL,
        O_ENTRY_D     TIMESTAMP  NOT NULL,
        O_CARRIER_ID  SMALLINT   NOT NULL,
        O_OL_CNT      SMALLINT   NOT NULL,
        O_ALL_LOCAL   SMALLINT   NOT NULL,
        O_ID          INTEGER    NOT NULL,
        O_W_ID        INTEGER    NOT NULL,
        O_D_ID        SMALLINT   NOT NULL
        )
        IN ts_order_014
        INDEX IN is_order_014
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 10557 ENDING AT 11368,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS15;
CREATE TABLE ORDERS15
        (
        O_C_ID        INTEGER    NOT NULL,
        O_ENTRY_D     TIMESTAMP  NOT NULL,
        O_CARRIER_ID  SMALLINT   NOT NULL,
        O_OL_CNT      SMALLINT   NOT NULL,
        O_ALL_LOCAL   SMALLINT   NOT NULL,
        O_ID          INTEGER    NOT NULL,
        O_W_ID        INTEGER    NOT NULL,
        O_D_ID        SMALLINT   NOT NULL
        )
        IN ts_order_015
        INDEX IN is_order_015
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 11369 ENDING AT 12180,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS16;
CREATE TABLE ORDERS16
        (
        O_C_ID        INTEGER    NOT NULL,
        O_ENTRY_D     TIMESTAMP  NOT NULL,
        O_CARRIER_ID  SMALLINT   NOT NULL,
        O_OL_CNT      SMALLINT   NOT NULL,
        O_ALL_LOCAL   SMALLINT   NOT NULL,
        O_ID          INTEGER    NOT NULL,
        O_W_ID        INTEGER    NOT NULL,
        O_D_ID        SMALLINT   NOT NULL
        )
        IN ts_order_016
        INDEX IN is_order_016
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 12181 ENDING AT 12992,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS17;
CREATE TABLE ORDERS17
        (
        O_C_ID        INTEGER    NOT NULL,
        O_ENTRY_D     TIMESTAMP  NOT NULL,
        O_CARRIER_ID  SMALLINT   NOT NULL,
        O_OL_CNT      SMALLINT   NOT NULL,
        O_ALL_LOCAL   SMALLINT   NOT NULL,
        O_ID          INTEGER    NOT NULL,
        O_W_ID        INTEGER    NOT NULL,
        O_D_ID        SMALLINT   NOT NULL
        )
        IN ts_order_017
        INDEX IN is_order_017
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 12993 ENDING AT 13804,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS18;
CREATE TABLE ORDERS18
        (
        O_C_ID        INTEGER    NOT NULL,
        O_ENTRY_D     TIMESTAMP  NOT NULL,
        O_CARRIER_ID  SMALLINT   NOT NULL,
        O_OL_CNT      SMALLINT   NOT NULL,
        O_ALL_LOCAL   SMALLINT   NOT NULL,
        O_ID          INTEGER    NOT NULL,
        O_W_ID        INTEGER    NOT NULL,
        O_D_ID        SMALLINT   NOT NULL
        )
        IN ts_order_018
        INDEX IN is_order_018
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 13805 ENDING AT 14616,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS19;
CREATE TABLE ORDERS19
        (
        O_C_ID        INTEGER    NOT NULL,
        O_ENTRY_D     TIMESTAMP  NOT NULL,
        O_CARRIER_ID  SMALLINT   NOT NULL,
        O_OL_CNT      SMALLINT   NOT NULL,
        O_ALL_LOCAL   SMALLINT   NOT NULL,
        O_ID          INTEGER    NOT NULL,
```

```
O_W_ID        INTEGER    NOT NULL,
O_D_ID        SMALLINT   NOT NULL
)
IN ts_order_019
INDEX IN is_order_019
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 1 ENDING AT 3695,
O_W_ID STARTING FROM 14617 ENDING AT 15428,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS20;
CREATE TABLE ORDERS20
(
O_C_ID        INTEGER    NOT NULL,
O_ENTRY_D     TIMESTAMP  NOT NULL,
O_CARRIER_ID  SMALLINT   NOT NULL,
O_OL_CNT      SMALLINT   NOT NULL,
O_ALL_LOCAL   SMALLINT   NOT NULL,
O_ID          INTEGER    NOT NULL,
O_W_ID        INTEGER    NOT NULL,
O_D_ID        SMALLINT   NOT NULL
)
IN ts_order_020
INDEX IN is_order_020
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 1 ENDING AT 3695,
O_W_ID STARTING FROM 15429 ENDING AT 16240,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS21;
CREATE TABLE ORDERS21
(
O_C_ID        INTEGER    NOT NULL,
O_ENTRY_D     TIMESTAMP  NOT NULL,
O_CARRIER_ID  SMALLINT   NOT NULL,
O_OL_CNT      SMALLINT   NOT NULL,
O_ALL_LOCAL   SMALLINT   NOT NULL,
O_ID          INTEGER    NOT NULL,
O_W_ID        INTEGER    NOT NULL,
O_D_ID        SMALLINT   NOT NULL
)
IN ts_order_021
INDEX IN is_order_021
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 1 ENDING AT 3695,
O_W_ID STARTING FROM 16241 ENDING AT 17052,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS22;
CREATE TABLE ORDERS22
(
O_C_ID        INTEGER    NOT NULL,
O_ENTRY_D     TIMESTAMP  NOT NULL,
O_CARRIER_ID  SMALLINT   NOT NULL,
O_OL_CNT      SMALLINT   NOT NULL,
O_ALL_LOCAL   SMALLINT   NOT NULL,
O_ID          INTEGER    NOT NULL,
O_W_ID        INTEGER    NOT NULL,
O_D_ID        SMALLINT   NOT NULL
)
IN ts_order_022
INDEX IN is_order_022
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 1 ENDING AT 3695,
O_W_ID STARTING FROM 17053 ENDING AT 17864,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS23;
CREATE TABLE ORDERS23
(
O_C_ID        INTEGER    NOT NULL,
O_ENTRY_D     TIMESTAMP  NOT NULL,
O_CARRIER_ID  SMALLINT   NOT NULL,
O_OL_CNT      SMALLINT   NOT NULL,
O_ALL_LOCAL   SMALLINT   NOT NULL,
O_ID          INTEGER    NOT NULL,
O_W_ID        INTEGER    NOT NULL,
O_D_ID        SMALLINT   NOT NULL
)
IN ts_order_023
INDEX IN is_order_023
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 1 ENDING AT 3695,
O_W_ID STARTING FROM 17865 ENDING AT 18676,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS24;
CREATE TABLE ORDERS24
(
O_C_ID        INTEGER    NOT NULL,
O_ENTRY_D     TIMESTAMP  NOT NULL,
O_CARRIER_ID  SMALLINT   NOT NULL,
O_OL_CNT      SMALLINT   NOT NULL,
O_ALL_LOCAL   SMALLINT   NOT NULL,
O_ID          INTEGER    NOT NULL,
O_W_ID        INTEGER    NOT NULL,
O_D_ID        SMALLINT   NOT NULL
)
IN ts_order_024
INDEX IN is_order_024
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 1 ENDING AT 3695,
O_W_ID STARTING FROM 18677 ENDING AT 19488,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS25;
CREATE TABLE ORDERS25
(
O_C_ID        INTEGER    NOT NULL,
O_ENTRY_D     TIMESTAMP  NOT NULL,
O_CARRIER_ID  SMALLINT   NOT NULL,
O_OL_CNT      SMALLINT   NOT NULL,
O_ALL_LOCAL   SMALLINT   NOT NULL,
O_ID          INTEGER    NOT NULL,
O_W_ID        INTEGER    NOT NULL,
O_D_ID        SMALLINT   NOT NULL
)
IN ts_order_025
INDEX IN is_order_025
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 1 ENDING AT 3695,
O_W_ID STARTING FROM 19489 ENDING AT 20300,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS26;
CREATE TABLE ORDERS26
(
O_C_ID        INTEGER    NOT NULL,
O_ENTRY_D     TIMESTAMP  NOT NULL,
O_CARRIER_ID  SMALLINT   NOT NULL,
O_OL_CNT      SMALLINT   NOT NULL,
O_ALL_LOCAL   SMALLINT   NOT NULL,
O_ID          INTEGER    NOT NULL,
O_W_ID        INTEGER    NOT NULL,
O_D_ID        SMALLINT   NOT NULL
)
IN ts_order_026
INDEX IN is_order_026
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 1 ENDING AT 3695,
O_W_ID STARTING FROM 20301 ENDING AT 21112,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS27;
CREATE TABLE ORDERS27
(
O_C_ID        INTEGER    NOT NULL,
O_ENTRY_D     TIMESTAMP  NOT NULL,
O_CARRIER_ID  SMALLINT   NOT NULL,
O_OL_CNT      SMALLINT   NOT NULL,
O_ALL_LOCAL   SMALLINT   NOT NULL,
O_ID          INTEGER    NOT NULL,
O_W_ID        INTEGER    NOT NULL,
O_D_ID        SMALLINT   NOT NULL
)
IN ts_order_027
INDEX IN is_order_027
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 1 ENDING AT 3695,
O_W_ID STARTING FROM 21113 ENDING AT 21924,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS28;
CREATE TABLE ORDERS28
(
O_C_ID        INTEGER    NOT NULL,
O_ENTRY_D     TIMESTAMP  NOT NULL,
O_CARRIER_ID  SMALLINT   NOT NULL,
O_OL_CNT      SMALLINT   NOT NULL,
O_ALL_LOCAL   SMALLINT   NOT NULL,
O_ID          INTEGER    NOT NULL,
O_W_ID        INTEGER    NOT NULL,
O_D_ID        SMALLINT   NOT NULL
)
IN ts_order_028
INDEX IN is_order_028
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 1 ENDING AT 3695,
O_W_ID STARTING FROM 21925 ENDING AT 22736,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS29;
CREATE TABLE ORDERS29
(
O_C_ID        INTEGER    NOT NULL,
O_ENTRY_D     TIMESTAMP  NOT NULL,
O_CARRIER_ID  SMALLINT   NOT NULL,
O_OL_CNT      SMALLINT   NOT NULL,
O_ALL_LOCAL   SMALLINT   NOT NULL,
```

```
        O_ID          INTEGER    NOT NULL,
        O_W_ID         INTEGER    NOT NULL,
        O_D_ID        SMALLINT    NOT NULL
        )
        IN ts_order_029
        INDEX IN is_order_029
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 22737 ENDING AT 23548,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS30;
CREATE TABLE ORDERS30
        (
        O_C_ID        INTEGER    NOT NULL,
        O_ENTRY_D      TIMESTAMP   NOT NULL,
        O_CARRIER_ID   SMALLINT    NOT NULL,
        O_OL_CNT       SMALLINT    NOT NULL,
        O_ALL_LOCAL    SMALLINT    NOT NULL,
        O_ID          INTEGER    NOT NULL,
        O_W_ID         INTEGER    NOT NULL,
        O_D_ID        SMALLINT    NOT NULL
        )
        IN ts_order_030
        INDEX IN is_order_030
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 23549 ENDING AT 24360,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS31;
CREATE TABLE ORDERS31
        (
        O_C_ID        INTEGER    NOT NULL,
        O_ENTRY_D      TIMESTAMP   NOT NULL,
        O_CARRIER_ID   SMALLINT    NOT NULL,
        O_OL_CNT       SMALLINT    NOT NULL,
        O_ALL_LOCAL    SMALLINT    NOT NULL,
        O_ID          INTEGER    NOT NULL,
        O_W_ID         INTEGER    NOT NULL,
        O_D_ID        SMALLINT    NOT NULL
        )
        IN ts_order_031
        INDEX IN is_order_031
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 24361 ENDING AT 25172,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS32;
CREATE TABLE ORDERS32
        (
        O_C_ID        INTEGER    NOT NULL,
        O_ENTRY_D      TIMESTAMP   NOT NULL,
        O_CARRIER_ID   SMALLINT    NOT NULL,
        O_OL_CNT       SMALLINT    NOT NULL,
        O_ALL_LOCAL    SMALLINT    NOT NULL,
        O_ID          INTEGER    NOT NULL,
        O_W_ID         INTEGER    NOT NULL,
        O_D_ID        SMALLINT    NOT NULL
        )
        IN ts_order_032
        INDEX IN is_order_032
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 25173 ENDING AT 25984,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS33;
CREATE TABLE ORDERS33
        (
        O_C_ID        INTEGER    NOT NULL,
        O_ENTRY_D      TIMESTAMP   NOT NULL,
        O_CARRIER_ID   SMALLINT    NOT NULL,
        O_OL_CNT       SMALLINT    NOT NULL,
        O_ALL_LOCAL    SMALLINT    NOT NULL,
        O_ID          INTEGER    NOT NULL,
        O_W_ID         INTEGER    NOT NULL,
        O_D_ID        SMALLINT    NOT NULL
        )
        IN ts_order_033
        INDEX IN is_order_033
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 25985 ENDING AT 26796,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS34;
CREATE TABLE ORDERS34
        (
        O_C_ID        INTEGER    NOT NULL,
        O_ENTRY_D      TIMESTAMP   NOT NULL,
        O_CARRIER_ID   SMALLINT    NOT NULL,
        O_OL_CNT       SMALLINT    NOT NULL,
        O_ALL_LOCAL    SMALLINT    NOT NULL,
        O_ID          INTEGER    NOT NULL,
        O_W_ID         INTEGER    NOT NULL,
        O_D_ID        SMALLINT    NOT NULL
        )
        IN ts_order_034
        INDEX IN is_order_034
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 26797 ENDING AT 27608,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS35;
CREATE TABLE ORDERS35
        (
        O_C_ID        INTEGER    NOT NULL,
        O_ENTRY_D      TIMESTAMP   NOT NULL,
        O_CARRIER_ID   SMALLINT    NOT NULL,
        O_OL_CNT       SMALLINT    NOT NULL,
        O_ALL_LOCAL    SMALLINT    NOT NULL,
        O_ID          INTEGER    NOT NULL,
        O_W_ID         INTEGER    NOT NULL,
        O_D_ID        SMALLINT    NOT NULL
        )
        IN ts_order_035
        INDEX IN is_order_035
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 27609 ENDING AT 28420,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS36;
CREATE TABLE ORDERS36
        (
        O_C_ID        INTEGER    NOT NULL,
        O_ENTRY_D      TIMESTAMP   NOT NULL,
        O_CARRIER_ID   SMALLINT    NOT NULL,
        O_OL_CNT       SMALLINT    NOT NULL,
        O_ALL_LOCAL    SMALLINT    NOT NULL,
        O_ID          INTEGER    NOT NULL,
        O_W_ID         INTEGER    NOT NULL,
        O_D_ID        SMALLINT    NOT NULL
        )
        IN ts_order_036
        INDEX IN is_order_036
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 28421 ENDING AT 29232,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS37;
CREATE TABLE ORDERS37
        (
        O_C_ID        INTEGER    NOT NULL,
        O_ENTRY_D      TIMESTAMP   NOT NULL,
        O_CARRIER_ID   SMALLINT    NOT NULL,
        O_OL_CNT       SMALLINT    NOT NULL,
        O_ALL_LOCAL    SMALLINT    NOT NULL,
        O_ID          INTEGER    NOT NULL,
        O_W_ID         INTEGER    NOT NULL,
        O_D_ID        SMALLINT    NOT NULL
        )
        IN ts_order_037
        INDEX IN is_order_037
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 29233 ENDING AT 30044,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS38;
CREATE TABLE ORDERS38
        (
        O_C_ID        INTEGER    NOT NULL,
        O_ENTRY_D      TIMESTAMP   NOT NULL,
        O_CARRIER_ID   SMALLINT    NOT NULL,
        O_OL_CNT       SMALLINT    NOT NULL,
        O_ALL_LOCAL    SMALLINT    NOT NULL,
        O_ID          INTEGER    NOT NULL,
        O_W_ID         INTEGER    NOT NULL,
        O_D_ID        SMALLINT    NOT NULL
        )
        IN ts_order_038
        INDEX IN is_order_038
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 30045 ENDING AT 30856,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS39;
CREATE TABLE ORDERS39
        (
        O_C_ID        INTEGER    NOT NULL,
        O_ENTRY_D      TIMESTAMP   NOT NULL,
        O_CARRIER_ID   SMALLINT    NOT NULL,
        O_OL_CNT       SMALLINT    NOT NULL,
```

```
O_ALL_LOCAL    SMALLINT   NOT NULL,
O_ID           INTEGER    NOT NULL,
O_W_ID         INTEGER    NOT NULL,
O_D_ID         SMALLINT   NOT NULL
)
IN ts_order_039
INDEX IN is_order_039
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 1 ENDING AT 3695,
O_W_ID STARTING FROM 30857 ENDING AT 31668,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS40;
CREATE TABLE ORDERS40
(
O_C_ID         INTEGER    NOT NULL,
O_ENTRY_D      TIMESTAMP  NOT NULL,
O_CARRIER_ID   SMALLINT   NOT NULL,
O_OL_CNT       SMALLINT   NOT NULL,
O_ALL_LOCAL    SMALLINT   NOT NULL,
O_ID           INTEGER    NOT NULL,
O_W_ID         INTEGER    NOT NULL,
O_D_ID         SMALLINT   NOT NULL
)
IN ts_order_040
INDEX IN is_order_040
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 1 ENDING AT 3695,
O_W_ID STARTING FROM 31669 ENDING AT 32480,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS41;
CREATE TABLE ORDERS41
(
O_C_ID         INTEGER    NOT NULL,
O_ENTRY_D      TIMESTAMP  NOT NULL,
O_CARRIER_ID   SMALLINT   NOT NULL,
O_OL_CNT       SMALLINT   NOT NULL,
O_ALL_LOCAL    SMALLINT   NOT NULL,
O_ID           INTEGER    NOT NULL,
O_W_ID         INTEGER    NOT NULL,
O_D_ID         SMALLINT   NOT NULL
)
IN ts_order_041
INDEX IN is_order_041
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 1 ENDING AT 3695,
O_W_ID STARTING FROM 32481 ENDING AT 33292,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS42;
CREATE TABLE ORDERS42
(
O_C_ID         INTEGER    NOT NULL,
O_ENTRY_D      TIMESTAMP  NOT NULL,
O_CARRIER_ID   SMALLINT   NOT NULL,
O_OL_CNT       SMALLINT   NOT NULL,
O_ALL_LOCAL    SMALLINT   NOT NULL,
O_ID           INTEGER    NOT NULL,
O_W_ID         INTEGER    NOT NULL,
O_D_ID         SMALLINT   NOT NULL
)
IN ts_order_042
INDEX IN is_order_042
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 1 ENDING AT 3695,
O_W_ID STARTING FROM 33293 ENDING AT 34104,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS43;
CREATE TABLE ORDERS43
(
O_C_ID         INTEGER    NOT NULL,
O_ENTRY_D      TIMESTAMP  NOT NULL,
O_CARRIER_ID   SMALLINT   NOT NULL,
O_OL_CNT       SMALLINT   NOT NULL,
O_ALL_LOCAL    SMALLINT   NOT NULL,
O_ID           INTEGER    NOT NULL,
O_W_ID         INTEGER    NOT NULL,
O_D_ID         SMALLINT   NOT NULL
)
IN ts_order_043
INDEX IN is_order_043
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 1 ENDING AT 3695,
O_W_ID STARTING FROM 34105 ENDING AT 34916,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS44;
CREATE TABLE ORDERS44
(
O_C_ID         INTEGER    NOT NULL,
O_ENTRY_D      TIMESTAMP  NOT NULL,
O_CARRIER_ID   SMALLINT   NOT NULL,
O_OL_CNT       SMALLINT   NOT NULL,
O_ALL_LOCAL    SMALLINT   NOT NULL,
O_ID           INTEGER    NOT NULL,
O_W_ID         INTEGER    NOT NULL,
O_D_ID         SMALLINT   NOT NULL
)
IN ts_order_044
INDEX IN is_order_044
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 1 ENDING AT 3695,
O_W_ID STARTING FROM 34917 ENDING AT 35728,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS45;
CREATE TABLE ORDERS45
(
O_C_ID         INTEGER    NOT NULL,
O_ENTRY_D      TIMESTAMP  NOT NULL,
O_CARRIER_ID   SMALLINT   NOT NULL,
O_OL_CNT       SMALLINT   NOT NULL,
O_ALL_LOCAL    SMALLINT   NOT NULL,
O_ID           INTEGER    NOT NULL,
O_W_ID         INTEGER    NOT NULL,
O_D_ID         SMALLINT   NOT NULL
)
IN ts_order_045
INDEX IN is_order_045
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 1 ENDING AT 3695,
O_W_ID STARTING FROM 35729 ENDING AT 36540,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS46;
CREATE TABLE ORDERS46
(
O_C_ID         INTEGER    NOT NULL,
O_ENTRY_D      TIMESTAMP  NOT NULL,
O_CARRIER_ID   SMALLINT   NOT NULL,
O_OL_CNT       SMALLINT   NOT NULL,
O_ALL_LOCAL    SMALLINT   NOT NULL,
O_ID           INTEGER    NOT NULL,
O_W_ID         INTEGER    NOT NULL,
O_D_ID         SMALLINT   NOT NULL
)
IN ts_order_046
INDEX IN is_order_046
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 1 ENDING AT 3695,
O_W_ID STARTING FROM 36541 ENDING AT 37352,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS47;
CREATE TABLE ORDERS47
(
O_C_ID         INTEGER    NOT NULL,
O_ENTRY_D      TIMESTAMP  NOT NULL,
O_CARRIER_ID   SMALLINT   NOT NULL,
O_OL_CNT       SMALLINT   NOT NULL,
O_ALL_LOCAL    SMALLINT   NOT NULL,
O_ID           INTEGER    NOT NULL,
O_W_ID         INTEGER    NOT NULL,
O_D_ID         SMALLINT   NOT NULL
)
IN ts_order_047
INDEX IN is_order_047
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 1 ENDING AT 3695,
O_W_ID STARTING FROM 37353 ENDING AT 38164,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS48;
CREATE TABLE ORDERS48
(
O_C_ID         INTEGER    NOT NULL,
O_ENTRY_D      TIMESTAMP  NOT NULL,
O_CARRIER_ID   SMALLINT   NOT NULL,
O_OL_CNT       SMALLINT   NOT NULL,
O_ALL_LOCAL    SMALLINT   NOT NULL,
O_ID           INTEGER    NOT NULL,
O_W_ID         INTEGER    NOT NULL,
O_D_ID         SMALLINT   NOT NULL
)
IN ts_order_048
INDEX IN is_order_048
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 1 ENDING AT 3695,
O_W_ID STARTING FROM 38165 ENDING AT 38976,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS49;
CREATE TABLE ORDERS49
(
O_C_ID         INTEGER    NOT NULL,
O_ENTRY_D      TIMESTAMP  NOT NULL,
O_CARRIER_ID   SMALLINT   NOT NULL,
```

```
        O_OL_CNT       SMALLINT    NOT NULL,
        O_ALL_LOCAL    SMALLINT    NOT NULL,
        O_ID           INTEGER     NOT NULL,
        O_W_ID         INTEGER     NOT NULL,
        O_D_ID         SMALLINT    NOT NULL
        )
        IN ts_order_049
        INDEX IN is_order_049
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 38977 ENDING AT 39788,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS50;
CREATE TABLE ORDERS50
        (
        O_C_ID         INTEGER     NOT NULL,
        O_ENTRY_D      TIMESTAMP   NOT NULL,
        O_CARRIER_ID   SMALLINT    NOT NULL,
        O_OL_CNT       SMALLINT    NOT NULL,
        O_ALL_LOCAL    SMALLINT    NOT NULL,
        O_ID           INTEGER     NOT NULL,
        O_W_ID         INTEGER     NOT NULL,
        O_D_ID         SMALLINT    NOT NULL
        )
        IN ts_order_050
        INDEX IN is_order_050
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 39789 ENDING AT 40600,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS51;
CREATE TABLE ORDERS51
        (
        O_C_ID         INTEGER     NOT NULL,
        O_ENTRY_D      TIMESTAMP   NOT NULL,
        O_CARRIER_ID   SMALLINT    NOT NULL,
        O_OL_CNT       SMALLINT    NOT NULL,
        O_ALL_LOCAL    SMALLINT    NOT NULL,
        O_ID           INTEGER     NOT NULL,
        O_W_ID         INTEGER     NOT NULL,
        O_D_ID         SMALLINT    NOT NULL
        )
        IN ts_order_051
        INDEX IN is_order_051
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 40601 ENDING AT 41412,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS52;
CREATE TABLE ORDERS52
        (
        O_C_ID         INTEGER     NOT NULL,
        O_ENTRY_D      TIMESTAMP   NOT NULL,
        O_CARRIER_ID   SMALLINT    NOT NULL,
        O_OL_CNT       SMALLINT    NOT NULL,
        O_ALL_LOCAL    SMALLINT    NOT NULL,
        O_ID           INTEGER     NOT NULL,
        O_W_ID         INTEGER     NOT NULL,
        O_D_ID         SMALLINT    NOT NULL
        )
        IN ts_order_052
        INDEX IN is_order_052
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 41413 ENDING AT 42224,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS53;
CREATE TABLE ORDERS53
        (
        O_C_ID         INTEGER     NOT NULL,
        O_ENTRY_D      TIMESTAMP   NOT NULL,
        O_CARRIER_ID   SMALLINT    NOT NULL,
        O_OL_CNT       SMALLINT    NOT NULL,
        O_ALL_LOCAL    SMALLINT    NOT NULL,
        O_ID           INTEGER     NOT NULL,
        O_W_ID         INTEGER     NOT NULL,
        O_D_ID         SMALLINT    NOT NULL
        )
        IN ts_order_053
        INDEX IN is_order_053
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 42225 ENDING AT 43036,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS54;
CREATE TABLE ORDERS54
        (
        O_C_ID         INTEGER     NOT NULL,
        O_ENTRY_D      TIMESTAMP   NOT NULL,
        O_CARRIER_ID   SMALLINT    NOT NULL,
        O_OL_CNT       SMALLINT    NOT NULL,
        O_ALL_LOCAL    SMALLINT    NOT NULL,
        O_ID           INTEGER     NOT NULL,
        O_W_ID         INTEGER     NOT NULL,
        O_D_ID         SMALLINT    NOT NULL
        )
        IN ts_order_054
        INDEX IN is_order_054
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 43037 ENDING AT 43848,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS55;
CREATE TABLE ORDERS55
        (
        O_C_ID         INTEGER     NOT NULL,
        O_ENTRY_D      TIMESTAMP   NOT NULL,
        O_CARRIER_ID   SMALLINT    NOT NULL,
        O_OL_CNT       SMALLINT    NOT NULL,
        O_ALL_LOCAL    SMALLINT    NOT NULL,
        O_ID           INTEGER     NOT NULL,
        O_W_ID         INTEGER     NOT NULL,
        O_D_ID         SMALLINT    NOT NULL
        )
        IN ts_order_055
        INDEX IN is_order_055
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 43849 ENDING AT 44660,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS56;
CREATE TABLE ORDERS56
        (
        O_C_ID         INTEGER     NOT NULL,
        O_ENTRY_D      TIMESTAMP   NOT NULL,
        O_CARRIER_ID   SMALLINT    NOT NULL,
        O_OL_CNT       SMALLINT    NOT NULL,
        O_ALL_LOCAL    SMALLINT    NOT NULL,
        O_ID           INTEGER     NOT NULL,
        O_W_ID         INTEGER     NOT NULL,
        O_D_ID         SMALLINT    NOT NULL
        )
        IN ts_order_056
        INDEX IN is_order_056
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 44661 ENDING AT 45472,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS57;
CREATE TABLE ORDERS57
        (
        O_C_ID         INTEGER     NOT NULL,
        O_ENTRY_D      TIMESTAMP   NOT NULL,
        O_CARRIER_ID   SMALLINT    NOT NULL,
        O_OL_CNT       SMALLINT    NOT NULL,
        O_ALL_LOCAL    SMALLINT    NOT NULL,
        O_ID           INTEGER     NOT NULL,
        O_W_ID         INTEGER     NOT NULL,
        O_D_ID         SMALLINT    NOT NULL
        )
        IN ts_order_057
        INDEX IN is_order_057
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 45473 ENDING AT 46284,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS58;
CREATE TABLE ORDERS58
        (
        O_C_ID         INTEGER     NOT NULL,
        O_ENTRY_D      TIMESTAMP   NOT NULL,
        O_CARRIER_ID   SMALLINT    NOT NULL,
        O_OL_CNT       SMALLINT    NOT NULL,
        O_ALL_LOCAL    SMALLINT    NOT NULL,
        O_ID           INTEGER     NOT NULL,
        O_W_ID         INTEGER     NOT NULL,
        O_D_ID         SMALLINT    NOT NULL
        )
        IN ts_order_058
        INDEX IN is_order_058
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 46285 ENDING AT 47096,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS59;
CREATE TABLE ORDERS59
        (
        O_C_ID         INTEGER     NOT NULL,
        O_ENTRY_D      TIMESTAMP   NOT NULL,
```

```
        O_CARRIER_ID   SMALLINT   NOT NULL,
        O_OL_CNT       SMALLINT   NOT NULL,
        O_ALL_LOCAL    SMALLINT   NOT NULL,
        O_ID           INTEGER    NOT NULL,
        O_W_ID         INTEGER    NOT NULL,
        O_D_ID         SMALLINT   NOT NULL
        )
        IN ts_order_059
        INDEX IN is_order_059
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 47097 ENDING AT 47908,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS60;
CREATE TABLE ORDERS60
        (
        O_C_ID         INTEGER    NOT NULL,
        O_ENTRY_D      TIMESTAMP  NOT NULL,
        O_CARRIER_ID   SMALLINT   NOT NULL,
        O_OL_CNT       SMALLINT   NOT NULL,
        O_ALL_LOCAL    SMALLINT   NOT NULL,
        O_ID           INTEGER    NOT NULL,
        O_W_ID         INTEGER    NOT NULL,
        O_D_ID         SMALLINT   NOT NULL
        )
        IN ts_order_060
        INDEX IN is_order_060
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 47909 ENDING AT 48720,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS61;
CREATE TABLE ORDERS61
        (
        O_C_ID         INTEGER    NOT NULL,
        O_ENTRY_D      TIMESTAMP  NOT NULL,
        O_CARRIER_ID   SMALLINT   NOT NULL,
        O_OL_CNT       SMALLINT   NOT NULL,
        O_ALL_LOCAL    SMALLINT   NOT NULL,
        O_ID           INTEGER    NOT NULL,
        O_W_ID         INTEGER    NOT NULL,
        O_D_ID         SMALLINT   NOT NULL
        )
        IN ts_order_061
        INDEX IN is_order_061
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 48721 ENDING AT 49532,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS62;
CREATE TABLE ORDERS62
        (
        O_C_ID         INTEGER    NOT NULL,
        O_ENTRY_D      TIMESTAMP  NOT NULL,
        O_CARRIER_ID   SMALLINT   NOT NULL,
        O_OL_CNT       SMALLINT   NOT NULL,
        O_ALL_LOCAL    SMALLINT   NOT NULL,
        O_ID           INTEGER    NOT NULL,
        O_W_ID         INTEGER    NOT NULL,
        O_D_ID         SMALLINT   NOT NULL
        )
        IN ts_order_062
        INDEX IN is_order_062
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 49533 ENDING AT 50344,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS63;
CREATE TABLE ORDERS63
        (
        O_C_ID         INTEGER    NOT NULL,
        O_ENTRY_D      TIMESTAMP  NOT NULL,
        O_CARRIER_ID   SMALLINT   NOT NULL,
        O_OL_CNT       SMALLINT   NOT NULL,
        O_ALL_LOCAL    SMALLINT   NOT NULL,
        O_ID           INTEGER    NOT NULL,
        O_W_ID         INTEGER    NOT NULL,
        O_D_ID         SMALLINT   NOT NULL
        )
        IN ts_order_063
        INDEX IN is_order_063
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 50345 ENDING AT 51156,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS64;
CREATE TABLE ORDERS64
        (
        O_C_ID         INTEGER    NOT NULL,
        O_ENTRY_D      TIMESTAMP  NOT NULL,
        O_CARRIER_ID   SMALLINT   NOT NULL,
        O_OL_CNT       SMALLINT   NOT NULL,
        O_ALL_LOCAL    SMALLINT   NOT NULL,
        O_ID           INTEGER    NOT NULL,
        O_W_ID         INTEGER    NOT NULL,
        O_D_ID         SMALLINT   NOT NULL
        )
        IN ts_order_064
        INDEX IN is_order_064
        ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 1 ENDING AT 3695,
        O_W_ID STARTING FROM 51157 ENDING AT 51968,
        O_D_ID STARTING FROM 1 ENDING AT 10
        )
        ALLOW OVERFLOW;
connect reset;
```

DDL/CRTB_ORDER_LINE.ddl

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE1;
CREATE TABLE ORDER_LINE1
        (
        OL_DELIVERY_D   TIMESTAMP   NOT NULL,
        OL_AMOUNT       DECIMAL(6,2) NOT NULL,
        OL_I_ID         INTEGER     NOT NULL,
        OL_SUPPLY_W_ID  INTEGER     NOT NULL,
        OL_QUANTITY     SMALLINT    NOT NULL,
        OL_DIST_INFO    CHAR(24)    NOT NULL,
        OL_O_ID         INTEGER     NOT NULL,
        OL_D_ID         SMALLINT    NOT NULL,
        OL_W_ID         INTEGER     NOT NULL,
        OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_001
        INDEX IN ts_orderline_001
        ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 1 ENDING AT 812,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3695,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE2;
CREATE TABLE ORDER_LINE2
        (
        OL_DELIVERY_D   TIMESTAMP   NOT NULL,
        OL_AMOUNT       DECIMAL(6,2) NOT NULL,
        OL_I_ID         INTEGER     NOT NULL,
        OL_SUPPLY_W_ID  INTEGER     NOT NULL,
        OL_QUANTITY     SMALLINT    NOT NULL,
        OL_DIST_INFO    CHAR(24)    NOT NULL,
        OL_O_ID         INTEGER     NOT NULL,
        OL_D_ID         SMALLINT    NOT NULL,
        OL_W_ID         INTEGER     NOT NULL,
        OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_002
        INDEX IN ts_orderline_002
        ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 813 ENDING AT 1624,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3695,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE3;
CREATE TABLE ORDER_LINE3
        (
        OL_DELIVERY_D   TIMESTAMP   NOT NULL,
        OL_AMOUNT       DECIMAL(6,2) NOT NULL,
        OL_I_ID         INTEGER     NOT NULL,
        OL_SUPPLY_W_ID  INTEGER     NOT NULL,
        OL_QUANTITY     SMALLINT    NOT NULL,
        OL_DIST_INFO    CHAR(24)    NOT NULL,
        OL_O_ID         INTEGER     NOT NULL,
        OL_D_ID         SMALLINT    NOT NULL,
        OL_W_ID         INTEGER     NOT NULL,
        OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_003
        INDEX IN ts_orderline_003
        ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 1625 ENDING AT 2436,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3695,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE4;
CREATE TABLE ORDER_LINE4
        (
        OL_DELIVERY_D   TIMESTAMP   NOT NULL,
        OL_AMOUNT       DECIMAL(6,2) NOT NULL,
        OL_I_ID         INTEGER     NOT NULL,
        OL_SUPPLY_W_ID  INTEGER     NOT NULL,
        OL_QUANTITY     SMALLINT    NOT NULL,
        OL_DIST_INFO    CHAR(24)    NOT NULL,
        OL_O_ID         INTEGER     NOT NULL,
```

```
        OL_D_ID        SMALLINT    NOT NULL,
        OL_W_ID        INTEGER     NOT NULL,
        OL_NUMBER      SMALLINT    NOT NULL
        )
        IN ts_orderline_004
        INDEX IN ts_orderline_004
        ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 2437 ENDING AT 3248,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3695,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE5;
CREATE TABLE ORDER_LINE5
        (
        OL_DELIVERY_D   TIMESTAMP   NOT NULL,
        OL_AMOUNT       DECIMAL(6,2) NOT NULL,
        OL_I_ID         INTEGER     NOT NULL,
        OL_SUPPLY_W_ID  INTEGER     NOT NULL,
        OL_QUANTITY     SMALLINT    NOT NULL,
        OL_DIST_INFO    CHAR(24)    NOT NULL,
        OL_O_ID         INTEGER     NOT NULL,
        OL_D_ID         SMALLINT    NOT NULL,
        OL_W_ID         INTEGER     NOT NULL,
        OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_005
        INDEX IN ts_orderline_005
        ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 3249 ENDING AT 4060,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3695,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE6;
CREATE TABLE ORDER_LINE6
        (
        OL_DELIVERY_D   TIMESTAMP   NOT NULL,
        OL_AMOUNT       DECIMAL(6,2) NOT NULL,
        OL_I_ID         INTEGER     NOT NULL,
        OL_SUPPLY_W_ID  INTEGER     NOT NULL,
        OL_QUANTITY     SMALLINT    NOT NULL,
        OL_DIST_INFO    CHAR(24)    NOT NULL,
        OL_O_ID         INTEGER     NOT NULL,
        OL_D_ID         SMALLINT    NOT NULL,
        OL_W_ID         INTEGER     NOT NULL,
        OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_006
        INDEX IN ts_orderline_006
        ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 4061 ENDING AT 4872,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3695,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE7;
CREATE TABLE ORDER_LINE7
        (
        OL_DELIVERY_D   TIMESTAMP   NOT NULL,
        OL_AMOUNT       DECIMAL(6,2) NOT NULL,
        OL_I_ID         INTEGER     NOT NULL,
        OL_SUPPLY_W_ID  INTEGER     NOT NULL,
        OL_QUANTITY     SMALLINT    NOT NULL,
        OL_DIST_INFO    CHAR(24)    NOT NULL,
        OL_O_ID         INTEGER     NOT NULL,
        OL_D_ID         SMALLINT    NOT NULL,
        OL_W_ID         INTEGER     NOT NULL,
        OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_007
        INDEX IN ts_orderline_007
        ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 4873 ENDING AT 5684,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3695,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE8;
CREATE TABLE ORDER_LINE8
        (
        OL_DELIVERY_D   TIMESTAMP   NOT NULL,
        OL_AMOUNT       DECIMAL(6,2) NOT NULL,
        OL_I_ID         INTEGER     NOT NULL,
        OL_SUPPLY_W_ID  INTEGER     NOT NULL,
        OL_QUANTITY     SMALLINT    NOT NULL,
        OL_DIST_INFO    CHAR(24)    NOT NULL,
        OL_O_ID         INTEGER     NOT NULL,
        OL_D_ID         SMALLINT    NOT NULL,
        OL_W_ID         INTEGER     NOT NULL,
        OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_008
        INDEX IN ts_orderline_008
        ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 5685 ENDING AT 6496,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3695,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE9;
CREATE TABLE ORDER_LINE9
        (
        OL_DELIVERY_D   TIMESTAMP   NOT NULL,
        OL_AMOUNT       DECIMAL(6,2) NOT NULL,
        OL_I_ID         INTEGER     NOT NULL,
        OL_SUPPLY_W_ID  INTEGER     NOT NULL,
        OL_QUANTITY     SMALLINT    NOT NULL,
        OL_DIST_INFO    CHAR(24)    NOT NULL,
        OL_O_ID         INTEGER     NOT NULL,
        OL_D_ID         SMALLINT    NOT NULL,
        OL_W_ID         INTEGER     NOT NULL,
        OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_009
        INDEX IN ts_orderline_009
        ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 6497 ENDING AT 7308,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3695,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE10;
CREATE TABLE ORDER_LINE10
        (
        OL_DELIVERY_D   TIMESTAMP   NOT NULL,
        OL_AMOUNT       DECIMAL(6,2) NOT NULL,
        OL_I_ID         INTEGER     NOT NULL,
        OL_SUPPLY_W_ID  INTEGER     NOT NULL,
        OL_QUANTITY     SMALLINT    NOT NULL,
        OL_DIST_INFO    CHAR(24)    NOT NULL,
        OL_O_ID         INTEGER     NOT NULL,
        OL_D_ID         SMALLINT    NOT NULL,
        OL_W_ID         INTEGER     NOT NULL,
        OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_010
        INDEX IN ts_orderline_010
        ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 7309 ENDING AT 8120,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3695,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE11;
CREATE TABLE ORDER_LINE11
        (
        OL_DELIVERY_D   TIMESTAMP   NOT NULL,
        OL_AMOUNT       DECIMAL(6,2) NOT NULL,
        OL_I_ID         INTEGER     NOT NULL,
        OL_SUPPLY_W_ID  INTEGER     NOT NULL,
        OL_QUANTITY     SMALLINT    NOT NULL,
        OL_DIST_INFO    CHAR(24)    NOT NULL,
        OL_O_ID         INTEGER     NOT NULL,
        OL_D_ID         SMALLINT    NOT NULL,
        OL_W_ID         INTEGER     NOT NULL,
        OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_011
        INDEX IN ts_orderline_011
        ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 8121 ENDING AT 8932,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3695,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE12;
CREATE TABLE ORDER_LINE12
        (
        OL_DELIVERY_D   TIMESTAMP   NOT NULL,
        OL_AMOUNT       DECIMAL(6,2) NOT NULL,
        OL_I_ID         INTEGER     NOT NULL,
        OL_SUPPLY_W_ID  INTEGER     NOT NULL,
        OL_QUANTITY     SMALLINT    NOT NULL,
        OL_DIST_INFO    CHAR(24)    NOT NULL,
        OL_O_ID         INTEGER     NOT NULL,
        OL_D_ID         SMALLINT    NOT NULL,
        OL_W_ID         INTEGER     NOT NULL,
        OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_012
        INDEX IN ts_orderline_012
        ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 8933 ENDING AT 9744,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3695,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE13;
CREATE TABLE ORDER_LINE13
        (
        OL_DELIVERY_D   TIMESTAMP   NOT NULL,
```

```
 OL_AMOUNT       DECIMAL(6,2) NOT NULL,
 OL_I_ID         INTEGER     NOT NULL,
 OL_SUPPLY_W_ID  INTEGER     NOT NULL,
 OL_QUANTITY     SMALLINT    NOT NULL,
 OL_DIST_INFO    CHAR(24)    NOT NULL,
 OL_O_ID         INTEGER     NOT NULL,
 OL_D_ID         SMALLINT    NOT NULL,
 OL_W_ID         INTEGER     NOT NULL,
 OL_NUMBER       SMALLINT    NOT NULL
)
IN ts_orderline_013
INDEX IN ts_orderline_013
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 9745 ENDING AT 10556,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3695,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE14;
CREATE TABLE ORDER_LINE14
(
 OL_DELIVERY_D   TIMESTAMP   NOT NULL,
 OL_AMOUNT       DECIMAL(6,2) NOT NULL,
 OL_I_ID         INTEGER     NOT NULL,
 OL_SUPPLY_W_ID  INTEGER     NOT NULL,
 OL_QUANTITY     SMALLINT    NOT NULL,
 OL_DIST_INFO    CHAR(24)    NOT NULL,
 OL_O_ID         INTEGER     NOT NULL,
 OL_D_ID         SMALLINT    NOT NULL,
 OL_W_ID         INTEGER     NOT NULL,
 OL_NUMBER       SMALLINT    NOT NULL
)
IN ts_orderline_014
INDEX IN ts_orderline_014
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 10557 ENDING AT 11368,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3695,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE15;
CREATE TABLE ORDER_LINE15
(
 OL_DELIVERY_D   TIMESTAMP   NOT NULL,
 OL_AMOUNT       DECIMAL(6,2) NOT NULL,
 OL_I_ID         INTEGER     NOT NULL,
 OL_SUPPLY_W_ID  INTEGER     NOT NULL,
 OL_QUANTITY     SMALLINT    NOT NULL,
 OL_DIST_INFO    CHAR(24)    NOT NULL,
 OL_O_ID         INTEGER     NOT NULL,
 OL_D_ID         SMALLINT    NOT NULL,
 OL_W_ID         INTEGER     NOT NULL,
 OL_NUMBER       SMALLINT    NOT NULL
)
IN ts_orderline_015
INDEX IN ts_orderline_015
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 11369 ENDING AT 12180,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3695,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE16;
CREATE TABLE ORDER_LINE16
(
 OL_DELIVERY_D   TIMESTAMP   NOT NULL,
 OL_AMOUNT       DECIMAL(6,2) NOT NULL,
 OL_I_ID         INTEGER     NOT NULL,
 OL_SUPPLY_W_ID  INTEGER     NOT NULL,
 OL_QUANTITY     SMALLINT    NOT NULL,
 OL_DIST_INFO    CHAR(24)    NOT NULL,
 OL_O_ID         INTEGER     NOT NULL,
 OL_D_ID         SMALLINT    NOT NULL,
 OL_W_ID         INTEGER     NOT NULL,
 OL_NUMBER       SMALLINT    NOT NULL
)
IN ts_orderline_016
INDEX IN ts_orderline_016
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 12181 ENDING AT 12992,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3695,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE17;
CREATE TABLE ORDER_LINE17
(
 OL_DELIVERY_D   TIMESTAMP   NOT NULL,
 OL_AMOUNT       DECIMAL(6,2) NOT NULL,
 OL_I_ID         INTEGER     NOT NULL,
 OL_SUPPLY_W_ID  INTEGER     NOT NULL,
 OL_QUANTITY     SMALLINT    NOT NULL,
 OL_DIST_INFO    CHAR(24)    NOT NULL,
 OL_O_ID         INTEGER     NOT NULL,
 OL_D_ID         SMALLINT    NOT NULL,
 OL_W_ID         INTEGER     NOT NULL,
 OL_NUMBER       SMALLINT    NOT NULL
)
IN ts_orderline_017
INDEX IN ts_orderline_017
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 12993 ENDING AT 13804,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3695,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE18;
CREATE TABLE ORDER_LINE18
(
 OL_DELIVERY_D   TIMESTAMP   NOT NULL,
 OL_AMOUNT       DECIMAL(6,2) NOT NULL,
 OL_I_ID         INTEGER     NOT NULL,
 OL_SUPPLY_W_ID  INTEGER     NOT NULL,
 OL_QUANTITY     SMALLINT    NOT NULL,
 OL_DIST_INFO    CHAR(24)    NOT NULL,
 OL_O_ID         INTEGER     NOT NULL,
 OL_D_ID         SMALLINT    NOT NULL,
 OL_W_ID         INTEGER     NOT NULL,
 OL_NUMBER       SMALLINT    NOT NULL
)
IN ts_orderline_018
INDEX IN ts_orderline_018
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 13805 ENDING AT 14616,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3695,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE19;
CREATE TABLE ORDER_LINE19
(
 OL_DELIVERY_D   TIMESTAMP   NOT NULL,
 OL_AMOUNT       DECIMAL(6,2) NOT NULL,
 OL_I_ID         INTEGER     NOT NULL,
 OL_SUPPLY_W_ID  INTEGER     NOT NULL,
 OL_QUANTITY     SMALLINT    NOT NULL,
 OL_DIST_INFO    CHAR(24)    NOT NULL,
 OL_O_ID         INTEGER     NOT NULL,
 OL_D_ID         SMALLINT    NOT NULL,
 OL_W_ID         INTEGER     NOT NULL,
 OL_NUMBER       SMALLINT    NOT NULL
)
IN ts_orderline_019
INDEX IN ts_orderline_019
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 14617 ENDING AT 15428,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3695,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE20;
CREATE TABLE ORDER_LINE20
(
 OL_DELIVERY_D   TIMESTAMP   NOT NULL,
 OL_AMOUNT       DECIMAL(6,2) NOT NULL,
 OL_I_ID         INTEGER     NOT NULL,
 OL_SUPPLY_W_ID  INTEGER     NOT NULL,
 OL_QUANTITY     SMALLINT    NOT NULL,
 OL_DIST_INFO    CHAR(24)    NOT NULL,
 OL_O_ID         INTEGER     NOT NULL,
 OL_D_ID         SMALLINT    NOT NULL,
 OL_W_ID         INTEGER     NOT NULL,
 OL_NUMBER       SMALLINT    NOT NULL
)
IN ts_orderline_020
INDEX IN ts_orderline_020
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 15429 ENDING AT 16240,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3695,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE21;
CREATE TABLE ORDER_LINE21
(
 OL_DELIVERY_D   TIMESTAMP   NOT NULL,
 OL_AMOUNT       DECIMAL(6,2) NOT NULL,
 OL_I_ID         INTEGER     NOT NULL,
 OL_SUPPLY_W_ID  INTEGER     NOT NULL,
 OL_QUANTITY     SMALLINT    NOT NULL,
 OL_DIST_INFO    CHAR(24)    NOT NULL,
 OL_O_ID         INTEGER     NOT NULL,
 OL_D_ID         SMALLINT    NOT NULL,
 OL_W_ID         INTEGER     NOT NULL,
 OL_NUMBER       SMALLINT    NOT NULL
)
IN ts_orderline_021
INDEX IN ts_orderline_021
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 16241 ENDING AT 17052,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3695,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
```

```
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE22;
CREATE TABLE ORDER_LINE22
        (
         OL_DELIVERY_D   TIMESTAMP   NOT NULL,
         OL_AMOUNT       DECIMAL(6,2) NOT NULL,
         OL_I_ID         INTEGER     NOT NULL,
         OL_SUPPLY_W_ID  INTEGER     NOT NULL,
         OL_QUANTITY     SMALLINT    NOT NULL,
         OL_DIST_INFO    CHAR(24)    NOT NULL,
         OL_O_ID         INTEGER     NOT NULL,
         OL_D_ID         SMALLINT    NOT NULL,
         OL_W_ID         INTEGER     NOT NULL,
         OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_022
        INDEX IN ts_orderline_022
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 17053 ENDING AT 17864,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3695,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE23;
CREATE TABLE ORDER_LINE23
        (
         OL_DELIVERY_D   TIMESTAMP   NOT NULL,
         OL_AMOUNT       DECIMAL(6,2) NOT NULL,
         OL_I_ID         INTEGER     NOT NULL,
         OL_SUPPLY_W_ID  INTEGER     NOT NULL,
         OL_QUANTITY     SMALLINT    NOT NULL,
         OL_DIST_INFO    CHAR(24)    NOT NULL,
         OL_O_ID         INTEGER     NOT NULL,
         OL_D_ID         SMALLINT    NOT NULL,
         OL_W_ID         INTEGER     NOT NULL,
         OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_023
        INDEX IN ts_orderline_023
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 17865 ENDING AT 18676,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3695,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE24;
CREATE TABLE ORDER_LINE24
        (
         OL_DELIVERY_D   TIMESTAMP   NOT NULL,
         OL_AMOUNT       DECIMAL(6,2) NOT NULL,
         OL_I_ID         INTEGER     NOT NULL,
         OL_SUPPLY_W_ID  INTEGER     NOT NULL,
         OL_QUANTITY     SMALLINT    NOT NULL,
         OL_DIST_INFO    CHAR(24)    NOT NULL,
         OL_O_ID         INTEGER     NOT NULL,
         OL_D_ID         SMALLINT    NOT NULL,
         OL_W_ID         INTEGER     NOT NULL,
         OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_024
        INDEX IN ts_orderline_024
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 18677 ENDING AT 19488,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3695,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE25;
CREATE TABLE ORDER_LINE25
        (
         OL_DELIVERY_D   TIMESTAMP   NOT NULL,
         OL_AMOUNT       DECIMAL(6,2) NOT NULL,
         OL_I_ID         INTEGER     NOT NULL,
         OL_SUPPLY_W_ID  INTEGER     NOT NULL,
         OL_QUANTITY     SMALLINT    NOT NULL,
         OL_DIST_INFO    CHAR(24)    NOT NULL,
         OL_O_ID         INTEGER     NOT NULL,
         OL_D_ID         SMALLINT    NOT NULL,
         OL_W_ID         INTEGER     NOT NULL,
         OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_025
        INDEX IN ts_orderline_025
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 19489 ENDING AT 20300,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3695,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE26;
CREATE TABLE ORDER_LINE26
        (
         OL_DELIVERY_D   TIMESTAMP   NOT NULL,
         OL_AMOUNT       DECIMAL(6,2) NOT NULL,
         OL_I_ID         INTEGER     NOT NULL,
         OL_SUPPLY_W_ID  INTEGER     NOT NULL,
         OL_QUANTITY     SMALLINT    NOT NULL,
         OL_DIST_INFO    CHAR(24)    NOT NULL,
         OL_O_ID         INTEGER     NOT NULL,
         OL_D_ID         SMALLINT    NOT NULL,
         OL_W_ID         INTEGER     NOT NULL,
         OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_026
        INDEX IN ts_orderline_026
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 20301 ENDING AT 21112,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3695,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE27;
CREATE TABLE ORDER_LINE27
        (
         OL_DELIVERY_D   TIMESTAMP   NOT NULL,
         OL_AMOUNT       DECIMAL(6,2) NOT NULL,
         OL_I_ID         INTEGER     NOT NULL,
         OL_SUPPLY_W_ID  INTEGER     NOT NULL,
         OL_QUANTITY     SMALLINT    NOT NULL,
         OL_DIST_INFO    CHAR(24)    NOT NULL,
         OL_O_ID         INTEGER     NOT NULL,
         OL_D_ID         SMALLINT    NOT NULL,
         OL_W_ID         INTEGER     NOT NULL,
         OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_027
        INDEX IN ts_orderline_027
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 21113 ENDING AT 21924,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3695,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE28;
CREATE TABLE ORDER_LINE28
        (
         OL_DELIVERY_D   TIMESTAMP   NOT NULL,
         OL_AMOUNT       DECIMAL(6,2) NOT NULL,
         OL_I_ID         INTEGER     NOT NULL,
         OL_SUPPLY_W_ID  INTEGER     NOT NULL,
         OL_QUANTITY     SMALLINT    NOT NULL,
         OL_DIST_INFO    CHAR(24)    NOT NULL,
         OL_O_ID         INTEGER     NOT NULL,
         OL_D_ID         SMALLINT    NOT NULL,
         OL_W_ID         INTEGER     NOT NULL,
         OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_028
        INDEX IN ts_orderline_028
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 21925 ENDING AT 22736,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3695,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE29;
CREATE TABLE ORDER_LINE29
        (
         OL_DELIVERY_D   TIMESTAMP   NOT NULL,
         OL_AMOUNT       DECIMAL(6,2) NOT NULL,
         OL_I_ID         INTEGER     NOT NULL,
         OL_SUPPLY_W_ID  INTEGER     NOT NULL,
         OL_QUANTITY     SMALLINT    NOT NULL,
         OL_DIST_INFO    CHAR(24)    NOT NULL,
         OL_O_ID         INTEGER     NOT NULL,
         OL_D_ID         SMALLINT    NOT NULL,
         OL_W_ID         INTEGER     NOT NULL,
         OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_029
        INDEX IN ts_orderline_029
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 22737 ENDING AT 23548,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3695,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE30;
CREATE TABLE ORDER_LINE30
        (
         OL_DELIVERY_D   TIMESTAMP   NOT NULL,
         OL_AMOUNT       DECIMAL(6,2) NOT NULL,
         OL_I_ID         INTEGER     NOT NULL,
         OL_SUPPLY_W_ID  INTEGER     NOT NULL,
         OL_QUANTITY     SMALLINT    NOT NULL,
         OL_DIST_INFO    CHAR(24)    NOT NULL,
         OL_O_ID         INTEGER     NOT NULL,
         OL_D_ID         SMALLINT    NOT NULL,
         OL_W_ID         INTEGER     NOT NULL,
         OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_030
        INDEX IN ts_orderline_030
        ORGANIZE BY KEY SEQUENCE (
```

```
            OL_W_ID STARTING FROM 23549 ENDING AT 24360,
            OL_D_ID STARTING FROM 1 ENDING AT 10,
            OL_O_ID STARTING FROM 1 ENDING AT 3695,
            OL_NUMBER STARTING FROM 1 ENDING AT 15
            )
            ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE31;
CREATE TABLE ORDER_LINE31
            (
            OL_DELIVERY_D   TIMESTAMP   NOT NULL,
            OL_AMOUNT       DECIMAL(6,2) NOT NULL,
            OL_I_ID         INTEGER     NOT NULL,
            OL_SUPPLY_W_ID  INTEGER     NOT NULL,
            OL_QUANTITY     SMALLINT    NOT NULL,
            OL_DIST_INFO    CHAR(24)    NOT NULL,
            OL_O_ID         INTEGER     NOT NULL,
            OL_D_ID         SMALLINT    NOT NULL,
            OL_W_ID         INTEGER     NOT NULL,
            OL_NUMBER       SMALLINT    NOT NULL
            )
            IN ts_orderline_031
            INDEX IN ts_orderline_031
            ORGANIZE BY KEY SEQUENCE (
            OL_W_ID STARTING FROM 24361 ENDING AT 25172,
            OL_D_ID STARTING FROM 1 ENDING AT 10,
            OL_O_ID STARTING FROM 1 ENDING AT 3695,
            OL_NUMBER STARTING FROM 1 ENDING AT 15
            )
            ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE32;
CREATE TABLE ORDER_LINE32
            (
            OL_DELIVERY_D   TIMESTAMP   NOT NULL,
            OL_AMOUNT       DECIMAL(6,2) NOT NULL,
            OL_I_ID         INTEGER     NOT NULL,
            OL_SUPPLY_W_ID  INTEGER     NOT NULL,
            OL_QUANTITY     SMALLINT    NOT NULL,
            OL_DIST_INFO    CHAR(24)    NOT NULL,
            OL_O_ID         INTEGER     NOT NULL,
            OL_D_ID         SMALLINT    NOT NULL,
            OL_W_ID         INTEGER     NOT NULL,
            OL_NUMBER       SMALLINT    NOT NULL
            )
            IN ts_orderline_032
            INDEX IN ts_orderline_032
            ORGANIZE BY KEY SEQUENCE (
            OL_W_ID STARTING FROM 25173 ENDING AT 25984,
            OL_D_ID STARTING FROM 1 ENDING AT 10,
            OL_O_ID STARTING FROM 1 ENDING AT 3695,
            OL_NUMBER STARTING FROM 1 ENDING AT 15
            )
            ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE33;
CREATE TABLE ORDER_LINE33
            (
            OL_DELIVERY_D   TIMESTAMP   NOT NULL,
            OL_AMOUNT       DECIMAL(6,2) NOT NULL,
            OL_I_ID         INTEGER     NOT NULL,
            OL_SUPPLY_W_ID  INTEGER     NOT NULL,
            OL_QUANTITY     SMALLINT    NOT NULL,
            OL_DIST_INFO    CHAR(24)    NOT NULL,
            OL_O_ID         INTEGER     NOT NULL,
            OL_D_ID         SMALLINT    NOT NULL,
            OL_W_ID         INTEGER     NOT NULL,
            OL_NUMBER       SMALLINT    NOT NULL
            )
            IN ts_orderline_033
            INDEX IN ts_orderline_033
            ORGANIZE BY KEY SEQUENCE (
            OL_W_ID STARTING FROM 25985 ENDING AT 26796,
            OL_D_ID STARTING FROM 1 ENDING AT 10,
            OL_O_ID STARTING FROM 1 ENDING AT 3695,
            OL_NUMBER STARTING FROM 1 ENDING AT 15
            )
            ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE34;
CREATE TABLE ORDER_LINE34
            (
            OL_DELIVERY_D   TIMESTAMP   NOT NULL,
            OL_AMOUNT       DECIMAL(6,2) NOT NULL,
            OL_I_ID         INTEGER     NOT NULL,
            OL_SUPPLY_W_ID  INTEGER     NOT NULL,
            OL_QUANTITY     SMALLINT    NOT NULL,
            OL_DIST_INFO    CHAR(24)    NOT NULL,
            OL_O_ID         INTEGER     NOT NULL,
            OL_D_ID         SMALLINT    NOT NULL,
            OL_W_ID         INTEGER     NOT NULL,
            OL_NUMBER       SMALLINT    NOT NULL
            )
            IN ts_orderline_034
            INDEX IN ts_orderline_034
            ORGANIZE BY KEY SEQUENCE (
            OL_W_ID STARTING FROM 26797 ENDING AT 27608,
            OL_D_ID STARTING FROM 1 ENDING AT 10,
            OL_O_ID STARTING FROM 1 ENDING AT 3695,
            OL_NUMBER STARTING FROM 1 ENDING AT 15
            )
            ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE35;
CREATE TABLE ORDER_LINE35
            (
            OL_DELIVERY_D   TIMESTAMP   NOT NULL,
            OL_AMOUNT       DECIMAL(6,2) NOT NULL,
            OL_I_ID         INTEGER     NOT NULL,
            OL_SUPPLY_W_ID  INTEGER     NOT NULL,
            OL_QUANTITY     SMALLINT    NOT NULL,
            OL_DIST_INFO    CHAR(24)    NOT NULL,
            OL_O_ID         INTEGER     NOT NULL,
            OL_D_ID         SMALLINT    NOT NULL,
            OL_W_ID         INTEGER     NOT NULL,
            OL_NUMBER       SMALLINT    NOT NULL
            )
            IN ts_orderline_035
            INDEX IN ts_orderline_035
            ORGANIZE BY KEY SEQUENCE (
            OL_W_ID STARTING FROM 27609 ENDING AT 28420,
            OL_D_ID STARTING FROM 1 ENDING AT 10,
            OL_O_ID STARTING FROM 1 ENDING AT 3695,
            OL_NUMBER STARTING FROM 1 ENDING AT 15
            )
            ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE36;
CREATE TABLE ORDER_LINE36
            (
            OL_DELIVERY_D   TIMESTAMP   NOT NULL,
            OL_AMOUNT       DECIMAL(6,2) NOT NULL,
            OL_I_ID         INTEGER     NOT NULL,
            OL_SUPPLY_W_ID  INTEGER     NOT NULL,
            OL_QUANTITY     SMALLINT    NOT NULL,
            OL_DIST_INFO    CHAR(24)    NOT NULL,
            OL_O_ID         INTEGER     NOT NULL,
            OL_D_ID         SMALLINT    NOT NULL,
            OL_W_ID         INTEGER     NOT NULL,
            OL_NUMBER       SMALLINT    NOT NULL
            )
            IN ts_orderline_036
            INDEX IN ts_orderline_036
            ORGANIZE BY KEY SEQUENCE (
            OL_W_ID STARTING FROM 28421 ENDING AT 29232,
            OL_D_ID STARTING FROM 1 ENDING AT 10,
            OL_O_ID STARTING FROM 1 ENDING AT 3695,
            OL_NUMBER STARTING FROM 1 ENDING AT 15
            )
            ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE37;
CREATE TABLE ORDER_LINE37
            (
            OL_DELIVERY_D   TIMESTAMP   NOT NULL,
            OL_AMOUNT       DECIMAL(6,2) NOT NULL,
            OL_I_ID         INTEGER     NOT NULL,
            OL_SUPPLY_W_ID  INTEGER     NOT NULL,
            OL_QUANTITY     SMALLINT    NOT NULL,
            OL_DIST_INFO    CHAR(24)    NOT NULL,
            OL_O_ID         INTEGER     NOT NULL,
            OL_D_ID         SMALLINT    NOT NULL,
            OL_W_ID         INTEGER     NOT NULL,
            OL_NUMBER       SMALLINT    NOT NULL
            )
            IN ts_orderline_037
            INDEX IN ts_orderline_037
            ORGANIZE BY KEY SEQUENCE (
            OL_W_ID STARTING FROM 29233 ENDING AT 30044,
            OL_D_ID STARTING FROM 1 ENDING AT 10,
            OL_O_ID STARTING FROM 1 ENDING AT 3695,
            OL_NUMBER STARTING FROM 1 ENDING AT 15
            )
            ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE38;
CREATE TABLE ORDER_LINE38
            (
            OL_DELIVERY_D   TIMESTAMP   NOT NULL,
            OL_AMOUNT       DECIMAL(6,2) NOT NULL,
            OL_I_ID         INTEGER     NOT NULL,
            OL_SUPPLY_W_ID  INTEGER     NOT NULL,
            OL_QUANTITY     SMALLINT    NOT NULL,
            OL_DIST_INFO    CHAR(24)    NOT NULL,
            OL_O_ID         INTEGER     NOT NULL,
            OL_D_ID         SMALLINT    NOT NULL,
            OL_W_ID         INTEGER     NOT NULL,
            OL_NUMBER       SMALLINT    NOT NULL
            )
            IN ts_orderline_038
            INDEX IN ts_orderline_038
            ORGANIZE BY KEY SEQUENCE (
            OL_W_ID STARTING FROM 30045 ENDING AT 30856,
            OL_D_ID STARTING FROM 1 ENDING AT 10,
            OL_O_ID STARTING FROM 1 ENDING AT 3695,
            OL_NUMBER STARTING FROM 1 ENDING AT 15
            )
            ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE39;
CREATE TABLE ORDER_LINE39
            (
            OL_DELIVERY_D   TIMESTAMP   NOT NULL,
            OL_AMOUNT       DECIMAL(6,2) NOT NULL,
            OL_I_ID         INTEGER     NOT NULL,
            OL_SUPPLY_W_ID  INTEGER     NOT NULL,
            OL_QUANTITY     SMALLINT    NOT NULL,
            OL_DIST_INFO    CHAR(24)    NOT NULL,
            OL_O_ID         INTEGER     NOT NULL,
            OL_D_ID         SMALLINT    NOT NULL,
```

```
        OL_W_ID         INTEGER     NOT NULL,
        OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_039
        INDEX IN ts_orderline_039
        ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 30857 ENDING AT 31668,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3695,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE40;
CREATE TABLE ORDER_LINE40
        (
        OL_DELIVERY_D   TIMESTAMP   NOT NULL,
        OL_AMOUNT       DECIMAL(6,2) NOT NULL,
        OL_I_ID         INTEGER     NOT NULL,
        OL_SUPPLY_W_ID  INTEGER     NOT NULL,
        OL_QUANTITY     SMALLINT    NOT NULL,
        OL_DIST_INFO    CHAR(24)    NOT NULL,
        OL_O_ID         INTEGER     NOT NULL,
        OL_D_ID         SMALLINT    NOT NULL,
        OL_W_ID         INTEGER     NOT NULL,
        OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_040
        INDEX IN ts_orderline_040
        ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 31669 ENDING AT 32480,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3695,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE41;
CREATE TABLE ORDER_LINE41
        (
        OL_DELIVERY_D   TIMESTAMP   NOT NULL,
        OL_AMOUNT       DECIMAL(6,2) NOT NULL,
        OL_I_ID         INTEGER     NOT NULL,
        OL_SUPPLY_W_ID  INTEGER     NOT NULL,
        OL_QUANTITY     SMALLINT    NOT NULL,
        OL_DIST_INFO    CHAR(24)    NOT NULL,
        OL_O_ID         INTEGER     NOT NULL,
        OL_D_ID         SMALLINT    NOT NULL,
        OL_W_ID         INTEGER     NOT NULL,
        OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_041
        INDEX IN ts_orderline_041
        ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 32481 ENDING AT 33292,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3695,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE42;
CREATE TABLE ORDER_LINE42
        (
        OL_DELIVERY_D   TIMESTAMP   NOT NULL,
        OL_AMOUNT       DECIMAL(6,2) NOT NULL,
        OL_I_ID         INTEGER     NOT NULL,
        OL_SUPPLY_W_ID  INTEGER     NOT NULL,
        OL_QUANTITY     SMALLINT    NOT NULL,
        OL_DIST_INFO    CHAR(24)    NOT NULL,
        OL_O_ID         INTEGER     NOT NULL,
        OL_D_ID         SMALLINT    NOT NULL,
        OL_W_ID         INTEGER     NOT NULL,
        OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_042
        INDEX IN ts_orderline_042
        ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 33293 ENDING AT 34104,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3695,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE43;
CREATE TABLE ORDER_LINE43
        (
        OL_DELIVERY_D   TIMESTAMP   NOT NULL,
        OL_AMOUNT       DECIMAL(6,2) NOT NULL,
        OL_I_ID         INTEGER     NOT NULL,
        OL_SUPPLY_W_ID  INTEGER     NOT NULL,
        OL_QUANTITY     SMALLINT    NOT NULL,
        OL_DIST_INFO    CHAR(24)    NOT NULL,
        OL_O_ID         INTEGER     NOT NULL,
        OL_D_ID         SMALLINT    NOT NULL,
        OL_W_ID         INTEGER     NOT NULL,
        OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_043
        INDEX IN ts_orderline_043
        ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 34105 ENDING AT 34916,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3695,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE44;
CREATE TABLE ORDER_LINE44
        (
        OL_DELIVERY_D   TIMESTAMP   NOT NULL,
        OL_AMOUNT       DECIMAL(6,2) NOT NULL,
        OL_I_ID         INTEGER     NOT NULL,
        OL_SUPPLY_W_ID  INTEGER     NOT NULL,
        OL_QUANTITY     SMALLINT    NOT NULL,
        OL_DIST_INFO    CHAR(24)    NOT NULL,
        OL_O_ID         INTEGER     NOT NULL,
        OL_D_ID         SMALLINT    NOT NULL,
        OL_W_ID         INTEGER     NOT NULL,
        OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_044
        INDEX IN ts_orderline_044
        ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 34917 ENDING AT 35728,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3695,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE45;
CREATE TABLE ORDER_LINE45
        (
        OL_DELIVERY_D   TIMESTAMP   NOT NULL,
        OL_AMOUNT       DECIMAL(6,2) NOT NULL,
        OL_I_ID         INTEGER     NOT NULL,
        OL_SUPPLY_W_ID  INTEGER     NOT NULL,
        OL_QUANTITY     SMALLINT    NOT NULL,
        OL_DIST_INFO    CHAR(24)    NOT NULL,
        OL_O_ID         INTEGER     NOT NULL,
        OL_D_ID         SMALLINT    NOT NULL,
        OL_W_ID         INTEGER     NOT NULL,
        OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_045
        INDEX IN ts_orderline_045
        ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 35729 ENDING AT 36540,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3695,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE46;
CREATE TABLE ORDER_LINE46
        (
        OL_DELIVERY_D   TIMESTAMP   NOT NULL,
        OL_AMOUNT       DECIMAL(6,2) NOT NULL,
        OL_I_ID         INTEGER     NOT NULL,
        OL_SUPPLY_W_ID  INTEGER     NOT NULL,
        OL_QUANTITY     SMALLINT    NOT NULL,
        OL_DIST_INFO    CHAR(24)    NOT NULL,
        OL_O_ID         INTEGER     NOT NULL,
        OL_D_ID         SMALLINT    NOT NULL,
        OL_W_ID         INTEGER     NOT NULL,
        OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_046
        INDEX IN ts_orderline_046
        ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 36541 ENDING AT 37352,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3695,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE47;
CREATE TABLE ORDER_LINE47
        (
        OL_DELIVERY_D   TIMESTAMP   NOT NULL,
        OL_AMOUNT       DECIMAL(6,2) NOT NULL,
        OL_I_ID         INTEGER     NOT NULL,
        OL_SUPPLY_W_ID  INTEGER     NOT NULL,
        OL_QUANTITY     SMALLINT    NOT NULL,
        OL_DIST_INFO    CHAR(24)    NOT NULL,
        OL_O_ID         INTEGER     NOT NULL,
        OL_D_ID         SMALLINT    NOT NULL,
        OL_W_ID         INTEGER     NOT NULL,
        OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_047
        INDEX IN ts_orderline_047
        ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 37353 ENDING AT 38164,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 1 ENDING AT 3695,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE48;
CREATE TABLE ORDER_LINE48
        (
        OL_DELIVERY_D   TIMESTAMP   NOT NULL,
        OL_AMOUNT       DECIMAL(6,2) NOT NULL,
```

```
 OL_I_ID        INTEGER     NOT NULL,
 OL_SUPPLY_W_ID  INTEGER     NOT NULL,
 OL_QUANTITY     SMALLINT    NOT NULL,
 OL_DIST_INFO    CHAR(24)    NOT NULL,
 OL_O_ID        INTEGER     NOT NULL,
 OL_D_ID        SMALLINT    NOT NULL,
 OL_W_ID        INTEGER     NOT NULL,
 OL_NUMBER      SMALLINT    NOT NULL
)
IN ts_orderline_048
INDEX IN ts_orderline_048
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 38165 ENDING AT 38976,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3695,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE49;
CREATE TABLE ORDER_LINE49
(
 OL_DELIVERY_D   TIMESTAMP   NOT NULL,
 OL_AMOUNT       DECIMAL(6,2) NOT NULL,
 OL_I_ID        INTEGER     NOT NULL,
 OL_SUPPLY_W_ID  INTEGER     NOT NULL,
 OL_QUANTITY     SMALLINT    NOT NULL,
 OL_DIST_INFO    CHAR(24)    NOT NULL,
 OL_O_ID        INTEGER     NOT NULL,
 OL_D_ID        SMALLINT    NOT NULL,
 OL_W_ID        INTEGER     NOT NULL,
 OL_NUMBER      SMALLINT    NOT NULL
)
IN ts_orderline_049
INDEX IN ts_orderline_049
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 38977 ENDING AT 39788,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3695,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE50;
CREATE TABLE ORDER_LINE50
(
 OL_DELIVERY_D   TIMESTAMP   NOT NULL,
 OL_AMOUNT       DECIMAL(6,2) NOT NULL,
 OL_I_ID        INTEGER     NOT NULL,
 OL_SUPPLY_W_ID  INTEGER     NOT NULL,
 OL_QUANTITY     SMALLINT    NOT NULL,
 OL_DIST_INFO    CHAR(24)    NOT NULL,
 OL_O_ID        INTEGER     NOT NULL,
 OL_D_ID        SMALLINT    NOT NULL,
 OL_W_ID        INTEGER     NOT NULL,
 OL_NUMBER      SMALLINT    NOT NULL
)
IN ts_orderline_050
INDEX IN ts_orderline_050
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 39789 ENDING AT 40600,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3695,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE51;
CREATE TABLE ORDER_LINE51
(
 OL_DELIVERY_D   TIMESTAMP   NOT NULL,
 OL_AMOUNT       DECIMAL(6,2) NOT NULL,
 OL_I_ID        INTEGER     NOT NULL,
 OL_SUPPLY_W_ID  INTEGER     NOT NULL,
 OL_QUANTITY     SMALLINT    NOT NULL,
 OL_DIST_INFO    CHAR(24)    NOT NULL,
 OL_O_ID        INTEGER     NOT NULL,
 OL_D_ID        SMALLINT    NOT NULL,
 OL_W_ID        INTEGER     NOT NULL,
 OL_NUMBER      SMALLINT    NOT NULL
)
IN ts_orderline_051
INDEX IN ts_orderline_051
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 40601 ENDING AT 41412,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3695,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE52;
CREATE TABLE ORDER_LINE52
(
 OL_DELIVERY_D   TIMESTAMP   NOT NULL,
 OL_AMOUNT       DECIMAL(6,2) NOT NULL,
 OL_I_ID        INTEGER     NOT NULL,
 OL_SUPPLY_W_ID  INTEGER     NOT NULL,
 OL_QUANTITY     SMALLINT    NOT NULL,
 OL_DIST_INFO    CHAR(24)    NOT NULL,
 OL_O_ID        INTEGER     NOT NULL,
 OL_D_ID        SMALLINT    NOT NULL,
 OL_W_ID        INTEGER     NOT NULL,
 OL_NUMBER      SMALLINT    NOT NULL
)
IN ts_orderline_052
INDEX IN ts_orderline_052
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 41413 ENDING AT 42224,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3695,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE53;
CREATE TABLE ORDER_LINE53
(
 OL_DELIVERY_D   TIMESTAMP   NOT NULL,
 OL_AMOUNT       DECIMAL(6,2) NOT NULL,
 OL_I_ID        INTEGER     NOT NULL,
 OL_SUPPLY_W_ID  INTEGER     NOT NULL,
 OL_QUANTITY     SMALLINT    NOT NULL,
 OL_DIST_INFO    CHAR(24)    NOT NULL,
 OL_O_ID        INTEGER     NOT NULL,
 OL_D_ID        SMALLINT    NOT NULL,
 OL_W_ID        INTEGER     NOT NULL,
 OL_NUMBER      SMALLINT    NOT NULL
)
IN ts_orderline_053
INDEX IN ts_orderline_053
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 42225 ENDING AT 43036,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3695,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE54;
CREATE TABLE ORDER_LINE54
(
 OL_DELIVERY_D   TIMESTAMP   NOT NULL,
 OL_AMOUNT       DECIMAL(6,2) NOT NULL,
 OL_I_ID        INTEGER     NOT NULL,
 OL_SUPPLY_W_ID  INTEGER     NOT NULL,
 OL_QUANTITY     SMALLINT    NOT NULL,
 OL_DIST_INFO    CHAR(24)    NOT NULL,
 OL_O_ID        INTEGER     NOT NULL,
 OL_D_ID        SMALLINT    NOT NULL,
 OL_W_ID        INTEGER     NOT NULL,
 OL_NUMBER      SMALLINT    NOT NULL
)
IN ts_orderline_054
INDEX IN ts_orderline_054
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 43037 ENDING AT 43848,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3695,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE55;
CREATE TABLE ORDER_LINE55
(
 OL_DELIVERY_D   TIMESTAMP   NOT NULL,
 OL_AMOUNT       DECIMAL(6,2) NOT NULL,
 OL_I_ID        INTEGER     NOT NULL,
 OL_SUPPLY_W_ID  INTEGER     NOT NULL,
 OL_QUANTITY     SMALLINT    NOT NULL,
 OL_DIST_INFO    CHAR(24)    NOT NULL,
 OL_O_ID        INTEGER     NOT NULL,
 OL_D_ID        SMALLINT    NOT NULL,
 OL_W_ID        INTEGER     NOT NULL,
 OL_NUMBER      SMALLINT    NOT NULL
)
IN ts_orderline_055
INDEX IN ts_orderline_055
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 43849 ENDING AT 44660,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3695,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE56;
CREATE TABLE ORDER_LINE56
(
 OL_DELIVERY_D   TIMESTAMP   NOT NULL,
 OL_AMOUNT       DECIMAL(6,2) NOT NULL,
 OL_I_ID        INTEGER     NOT NULL,
 OL_SUPPLY_W_ID  INTEGER     NOT NULL,
 OL_QUANTITY     SMALLINT    NOT NULL,
 OL_DIST_INFO    CHAR(24)    NOT NULL,
 OL_O_ID        INTEGER     NOT NULL,
 OL_D_ID        SMALLINT    NOT NULL,
 OL_W_ID        INTEGER     NOT NULL,
 OL_NUMBER      SMALLINT    NOT NULL
)
IN ts_orderline_056
INDEX IN ts_orderline_056
ORGANIZE BY KEY SEQUENCE (
 OL_W_ID STARTING FROM 44661 ENDING AT 45472,
 OL_D_ID STARTING FROM 1 ENDING AT 10,
 OL_O_ID STARTING FROM 1 ENDING AT 3695,
 OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
```

```
connect to TPCC in share mode;
DROP TABLE ORDER_LINE57;
CREATE TABLE ORDER_LINE57
        (
         OL_DELIVERY_D   TIMESTAMP   NOT NULL,
         OL_AMOUNT       DECIMAL(6,2) NOT NULL,
         OL_I_ID         INTEGER     NOT NULL,
         OL_SUPPLY_W_ID  INTEGER     NOT NULL,
         OL_QUANTITY     SMALLINT    NOT NULL,
         OL_DIST_INFO    CHAR(24)    NOT NULL,
         OL_O_ID         INTEGER     NOT NULL,
         OL_D_ID         SMALLINT    NOT NULL,
         OL_W_ID         INTEGER     NOT NULL,
         OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_057
        INDEX IN ts_orderline_057
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 45473 ENDING AT 46284,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3695,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE58;
CREATE TABLE ORDER_LINE58
        (
         OL_DELIVERY_D   TIMESTAMP   NOT NULL,
         OL_AMOUNT       DECIMAL(6,2) NOT NULL,
         OL_I_ID         INTEGER     NOT NULL,
         OL_SUPPLY_W_ID  INTEGER     NOT NULL,
         OL_QUANTITY     SMALLINT    NOT NULL,
         OL_DIST_INFO    CHAR(24)    NOT NULL,
         OL_O_ID         INTEGER     NOT NULL,
         OL_D_ID         SMALLINT    NOT NULL,
         OL_W_ID         INTEGER     NOT NULL,
         OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_058
        INDEX IN ts_orderline_058
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 46285 ENDING AT 47096,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3695,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE59;
CREATE TABLE ORDER_LINE59
        (
         OL_DELIVERY_D   TIMESTAMP   NOT NULL,
         OL_AMOUNT       DECIMAL(6,2) NOT NULL,
         OL_I_ID         INTEGER     NOT NULL,
         OL_SUPPLY_W_ID  INTEGER     NOT NULL,
         OL_QUANTITY     SMALLINT    NOT NULL,
         OL_DIST_INFO    CHAR(24)    NOT NULL,
         OL_O_ID         INTEGER     NOT NULL,
         OL_D_ID         SMALLINT    NOT NULL,
         OL_W_ID         INTEGER     NOT NULL,
         OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_059
        INDEX IN ts_orderline_059
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 47097 ENDING AT 47908,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3695,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE60;
CREATE TABLE ORDER_LINE60
        (
         OL_DELIVERY_D   TIMESTAMP   NOT NULL,
         OL_AMOUNT       DECIMAL(6,2) NOT NULL,
         OL_I_ID         INTEGER     NOT NULL,
         OL_SUPPLY_W_ID  INTEGER     NOT NULL,
         OL_QUANTITY     SMALLINT    NOT NULL,
         OL_DIST_INFO    CHAR(24)    NOT NULL,
         OL_O_ID         INTEGER     NOT NULL,
         OL_D_ID         SMALLINT    NOT NULL,
         OL_W_ID         INTEGER     NOT NULL,
         OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_060
        INDEX IN ts_orderline_060
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 47909 ENDING AT 48720,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3695,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE61;
CREATE TABLE ORDER_LINE61
        (
         OL_DELIVERY_D   TIMESTAMP   NOT NULL,
         OL_AMOUNT       DECIMAL(6,2) NOT NULL,
         OL_I_ID         INTEGER     NOT NULL,
         OL_SUPPLY_W_ID  INTEGER     NOT NULL,
         OL_QUANTITY     SMALLINT    NOT NULL,
         OL_DIST_INFO    CHAR(24)    NOT NULL,
         OL_O_ID         INTEGER     NOT NULL,
         OL_D_ID         SMALLINT    NOT NULL,
         OL_W_ID         INTEGER     NOT NULL,
         OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_061
        INDEX IN ts_orderline_061
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 48721 ENDING AT 49532,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3695,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE62;
CREATE TABLE ORDER_LINE62
        (
         OL_DELIVERY_D   TIMESTAMP   NOT NULL,
         OL_AMOUNT       DECIMAL(6,2) NOT NULL,
         OL_I_ID         INTEGER     NOT NULL,
         OL_SUPPLY_W_ID  INTEGER     NOT NULL,
         OL_QUANTITY     SMALLINT    NOT NULL,
         OL_DIST_INFO    CHAR(24)    NOT NULL,
         OL_O_ID         INTEGER     NOT NULL,
         OL_D_ID         SMALLINT    NOT NULL,
         OL_W_ID         INTEGER     NOT NULL,
         OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_062
        INDEX IN ts_orderline_062
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 49533 ENDING AT 50344,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3695,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE63;
CREATE TABLE ORDER_LINE63
        (
         OL_DELIVERY_D   TIMESTAMP   NOT NULL,
         OL_AMOUNT       DECIMAL(6,2) NOT NULL,
         OL_I_ID         INTEGER     NOT NULL,
         OL_SUPPLY_W_ID  INTEGER     NOT NULL,
         OL_QUANTITY     SMALLINT    NOT NULL,
         OL_DIST_INFO    CHAR(24)    NOT NULL,
         OL_O_ID         INTEGER     NOT NULL,
         OL_D_ID         SMALLINT    NOT NULL,
         OL_W_ID         INTEGER     NOT NULL,
         OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_063
        INDEX IN ts_orderline_063
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 50345 ENDING AT 51156,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3695,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE64;
CREATE TABLE ORDER_LINE64
        (
         OL_DELIVERY_D   TIMESTAMP   NOT NULL,
         OL_AMOUNT       DECIMAL(6,2) NOT NULL,
         OL_I_ID         INTEGER     NOT NULL,
         OL_SUPPLY_W_ID  INTEGER     NOT NULL,
         OL_QUANTITY     SMALLINT    NOT NULL,
         OL_DIST_INFO    CHAR(24)    NOT NULL,
         OL_O_ID         INTEGER     NOT NULL,
         OL_D_ID         SMALLINT    NOT NULL,
         OL_W_ID         INTEGER     NOT NULL,
         OL_NUMBER       SMALLINT    NOT NULL
        )
        IN ts_orderline_064
        INDEX IN ts_orderline_064
        ORGANIZE BY KEY SEQUENCE (
         OL_W_ID STARTING FROM 51157 ENDING AT 51968,
         OL_D_ID STARTING FROM 1 ENDING AT 10,
         OL_O_ID STARTING FROM 1 ENDING AT 3695,
         OL_NUMBER STARTING FROM 1 ENDING AT 15
        )
        ALLOW OVERFLOW;
connect reset;
```

DDL/CRTB_STOCK.ddl

```
connect to TPCC in share mode;
DROP TABLE STOCK1;
CREATE TABLE STOCK1
        (
         S_REMOTE_CNT    INTEGER     NOT NULL,
         S_QUANTITY      INTEGER     NOT NULL,
         S_ORDER_CNT     INTEGER     NOT NULL,
         S_YTD           INTEGER     NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)    NOT NULL,
         S_DIST_02       CHAR(24)    NOT NULL,
         S_DIST_03       CHAR(24)    NOT NULL,
```

```
        S_DIST_04      CHAR(24)    NOT NULL,
        S_DIST_05      CHAR(24)    NOT NULL,
        S_DIST_06      CHAR(24)    NOT NULL,
        S_DIST_07      CHAR(24)    NOT NULL,
        S_DIST_08      CHAR(24)    NOT NULL,
        S_DIST_09      CHAR(24)    NOT NULL,
        S_DIST_10      CHAR(24)    NOT NULL,
        S_I_ID         INTEGER     NOT NULL,
        S_W_ID         INTEGER     NOT NULL
        )
        IN ts_stock_001
        INDEX IN ts_stock_001
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 1 ENDING AT 812
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK2;
CREATE TABLE STOCK2
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
        )
        IN ts_stock_002
        INDEX IN ts_stock_002
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 813 ENDING AT 1624
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK3;
CREATE TABLE STOCK3
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
        )
        IN ts_stock_003
        INDEX IN ts_stock_003
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 1625 ENDING AT 2436
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK4;
CREATE TABLE STOCK4
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
        )
        IN ts_stock_004
        INDEX IN ts_stock_004
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 2437 ENDING AT 3248
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK5;
CREATE TABLE STOCK5
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
        )
        IN ts_stock_005
        INDEX IN ts_stock_005
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 3249 ENDING AT 4060
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK6;
CREATE TABLE STOCK6
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
        )
        IN ts_stock_006
        INDEX IN ts_stock_006
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 4061 ENDING AT 4872
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK7;
CREATE TABLE STOCK7
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
        )
        IN ts_stock_007
        INDEX IN ts_stock_007
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 4873 ENDING AT 5684
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK8;
CREATE TABLE STOCK8
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
```

```
)
IN ts_stock_008
INDEX IN ts_stock_008
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 5685 ENDING AT 6496
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK9;
CREATE TABLE STOCK9
(
 S_REMOTE_CNT    INTEGER    NOT NULL,
 S_QUANTITY      INTEGER    NOT NULL,
 S_ORDER_CNT     INTEGER    NOT NULL,
 S_YTD           INTEGER    NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)   NOT NULL,
 S_DIST_02       CHAR(24)   NOT NULL,
 S_DIST_03       CHAR(24)   NOT NULL,
 S_DIST_04       CHAR(24)   NOT NULL,
 S_DIST_05       CHAR(24)   NOT NULL,
 S_DIST_06       CHAR(24)   NOT NULL,
 S_DIST_07       CHAR(24)   NOT NULL,
 S_DIST_08       CHAR(24)   NOT NULL,
 S_DIST_09       CHAR(24)   NOT NULL,
 S_DIST_10       CHAR(24)   NOT NULL,
 S_I_ID          INTEGER    NOT NULL,
 S_W_ID          INTEGER    NOT NULL
)
IN ts_stock_009
INDEX IN ts_stock_009
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 6497 ENDING AT 7308
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK10;
CREATE TABLE STOCK10
(
 S_REMOTE_CNT    INTEGER    NOT NULL,
 S_QUANTITY      INTEGER    NOT NULL,
 S_ORDER_CNT     INTEGER    NOT NULL,
 S_YTD           INTEGER    NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)   NOT NULL,
 S_DIST_02       CHAR(24)   NOT NULL,
 S_DIST_03       CHAR(24)   NOT NULL,
 S_DIST_04       CHAR(24)   NOT NULL,
 S_DIST_05       CHAR(24)   NOT NULL,
 S_DIST_06       CHAR(24)   NOT NULL,
 S_DIST_07       CHAR(24)   NOT NULL,
 S_DIST_08       CHAR(24)   NOT NULL,
 S_DIST_09       CHAR(24)   NOT NULL,
 S_DIST_10       CHAR(24)   NOT NULL,
 S_I_ID          INTEGER    NOT NULL,
 S_W_ID          INTEGER    NOT NULL
)
IN ts_stock_010
INDEX IN ts_stock_010
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 7309 ENDING AT 8120
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK11;
CREATE TABLE STOCK11
(
 S_REMOTE_CNT    INTEGER    NOT NULL,
 S_QUANTITY      INTEGER    NOT NULL,
 S_ORDER_CNT     INTEGER    NOT NULL,
 S_YTD           INTEGER    NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)   NOT NULL,
 S_DIST_02       CHAR(24)   NOT NULL,
 S_DIST_03       CHAR(24)   NOT NULL,
 S_DIST_04       CHAR(24)   NOT NULL,
 S_DIST_05       CHAR(24)   NOT NULL,
 S_DIST_06       CHAR(24)   NOT NULL,
 S_DIST_07       CHAR(24)   NOT NULL,
 S_DIST_08       CHAR(24)   NOT NULL,
 S_DIST_09       CHAR(24)   NOT NULL,
 S_DIST_10       CHAR(24)   NOT NULL,
 S_I_ID          INTEGER    NOT NULL,
 S_W_ID          INTEGER    NOT NULL
)
IN ts_stock_011
INDEX IN ts_stock_011
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 8121 ENDING AT 8932
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK12;
CREATE TABLE STOCK12
(
 S_REMOTE_CNT    INTEGER    NOT NULL,
 S_QUANTITY      INTEGER    NOT NULL,
 S_ORDER_CNT     INTEGER    NOT NULL,
 S_YTD           INTEGER    NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)   NOT NULL,
 S_DIST_02       CHAR(24)   NOT NULL,
 S_DIST_03       CHAR(24)   NOT NULL,
 S_DIST_04       CHAR(24)   NOT NULL,
 S_DIST_05       CHAR(24)   NOT NULL,
 S_DIST_06       CHAR(24)   NOT NULL,
 S_DIST_07       CHAR(24)   NOT NULL,
 S_DIST_08       CHAR(24)   NOT NULL,
 S_DIST_09       CHAR(24)   NOT NULL,
 S_DIST_10       CHAR(24)   NOT NULL,
 S_I_ID          INTEGER    NOT NULL,
 S_W_ID          INTEGER    NOT NULL
)
IN ts_stock_012
INDEX IN ts_stock_012
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 8933 ENDING AT 9744
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK13;
CREATE TABLE STOCK13
(
 S_REMOTE_CNT    INTEGER    NOT NULL,
 S_QUANTITY      INTEGER    NOT NULL,
 S_ORDER_CNT     INTEGER    NOT NULL,
 S_YTD           INTEGER    NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)   NOT NULL,
 S_DIST_02       CHAR(24)   NOT NULL,
 S_DIST_03       CHAR(24)   NOT NULL,
 S_DIST_04       CHAR(24)   NOT NULL,
 S_DIST_05       CHAR(24)   NOT NULL,
 S_DIST_06       CHAR(24)   NOT NULL,
 S_DIST_07       CHAR(24)   NOT NULL,
 S_DIST_08       CHAR(24)   NOT NULL,
 S_DIST_09       CHAR(24)   NOT NULL,
 S_DIST_10       CHAR(24)   NOT NULL,
 S_I_ID          INTEGER    NOT NULL,
 S_W_ID          INTEGER    NOT NULL
)
IN ts_stock_013
INDEX IN ts_stock_013
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 9745 ENDING AT 10556
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK14;
CREATE TABLE STOCK14
(
 S_REMOTE_CNT    INTEGER    NOT NULL,
 S_QUANTITY      INTEGER    NOT NULL,
 S_ORDER_CNT     INTEGER    NOT NULL,
 S_YTD           INTEGER    NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)   NOT NULL,
 S_DIST_02       CHAR(24)   NOT NULL,
 S_DIST_03       CHAR(24)   NOT NULL,
 S_DIST_04       CHAR(24)   NOT NULL,
 S_DIST_05       CHAR(24)   NOT NULL,
 S_DIST_06       CHAR(24)   NOT NULL,
 S_DIST_07       CHAR(24)   NOT NULL,
 S_DIST_08       CHAR(24)   NOT NULL,
 S_DIST_09       CHAR(24)   NOT NULL,
 S_DIST_10       CHAR(24)   NOT NULL,
 S_I_ID          INTEGER    NOT NULL,
 S_W_ID          INTEGER    NOT NULL
)
IN ts_stock_014
INDEX IN ts_stock_014
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 10557 ENDING AT 11368
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK15;
CREATE TABLE STOCK15
(
 S_REMOTE_CNT    INTEGER    NOT NULL,
 S_QUANTITY      INTEGER    NOT NULL,
 S_ORDER_CNT     INTEGER    NOT NULL,
 S_YTD           INTEGER    NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)   NOT NULL,
 S_DIST_02       CHAR(24)   NOT NULL,
 S_DIST_03       CHAR(24)   NOT NULL,
 S_DIST_04       CHAR(24)   NOT NULL,
 S_DIST_05       CHAR(24)   NOT NULL,
 S_DIST_06       CHAR(24)   NOT NULL,
 S_DIST_07       CHAR(24)   NOT NULL,
 S_DIST_08       CHAR(24)   NOT NULL,
 S_DIST_09       CHAR(24)   NOT NULL,
 S_DIST_10       CHAR(24)   NOT NULL,
 S_I_ID          INTEGER    NOT NULL,
 S_W_ID          INTEGER    NOT NULL
)
IN ts_stock_015
INDEX IN ts_stock_015
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 11369 ENDING AT 12180
)
ALLOW OVERFLOW;
connect reset;
```

```
connect to TPCC in share mode;
DROP TABLE STOCK16;
CREATE TABLE STOCK16
        (
         S_REMOTE_CNT    INTEGER    NOT NULL,
         S_QUANTITY      INTEGER    NOT NULL,
         S_ORDER_CNT     INTEGER    NOT NULL,
         S_YTD           INTEGER    NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)   NOT NULL,
         S_DIST_02       CHAR(24)   NOT NULL,
         S_DIST_03       CHAR(24)   NOT NULL,
         S_DIST_04       CHAR(24)   NOT NULL,
         S_DIST_05       CHAR(24)   NOT NULL,
         S_DIST_06       CHAR(24)   NOT NULL,
         S_DIST_07       CHAR(24)   NOT NULL,
         S_DIST_08       CHAR(24)   NOT NULL,
         S_DIST_09       CHAR(24)   NOT NULL,
         S_DIST_10       CHAR(24)   NOT NULL,
         S_I_ID          INTEGER    NOT NULL,
         S_W_ID          INTEGER    NOT NULL
        )
        IN ts_stock_016
        INDEX IN ts_stock_016
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 12181 ENDING AT 12992
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK17;
CREATE TABLE STOCK17
        (
         S_REMOTE_CNT    INTEGER    NOT NULL,
         S_QUANTITY      INTEGER    NOT NULL,
         S_ORDER_CNT     INTEGER    NOT NULL,
         S_YTD           INTEGER    NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)   NOT NULL,
         S_DIST_02       CHAR(24)   NOT NULL,
         S_DIST_03       CHAR(24)   NOT NULL,
         S_DIST_04       CHAR(24)   NOT NULL,
         S_DIST_05       CHAR(24)   NOT NULL,
         S_DIST_06       CHAR(24)   NOT NULL,
         S_DIST_07       CHAR(24)   NOT NULL,
         S_DIST_08       CHAR(24)   NOT NULL,
         S_DIST_09       CHAR(24)   NOT NULL,
         S_DIST_10       CHAR(24)   NOT NULL,
         S_I_ID          INTEGER    NOT NULL,
         S_W_ID          INTEGER    NOT NULL
        )
        IN ts_stock_017
        INDEX IN ts_stock_017
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 12993 ENDING AT 13804
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK18;
CREATE TABLE STOCK18
        (
         S_REMOTE_CNT    INTEGER    NOT NULL,
         S_QUANTITY      INTEGER    NOT NULL,
         S_ORDER_CNT     INTEGER    NOT NULL,
         S_YTD           INTEGER    NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)   NOT NULL,
         S_DIST_02       CHAR(24)   NOT NULL,
         S_DIST_03       CHAR(24)   NOT NULL,
         S_DIST_04       CHAR(24)   NOT NULL,
         S_DIST_05       CHAR(24)   NOT NULL,
         S_DIST_06       CHAR(24)   NOT NULL,
         S_DIST_07       CHAR(24)   NOT NULL,
         S_DIST_08       CHAR(24)   NOT NULL,
         S_DIST_09       CHAR(24)   NOT NULL,
         S_DIST_10       CHAR(24)   NOT NULL,
         S_I_ID          INTEGER    NOT NULL,
         S_W_ID          INTEGER    NOT NULL
        )
        IN ts_stock_018
        INDEX IN ts_stock_018
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 13805 ENDING AT 14616
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK19;
CREATE TABLE STOCK19
        (
         S_REMOTE_CNT    INTEGER    NOT NULL,
         S_QUANTITY      INTEGER    NOT NULL,
         S_ORDER_CNT     INTEGER    NOT NULL,
         S_YTD           INTEGER    NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)   NOT NULL,
         S_DIST_02       CHAR(24)   NOT NULL,
         S_DIST_03       CHAR(24)   NOT NULL,
         S_DIST_04       CHAR(24)   NOT NULL,
         S_DIST_05       CHAR(24)   NOT NULL,
         S_DIST_06       CHAR(24)   NOT NULL,
         S_DIST_07       CHAR(24)   NOT NULL,
         S_DIST_08       CHAR(24)   NOT NULL,
         S_DIST_09       CHAR(24)   NOT NULL,
         S_DIST_10       CHAR(24)   NOT NULL,
         S_I_ID          INTEGER    NOT NULL,
         S_W_ID          INTEGER    NOT NULL
        )
        IN ts_stock_019
        INDEX IN ts_stock_019
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 14617 ENDING AT 15428
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK20;
CREATE TABLE STOCK20
        (
         S_REMOTE_CNT    INTEGER    NOT NULL,
         S_QUANTITY      INTEGER    NOT NULL,
         S_ORDER_CNT     INTEGER    NOT NULL,
         S_YTD           INTEGER    NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)   NOT NULL,
         S_DIST_02       CHAR(24)   NOT NULL,
         S_DIST_03       CHAR(24)   NOT NULL,
         S_DIST_04       CHAR(24)   NOT NULL,
         S_DIST_05       CHAR(24)   NOT NULL,
         S_DIST_06       CHAR(24)   NOT NULL,
         S_DIST_07       CHAR(24)   NOT NULL,
         S_DIST_08       CHAR(24)   NOT NULL,
         S_DIST_09       CHAR(24)   NOT NULL,
         S_DIST_10       CHAR(24)   NOT NULL,
         S_I_ID          INTEGER    NOT NULL,
         S_W_ID          INTEGER    NOT NULL
        )
        IN ts_stock_020
        INDEX IN ts_stock_020
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 15429 ENDING AT 16240
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK21;
CREATE TABLE STOCK21
        (
         S_REMOTE_CNT    INTEGER    NOT NULL,
         S_QUANTITY      INTEGER    NOT NULL,
         S_ORDER_CNT     INTEGER    NOT NULL,
         S_YTD           INTEGER    NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)   NOT NULL,
         S_DIST_02       CHAR(24)   NOT NULL,
         S_DIST_03       CHAR(24)   NOT NULL,
         S_DIST_04       CHAR(24)   NOT NULL,
         S_DIST_05       CHAR(24)   NOT NULL,
         S_DIST_06       CHAR(24)   NOT NULL,
         S_DIST_07       CHAR(24)   NOT NULL,
         S_DIST_08       CHAR(24)   NOT NULL,
         S_DIST_09       CHAR(24)   NOT NULL,
         S_DIST_10       CHAR(24)   NOT NULL,
         S_I_ID          INTEGER    NOT NULL,
         S_W_ID          INTEGER    NOT NULL
        )
        IN ts_stock_021
        INDEX IN ts_stock_021
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 16241 ENDING AT 17052
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK22;
CREATE TABLE STOCK22
        (
         S_REMOTE_CNT    INTEGER    NOT NULL,
         S_QUANTITY      INTEGER    NOT NULL,
         S_ORDER_CNT     INTEGER    NOT NULL,
         S_YTD           INTEGER    NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)   NOT NULL,
         S_DIST_02       CHAR(24)   NOT NULL,
         S_DIST_03       CHAR(24)   NOT NULL,
         S_DIST_04       CHAR(24)   NOT NULL,
         S_DIST_05       CHAR(24)   NOT NULL,
         S_DIST_06       CHAR(24)   NOT NULL,
         S_DIST_07       CHAR(24)   NOT NULL,
         S_DIST_08       CHAR(24)   NOT NULL,
         S_DIST_09       CHAR(24)   NOT NULL,
         S_DIST_10       CHAR(24)   NOT NULL,
         S_I_ID          INTEGER    NOT NULL,
         S_W_ID          INTEGER    NOT NULL
        )
        IN ts_stock_022
        INDEX IN ts_stock_022
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 17053 ENDING AT 17864
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK23;
CREATE TABLE STOCK23
        (
         S_REMOTE_CNT    INTEGER    NOT NULL,
         S_QUANTITY      INTEGER    NOT NULL,
         S_ORDER_CNT     INTEGER    NOT NULL,
         S_YTD           INTEGER    NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
```

```
        S_DIST_01      CHAR(24)    NOT NULL,
        S_DIST_02      CHAR(24)    NOT NULL,
        S_DIST_03      CHAR(24)    NOT NULL,
        S_DIST_04      CHAR(24)    NOT NULL,
        S_DIST_05      CHAR(24)    NOT NULL,
        S_DIST_06      CHAR(24)    NOT NULL,
        S_DIST_07      CHAR(24)    NOT NULL,
        S_DIST_08      CHAR(24)    NOT NULL,
        S_DIST_09      CHAR(24)    NOT NULL,
        S_DIST_10      CHAR(24)    NOT NULL,
        S_I_ID         INTEGER     NOT NULL,
        S_W_ID         INTEGER     NOT NULL
        )
        IN ts_stock_023
        INDEX IN ts_stock_023
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 17865 ENDING AT 18676
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK24;
CREATE TABLE STOCK24
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
        )
        IN ts_stock_024
        INDEX IN ts_stock_024
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 18677 ENDING AT 19488
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK25;
CREATE TABLE STOCK25
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
        )
        IN ts_stock_025
        INDEX IN ts_stock_025
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 19489 ENDING AT 20300
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK26;
CREATE TABLE STOCK26
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
        )
        IN ts_stock_026
        INDEX IN ts_stock_026
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 20301 ENDING AT 21112
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK27;
CREATE TABLE STOCK27
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
        )
        IN ts_stock_027
        INDEX IN ts_stock_027
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 21113 ENDING AT 21924
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK28;
CREATE TABLE STOCK28
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
        )
        IN ts_stock_028
        INDEX IN ts_stock_028
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 21925 ENDING AT 22736
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK29;
CREATE TABLE STOCK29
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
        S_DIST_10       CHAR(24)    NOT NULL,
        S_I_ID          INTEGER     NOT NULL,
        S_W_ID          INTEGER     NOT NULL
        )
        IN ts_stock_029
        INDEX IN ts_stock_029
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 22737 ENDING AT 23548
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK30;
CREATE TABLE STOCK30
        (
        S_REMOTE_CNT    INTEGER     NOT NULL,
        S_QUANTITY      INTEGER     NOT NULL,
        S_ORDER_CNT     INTEGER     NOT NULL,
        S_YTD           INTEGER     NOT NULL,
        S_DATA          VARCHAR(50) NOT NULL,
        S_DIST_01       CHAR(24)    NOT NULL,
        S_DIST_02       CHAR(24)    NOT NULL,
        S_DIST_03       CHAR(24)    NOT NULL,
        S_DIST_04       CHAR(24)    NOT NULL,
        S_DIST_05       CHAR(24)    NOT NULL,
        S_DIST_06       CHAR(24)    NOT NULL,
        S_DIST_07       CHAR(24)    NOT NULL,
        S_DIST_08       CHAR(24)    NOT NULL,
        S_DIST_09       CHAR(24)    NOT NULL,
```

```
        S_DIST_10      CHAR(24)   NOT NULL,
        S_I_ID         INTEGER    NOT NULL,
        S_W_ID         INTEGER    NOT NULL
        )
        IN ts_stock_030
        INDEX IN ts_stock_030
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 23549 ENDING AT 24360
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK31;
CREATE TABLE STOCK31
        (
        S_REMOTE_CNT   INTEGER    NOT NULL,
        S_QUANTITY     INTEGER    NOT NULL,
        S_ORDER_CNT    INTEGER    NOT NULL,
        S_YTD          INTEGER    NOT NULL,
        S_DATA         VARCHAR(50) NOT NULL,
        S_DIST_01      CHAR(24)   NOT NULL,
        S_DIST_02      CHAR(24)   NOT NULL,
        S_DIST_03      CHAR(24)   NOT NULL,
        S_DIST_04      CHAR(24)   NOT NULL,
        S_DIST_05      CHAR(24)   NOT NULL,
        S_DIST_06      CHAR(24)   NOT NULL,
        S_DIST_07      CHAR(24)   NOT NULL,
        S_DIST_08      CHAR(24)   NOT NULL,
        S_DIST_09      CHAR(24)   NOT NULL,
        S_DIST_10      CHAR(24)   NOT NULL,
        S_I_ID         INTEGER    NOT NULL,
        S_W_ID         INTEGER    NOT NULL
        )
        IN ts_stock_031
        INDEX IN ts_stock_031
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 24361 ENDING AT 25172
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK32;
CREATE TABLE STOCK32
        (
        S_REMOTE_CNT   INTEGER    NOT NULL,
        S_QUANTITY     INTEGER    NOT NULL,
        S_ORDER_CNT    INTEGER    NOT NULL,
        S_YTD          INTEGER    NOT NULL,
        S_DATA         VARCHAR(50) NOT NULL,
        S_DIST_01      CHAR(24)   NOT NULL,
        S_DIST_02      CHAR(24)   NOT NULL,
        S_DIST_03      CHAR(24)   NOT NULL,
        S_DIST_04      CHAR(24)   NOT NULL,
        S_DIST_05      CHAR(24)   NOT NULL,
        S_DIST_06      CHAR(24)   NOT NULL,
        S_DIST_07      CHAR(24)   NOT NULL,
        S_DIST_08      CHAR(24)   NOT NULL,
        S_DIST_09      CHAR(24)   NOT NULL,
        S_DIST_10      CHAR(24)   NOT NULL,
        S_I_ID         INTEGER    NOT NULL,
        S_W_ID         INTEGER    NOT NULL
        )
        IN ts_stock_032
        INDEX IN ts_stock_032
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 25173 ENDING AT 25984
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK33;
CREATE TABLE STOCK33
        (
        S_REMOTE_CNT   INTEGER    NOT NULL,
        S_QUANTITY     INTEGER    NOT NULL,
        S_ORDER_CNT    INTEGER    NOT NULL,
        S_YTD          INTEGER    NOT NULL,
        S_DATA         VARCHAR(50) NOT NULL,
        S_DIST_01      CHAR(24)   NOT NULL,
        S_DIST_02      CHAR(24)   NOT NULL,
        S_DIST_03      CHAR(24)   NOT NULL,
        S_DIST_04      CHAR(24)   NOT NULL,
        S_DIST_05      CHAR(24)   NOT NULL,
        S_DIST_06      CHAR(24)   NOT NULL,
        S_DIST_07      CHAR(24)   NOT NULL,
        S_DIST_08      CHAR(24)   NOT NULL,
        S_DIST_09      CHAR(24)   NOT NULL,
        S_DIST_10      CHAR(24)   NOT NULL,
        S_I_ID         INTEGER    NOT NULL,
        S_W_ID         INTEGER    NOT NULL
        )
        IN ts_stock_033
        INDEX IN ts_stock_033
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 25985 ENDING AT 26796
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK34;
CREATE TABLE STOCK34
        (
        S_REMOTE_CNT   INTEGER    NOT NULL,
        S_QUANTITY     INTEGER    NOT NULL,
        S_ORDER_CNT    INTEGER    NOT NULL,
        S_YTD          INTEGER    NOT NULL,
        S_DATA         VARCHAR(50) NOT NULL,
        S_DIST_01      CHAR(24)   NOT NULL,
        S_DIST_02      CHAR(24)   NOT NULL,
        S_DIST_03      CHAR(24)   NOT NULL,
        S_DIST_04      CHAR(24)   NOT NULL,
        S_DIST_05      CHAR(24)   NOT NULL,
        S_DIST_06      CHAR(24)   NOT NULL,
        S_DIST_07      CHAR(24)   NOT NULL,
        S_DIST_08      CHAR(24)   NOT NULL,
        S_DIST_09      CHAR(24)   NOT NULL,
        S_DIST_10      CHAR(24)   NOT NULL,
        S_I_ID         INTEGER    NOT NULL,
        S_W_ID         INTEGER    NOT NULL
        )
        IN ts_stock_034
        INDEX IN ts_stock_034
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 26797 ENDING AT 27608
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK35;
CREATE TABLE STOCK35
        (
        S_REMOTE_CNT   INTEGER    NOT NULL,
        S_QUANTITY     INTEGER    NOT NULL,
        S_ORDER_CNT    INTEGER    NOT NULL,
        S_YTD          INTEGER    NOT NULL,
        S_DATA         VARCHAR(50) NOT NULL,
        S_DIST_01      CHAR(24)   NOT NULL,
        S_DIST_02      CHAR(24)   NOT NULL,
        S_DIST_03      CHAR(24)   NOT NULL,
        S_DIST_04      CHAR(24)   NOT NULL,
        S_DIST_05      CHAR(24)   NOT NULL,
        S_DIST_06      CHAR(24)   NOT NULL,
        S_DIST_07      CHAR(24)   NOT NULL,
        S_DIST_08      CHAR(24)   NOT NULL,
        S_DIST_09      CHAR(24)   NOT NULL,
        S_DIST_10      CHAR(24)   NOT NULL,
        S_I_ID         INTEGER    NOT NULL,
        S_W_ID         INTEGER    NOT NULL
        )
        IN ts_stock_035
        INDEX IN ts_stock_035
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 27609 ENDING AT 28420
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK36;
CREATE TABLE STOCK36
        (
        S_REMOTE_CNT   INTEGER    NOT NULL,
        S_QUANTITY     INTEGER    NOT NULL,
        S_ORDER_CNT    INTEGER    NOT NULL,
        S_YTD          INTEGER    NOT NULL,
        S_DATA         VARCHAR(50) NOT NULL,
        S_DIST_01      CHAR(24)   NOT NULL,
        S_DIST_02      CHAR(24)   NOT NULL,
        S_DIST_03      CHAR(24)   NOT NULL,
        S_DIST_04      CHAR(24)   NOT NULL,
        S_DIST_05      CHAR(24)   NOT NULL,
        S_DIST_06      CHAR(24)   NOT NULL,
        S_DIST_07      CHAR(24)   NOT NULL,
        S_DIST_08      CHAR(24)   NOT NULL,
        S_DIST_09      CHAR(24)   NOT NULL,
        S_DIST_10      CHAR(24)   NOT NULL,
        S_I_ID         INTEGER    NOT NULL,
        S_W_ID         INTEGER    NOT NULL
        )
        IN ts_stock_036
        INDEX IN ts_stock_036
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 28421 ENDING AT 29232
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK37;
CREATE TABLE STOCK37
        (
        S_REMOTE_CNT   INTEGER    NOT NULL,
        S_QUANTITY     INTEGER    NOT NULL,
        S_ORDER_CNT    INTEGER    NOT NULL,
        S_YTD          INTEGER    NOT NULL,
        S_DATA         VARCHAR(50) NOT NULL,
        S_DIST_01      CHAR(24)   NOT NULL,
        S_DIST_02      CHAR(24)   NOT NULL,
        S_DIST_03      CHAR(24)   NOT NULL,
        S_DIST_04      CHAR(24)   NOT NULL,
        S_DIST_05      CHAR(24)   NOT NULL,
        S_DIST_06      CHAR(24)   NOT NULL,
        S_DIST_07      CHAR(24)   NOT NULL,
        S_DIST_08      CHAR(24)   NOT NULL,
        S_DIST_09      CHAR(24)   NOT NULL,
        S_DIST_10      CHAR(24)   NOT NULL,
        S_I_ID         INTEGER    NOT NULL,
        S_W_ID         INTEGER    NOT NULL
        )
        IN ts_stock_037
        INDEX IN ts_stock_037
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 29233 ENDING AT 30044
```

```
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK38;
CREATE TABLE STOCK38
        (
         S_REMOTE_CNT    INTEGER    NOT NULL,
         S_QUANTITY      INTEGER    NOT NULL,
         S_ORDER_CNT     INTEGER    NOT NULL,
         S_YTD           INTEGER    NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)   NOT NULL,
         S_DIST_02       CHAR(24)   NOT NULL,
         S_DIST_03       CHAR(24)   NOT NULL,
         S_DIST_04       CHAR(24)   NOT NULL,
         S_DIST_05       CHAR(24)   NOT NULL,
         S_DIST_06       CHAR(24)   NOT NULL,
         S_DIST_07       CHAR(24)   NOT NULL,
         S_DIST_08       CHAR(24)   NOT NULL,
         S_DIST_09       CHAR(24)   NOT NULL,
         S_DIST_10       CHAR(24)   NOT NULL,
         S_I_ID          INTEGER    NOT NULL,
         S_W_ID          INTEGER    NOT NULL
        )
        IN ts_stock_038
        INDEX IN ts_stock_038
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 30045 ENDING AT 30856
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK39;
CREATE TABLE STOCK39
        (
         S_REMOTE_CNT    INTEGER    NOT NULL,
         S_QUANTITY      INTEGER    NOT NULL,
         S_ORDER_CNT     INTEGER    NOT NULL,
         S_YTD           INTEGER    NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)   NOT NULL,
         S_DIST_02       CHAR(24)   NOT NULL,
         S_DIST_03       CHAR(24)   NOT NULL,
         S_DIST_04       CHAR(24)   NOT NULL,
         S_DIST_05       CHAR(24)   NOT NULL,
         S_DIST_06       CHAR(24)   NOT NULL,
         S_DIST_07       CHAR(24)   NOT NULL,
         S_DIST_08       CHAR(24)   NOT NULL,
         S_DIST_09       CHAR(24)   NOT NULL,
         S_DIST_10       CHAR(24)   NOT NULL,
         S_I_ID          INTEGER    NOT NULL,
         S_W_ID          INTEGER    NOT NULL
        )
        IN ts_stock_039
        INDEX IN ts_stock_039
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 30857 ENDING AT 31668
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK40;
CREATE TABLE STOCK40
        (
         S_REMOTE_CNT    INTEGER    NOT NULL,
         S_QUANTITY      INTEGER    NOT NULL,
         S_ORDER_CNT     INTEGER    NOT NULL,
         S_YTD           INTEGER    NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)   NOT NULL,
         S_DIST_02       CHAR(24)   NOT NULL,
         S_DIST_03       CHAR(24)   NOT NULL,
         S_DIST_04       CHAR(24)   NOT NULL,
         S_DIST_05       CHAR(24)   NOT NULL,
         S_DIST_06       CHAR(24)   NOT NULL,
         S_DIST_07       CHAR(24)   NOT NULL,
         S_DIST_08       CHAR(24)   NOT NULL,
         S_DIST_09       CHAR(24)   NOT NULL,
         S_DIST_10       CHAR(24)   NOT NULL,
         S_I_ID          INTEGER    NOT NULL,
         S_W_ID          INTEGER    NOT NULL
        )
        IN ts_stock_040
        INDEX IN ts_stock_040
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 31669 ENDING AT 32480
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK41;
CREATE TABLE STOCK41
        (
         S_REMOTE_CNT    INTEGER    NOT NULL,
         S_QUANTITY      INTEGER    NOT NULL,
         S_ORDER_CNT     INTEGER    NOT NULL,
         S_YTD           INTEGER    NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)   NOT NULL,
         S_DIST_02       CHAR(24)   NOT NULL,
         S_DIST_03       CHAR(24)   NOT NULL,
         S_DIST_04       CHAR(24)   NOT NULL,
         S_DIST_05       CHAR(24)   NOT NULL,
         S_DIST_06       CHAR(24)   NOT NULL,
         S_DIST_07       CHAR(24)   NOT NULL,
         S_DIST_08       CHAR(24)   NOT NULL,
         S_DIST_09       CHAR(24)   NOT NULL,
         S_DIST_10       CHAR(24)   NOT NULL,
         S_I_ID          INTEGER    NOT NULL,
         S_W_ID          INTEGER    NOT NULL
        )
        IN ts_stock_041
        INDEX IN ts_stock_041
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 32481 ENDING AT 33292
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK42;
CREATE TABLE STOCK42
        (
         S_REMOTE_CNT    INTEGER    NOT NULL,
         S_QUANTITY      INTEGER    NOT NULL,
         S_ORDER_CNT     INTEGER    NOT NULL,
         S_YTD           INTEGER    NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)   NOT NULL,
         S_DIST_02       CHAR(24)   NOT NULL,
         S_DIST_03       CHAR(24)   NOT NULL,
         S_DIST_04       CHAR(24)   NOT NULL,
         S_DIST_05       CHAR(24)   NOT NULL,
         S_DIST_06       CHAR(24)   NOT NULL,
         S_DIST_07       CHAR(24)   NOT NULL,
         S_DIST_08       CHAR(24)   NOT NULL,
         S_DIST_09       CHAR(24)   NOT NULL,
         S_DIST_10       CHAR(24)   NOT NULL,
         S_I_ID          INTEGER    NOT NULL,
         S_W_ID          INTEGER    NOT NULL
        )
        IN ts_stock_042
        INDEX IN ts_stock_042
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 33293 ENDING AT 34104
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK43;
CREATE TABLE STOCK43
        (
         S_REMOTE_CNT    INTEGER    NOT NULL,
         S_QUANTITY      INTEGER    NOT NULL,
         S_ORDER_CNT     INTEGER    NOT NULL,
         S_YTD           INTEGER    NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)   NOT NULL,
         S_DIST_02       CHAR(24)   NOT NULL,
         S_DIST_03       CHAR(24)   NOT NULL,
         S_DIST_04       CHAR(24)   NOT NULL,
         S_DIST_05       CHAR(24)   NOT NULL,
         S_DIST_06       CHAR(24)   NOT NULL,
         S_DIST_07       CHAR(24)   NOT NULL,
         S_DIST_08       CHAR(24)   NOT NULL,
         S_DIST_09       CHAR(24)   NOT NULL,
         S_DIST_10       CHAR(24)   NOT NULL,
         S_I_ID          INTEGER    NOT NULL,
         S_W_ID          INTEGER    NOT NULL
        )
        IN ts_stock_043
        INDEX IN ts_stock_043
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 34105 ENDING AT 34916
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK44;
CREATE TABLE STOCK44
        (
         S_REMOTE_CNT    INTEGER    NOT NULL,
         S_QUANTITY      INTEGER    NOT NULL,
         S_ORDER_CNT     INTEGER    NOT NULL,
         S_YTD           INTEGER    NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)   NOT NULL,
         S_DIST_02       CHAR(24)   NOT NULL,
         S_DIST_03       CHAR(24)   NOT NULL,
         S_DIST_04       CHAR(24)   NOT NULL,
         S_DIST_05       CHAR(24)   NOT NULL,
         S_DIST_06       CHAR(24)   NOT NULL,
         S_DIST_07       CHAR(24)   NOT NULL,
         S_DIST_08       CHAR(24)   NOT NULL,
         S_DIST_09       CHAR(24)   NOT NULL,
         S_DIST_10       CHAR(24)   NOT NULL,
         S_I_ID          INTEGER    NOT NULL,
         S_W_ID          INTEGER    NOT NULL
        )
        IN ts_stock_044
        INDEX IN ts_stock_044
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 34917 ENDING AT 35728
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK45;
CREATE TABLE STOCK45
        (
         S_REMOTE_CNT    INTEGER    NOT NULL,
         S_QUANTITY      INTEGER    NOT NULL,
```

```
        S_ORDER_CNT    INTEGER    NOT NULL,
        S_YTD          INTEGER    NOT NULL,
        S_DATA         VARCHAR(50) NOT NULL,
        S_DIST_01      CHAR(24)   NOT NULL,
        S_DIST_02      CHAR(24)   NOT NULL,
        S_DIST_03      CHAR(24)   NOT NULL,
        S_DIST_04      CHAR(24)   NOT NULL,
        S_DIST_05      CHAR(24)   NOT NULL,
        S_DIST_06      CHAR(24)   NOT NULL,
        S_DIST_07      CHAR(24)   NOT NULL,
        S_DIST_08      CHAR(24)   NOT NULL,
        S_DIST_09      CHAR(24)   NOT NULL,
        S_DIST_10      CHAR(24)   NOT NULL,
        S_I_ID         INTEGER    NOT NULL,
        S_W_ID         INTEGER    NOT NULL
        )
        IN ts_stock_045
        INDEX IN ts_stock_045
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 35729 ENDING AT 36540
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK46;
CREATE TABLE STOCK46
        (
        S_REMOTE_CNT   INTEGER    NOT NULL,
        S_QUANTITY     INTEGER    NOT NULL,
        S_ORDER_CNT    INTEGER    NOT NULL,
        S_YTD          INTEGER    NOT NULL,
        S_DATA         VARCHAR(50) NOT NULL,
        S_DIST_01      CHAR(24)   NOT NULL,
        S_DIST_02      CHAR(24)   NOT NULL,
        S_DIST_03      CHAR(24)   NOT NULL,
        S_DIST_04      CHAR(24)   NOT NULL,
        S_DIST_05      CHAR(24)   NOT NULL,
        S_DIST_06      CHAR(24)   NOT NULL,
        S_DIST_07      CHAR(24)   NOT NULL,
        S_DIST_08      CHAR(24)   NOT NULL,
        S_DIST_09      CHAR(24)   NOT NULL,
        S_DIST_10      CHAR(24)   NOT NULL,
        S_I_ID         INTEGER    NOT NULL,
        S_W_ID         INTEGER    NOT NULL
        )
        IN ts_stock_046
        INDEX IN ts_stock_046
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 36541 ENDING AT 37352
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK47;
CREATE TABLE STOCK47
        (
        S_REMOTE_CNT   INTEGER    NOT NULL,
        S_QUANTITY     INTEGER    NOT NULL,
        S_ORDER_CNT    INTEGER    NOT NULL,
        S_YTD          INTEGER    NOT NULL,
        S_DATA         VARCHAR(50) NOT NULL,
        S_DIST_01      CHAR(24)   NOT NULL,
        S_DIST_02      CHAR(24)   NOT NULL,
        S_DIST_03      CHAR(24)   NOT NULL,
        S_DIST_04      CHAR(24)   NOT NULL,
        S_DIST_05      CHAR(24)   NOT NULL,
        S_DIST_06      CHAR(24)   NOT NULL,
        S_DIST_07      CHAR(24)   NOT NULL,
        S_DIST_08      CHAR(24)   NOT NULL,
        S_DIST_09      CHAR(24)   NOT NULL,
        S_DIST_10      CHAR(24)   NOT NULL,
        S_I_ID         INTEGER    NOT NULL,
        S_W_ID         INTEGER    NOT NULL
        )
        IN ts_stock_047
        INDEX IN ts_stock_047
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 37353 ENDING AT 38164
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK48;
CREATE TABLE STOCK48
        (
        S_REMOTE_CNT   INTEGER    NOT NULL,
        S_QUANTITY     INTEGER    NOT NULL,
        S_ORDER_CNT    INTEGER    NOT NULL,
        S_YTD          INTEGER    NOT NULL,
        S_DATA         VARCHAR(50) NOT NULL,
        S_DIST_01      CHAR(24)   NOT NULL,
        S_DIST_02      CHAR(24)   NOT NULL,
        S_DIST_03      CHAR(24)   NOT NULL,
        S_DIST_04      CHAR(24)   NOT NULL,
        S_DIST_05      CHAR(24)   NOT NULL,
        S_DIST_06      CHAR(24)   NOT NULL,
        S_DIST_07      CHAR(24)   NOT NULL,
        S_DIST_08      CHAR(24)   NOT NULL,
        S_DIST_09      CHAR(24)   NOT NULL,
        S_DIST_10      CHAR(24)   NOT NULL,
        S_I_ID         INTEGER    NOT NULL,
        S_W_ID         INTEGER    NOT NULL
        )
        IN ts_stock_048
        INDEX IN ts_stock_048
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 38165 ENDING AT 38976
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK49;
CREATE TABLE STOCK49
        (
        S_REMOTE_CNT   INTEGER    NOT NULL,
        S_QUANTITY     INTEGER    NOT NULL,
        S_ORDER_CNT    INTEGER    NOT NULL,
        S_YTD          INTEGER    NOT NULL,
        S_DATA         VARCHAR(50) NOT NULL,
        S_DIST_01      CHAR(24)   NOT NULL,
        S_DIST_02      CHAR(24)   NOT NULL,
        S_DIST_03      CHAR(24)   NOT NULL,
        S_DIST_04      CHAR(24)   NOT NULL,
        S_DIST_05      CHAR(24)   NOT NULL,
        S_DIST_06      CHAR(24)   NOT NULL,
        S_DIST_07      CHAR(24)   NOT NULL,
        S_DIST_08      CHAR(24)   NOT NULL,
        S_DIST_09      CHAR(24)   NOT NULL,
        S_DIST_10      CHAR(24)   NOT NULL,
        S_I_ID         INTEGER    NOT NULL,
        S_W_ID         INTEGER    NOT NULL
        )
        IN ts_stock_049
        INDEX IN ts_stock_049
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 38977 ENDING AT 39788
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK50;
CREATE TABLE STOCK50
        (
        S_REMOTE_CNT   INTEGER    NOT NULL,
        S_QUANTITY     INTEGER    NOT NULL,
        S_ORDER_CNT    INTEGER    NOT NULL,
        S_YTD          INTEGER    NOT NULL,
        S_DATA         VARCHAR(50) NOT NULL,
        S_DIST_01      CHAR(24)   NOT NULL,
        S_DIST_02      CHAR(24)   NOT NULL,
        S_DIST_03      CHAR(24)   NOT NULL,
        S_DIST_04      CHAR(24)   NOT NULL,
        S_DIST_05      CHAR(24)   NOT NULL,
        S_DIST_06      CHAR(24)   NOT NULL,
        S_DIST_07      CHAR(24)   NOT NULL,
        S_DIST_08      CHAR(24)   NOT NULL,
        S_DIST_09      CHAR(24)   NOT NULL,
        S_DIST_10      CHAR(24)   NOT NULL,
        S_I_ID         INTEGER    NOT NULL,
        S_W_ID         INTEGER    NOT NULL
        )
        IN ts_stock_050
        INDEX IN ts_stock_050
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 39789 ENDING AT 40600
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK51;
CREATE TABLE STOCK51
        (
        S_REMOTE_CNT   INTEGER    NOT NULL,
        S_QUANTITY     INTEGER    NOT NULL,
        S_ORDER_CNT    INTEGER    NOT NULL,
        S_YTD          INTEGER    NOT NULL,
        S_DATA         VARCHAR(50) NOT NULL,
        S_DIST_01      CHAR(24)   NOT NULL,
        S_DIST_02      CHAR(24)   NOT NULL,
        S_DIST_03      CHAR(24)   NOT NULL,
        S_DIST_04      CHAR(24)   NOT NULL,
        S_DIST_05      CHAR(24)   NOT NULL,
        S_DIST_06      CHAR(24)   NOT NULL,
        S_DIST_07      CHAR(24)   NOT NULL,
        S_DIST_08      CHAR(24)   NOT NULL,
        S_DIST_09      CHAR(24)   NOT NULL,
        S_DIST_10      CHAR(24)   NOT NULL,
        S_I_ID         INTEGER    NOT NULL,
        S_W_ID         INTEGER    NOT NULL
        )
        IN ts_stock_051
        INDEX IN ts_stock_051
        ORGANIZE BY KEY SEQUENCE (
        S_I_ID STARTING FROM 1 ENDING AT 100000,
        S_W_ID STARTING FROM 40601 ENDING AT 41412
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK52;
CREATE TABLE STOCK52
        (
        S_REMOTE_CNT   INTEGER    NOT NULL,
        S_QUANTITY     INTEGER    NOT NULL,
        S_ORDER_CNT    INTEGER    NOT NULL,
        S_YTD          INTEGER    NOT NULL,
        S_DATA         VARCHAR(50) NOT NULL,
        S_DIST_01      CHAR(24)   NOT NULL,
        S_DIST_02      CHAR(24)   NOT NULL,
        S_DIST_03      CHAR(24)   NOT NULL,
        S_DIST_04      CHAR(24)   NOT NULL,
        S_DIST_05      CHAR(24)   NOT NULL,
        S_DIST_06      CHAR(24)   NOT NULL,
```

```
 S_DIST_07      CHAR(24)    NOT NULL,
 S_DIST_08      CHAR(24)    NOT NULL,
 S_DIST_09      CHAR(24)    NOT NULL,
 S_DIST_10      CHAR(24)    NOT NULL,
 S_I_ID         INTEGER     NOT NULL,
 S_W_ID         INTEGER     NOT NULL
)
IN ts_stock_052
INDEX IN ts_stock_052
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 41413 ENDING AT 42224
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK53;
CREATE TABLE STOCK53
(
 S_REMOTE_CNT    INTEGER     NOT NULL,
 S_QUANTITY      INTEGER     NOT NULL,
 S_ORDER_CNT     INTEGER     NOT NULL,
 S_YTD           INTEGER     NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)    NOT NULL,
 S_DIST_02       CHAR(24)    NOT NULL,
 S_DIST_03       CHAR(24)    NOT NULL,
 S_DIST_04       CHAR(24)    NOT NULL,
 S_DIST_05       CHAR(24)    NOT NULL,
 S_DIST_06       CHAR(24)    NOT NULL,
 S_DIST_07       CHAR(24)    NOT NULL,
 S_DIST_08       CHAR(24)    NOT NULL,
 S_DIST_09       CHAR(24)    NOT NULL,
 S_DIST_10       CHAR(24)    NOT NULL,
 S_I_ID          INTEGER     NOT NULL,
 S_W_ID          INTEGER     NOT NULL
)
IN ts_stock_053
INDEX IN ts_stock_053
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 42225 ENDING AT 43036
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK54;
CREATE TABLE STOCK54
(
 S_REMOTE_CNT    INTEGER     NOT NULL,
 S_QUANTITY      INTEGER     NOT NULL,
 S_ORDER_CNT     INTEGER     NOT NULL,
 S_YTD           INTEGER     NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)    NOT NULL,
 S_DIST_02       CHAR(24)    NOT NULL,
 S_DIST_03       CHAR(24)    NOT NULL,
 S_DIST_04       CHAR(24)    NOT NULL,
 S_DIST_05       CHAR(24)    NOT NULL,
 S_DIST_06       CHAR(24)    NOT NULL,
 S_DIST_07       CHAR(24)    NOT NULL,
 S_DIST_08       CHAR(24)    NOT NULL,
 S_DIST_09       CHAR(24)    NOT NULL,
 S_DIST_10       CHAR(24)    NOT NULL,
 S_I_ID          INTEGER     NOT NULL,
 S_W_ID          INTEGER     NOT NULL
)
IN ts_stock_054
INDEX IN ts_stock_054
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 43037 ENDING AT 43848
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK55;
CREATE TABLE STOCK55
(
 S_REMOTE_CNT    INTEGER     NOT NULL,
 S_QUANTITY      INTEGER     NOT NULL,
 S_ORDER_CNT     INTEGER     NOT NULL,
 S_YTD           INTEGER     NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)    NOT NULL,
 S_DIST_02       CHAR(24)    NOT NULL,
 S_DIST_03       CHAR(24)    NOT NULL,
 S_DIST_04       CHAR(24)    NOT NULL,
 S_DIST_05       CHAR(24)    NOT NULL,
 S_DIST_06       CHAR(24)    NOT NULL,
 S_DIST_07       CHAR(24)    NOT NULL,
 S_DIST_08       CHAR(24)    NOT NULL,
 S_DIST_09       CHAR(24)    NOT NULL,
 S_DIST_10       CHAR(24)    NOT NULL,
 S_I_ID          INTEGER     NOT NULL,
 S_W_ID          INTEGER     NOT NULL
)
IN ts_stock_055
INDEX IN ts_stock_055
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 43849 ENDING AT 44660
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK56;
CREATE TABLE STOCK56
(
 S_REMOTE_CNT    INTEGER     NOT NULL,
 S_QUANTITY      INTEGER     NOT NULL,
 S_ORDER_CNT     INTEGER     NOT NULL,
 S_YTD           INTEGER     NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)    NOT NULL,
 S_DIST_02       CHAR(24)    NOT NULL,
 S_DIST_03       CHAR(24)    NOT NULL,
 S_DIST_04       CHAR(24)    NOT NULL,
 S_DIST_05       CHAR(24)    NOT NULL,
 S_DIST_06       CHAR(24)    NOT NULL,
 S_DIST_07       CHAR(24)    NOT NULL,
 S_DIST_08       CHAR(24)    NOT NULL,
 S_DIST_09       CHAR(24)    NOT NULL,
 S_DIST_10       CHAR(24)    NOT NULL,
 S_I_ID          INTEGER     NOT NULL,
 S_W_ID          INTEGER     NOT NULL
)
IN ts_stock_056
INDEX IN ts_stock_056
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 44661 ENDING AT 45472
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK57;
CREATE TABLE STOCK57
(
 S_REMOTE_CNT    INTEGER     NOT NULL,
 S_QUANTITY      INTEGER     NOT NULL,
 S_ORDER_CNT     INTEGER     NOT NULL,
 S_YTD           INTEGER     NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)    NOT NULL,
 S_DIST_02       CHAR(24)    NOT NULL,
 S_DIST_03       CHAR(24)    NOT NULL,
 S_DIST_04       CHAR(24)    NOT NULL,
 S_DIST_05       CHAR(24)    NOT NULL,
 S_DIST_06       CHAR(24)    NOT NULL,
 S_DIST_07       CHAR(24)    NOT NULL,
 S_DIST_08       CHAR(24)    NOT NULL,
 S_DIST_09       CHAR(24)    NOT NULL,
 S_DIST_10       CHAR(24)    NOT NULL,
 S_I_ID          INTEGER     NOT NULL,
 S_W_ID          INTEGER     NOT NULL
)
IN ts_stock_057
INDEX IN ts_stock_057
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 45473 ENDING AT 46284
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK58;
CREATE TABLE STOCK58
(
 S_REMOTE_CNT    INTEGER     NOT NULL,
 S_QUANTITY      INTEGER     NOT NULL,
 S_ORDER_CNT     INTEGER     NOT NULL,
 S_YTD           INTEGER     NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)    NOT NULL,
 S_DIST_02       CHAR(24)    NOT NULL,
 S_DIST_03       CHAR(24)    NOT NULL,
 S_DIST_04       CHAR(24)    NOT NULL,
 S_DIST_05       CHAR(24)    NOT NULL,
 S_DIST_06       CHAR(24)    NOT NULL,
 S_DIST_07       CHAR(24)    NOT NULL,
 S_DIST_08       CHAR(24)    NOT NULL,
 S_DIST_09       CHAR(24)    NOT NULL,
 S_DIST_10       CHAR(24)    NOT NULL,
 S_I_ID          INTEGER     NOT NULL,
 S_W_ID          INTEGER     NOT NULL
)
IN ts_stock_058
INDEX IN ts_stock_058
ORGANIZE BY KEY SEQUENCE (
 S_I_ID STARTING FROM 1 ENDING AT 100000,
 S_W_ID STARTING FROM 46285 ENDING AT 47096
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK59;
CREATE TABLE STOCK59
(
 S_REMOTE_CNT    INTEGER     NOT NULL,
 S_QUANTITY      INTEGER     NOT NULL,
 S_ORDER_CNT     INTEGER     NOT NULL,
 S_YTD           INTEGER     NOT NULL,
 S_DATA          VARCHAR(50) NOT NULL,
 S_DIST_01       CHAR(24)    NOT NULL,
 S_DIST_02       CHAR(24)    NOT NULL,
 S_DIST_03       CHAR(24)    NOT NULL,
 S_DIST_04       CHAR(24)    NOT NULL,
 S_DIST_05       CHAR(24)    NOT NULL,
 S_DIST_06       CHAR(24)    NOT NULL,
 S_DIST_07       CHAR(24)    NOT NULL,
 S_DIST_08       CHAR(24)    NOT NULL,
 S_DIST_09       CHAR(24)    NOT NULL,
 S_DIST_10       CHAR(24)    NOT NULL,
 S_I_ID          INTEGER     NOT NULL,
 S_W_ID          INTEGER     NOT NULL
)
IN ts_stock_059
INDEX IN ts_stock_059
```

```
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 47097 ENDING AT 47908
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK60;
CREATE TABLE STOCK60
        (
         S_REMOTE_CNT    INTEGER    NOT NULL,
         S_QUANTITY      INTEGER    NOT NULL,
         S_ORDER_CNT     INTEGER    NOT NULL,
         S_YTD           INTEGER    NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)   NOT NULL,
         S_DIST_02       CHAR(24)   NOT NULL,
         S_DIST_03       CHAR(24)   NOT NULL,
         S_DIST_04       CHAR(24)   NOT NULL,
         S_DIST_05       CHAR(24)   NOT NULL,
         S_DIST_06       CHAR(24)   NOT NULL,
         S_DIST_07       CHAR(24)   NOT NULL,
         S_DIST_08       CHAR(24)   NOT NULL,
         S_DIST_09       CHAR(24)   NOT NULL,
         S_DIST_10       CHAR(24)   NOT NULL,
         S_I_ID          INTEGER    NOT NULL,
         S_W_ID          INTEGER    NOT NULL
        )
        IN ts_stock_060
        INDEX IN ts_stock_060
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 47909 ENDING AT 48720
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK61;
CREATE TABLE STOCK61
        (
         S_REMOTE_CNT    INTEGER    NOT NULL,
         S_QUANTITY      INTEGER    NOT NULL,
         S_ORDER_CNT     INTEGER    NOT NULL,
         S_YTD           INTEGER    NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)   NOT NULL,
         S_DIST_02       CHAR(24)   NOT NULL,
         S_DIST_03       CHAR(24)   NOT NULL,
         S_DIST_04       CHAR(24)   NOT NULL,
         S_DIST_05       CHAR(24)   NOT NULL,
         S_DIST_06       CHAR(24)   NOT NULL,
         S_DIST_07       CHAR(24)   NOT NULL,
         S_DIST_08       CHAR(24)   NOT NULL,
         S_DIST_09       CHAR(24)   NOT NULL,
         S_DIST_10       CHAR(24)   NOT NULL,
         S_I_ID          INTEGER    NOT NULL,
         S_W_ID          INTEGER    NOT NULL
        )
        IN ts_stock_061
        INDEX IN ts_stock_061
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 48721 ENDING AT 49532
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK62;
CREATE TABLE STOCK62
        (
         S_REMOTE_CNT    INTEGER    NOT NULL,
         S_QUANTITY      INTEGER    NOT NULL,
         S_ORDER_CNT     INTEGER    NOT NULL,
         S_YTD           INTEGER    NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)   NOT NULL,
         S_DIST_02       CHAR(24)   NOT NULL,
         S_DIST_03       CHAR(24)   NOT NULL,
         S_DIST_04       CHAR(24)   NOT NULL,
         S_DIST_05       CHAR(24)   NOT NULL,
         S_DIST_06       CHAR(24)   NOT NULL,
         S_DIST_07       CHAR(24)   NOT NULL,
         S_DIST_08       CHAR(24)   NOT NULL,
         S_DIST_09       CHAR(24)   NOT NULL,
         S_DIST_10       CHAR(24)   NOT NULL,
         S_I_ID          INTEGER    NOT NULL,
         S_W_ID          INTEGER    NOT NULL
        )
        IN ts_stock_062
        INDEX IN ts_stock_062
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 49533 ENDING AT 50344
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK63;
CREATE TABLE STOCK63
        (
         S_REMOTE_CNT    INTEGER    NOT NULL,
         S_QUANTITY      INTEGER    NOT NULL,
         S_ORDER_CNT     INTEGER    NOT NULL,
         S_YTD           INTEGER    NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)   NOT NULL,
         S_DIST_02       CHAR(24)   NOT NULL,
         S_DIST_03       CHAR(24)   NOT NULL,
         S_DIST_04       CHAR(24)   NOT NULL,
         S_DIST_05       CHAR(24)   NOT NULL,
         S_DIST_06       CHAR(24)   NOT NULL,
         S_DIST_07       CHAR(24)   NOT NULL,
         S_DIST_08       CHAR(24)   NOT NULL,
         S_DIST_09       CHAR(24)   NOT NULL,
         S_DIST_10       CHAR(24)   NOT NULL,
         S_I_ID          INTEGER    NOT NULL,
         S_W_ID          INTEGER    NOT NULL
        )
        IN ts_stock_063
        INDEX IN ts_stock_063
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 50345 ENDING AT 51156
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK64;
CREATE TABLE STOCK64
        (
         S_REMOTE_CNT    INTEGER    NOT NULL,
         S_QUANTITY      INTEGER    NOT NULL,
         S_ORDER_CNT     INTEGER    NOT NULL,
         S_YTD           INTEGER    NOT NULL,
         S_DATA          VARCHAR(50) NOT NULL,
         S_DIST_01       CHAR(24)   NOT NULL,
         S_DIST_02       CHAR(24)   NOT NULL,
         S_DIST_03       CHAR(24)   NOT NULL,
         S_DIST_04       CHAR(24)   NOT NULL,
         S_DIST_05       CHAR(24)   NOT NULL,
         S_DIST_06       CHAR(24)   NOT NULL,
         S_DIST_07       CHAR(24)   NOT NULL,
         S_DIST_08       CHAR(24)   NOT NULL,
         S_DIST_09       CHAR(24)   NOT NULL,
         S_DIST_10       CHAR(24)   NOT NULL,
         S_I_ID          INTEGER    NOT NULL,
         S_W_ID          INTEGER    NOT NULL
        )
        IN ts_stock_064
        INDEX IN ts_stock_064
        ORGANIZE BY KEY SEQUENCE (
         S_I_ID STARTING FROM 1 ENDING AT 100000,
         S_W_ID STARTING FROM 51157 ENDING AT 51968
        )
        ALLOW OVERFLOW;
connect reset;
```

DDL/CRTB_WAREHOUSE.ddl

```
connect to TPCC in share mode;
DROP TABLE WAREHOUSE1;
CREATE TABLE WAREHOUSE1
        (
         W_NAME      CHAR(10)       NOT NULL,
         W_STREET_1  CHAR(20)       NOT NULL,
         W_STREET_2  CHAR(20)       NOT NULL,
         W_CITY      CHAR(20)       NOT NULL,
         W_STATE     CHAR(2)        NOT NULL,
         W_ZIP       CHAR(9)        NOT NULL,
         W_TAX       REAL           NOT NULL,
         W_YTD       DECIMAL(12,2)  NOT NULL,
         W_ID        INTEGER        NOT NULL
        )
        IN ts_ware_001
        INDEX IN ts_ware_001
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 1 ENDING AT 1624
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE2;
CREATE TABLE WAREHOUSE2
        (
         W_NAME      CHAR(10)       NOT NULL,
         W_STREET_1  CHAR(20)       NOT NULL,
         W_STREET_2  CHAR(20)       NOT NULL,
         W_CITY      CHAR(20)       NOT NULL,
         W_STATE     CHAR(2)        NOT NULL,
         W_ZIP       CHAR(9)        NOT NULL,
         W_TAX       REAL           NOT NULL,
         W_YTD       DECIMAL(12,2)  NOT NULL,
         W_ID        INTEGER        NOT NULL
        )
        IN ts_ware_002
        INDEX IN ts_ware_002
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 1625 ENDING AT 3248
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE3;
CREATE TABLE WAREHOUSE3
        (
         W_NAME      CHAR(10)       NOT NULL,
         W_STREET_1  CHAR(20)       NOT NULL,
         W_STREET_2  CHAR(20)       NOT NULL,
         W_CITY      CHAR(20)       NOT NULL,
         W_STATE     CHAR(2)        NOT NULL,
         W_ZIP       CHAR(9)        NOT NULL,
         W_TAX       REAL           NOT NULL,
         W_YTD       DECIMAL(12,2)  NOT NULL,
         W_ID        INTEGER        NOT NULL
        )
```

```
IN ts_ware_003
INDEX IN ts_ware_003
ORGANIZE BY KEY SEQUENCE (
 W_ID STARTING FROM 3249 ENDING AT 4872
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE4;
CREATE TABLE WAREHOUSE4
(
 W_NAME     CHAR(10)       NOT NULL,
 W_STREET_1  CHAR(20)       NOT NULL,
 W_STREET_2  CHAR(20)       NOT NULL,
 W_CITY     CHAR(20)       NOT NULL,
 W_STATE    CHAR(2)        NOT NULL,
 W_ZIP      CHAR(9)        NOT NULL,
 W_TAX      REAL           NOT NULL,
 W_YTD      DECIMAL(12,2)  NOT NULL,
 W_ID       INTEGER        NOT NULL
)
IN ts_ware_004
INDEX IN ts_ware_004
ORGANIZE BY KEY SEQUENCE (
 W_ID STARTING FROM 4873 ENDING AT 6496
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE5;
CREATE TABLE WAREHOUSE5
(
 W_NAME     CHAR(10)       NOT NULL,
 W_STREET_1  CHAR(20)       NOT NULL,
 W_STREET_2  CHAR(20)       NOT NULL,
 W_CITY     CHAR(20)       NOT NULL,
 W_STATE    CHAR(2)        NOT NULL,
 W_ZIP      CHAR(9)        NOT NULL,
 W_TAX      REAL           NOT NULL,
 W_YTD      DECIMAL(12,2)  NOT NULL,
 W_ID       INTEGER        NOT NULL
)
IN ts_ware_005
INDEX IN ts_ware_005
ORGANIZE BY KEY SEQUENCE (
 W_ID STARTING FROM 6497 ENDING AT 8120
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE6;
CREATE TABLE WAREHOUSE6
(
 W_NAME     CHAR(10)       NOT NULL,
 W_STREET_1  CHAR(20)       NOT NULL,
 W_STREET_2  CHAR(20)       NOT NULL,
 W_CITY     CHAR(20)       NOT NULL,
 W_STATE    CHAR(2)        NOT NULL,
 W_ZIP      CHAR(9)        NOT NULL,
 W_TAX      REAL           NOT NULL,
 W_YTD      DECIMAL(12,2)  NOT NULL,
 W_ID       INTEGER        NOT NULL
)
IN ts_ware_006
INDEX IN ts_ware_006
ORGANIZE BY KEY SEQUENCE (
 W_ID STARTING FROM 8121 ENDING AT 9744
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE7;
CREATE TABLE WAREHOUSE7
(
 W_NAME     CHAR(10)       NOT NULL,
 W_STREET_1  CHAR(20)       NOT NULL,
 W_STREET_2  CHAR(20)       NOT NULL,
 W_CITY     CHAR(20)       NOT NULL,
 W_STATE    CHAR(2)        NOT NULL,
 W_ZIP      CHAR(9)        NOT NULL,
 W_TAX      REAL           NOT NULL,
 W_YTD      DECIMAL(12,2)  NOT NULL,
 W_ID       INTEGER        NOT NULL
)
IN ts_ware_007
INDEX IN ts_ware_007
ORGANIZE BY KEY SEQUENCE (
 W_ID STARTING FROM 9745 ENDING AT 11368
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE8;
CREATE TABLE WAREHOUSE8
(
 W_NAME     CHAR(10)       NOT NULL,
 W_STREET_1  CHAR(20)       NOT NULL,
 W_STREET_2  CHAR(20)       NOT NULL,
 W_CITY     CHAR(20)       NOT NULL,
 W_STATE    CHAR(2)        NOT NULL,
 W_ZIP      CHAR(9)        NOT NULL,
 W_TAX      REAL           NOT NULL,
 W_YTD      DECIMAL(12,2)  NOT NULL,
 W_ID       INTEGER        NOT NULL
)
IN ts_ware_008
INDEX IN ts_ware_008
ORGANIZE BY KEY SEQUENCE (
 W_ID STARTING FROM 11369 ENDING AT 12992
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE9;
CREATE TABLE WAREHOUSE9
(
 W_NAME     CHAR(10)       NOT NULL,
 W_STREET_1  CHAR(20)       NOT NULL,
 W_STREET_2  CHAR(20)       NOT NULL,
 W_CITY     CHAR(20)       NOT NULL,
 W_STATE    CHAR(2)        NOT NULL,
 W_ZIP      CHAR(9)        NOT NULL,
 W_TAX      REAL           NOT NULL,
 W_YTD      DECIMAL(12,2)  NOT NULL,
 W_ID       INTEGER        NOT NULL
)
IN ts_ware_009
INDEX IN ts_ware_009
ORGANIZE BY KEY SEQUENCE (
 W_ID STARTING FROM 12993 ENDING AT 14616
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE10;
CREATE TABLE WAREHOUSE10
(
 W_NAME     CHAR(10)       NOT NULL,
 W_STREET_1  CHAR(20)       NOT NULL,
 W_STREET_2  CHAR(20)       NOT NULL,
 W_CITY     CHAR(20)       NOT NULL,
 W_STATE    CHAR(2)        NOT NULL,
 W_ZIP      CHAR(9)        NOT NULL,
 W_TAX      REAL           NOT NULL,
 W_YTD      DECIMAL(12,2)  NOT NULL,
 W_ID       INTEGER        NOT NULL
)
IN ts_ware_010
INDEX IN ts_ware_010
ORGANIZE BY KEY SEQUENCE (
 W_ID STARTING FROM 14617 ENDING AT 16240
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE11;
CREATE TABLE WAREHOUSE11
(
 W_NAME     CHAR(10)       NOT NULL,
 W_STREET_1  CHAR(20)       NOT NULL,
 W_STREET_2  CHAR(20)       NOT NULL,
 W_CITY     CHAR(20)       NOT NULL,
 W_STATE    CHAR(2)        NOT NULL,
 W_ZIP      CHAR(9)        NOT NULL,
 W_TAX      REAL           NOT NULL,
 W_YTD      DECIMAL(12,2)  NOT NULL,
 W_ID       INTEGER        NOT NULL
)
IN ts_ware_011
INDEX IN ts_ware_011
ORGANIZE BY KEY SEQUENCE (
 W_ID STARTING FROM 16241 ENDING AT 17864
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE12;
CREATE TABLE WAREHOUSE12
(
 W_NAME     CHAR(10)       NOT NULL,
 W_STREET_1  CHAR(20)       NOT NULL,
 W_STREET_2  CHAR(20)       NOT NULL,
 W_CITY     CHAR(20)       NOT NULL,
 W_STATE    CHAR(2)        NOT NULL,
 W_ZIP      CHAR(9)        NOT NULL,
 W_TAX      REAL           NOT NULL,
 W_YTD      DECIMAL(12,2)  NOT NULL,
 W_ID       INTEGER        NOT NULL
)
IN ts_ware_012
INDEX IN ts_ware_012
ORGANIZE BY KEY SEQUENCE (
 W_ID STARTING FROM 17865 ENDING AT 19488
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE13;
CREATE TABLE WAREHOUSE13
(
 W_NAME     CHAR(10)       NOT NULL,
 W_STREET_1  CHAR(20)       NOT NULL,
 W_STREET_2  CHAR(20)       NOT NULL,
 W_CITY     CHAR(20)       NOT NULL,
 W_STATE    CHAR(2)        NOT NULL,
 W_ZIP      CHAR(9)        NOT NULL,
 W_TAX      REAL           NOT NULL,
 W_YTD      DECIMAL(12,2)  NOT NULL,
 W_ID       INTEGER        NOT NULL
)
IN ts_ware_013
INDEX IN ts_ware_013
ORGANIZE BY KEY SEQUENCE (
 W_ID STARTING FROM 19489 ENDING AT 21112
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE14;
```

```
CREATE TABLE WAREHOUSE14
        (
         W_NAME     CHAR(10)       NOT NULL,
         W_STREET_1  CHAR(20)       NOT NULL,
         W_STREET_2  CHAR(20)       NOT NULL,
         W_CITY     CHAR(20)       NOT NULL,
         W_STATE    CHAR(2)        NOT NULL,
         W_ZIP      CHAR(9)        NOT NULL,
         W_TAX      REAL           NOT NULL,
         W_YTD      DECIMAL(12,2)  NOT NULL,
         W_ID       INTEGER        NOT NULL
        )
        IN ts_ware_014
        INDEX IN ts_ware_014
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 21113 ENDING AT 22736
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE15;
CREATE TABLE WAREHOUSE15
        (
         W_NAME     CHAR(10)       NOT NULL,
         W_STREET_1  CHAR(20)       NOT NULL,
         W_STREET_2  CHAR(20)       NOT NULL,
         W_CITY     CHAR(20)       NOT NULL,
         W_STATE    CHAR(2)        NOT NULL,
         W_ZIP      CHAR(9)        NOT NULL,
         W_TAX      REAL           NOT NULL,
         W_YTD      DECIMAL(12,2)  NOT NULL,
         W_ID       INTEGER        NOT NULL
        )
        IN ts_ware_015
        INDEX IN ts_ware_015
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 22737 ENDING AT 24360
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE16;
CREATE TABLE WAREHOUSE16
        (
         W_NAME     CHAR(10)       NOT NULL,
         W_STREET_1  CHAR(20)       NOT NULL,
         W_STREET_2  CHAR(20)       NOT NULL,
         W_CITY     CHAR(20)       NOT NULL,
         W_STATE    CHAR(2)        NOT NULL,
         W_ZIP      CHAR(9)        NOT NULL,
         W_TAX      REAL           NOT NULL,
         W_YTD      DECIMAL(12,2)  NOT NULL,
         W_ID       INTEGER        NOT NULL
        )
        IN ts_ware_016
        INDEX IN ts_ware_016
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 24361 ENDING AT 25984
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE17;
CREATE TABLE WAREHOUSE17
        (
         W_NAME     CHAR(10)       NOT NULL,
         W_STREET_1  CHAR(20)       NOT NULL,
         W_STREET_2  CHAR(20)       NOT NULL,
         W_CITY     CHAR(20)       NOT NULL,
         W_STATE    CHAR(2)        NOT NULL,
         W_ZIP      CHAR(9)        NOT NULL,
         W_TAX      REAL           NOT NULL,
         W_YTD      DECIMAL(12,2)  NOT NULL,
         W_ID       INTEGER        NOT NULL
        )
        IN ts_ware_017
        INDEX IN ts_ware_017
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 25985 ENDING AT 27608
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE18;
CREATE TABLE WAREHOUSE18
        (
         W_NAME     CHAR(10)       NOT NULL,
         W_STREET_1  CHAR(20)       NOT NULL,
         W_STREET_2  CHAR(20)       NOT NULL,
         W_CITY     CHAR(20)       NOT NULL,
         W_STATE    CHAR(2)        NOT NULL,
         W_ZIP      CHAR(9)        NOT NULL,
         W_TAX      REAL           NOT NULL,
         W_YTD      DECIMAL(12,2)  NOT NULL,
         W_ID       INTEGER        NOT NULL
        )
        IN ts_ware_018
        INDEX IN ts_ware_018
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 27609 ENDING AT 29232
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE19;
CREATE TABLE WAREHOUSE19
        (
         W_NAME     CHAR(10)       NOT NULL,
         W_STREET_1  CHAR(20)       NOT NULL,
         W_STREET_2  CHAR(20)       NOT NULL,
         W_CITY     CHAR(20)       NOT NULL,
         W_STATE    CHAR(2)        NOT NULL,
         W_ZIP      CHAR(9)        NOT NULL,
         W_TAX      REAL           NOT NULL,
         W_YTD      DECIMAL(12,2)  NOT NULL,
         W_ID       INTEGER        NOT NULL
        )
        IN ts_ware_019
        INDEX IN ts_ware_019
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 29233 ENDING AT 30856
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE20;
CREATE TABLE WAREHOUSE20
        (
         W_NAME     CHAR(10)       NOT NULL,
         W_STREET_1  CHAR(20)       NOT NULL,
         W_STREET_2  CHAR(20)       NOT NULL,
         W_CITY     CHAR(20)       NOT NULL,
         W_STATE    CHAR(2)        NOT NULL,
         W_ZIP      CHAR(9)        NOT NULL,
         W_TAX      REAL           NOT NULL,
         W_YTD      DECIMAL(12,2)  NOT NULL,
         W_ID       INTEGER        NOT NULL
        )
        IN ts_ware_020
        INDEX IN ts_ware_020
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 30857 ENDING AT 32480
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE21;
CREATE TABLE WAREHOUSE21
        (
         W_NAME     CHAR(10)       NOT NULL,
         W_STREET_1  CHAR(20)       NOT NULL,
         W_STREET_2  CHAR(20)       NOT NULL,
         W_CITY     CHAR(20)       NOT NULL,
         W_STATE    CHAR(2)        NOT NULL,
         W_ZIP      CHAR(9)        NOT NULL,
         W_TAX      REAL           NOT NULL,
         W_YTD      DECIMAL(12,2)  NOT NULL,
         W_ID       INTEGER        NOT NULL
        )
        IN ts_ware_021
        INDEX IN ts_ware_021
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 32481 ENDING AT 34104
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE22;
CREATE TABLE WAREHOUSE22
        (
         W_NAME     CHAR(10)       NOT NULL,
         W_STREET_1  CHAR(20)       NOT NULL,
         W_STREET_2  CHAR(20)       NOT NULL,
         W_CITY     CHAR(20)       NOT NULL,
         W_STATE    CHAR(2)        NOT NULL,
         W_ZIP      CHAR(9)        NOT NULL,
         W_TAX      REAL           NOT NULL,
         W_YTD      DECIMAL(12,2)  NOT NULL,
         W_ID       INTEGER        NOT NULL
        )
        IN ts_ware_022
        INDEX IN ts_ware_022
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 34105 ENDING AT 35728
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE23;
CREATE TABLE WAREHOUSE23
        (
         W_NAME     CHAR(10)       NOT NULL,
         W_STREET_1  CHAR(20)       NOT NULL,
         W_STREET_2  CHAR(20)       NOT NULL,
         W_CITY     CHAR(20)       NOT NULL,
         W_STATE    CHAR(2)        NOT NULL,
         W_ZIP      CHAR(9)        NOT NULL,
         W_TAX      REAL           NOT NULL,
         W_YTD      DECIMAL(12,2)  NOT NULL,
         W_ID       INTEGER        NOT NULL
        )
        IN ts_ware_023
        INDEX IN ts_ware_023
        ORGANIZE BY KEY SEQUENCE (
         W_ID STARTING FROM 35729 ENDING AT 37352
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE24;
CREATE TABLE WAREHOUSE24
        (
         W_NAME     CHAR(10)       NOT NULL,
         W_STREET_1  CHAR(20)       NOT NULL,
         W_STREET_2  CHAR(20)       NOT NULL,
         W_CITY     CHAR(20)       NOT NULL,
         W_STATE    CHAR(2)        NOT NULL,
         W_ZIP      CHAR(9)        NOT NULL,
         W_TAX      REAL           NOT NULL,
```

```
        W_YTD      DECIMAL(12,2)   NOT NULL,
        W_ID       INTEGER         NOT NULL
        )
        IN ts_ware_024
        INDEX IN ts_ware_024
        ORGANIZE BY KEY SEQUENCE (
        W_ID STARTING FROM 37353 ENDING AT 38976
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE25;
CREATE TABLE WAREHOUSE25
        (
        W_NAME     CHAR(10)       NOT NULL,
        W_STREET_1  CHAR(20)      NOT NULL,
        W_STREET_2  CHAR(20)      NOT NULL,
        W_CITY     CHAR(20)       NOT NULL,
        W_STATE    CHAR(2)        NOT NULL,
        W_ZIP      CHAR(9)        NOT NULL,
        W_TAX      REAL           NOT NULL,
        W_YTD      DECIMAL(12,2)   NOT NULL,
        W_ID       INTEGER         NOT NULL
        )
        IN ts_ware_025
        INDEX IN ts_ware_025
        ORGANIZE BY KEY SEQUENCE (
        W_ID STARTING FROM 38977 ENDING AT 40600
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE26;
CREATE TABLE WAREHOUSE26
        (
        W_NAME     CHAR(10)       NOT NULL,
        W_STREET_1  CHAR(20)      NOT NULL,
        W_STREET_2  CHAR(20)      NOT NULL,
        W_CITY     CHAR(20)       NOT NULL,
        W_STATE    CHAR(2)        NOT NULL,
        W_ZIP      CHAR(9)        NOT NULL,
        W_TAX      REAL           NOT NULL,
        W_YTD      DECIMAL(12,2)   NOT NULL,
        W_ID       INTEGER         NOT NULL
        )
        IN ts_ware_026
        INDEX IN ts_ware_026
        ORGANIZE BY KEY SEQUENCE (
        W_ID STARTING FROM 40601 ENDING AT 42224
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE27;
CREATE TABLE WAREHOUSE27
        (
        W_NAME     CHAR(10)       NOT NULL,
        W_STREET_1  CHAR(20)      NOT NULL,
        W_STREET_2  CHAR(20)      NOT NULL,
        W_CITY     CHAR(20)       NOT NULL,
        W_STATE    CHAR(2)        NOT NULL,
        W_ZIP      CHAR(9)        NOT NULL,
        W_TAX      REAL           NOT NULL,
        W_YTD      DECIMAL(12,2)   NOT NULL,
        W_ID       INTEGER         NOT NULL
        )
        IN ts_ware_027
        INDEX IN ts_ware_027
        ORGANIZE BY KEY SEQUENCE (
        W_ID STARTING FROM 42225 ENDING AT 43848
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE28;
CREATE TABLE WAREHOUSE28
        (
        W_NAME     CHAR(10)       NOT NULL,
        W_STREET_1  CHAR(20)      NOT NULL,
        W_STREET_2  CHAR(20)      NOT NULL,
        W_CITY     CHAR(20)       NOT NULL,
        W_STATE    CHAR(2)        NOT NULL,
        W_ZIP      CHAR(9)        NOT NULL,
        W_TAX      REAL           NOT NULL,
        W_YTD      DECIMAL(12,2)   NOT NULL,
        W_ID       INTEGER         NOT NULL
        )
        IN ts_ware_028
        INDEX IN ts_ware_028
        ORGANIZE BY KEY SEQUENCE (
        W_ID STARTING FROM 43849 ENDING AT 45472
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE29;
CREATE TABLE WAREHOUSE29
        (
        W_NAME     CHAR(10)       NOT NULL,
        W_STREET_1  CHAR(20)      NOT NULL,
        W_STREET_2  CHAR(20)      NOT NULL,
        W_CITY     CHAR(20)       NOT NULL,
        W_STATE    CHAR(2)        NOT NULL,
        W_ZIP      CHAR(9)        NOT NULL,
        W_TAX      REAL           NOT NULL,
        W_YTD      DECIMAL(12,2)   NOT NULL,
        W_ID       INTEGER         NOT NULL
        )
        IN ts_ware_029
        INDEX IN ts_ware_029
        ORGANIZE BY KEY SEQUENCE (
        W_ID STARTING FROM 45473 ENDING AT 47096
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE30;
CREATE TABLE WAREHOUSE30
        (
        W_NAME     CHAR(10)       NOT NULL,
        W_STREET_1  CHAR(20)      NOT NULL,
        W_STREET_2  CHAR(20)      NOT NULL,
        W_CITY     CHAR(20)       NOT NULL,
        W_STATE    CHAR(2)        NOT NULL,
        W_ZIP      CHAR(9)        NOT NULL,
        W_TAX      REAL           NOT NULL,
        W_YTD      DECIMAL(12,2)   NOT NULL,
        W_ID       INTEGER         NOT NULL
        )
        IN ts_ware_030
        INDEX IN ts_ware_030
        ORGANIZE BY KEY SEQUENCE (
        W_ID STARTING FROM 47097 ENDING AT 48720
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE31;
CREATE TABLE WAREHOUSE31
        (
        W_NAME     CHAR(10)       NOT NULL,
        W_STREET_1  CHAR(20)      NOT NULL,
        W_STREET_2  CHAR(20)      NOT NULL,
        W_CITY     CHAR(20)       NOT NULL,
        W_STATE    CHAR(2)        NOT NULL,
        W_ZIP      CHAR(9)        NOT NULL,
        W_TAX      REAL           NOT NULL,
        W_YTD      DECIMAL(12,2)   NOT NULL,
        W_ID       INTEGER         NOT NULL
        )
        IN ts_ware_031
        INDEX IN ts_ware_031
        ORGANIZE BY KEY SEQUENCE (
        W_ID STARTING FROM 48721 ENDING AT 50344
        )
        ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE32;
CREATE TABLE WAREHOUSE32
        (
        W_NAME     CHAR(10)       NOT NULL,
        W_STREET_1  CHAR(20)      NOT NULL,
        W_STREET_2  CHAR(20)      NOT NULL,
        W_CITY     CHAR(20)       NOT NULL,
        W_STATE    CHAR(2)        NOT NULL,
        W_ZIP      CHAR(9)        NOT NULL,
        W_TAX      REAL           NOT NULL,
        W_YTD      DECIMAL(12,2)   NOT NULL,
        W_ID       INTEGER         NOT NULL
        )
        IN ts_ware_032
        INDEX IN ts_ware_032
        ORGANIZE BY KEY SEQUENCE (
        W_ID STARTING FROM 50345 ENDING AT 51968
        )
        ALLOW OVERFLOW;
connect reset;
```

DDL/CRVW_CUSTOMER.ddl

```
connect to TPCC in share mode;
DROP VIEW CUSTOMER;
CREATE VIEW CUSTOMER
        (C_ID,
        C_STATE,
        C_ZIP,
        C_PHONE,
        C_SINCE,
        C_CREDIT_LIM,
        C_MIDDLE,
        C_CREDIT,
        C_DISCOUNT,
        C_DATA,
        C_LAST,
        C_FIRST,
        C_STREET_1,
        C_STREET_2,
        C_CITY,
        C_D_ID,
        C_W_ID,
        C_DELIVERY_CNT,
        C_BALANCE,
        C_YTD_PAYMENT,
        C_PAYMENT_CNT
        ) AS SELECT * FROM CUSTOMER1 UNION ALL
SELECT * FROM CUSTOMER2 UNION ALL
SELECT * FROM CUSTOMER3 UNION ALL
SELECT * FROM CUSTOMER4 UNION ALL
SELECT * FROM CUSTOMER5 UNION ALL
SELECT * FROM CUSTOMER6 UNION ALL
SELECT * FROM CUSTOMER7 UNION ALL
SELECT * FROM CUSTOMER8 UNION ALL
SELECT * FROM CUSTOMER9 UNION ALL
SELECT * FROM CUSTOMER10 UNION ALL
```

```
SELECT * FROM CUSTOMER11 UNION ALL
SELECT * FROM CUSTOMER12 UNION ALL
SELECT * FROM CUSTOMER13 UNION ALL
SELECT * FROM CUSTOMER14 UNION ALL
SELECT * FROM CUSTOMER15 UNION ALL
SELECT * FROM CUSTOMER16 UNION ALL
SELECT * FROM CUSTOMER17 UNION ALL
SELECT * FROM CUSTOMER18 UNION ALL
SELECT * FROM CUSTOMER19 UNION ALL
SELECT * FROM CUSTOMER20 UNION ALL
SELECT * FROM CUSTOMER21 UNION ALL
SELECT * FROM CUSTOMER22 UNION ALL
SELECT * FROM CUSTOMER23 UNION ALL
SELECT * FROM CUSTOMER24 UNION ALL
SELECT * FROM CUSTOMER25 UNION ALL
SELECT * FROM CUSTOMER26 UNION ALL
SELECT * FROM CUSTOMER27 UNION ALL
SELECT * FROM CUSTOMER28 UNION ALL
SELECT * FROM CUSTOMER29 UNION ALL
SELECT * FROM CUSTOMER30 UNION ALL
SELECT * FROM CUSTOMER31 UNION ALL
SELECT * FROM CUSTOMER32 UNION ALL
SELECT * FROM CUSTOMER33 UNION ALL
SELECT * FROM CUSTOMER34 UNION ALL
SELECT * FROM CUSTOMER35 UNION ALL
SELECT * FROM CUSTOMER36 UNION ALL
SELECT * FROM CUSTOMER37 UNION ALL
SELECT * FROM CUSTOMER38 UNION ALL
SELECT * FROM CUSTOMER39 UNION ALL
SELECT * FROM CUSTOMER40 UNION ALL
SELECT * FROM CUSTOMER41 UNION ALL
SELECT * FROM CUSTOMER42 UNION ALL
SELECT * FROM CUSTOMER43 UNION ALL
SELECT * FROM CUSTOMER44 UNION ALL
SELECT * FROM CUSTOMER45 UNION ALL
SELECT * FROM CUSTOMER46 UNION ALL
SELECT * FROM CUSTOMER47 UNION ALL
SELECT * FROM CUSTOMER48 UNION ALL
SELECT * FROM CUSTOMER49 UNION ALL
SELECT * FROM CUSTOMER50 UNION ALL
SELECT * FROM CUSTOMER51 UNION ALL
SELECT * FROM CUSTOMER52 UNION ALL
SELECT * FROM CUSTOMER53 UNION ALL
SELECT * FROM CUSTOMER54 UNION ALL
SELECT * FROM CUSTOMER55 UNION ALL
SELECT * FROM CUSTOMER56 UNION ALL
SELECT * FROM CUSTOMER57 UNION ALL
SELECT * FROM CUSTOMER58 UNION ALL
SELECT * FROM CUSTOMER59 UNION ALL
SELECT * FROM CUSTOMER60 UNION ALL
SELECT * FROM CUSTOMER61 UNION ALL
SELECT * FROM CUSTOMER62 UNION ALL
SELECT * FROM CUSTOMER63 UNION ALL
SELECT * FROM CUSTOMER64
WITH ROW MOVEMENT;
COMMIT WORK;
connect reset;
```

DDL/CRVW_DISTRICT.ddl

```
connect to TPCC in share mode;
DROP VIEW DISTRICT;
CREATE VIEW DISTRICT
        (D_NEXT_O_ID,
         D_TAX,
         D_YTD,
         D_NAME,
         D_STREET_1,
         D_STREET_2,
         D_CITY,
         D_STATE,
         D_ZIP,
         D_ID,
         D_W_ID
        ) AS SELECT * FROM DISTRICT1 UNION ALL
SELECT * FROM DISTRICT2 UNION ALL
SELECT * FROM DISTRICT3 UNION ALL
SELECT * FROM DISTRICT4 UNION ALL
SELECT * FROM DISTRICT5 UNION ALL
SELECT * FROM DISTRICT6 UNION ALL
SELECT * FROM DISTRICT7 UNION ALL
SELECT * FROM DISTRICT8 UNION ALL
SELECT * FROM DISTRICT9 UNION ALL
SELECT * FROM DISTRICT10 UNION ALL
SELECT * FROM DISTRICT11 UNION ALL
SELECT * FROM DISTRICT12 UNION ALL
SELECT * FROM DISTRICT13 UNION ALL
SELECT * FROM DISTRICT14 UNION ALL
SELECT * FROM DISTRICT15 UNION ALL
SELECT * FROM DISTRICT16 UNION ALL
SELECT * FROM DISTRICT17 UNION ALL
SELECT * FROM DISTRICT18 UNION ALL
SELECT * FROM DISTRICT19 UNION ALL
SELECT * FROM DISTRICT20 UNION ALL
SELECT * FROM DISTRICT21 UNION ALL
SELECT * FROM DISTRICT22 UNION ALL
SELECT * FROM DISTRICT23 UNION ALL
SELECT * FROM DISTRICT24 UNION ALL
SELECT * FROM DISTRICT25 UNION ALL
SELECT * FROM DISTRICT26 UNION ALL
SELECT * FROM DISTRICT27 UNION ALL
SELECT * FROM DISTRICT28 UNION ALL
SELECT * FROM DISTRICT29 UNION ALL
SELECT * FROM DISTRICT30 UNION ALL
SELECT * FROM DISTRICT31 UNION ALL
SELECT * FROM DISTRICT32
WITH ROW MOVEMENT;
COMMIT WORK;
connect reset;
```

DDL/CRVW_HISTORY.ddl

```
connect to TPCC in share mode;
DROP VIEW HISTORY;
CREATE VIEW HISTORY
        (H_C_ID,
         H_C_D_ID,
         H_C_W_ID,
         H_D_ID,
         H_W_ID,
         H_DATE,
         H_AMOUNT,
         H_DATA
        ) AS SELECT * FROM HISTORY1 UNION ALL
SELECT * FROM HISTORY2 UNION ALL
SELECT * FROM HISTORY3 UNION ALL
SELECT * FROM HISTORY4 UNION ALL
SELECT * FROM HISTORY5 UNION ALL
SELECT * FROM HISTORY6 UNION ALL
SELECT * FROM HISTORY7 UNION ALL
SELECT * FROM HISTORY8 UNION ALL
SELECT * FROM HISTORY9 UNION ALL
SELECT * FROM HISTORY10 UNION ALL
SELECT * FROM HISTORY11 UNION ALL
SELECT * FROM HISTORY12 UNION ALL
SELECT * FROM HISTORY13 UNION ALL
SELECT * FROM HISTORY14 UNION ALL
SELECT * FROM HISTORY15 UNION ALL
SELECT * FROM HISTORY16 UNION ALL
SELECT * FROM HISTORY17 UNION ALL
SELECT * FROM HISTORY18 UNION ALL
SELECT * FROM HISTORY19 UNION ALL
SELECT * FROM HISTORY20 UNION ALL
SELECT * FROM HISTORY21 UNION ALL
SELECT * FROM HISTORY22 UNION ALL
SELECT * FROM HISTORY23 UNION ALL
SELECT * FROM HISTORY24 UNION ALL
SELECT * FROM HISTORY25 UNION ALL
SELECT * FROM HISTORY26 UNION ALL
SELECT * FROM HISTORY27 UNION ALL
SELECT * FROM HISTORY28 UNION ALL
SELECT * FROM HISTORY29 UNION ALL
SELECT * FROM HISTORY30 UNION ALL
SELECT * FROM HISTORY31 UNION ALL
SELECT * FROM HISTORY32
WITH ROW MOVEMENT;
COMMIT WORK;
connect reset;
```

DDL/CRVW_NEW_ORDER.ddl

```
connect to TPCC in share mode;
DROP VIEW NEW_ORDER;
CREATE VIEW NEW_ORDER
        (NO_O_ID,
         NO_D_ID,
         NO_W_ID
        ) AS SELECT * FROM NEW_ORDERA1 UNION ALL
SELECT * FROM NEW_ORDERA2 UNION ALL
SELECT * FROM NEW_ORDERA3 UNION ALL
SELECT * FROM NEW_ORDERA4 UNION ALL
SELECT * FROM NEW_ORDERA5 UNION ALL
SELECT * FROM NEW_ORDERA6 UNION ALL
SELECT * FROM NEW_ORDERA7 UNION ALL
SELECT * FROM NEW_ORDERA8 UNION ALL
SELECT * FROM NEW_ORDERA9 UNION ALL
SELECT * FROM NEW_ORDERA10 UNION ALL
SELECT * FROM NEW_ORDERA11 UNION ALL
SELECT * FROM NEW_ORDERA12 UNION ALL
SELECT * FROM NEW_ORDERA13 UNION ALL
SELECT * FROM NEW_ORDERA14 UNION ALL
SELECT * FROM NEW_ORDERA15 UNION ALL
SELECT * FROM NEW_ORDERA16 UNION ALL
SELECT * FROM NEW_ORDERA17 UNION ALL
SELECT * FROM NEW_ORDERA18 UNION ALL
SELECT * FROM NEW_ORDERA19 UNION ALL
SELECT * FROM NEW_ORDERA20 UNION ALL
SELECT * FROM NEW_ORDERA21 UNION ALL
SELECT * FROM NEW_ORDERA22 UNION ALL
SELECT * FROM NEW_ORDERA23 UNION ALL
SELECT * FROM NEW_ORDERA24 UNION ALL
SELECT * FROM NEW_ORDERA25 UNION ALL
SELECT * FROM NEW_ORDERA26 UNION ALL
SELECT * FROM NEW_ORDERA27 UNION ALL
SELECT * FROM NEW_ORDERA28 UNION ALL
SELECT * FROM NEW_ORDERA29 UNION ALL
SELECT * FROM NEW_ORDERA30 UNION ALL
SELECT * FROM NEW_ORDERA31 UNION ALL
SELECT * FROM NEW_ORDERA32 UNION ALL
SELECT * FROM NEW_ORDERB1 UNION ALL
SELECT * FROM NEW_ORDERB2 UNION ALL
SELECT * FROM NEW_ORDERB3 UNION ALL
SELECT * FROM NEW_ORDERB4 UNION ALL
SELECT * FROM NEW_ORDERB5 UNION ALL
SELECT * FROM NEW_ORDERB6 UNION ALL
SELECT * FROM NEW_ORDERB7 UNION ALL
SELECT * FROM NEW_ORDERB8 UNION ALL
```

```
SELECT * FROM NEW_ORDERB9 UNION ALL
SELECT * FROM NEW_ORDERB10 UNION ALL
SELECT * FROM NEW_ORDERB11 UNION ALL
SELECT * FROM NEW_ORDERB12 UNION ALL
SELECT * FROM NEW_ORDERB13 UNION ALL
SELECT * FROM NEW_ORDERB14 UNION ALL
SELECT * FROM NEW_ORDERB15 UNION ALL
SELECT * FROM NEW_ORDERB16 UNION ALL
SELECT * FROM NEW_ORDERB17 UNION ALL
SELECT * FROM NEW_ORDERB18 UNION ALL
SELECT * FROM NEW_ORDERB19 UNION ALL
SELECT * FROM NEW_ORDERB20 UNION ALL
SELECT * FROM NEW_ORDERB21 UNION ALL
SELECT * FROM NEW_ORDERB22 UNION ALL
SELECT * FROM NEW_ORDERB23 UNION ALL
SELECT * FROM NEW_ORDERB24 UNION ALL
SELECT * FROM NEW_ORDERB25 UNION ALL
SELECT * FROM NEW_ORDERB26 UNION ALL
SELECT * FROM NEW_ORDERB27 UNION ALL
SELECT * FROM NEW_ORDERB28 UNION ALL
SELECT * FROM NEW_ORDERB29 UNION ALL
SELECT * FROM NEW_ORDERB30 UNION ALL
SELECT * FROM NEW_ORDERB31 UNION ALL
SELECT * FROM NEW_ORDERB32
WITH ROW MOVEMENT;
COMMIT WORK;
connect reset;
```

DDL/CRVW_ORDERS.ddl

```
connect to TPCC in share mode;
DROP VIEW ORDERS;
CREATE VIEW ORDERS
        (O_C_ID,
         O_ENTRY_D,
         O_CARRIER_ID,
         O_OL_CNT,
         O_ALL_LOCAL,
         O_ID,
         O_W_ID,
         O_D_ID
        ) AS SELECT * FROM ORDERS1 UNION ALL
SELECT * FROM ORDERS2 UNION ALL
SELECT * FROM ORDERS3 UNION ALL
SELECT * FROM ORDERS4 UNION ALL
SELECT * FROM ORDERS5 UNION ALL
SELECT * FROM ORDERS6 UNION ALL
SELECT * FROM ORDERS7 UNION ALL
SELECT * FROM ORDERS8 UNION ALL
SELECT * FROM ORDERS9 UNION ALL
SELECT * FROM ORDERS10 UNION ALL
SELECT * FROM ORDERS11 UNION ALL
SELECT * FROM ORDERS12 UNION ALL
SELECT * FROM ORDERS13 UNION ALL
SELECT * FROM ORDERS14 UNION ALL
SELECT * FROM ORDERS15 UNION ALL
SELECT * FROM ORDERS16 UNION ALL
SELECT * FROM ORDERS17 UNION ALL
SELECT * FROM ORDERS18 UNION ALL
SELECT * FROM ORDERS19 UNION ALL
SELECT * FROM ORDERS20 UNION ALL
SELECT * FROM ORDERS21 UNION ALL
SELECT * FROM ORDERS22 UNION ALL
SELECT * FROM ORDERS23 UNION ALL
SELECT * FROM ORDERS24 UNION ALL
SELECT * FROM ORDERS25 UNION ALL
SELECT * FROM ORDERS26 UNION ALL
SELECT * FROM ORDERS27 UNION ALL
SELECT * FROM ORDERS28 UNION ALL
SELECT * FROM ORDERS29 UNION ALL
SELECT * FROM ORDERS30 UNION ALL
SELECT * FROM ORDERS31 UNION ALL
SELECT * FROM ORDERS32 UNION ALL
SELECT * FROM ORDERS33 UNION ALL
SELECT * FROM ORDERS34 UNION ALL
SELECT * FROM ORDERS35 UNION ALL
SELECT * FROM ORDERS36 UNION ALL
SELECT * FROM ORDERS37 UNION ALL
SELECT * FROM ORDERS38 UNION ALL
SELECT * FROM ORDERS39 UNION ALL
SELECT * FROM ORDERS40 UNION ALL
SELECT * FROM ORDERS41 UNION ALL
SELECT * FROM ORDERS42 UNION ALL
SELECT * FROM ORDERS43 UNION ALL
SELECT * FROM ORDERS44 UNION ALL
SELECT * FROM ORDERS45 UNION ALL
SELECT * FROM ORDERS46 UNION ALL
SELECT * FROM ORDERS47 UNION ALL
SELECT * FROM ORDERS48 UNION ALL
SELECT * FROM ORDERS49 UNION ALL
SELECT * FROM ORDERS50 UNION ALL
SELECT * FROM ORDERS51 UNION ALL
SELECT * FROM ORDERS52 UNION ALL
SELECT * FROM ORDERS53 UNION ALL
SELECT * FROM ORDERS54 UNION ALL
SELECT * FROM ORDERS55 UNION ALL
SELECT * FROM ORDERS56 UNION ALL
SELECT * FROM ORDERS57 UNION ALL
SELECT * FROM ORDERS58 UNION ALL
SELECT * FROM ORDERS59 UNION ALL
SELECT * FROM ORDERS60 UNION ALL
SELECT * FROM ORDERS61 UNION ALL
SELECT * FROM ORDERS62 UNION ALL
SELECT * FROM ORDERS63 UNION ALL
SELECT * FROM ORDERS64
WITH ROW MOVEMENT;
COMMIT WORK;
connect reset;
```

DDL/CRVW_ORDER_LINE.ddl

```
connect to TPCC in share mode;
DROP VIEW ORDER_LINE;
CREATE VIEW ORDER_LINE
        (OL_DELIVERY_D,
         OL_AMOUNT,
         OL_I_ID,
         OL_SUPPLY_W_ID,
         OL_QUANTITY,
         OL_DIST_INFO,
         OL_O_ID,
         OL_D_ID,
         OL_W_ID,
         OL_NUMBER
        ) AS SELECT * FROM ORDER_LINE1 UNION ALL
SELECT * FROM ORDER_LINE2 UNION ALL
SELECT * FROM ORDER_LINE3 UNION ALL
SELECT * FROM ORDER_LINE4 UNION ALL
SELECT * FROM ORDER_LINE5 UNION ALL
SELECT * FROM ORDER_LINE6 UNION ALL
SELECT * FROM ORDER_LINE7 UNION ALL
SELECT * FROM ORDER_LINE8 UNION ALL
SELECT * FROM ORDER_LINE9 UNION ALL
SELECT * FROM ORDER_LINE10 UNION ALL
SELECT * FROM ORDER_LINE11 UNION ALL
SELECT * FROM ORDER_LINE12 UNION ALL
SELECT * FROM ORDER_LINE13 UNION ALL
SELECT * FROM ORDER_LINE14 UNION ALL
SELECT * FROM ORDER_LINE15 UNION ALL
SELECT * FROM ORDER_LINE16 UNION ALL
SELECT * FROM ORDER_LINE17 UNION ALL
SELECT * FROM ORDER_LINE18 UNION ALL
SELECT * FROM ORDER_LINE19 UNION ALL
SELECT * FROM ORDER_LINE20 UNION ALL
SELECT * FROM ORDER_LINE21 UNION ALL
SELECT * FROM ORDER_LINE22 UNION ALL
SELECT * FROM ORDER_LINE23 UNION ALL
SELECT * FROM ORDER_LINE24 UNION ALL
SELECT * FROM ORDER_LINE25 UNION ALL
SELECT * FROM ORDER_LINE26 UNION ALL
SELECT * FROM ORDER_LINE27 UNION ALL
SELECT * FROM ORDER_LINE28 UNION ALL
SELECT * FROM ORDER_LINE29 UNION ALL
SELECT * FROM ORDER_LINE30 UNION ALL
SELECT * FROM ORDER_LINE31 UNION ALL
SELECT * FROM ORDER_LINE32 UNION ALL
SELECT * FROM ORDER_LINE33 UNION ALL
SELECT * FROM ORDER_LINE34 UNION ALL
SELECT * FROM ORDER_LINE35 UNION ALL
SELECT * FROM ORDER_LINE36 UNION ALL
SELECT * FROM ORDER_LINE37 UNION ALL
SELECT * FROM ORDER_LINE38 UNION ALL
SELECT * FROM ORDER_LINE39 UNION ALL
SELECT * FROM ORDER_LINE40 UNION ALL
SELECT * FROM ORDER_LINE41 UNION ALL
SELECT * FROM ORDER_LINE42 UNION ALL
SELECT * FROM ORDER_LINE43 UNION ALL
SELECT * FROM ORDER_LINE44 UNION ALL
SELECT * FROM ORDER_LINE45 UNION ALL
SELECT * FROM ORDER_LINE46 UNION ALL
SELECT * FROM ORDER_LINE47 UNION ALL
SELECT * FROM ORDER_LINE48 UNION ALL
SELECT * FROM ORDER_LINE49 UNION ALL
SELECT * FROM ORDER_LINE50 UNION ALL
SELECT * FROM ORDER_LINE51 UNION ALL
SELECT * FROM ORDER_LINE52 UNION ALL
SELECT * FROM ORDER_LINE53 UNION ALL
SELECT * FROM ORDER_LINE54 UNION ALL
SELECT * FROM ORDER_LINE55 UNION ALL
SELECT * FROM ORDER_LINE56 UNION ALL
SELECT * FROM ORDER_LINE57 UNION ALL
SELECT * FROM ORDER_LINE58 UNION ALL
SELECT * FROM ORDER_LINE59 UNION ALL
SELECT * FROM ORDER_LINE60 UNION ALL
SELECT * FROM ORDER_LINE61 UNION ALL
SELECT * FROM ORDER_LINE62 UNION ALL
SELECT * FROM ORDER_LINE63 UNION ALL
SELECT * FROM ORDER_LINE64
WITH ROW MOVEMENT;
COMMIT WORK;
connect reset;
```

DDL/CRVW_STOCK.ddl

```
connect to TPCC in share mode;
DROP VIEW STOCK;
CREATE VIEW STOCK
        (S_REMOTE_CNT,
         S_QUANTITY,
         S_ORDER_CNT,
         S_YTD,
         S_DATA,
         S_DIST_01,
         S_DIST_02,
         S_DIST_03,
         S_DIST_04,
         S_DIST_05,
```

```
            S_DIST_06,
            S_DIST_07,
            S_DIST_08,
            S_DIST_09,
            S_DIST_10,
            S_I_ID,
            S_W_ID
           ) AS SELECT * FROM STOCK1 UNION ALL
SELECT * FROM STOCK2 UNION ALL
SELECT * FROM STOCK3 UNION ALL
SELECT * FROM STOCK4 UNION ALL
SELECT * FROM STOCK5 UNION ALL
SELECT * FROM STOCK6 UNION ALL
SELECT * FROM STOCK7 UNION ALL
SELECT * FROM STOCK8 UNION ALL
SELECT * FROM STOCK9 UNION ALL
SELECT * FROM STOCK10 UNION ALL
SELECT * FROM STOCK11 UNION ALL
SELECT * FROM STOCK12 UNION ALL
SELECT * FROM STOCK13 UNION ALL
SELECT * FROM STOCK14 UNION ALL
SELECT * FROM STOCK15 UNION ALL
SELECT * FROM STOCK16 UNION ALL
SELECT * FROM STOCK17 UNION ALL
SELECT * FROM STOCK18 UNION ALL
SELECT * FROM STOCK19 UNION ALL
SELECT * FROM STOCK20 UNION ALL
SELECT * FROM STOCK21 UNION ALL
SELECT * FROM STOCK22 UNION ALL
SELECT * FROM STOCK23 UNION ALL
SELECT * FROM STOCK24 UNION ALL
SELECT * FROM STOCK25 UNION ALL
SELECT * FROM STOCK26 UNION ALL
SELECT * FROM STOCK27 UNION ALL
SELECT * FROM STOCK28 UNION ALL
SELECT * FROM STOCK29 UNION ALL
SELECT * FROM STOCK30 UNION ALL
SELECT * FROM STOCK31 UNION ALL
SELECT * FROM STOCK32 UNION ALL
SELECT * FROM STOCK33 UNION ALL
SELECT * FROM STOCK34 UNION ALL
SELECT * FROM STOCK35 UNION ALL
SELECT * FROM STOCK36 UNION ALL
SELECT * FROM STOCK37 UNION ALL
SELECT * FROM STOCK38 UNION ALL
SELECT * FROM STOCK39 UNION ALL
SELECT * FROM STOCK40 UNION ALL
SELECT * FROM STOCK41 UNION ALL
SELECT * FROM STOCK42 UNION ALL
SELECT * FROM STOCK43 UNION ALL
SELECT * FROM STOCK44 UNION ALL
SELECT * FROM STOCK45 UNION ALL
SELECT * FROM STOCK46 UNION ALL
SELECT * FROM STOCK47 UNION ALL
SELECT * FROM STOCK48 UNION ALL
SELECT * FROM STOCK49 UNION ALL
SELECT * FROM STOCK50 UNION ALL
SELECT * FROM STOCK51 UNION ALL
SELECT * FROM STOCK52 UNION ALL
SELECT * FROM STOCK53 UNION ALL
SELECT * FROM STOCK54 UNION ALL
SELECT * FROM STOCK55 UNION ALL
SELECT * FROM STOCK56 UNION ALL
SELECT * FROM STOCK57 UNION ALL
SELECT * FROM STOCK58 UNION ALL
SELECT * FROM STOCK59 UNION ALL
SELECT * FROM STOCK60 UNION ALL
SELECT * FROM STOCK61 UNION ALL
SELECT * FROM STOCK62 UNION ALL
SELECT * FROM STOCK63 UNION ALL
SELECT * FROM STOCK64
WITH ROW MOVEMENT;
COMMIT WORK;
connect reset;
```

DDL/CRVW_WAREHOUSE.ddl

```
connect to TPCC in share mode;
DROP VIEW WAREHOUSE;
CREATE VIEW WAREHOUSE
            (W_NAME,
             W_STREET_1,
             W_STREET_2,
             W_CITY,
             W_STATE,
             W_ZIP,
             W_TAX,
             W_YTD,
             W_ID
            ) AS SELECT * FROM WAREHOUSE1 UNION ALL
SELECT * FROM WAREHOUSE2 UNION ALL
SELECT * FROM WAREHOUSE3 UNION ALL
SELECT * FROM WAREHOUSE4 UNION ALL
SELECT * FROM WAREHOUSE5 UNION ALL
SELECT * FROM WAREHOUSE6 UNION ALL
SELECT * FROM WAREHOUSE7 UNION ALL
SELECT * FROM WAREHOUSE8 UNION ALL
SELECT * FROM WAREHOUSE9 UNION ALL
SELECT * FROM WAREHOUSE10 UNION ALL
SELECT * FROM WAREHOUSE11 UNION ALL
SELECT * FROM WAREHOUSE12 UNION ALL
SELECT * FROM WAREHOUSE13 UNION ALL
SELECT * FROM WAREHOUSE14 UNION ALL
SELECT * FROM WAREHOUSE15 UNION ALL
SELECT * FROM WAREHOUSE16 UNION ALL
SELECT * FROM WAREHOUSE17 UNION ALL
SELECT * FROM WAREHOUSE18 UNION ALL
SELECT * FROM WAREHOUSE19 UNION ALL
SELECT * FROM WAREHOUSE20 UNION ALL
SELECT * FROM WAREHOUSE21 UNION ALL
SELECT * FROM WAREHOUSE22 UNION ALL
SELECT * FROM WAREHOUSE23 UNION ALL
SELECT * FROM WAREHOUSE24 UNION ALL
SELECT * FROM WAREHOUSE25 UNION ALL
SELECT * FROM WAREHOUSE26 UNION ALL
SELECT * FROM WAREHOUSE27 UNION ALL
SELECT * FROM WAREHOUSE28 UNION ALL
SELECT * FROM WAREHOUSE29 UNION ALL
SELECT * FROM WAREHOUSE30 UNION ALL
SELECT * FROM WAREHOUSE31 UNION ALL
SELECT * FROM WAREHOUSE32
WITH ROW MOVEMENT;
COMMIT WORK;
connect reset;
```

DDL/GEN_CUSTOMER_ALL.sh

```
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 1 812 -f1 /flats/F1_001/customer_001_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 813 1624 -f1 /flats/F1_002/customer_002_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 1625 2436 -f1 /flats/F1_003/customer_003_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 2437 3248 -f1 /flats/F1_004/customer_004_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 3249 4060 -f1 /flats/F1_005/customer_005_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 4061 4872 -f1 /flats/F1_006/customer_006_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 4873 5684 -f1 /flats/F1_007/customer_007_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 5685 6496 -f1 /flats/F1_008/customer_008_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 6497 7308 -f1 /flats/F1_009/customer_009_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 7309 8120 -f1 /flats/F1_010/customer_010_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 8121 8932 -f1 /flats/F1_011/customer_011_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 8933 9744 -f1 /flats/F1_012/customer_012_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 9745 10556 -f1 /flats/F1_013/customer_013_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 10557 11368 -f1 /flats/F1_014/customer_014_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 11369 12180 -f1 /flats/F1_015/customer_015_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 12181 12992 -f1 /flats/F1_016/customer_016_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 12993 13804 -f1 /flats/F1_017/customer_017_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 13805 14616 -f1 /flats/F1_018/customer_018_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 14617 15428 -f1 /flats/F1_019/customer_019_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 15429 16240 -f1 /flats/F1_020/customer_020_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 16241 17052 -f1 /flats/F1_021/customer_021_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 17053 17864 -f1 /flats/F1_022/customer_022_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 17865 18676 -f1 /flats/F1_023/customer_023_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 18677 19488 -f1 /flats/F1_024/customer_024_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 19489 20300 -f1 /flats/F1_025/customer_025_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 20301 21112 -f1 /flats/F1_026/customer_026_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 21113 21924 -f1 /flats/F1_027/customer_027_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 21925 22736 -f1 /flats/F1_028/customer_028_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 22737 23548 -f1 /flats/F1_029/customer_029_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 23549 24360 -f1 /flats/F1_030/customer_030_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 24361 25172 -f1 /flats/F1_031/customer_031_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 25173 25984 -f1 /flats/F1_032/customer_032_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 25985 26796 -f1 /flats/F1_033/customer_033_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 26797 27608 -f1 /flats/F1_034/customer_034_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 27609 28420 -f1 /flats/F1_035/customer_035_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 28421 29232 -f1 /flats/F1_036/customer_036_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 29233 30044 -f1 /flats/F1_037/customer_037_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 30045 30856 -f1 /flats/F1_038/customer_038_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 30857 31668 -f1 /flats/F1_039/customer_039_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 31669 32480 -f1 /flats/F1_040/customer_040_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 32481 33292 -f1 /flats/F1_041/customer_041_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 33293 34104 -f1 /flats/F1_042/customer_042_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 34105 34916 -f1 /flats/F1_043/customer_043_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 34917 35728 -f1 /flats/F1_044/customer_044_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 35729 36540 -f1 /flats/F1_045/customer_045_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 36541 37352 -f1 /flats/F1_046/customer_046_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 37353 38164 -f1 /flats/F1_047/customer_047_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 38165 38976 -f1 /flats/F1_048/customer_048_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 38977 39788 -f1 /flats/F1_049/customer_049_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 39789 40600 -f1 /flats/F1_050/customer_050_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 40601 41412 -f1 /flats/F1_051/customer_051_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 41413 42224 -f1 /flats/F1_052/customer_052_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 42225 43036 -f1 /flats/F1_053/customer_053_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 43037 43848 -f1 /flats/F1_054/customer_054_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 43849 44660 -f1 /flats/F1_055/customer_055_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 44661 45472 -f1 /flats/F1_056/customer_056_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 45473 46284 -f1 /flats/F1_057/customer_057_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 46285 47096 -f1 /flats/F1_058/customer_058_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 47097 47908 -f1 /flats/F1_059/customer_059_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 47909 48720 -f1 /flats/F1_060/customer_060_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 48721 49532 -f1 /flats/F1_061/customer_061_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 49533 50344 -f1 /flats/F1_062/customer_062_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 50345 51156 -f1 /flats/F1_063/customer_063_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 7 -r 51157 51968 -f1 /flats/F1_064/customer_064_1.dat
```

DDL/GEN_DISTRICT_ALL.sh

```
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 1 1624 -f1 /flats/F1_001/district_001_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 1625 3248 -f1 /flats/F1_002/district_002_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 3249 4872 -f1 /flats/F1_003/district_003_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 4873 6496 -f1 /flats/F1_004/district_004_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 6497 8120 -f1 /flats/F1_005/district_005_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 8121 9744 -f1 /flats/F1_006/district_006_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 9745 11368 -f1 /flats/F1_007/district_007_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 11369 12992 -f1 /flats/F1_008/district_008_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 12993 14616 -f1 /flats/F1_009/district_009_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 14617 16240 -f1 /flats/F1_010/district_010_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 16241 17864 -f1 /flats/F1_011/district_011_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 17865 19488 -f1 /flats/F1_012/district_012_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 19489 21112 -f1 /flats/F1_013/district_013_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 21113 22736 -f1 /flats/F1_014/district_014_1.dat
```

```
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 22737 24360 -f1 /flats/F1_015/district_015_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 24361 25984 -f1 /flats/F1_016/district_016_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 25985 27608 -f1 /flats/F1_017/district_017_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 27609 29232 -f1 /flats/F1_018/district_018_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 29233 30856 -f1 /flats/F1_019/district_019_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 30857 32480 -f1 /flats/F1_020/district_020_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 32481 34104 -f1 /flats/F1_021/district_021_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 34105 35728 -f1 /flats/F1_022/district_022_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 35729 37352 -f1 /flats/F1_023/district_023_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 37353 38976 -f1 /flats/F1_024/district_024_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 38977 40600 -f1 /flats/F1_025/district_025_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 40601 42224 -f1 /flats/F1_026/district_026_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 42225 43848 -f1 /flats/F1_027/district_027_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 43849 45472 -f1 /flats/F1_028/district_028_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 45473 47096 -f1 /flats/F1_029/district_029_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 47097 48720 -f1 /flats/F1_030/district_030_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 48721 50344 -f1 /flats/F1_031/district_031_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 4 -r 50345 51968 -f1 /flats/F1_032/district_032_1.dat
```

DDL/GEN_HISTORY_ALL.sh

```
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 1 1624 -f1 /flats/F1_001/history_001_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 1625 3248 -f1 /flats/F1_002/history_002_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 3249 4872 -f1 /flats/F1_003/history_003_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 4873 6496 -f1 /flats/F1_004/history_004_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 6497 8120 -f1 /flats/F1_005/history_005_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 8121 9744 -f1 /flats/F1_006/history_006_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 9745 11368 -f1 /flats/F1_007/history_007_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 11369 12992 -f1 /flats/F1_008/history_008_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 12993 14616 -f1 /flats/F1_009/history_009_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 14617 16240 -f1 /flats/F1_010/history_010_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 16241 17864 -f1 /flats/F1_011/history_011_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 17865 19488 -f1 /flats/F1_012/history_012_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 19489 21112 -f1 /flats/F1_013/history_013_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 21113 22736 -f1 /flats/F1_014/history_014_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 22737 24360 -f1 /flats/F1_015/history_015_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 24361 25984 -f1 /flats/F1_016/history_016_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 25985 27608 -f1 /flats/F1_017/history_017_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 27609 29232 -f1 /flats/F1_018/history_018_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 29233 30856 -f1 /flats/F1_019/history_019_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 30857 32480 -f1 /flats/F1_020/history_020_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 32481 34104 -f1 /flats/F1_021/history_021_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 34105 35728 -f1 /flats/F1_022/history_022_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 35729 37352 -f1 /flats/F1_023/history_023_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 37353 38976 -f1 /flats/F1_024/history_024_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 38977 40600 -f1 /flats/F1_025/history_025_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 40601 42224 -f1 /flats/F1_026/history_026_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 42225 43848 -f1 /flats/F1_027/history_027_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 43849 45472 -f1 /flats/F1_028/history_028_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 45473 47096 -f1 /flats/F1_029/history_029_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 47097 48720 -f1 /flats/F1_030/history_030_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 48721 50344 -f1 /flats/F1_031/history_031_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 8 -r 50345 51968 -f1 /flats/F1_032/history_032_1.dat
```

DDL/GEN_ITEM_ALL.sh

```
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 5 -f1 /flats/F1_001/item_1.dat
```

DDL/GEN_NEW_ORDER_ALL.sh

```
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 1 1624 -f1 /flats/F1_001/neworder_001_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 1625 3248 -f1 /flats/F1_002/neworder_002_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 3249 4872 -f1 /flats/F1_003/neworder_003_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 4873 6496 -f1 /flats/F1_004/neworder_004_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 6497 8120 -f1 /flats/F1_005/neworder_005_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 8121 9744 -f1 /flats/F1_006/neworder_006_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 9745 11368 -f1 /flats/F1_007/neworder_007_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 11369 12992 -f1 /flats/F1_008/neworder_008_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 12993 14616 -f1 /flats/F1_009/neworder_009_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 14617 16240 -f1 /flats/F1_010/neworder_010_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 16241 17864 -f1 /flats/F1_011/neworder_011_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 17865 19488 -f1 /flats/F1_012/neworder_012_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 19489 21112 -f1 /flats/F1_013/neworder_013_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 21113 22736 -f1 /flats/F1_014/neworder_014_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 22737 24360 -f1 /flats/F1_015/neworder_015_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 24361 25984 -f1 /flats/F1_016/neworder_016_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 25985 27608 -f1 /flats/F1_017/neworder_017_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 27609 29232 -f1 /flats/F1_018/neworder_018_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 29233 30856 -f1 /flats/F1_019/neworder_019_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 30857 32480 -f1 /flats/F1_020/neworder_020_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 32481 34104 -f1 /flats/F1_021/neworder_021_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 34105 35728 -f1 /flats/F1_022/neworder_022_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 35729 37352 -f1 /flats/F1_023/neworder_023_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 37353 38976 -f1 /flats/F1_024/neworder_024_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 38977 40600 -f1 /flats/F1_025/neworder_025_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 40601 42224 -f1 /flats/F1_026/neworder_026_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 42225 43848 -f1 /flats/F1_027/neworder_027_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 43849 45472 -f1 /flats/F1_028/neworder_028_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 45473 47096 -f1 /flats/F1_029/neworder_029_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 47097 48720 -f1 /flats/F1_030/neworder_030_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 48721 50344 -f1 /flats/F1_031/neworder_031_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 11 -r 50345 51968 -f1 /flats/F1_032/neworder_032_1.dat
```

DDL/GEN_ORDERS_ALL.sh

```
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 1 812 -f1 /flats/F1_001/orders_001_1.dat -f2
/flats/F1_001/orderline_001_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 813 1624 -f1 /flats/F1_002/orders_002_1.dat -f2
/flats/F1_002/orderline_002_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 1625 2436 -f1 /flats/F1_003/orders_003_1.dat -f2
/flats/F1_003/orderline_003_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 2437 3248 -f1 /flats/F1_004/orders_004_1.dat -f2
/flats/F1_004/orderline_004_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 3249 4060 -f1 /flats/F1_005/orders_005_1.dat -f2
/flats/F1_005/orderline_005_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 4061 4872 -f1 /flats/F1_006/orders_006_1.dat -f2
/flats/F1_006/orderline_006_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 4873 5684 -f1 /flats/F1_007/orders_007_1.dat -f2
/flats/F1_007/orderline_007_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 5685 6496 -f1 /flats/F1_008/orders_008_1.dat -f2
/flats/F1_008/orderline_008_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 6497 7308 -f1 /flats/F1_009/orders_009_1.dat -f2
/flats/F1_009/orderline_009_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 7309 8120 -f1 /flats/F1_010/orders_010_1.dat -f2
/flats/F1_010/orderline_010_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 8121 8932 -f1 /flats/F1_011/orders_011_1.dat -f2
/flats/F1_011/orderline_011_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 8933 9744 -f1 /flats/F1_012/orders_012_1.dat -f2
/flats/F1_012/orderline_012_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 9745 10556 -f1 /flats/F1_013/orders_013_1.dat -f2
/flats/F1_013/orderline_013_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 10557 11368 -f1 /flats/F1_014/orders_014_1.dat -f2
/flats/F1_014/orderline_014_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 11369 12180 -f1 /flats/F1_015/orders_015_1.dat -f2
/flats/F1_015/orderline_015_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 12181 12992 -f1 /flats/F1_016/orders_016_1.dat -f2
/flats/F1_016/orderline_016_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 12993 13804 -f1 /flats/F1_017/orders_017_1.dat -f2
/flats/F1_017/orderline_017_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 13805 14616 -f1 /flats/F1_018/orders_018_1.dat -f2
/flats/F1_018/orderline_018_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 14617 15428 -f1 /flats/F1_019/orders_019_1.dat -f2
/flats/F1_019/orderline_019_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 15429 16240 -f1 /flats/F1_020/orders_020_1.dat -f2
/flats/F1_020/orderline_020_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 16241 17052 -f1 /flats/F1_021/orders_021_1.dat -f2
/flats/F1_021/orderline_021_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 17053 17864 -f1 /flats/F1_022/orders_022_1.dat -f2
/flats/F1_022/orderline_022_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 17865 18676 -f1 /flats/F1_023/orders_023_1.dat -f2
/flats/F1_023/orderline_023_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 18677 19488 -f1 /flats/F1_024/orders_024_1.dat -f2
/flats/F1_024/orderline_024_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 19489 20300 -f1 /flats/F1_025/orders_025_1.dat -f2
/flats/F1_025/orderline_025_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 20301 21112 -f1 /flats/F1_026/orders_026_1.dat -f2
/flats/F1_026/orderline_026_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 21113 21924 -f1 /flats/F1_027/orders_027_1.dat -f2
/flats/F1_027/orderline_027_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 21925 22736 -f1 /flats/F1_028/orders_028_1.dat -f2
/flats/F1_028/orderline_028_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 22737 23548 -f1 /flats/F1_029/orders_029_1.dat -f2
/flats/F1_029/orderline_029_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 23549 24360 -f1 /flats/F1_030/orders_030_1.dat -f2
/flats/F1_030/orderline_030_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 24361 25172 -f1 /flats/F1_031/orders_031_1.dat -f2
/flats/F1_031/orderline_031_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 25173 25984 -f1 /flats/F1_032/orders_032_1.dat -f2
/flats/F1_032/orderline_032_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 25985 26796 -f1 /flats/F1_033/orders_033_1.dat -f2
/flats/F1_033/orderline_033_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 26797 27608 -f1 /flats/F1_034/orders_034_1.dat -f2
/flats/F1_034/orderline_034_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 27609 28420 -f1 /flats/F1_035/orders_035_1.dat -f2
/flats/F1_035/orderline_035_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 28421 29232 -f1 /flats/F1_036/orders_036_1.dat -f2
/flats/F1_036/orderline_036_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 29233 30044 -f1 /flats/F1_037/orders_037_1.dat -f2
/flats/F1_037/orderline_037_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 30045 30856 -f1 /flats/F1_038/orders_038_1.dat -f2
/flats/F1_038/orderline_038_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 30857 31668 -f1 /flats/F1_039/orders_039_1.dat -f2
/flats/F1_039/orderline_039_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 31669 32480 -f1 /flats/F1_040/orders_040_1.dat -f2
/flats/F1_040/orderline_040_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 32481 33292 -f1 /flats/F1_041/orders_041_1.dat -f2
/flats/F1_041/orderline_041_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 33293 34104 -f1 /flats/F1_042/orders_042_1.dat -f2
/flats/F1_042/orderline_042_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 34105 34916 -f1 /flats/F1_043/orders_043_1.dat -f2
/flats/F1_043/orderline_043_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 34917 35728 -f1 /flats/F1_044/orders_044_1.dat -f2
/flats/F1_044/orderline_044_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 35729 36540 -f1 /flats/F1_045/orders_045_1.dat -f2
/flats/F1_045/orderline_045_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 36541 37352 -f1 /flats/F1_046/orders_046_1.dat -f2
/flats/F1_046/orderline_046_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 37353 38164 -f1 /flats/F1_047/orders_047_1.dat -f2
/flats/F1_047/orderline_047_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 38165 38976 -f1 /flats/F1_048/orders_048_1.dat -f2
/flats/F1_048/orderline_048_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 38977 39788 -f1 /flats/F1_049/orders_049_1.dat -f2
/flats/F1_049/orderline_049_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 39789 40600 -f1 /flats/F1_050/orders_050_1.dat -f2
/flats/F1_050/orderline_050_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 40601 41412 -f1 /flats/F1_051/orders_051_1.dat -f2
/flats/F1_051/orderline_051_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 41413 42224 -f1 /flats/F1_052/orders_052_1.dat -f2
/flats/F1_052/orderline_052_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 42225 43036 -f1 /flats/F1_053/orders_053_1.dat -f2
/flats/F1_053/orderline_053_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 43037 43848 -f1 /flats/F1_054/orders_054_1.dat -f2
/flats/F1_054/orderline_054_1.dat
```

```
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 43849 44660 -f1 /flats/F1_055/orders_055_1.dat -f2
/flats/F1_055/orderline_055_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 44661 45472 -f1 /flats/F1_056/orders_056_1.dat -f2
/flats/F1_056/orderline_056_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 45473 46284 -f1 /flats/F1_057/orders_057_1.dat -f2
/flats/F1_057/orderline_057_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 46285 47096 -f1 /flats/F1_058/orders_058_1.dat -f2
/flats/F1_058/orderline_058_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 47097 47908 -f1 /flats/F1_059/orders_059_1.dat -f2
/flats/F1_059/orderline_059_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 47909 48720 -f1 /flats/F1_060/orders_060_1.dat -f2
/flats/F1_060/orderline_060_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 48721 49532 -f1 /flats/F1_061/orders_061_1.dat -f2
/flats/F1_061/orderline_061_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 49533 50344 -f1 /flats/F1_062/orders_062_1.dat -f2
/flats/F1_062/orderline_062_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 50345 51156 -f1 /flats/F1_063/orders_063_1.dat -f2
/flats/F1_063/orderline_063_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 9 -r 51157 51968 -f1 /flats/F1_064/orders_064_1.dat -f2
/flats/F1_064/orderline_064_1.dat
```

DDL/GEN_STOCK_ALL.sh

```
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 1 812 -f1 /flats/F1_001/stock_001_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 813 1624 -f1 /flats/F1_002/stock_002_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 1625 2436 -f1 /flats/F1_003/stock_003_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 2437 3248 -f1 /flats/F1_004/stock_004_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 3249 4060 -f1 /flats/F1_005/stock_005_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 4061 4872 -f1 /flats/F1_006/stock_006_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 4873 5684 -f1 /flats/F1_007/stock_007_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 5685 6496 -f1 /flats/F1_008/stock_008_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 6497 7308 -f1 /flats/F1_009/stock_009_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 7309 8120 -f1 /flats/F1_010/stock_010_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 8121 8932 -f1 /flats/F1_011/stock_011_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 8933 9744 -f1 /flats/F1_012/stock_012_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 9745 10556 -f1 /flats/F1_013/stock_013_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 10557 11368 -f1 /flats/F1_014/stock_014_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 11369 12180 -f1 /flats/F1_015/stock_015_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 12181 12992 -f1 /flats/F1_016/stock_016_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 12993 13804 -f1 /flats/F1_017/stock_017_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 13805 14616 -f1 /flats/F1_018/stock_018_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 14617 15428 -f1 /flats/F1_019/stock_019_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 15429 16240 -f1 /flats/F1_020/stock_020_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 16241 17052 -f1 /flats/F1_021/stock_021_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 17053 17864 -f1 /flats/F1_022/stock_022_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 17865 18676 -f1 /flats/F1_023/stock_023_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 18677 19488 -f1 /flats/F1_024/stock_024_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 19489 20300 -f1 /flats/F1_025/stock_025_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 20301 21112 -f1 /flats/F1_026/stock_026_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 21113 21924 -f1 /flats/F1_027/stock_027_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 21925 22736 -f1 /flats/F1_028/stock_028_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 22737 23548 -f1 /flats/F1_029/stock_029_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 23549 24360 -f1 /flats/F1_030/stock_030_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 24361 25172 -f1 /flats/F1_031/stock_031_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 25173 25984 -f1 /flats/F1_032/stock_032_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 25985 26796 -f1 /flats/F1_033/stock_033_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 26797 27608 -f1 /flats/F1_034/stock_034_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 27609 28420 -f1 /flats/F1_035/stock_035_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 28421 29232 -f1 /flats/F1_036/stock_036_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 29233 30044 -f1 /flats/F1_037/stock_037_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 30045 30856 -f1 /flats/F1_038/stock_038_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 30857 31668 -f1 /flats/F1_039/stock_039_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 31669 32480 -f1 /flats/F1_040/stock_040_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 32481 33292 -f1 /flats/F1_041/stock_041_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 33293 34104 -f1 /flats/F1_042/stock_042_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 34105 34916 -f1 /flats/F1_043/stock_043_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 34917 35728 -f1 /flats/F1_044/stock_044_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 35729 36540 -f1 /flats/F1_045/stock_045_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 36541 37352 -f1 /flats/F1_046/stock_046_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 37353 38164 -f1 /flats/F1_047/stock_047_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 38165 38976 -f1 /flats/F1_048/stock_048_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 38977 39788 -f1 /flats/F1_049/stock_049_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 39789 40600 -f1 /flats/F1_050/stock_050_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 40601 41412 -f1 /flats/F1_051/stock_051_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 41413 42224 -f1 /flats/F1_052/stock_052_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 42225 43036 -f1 /flats/F1_053/stock_053_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 43037 43848 -f1 /flats/F1_054/stock_054_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 43849 44660 -f1 /flats/F1_055/stock_055_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 44661 45472 -f1 /flats/F1_056/stock_056_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 45473 46284 -f1 /flats/F1_057/stock_057_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 46285 47096 -f1 /flats/F1_058/stock_058_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 47097 47908 -f1 /flats/F1_059/stock_059_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 47909 48720 -f1 /flats/F1_060/stock_060_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 48721 49532 -f1 /flats/F1_061/stock_061_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 49533 50344 -f1 /flats/F1_062/stock_062_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 50345 51156 -f1 /flats/F1_063/stock_063_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 6 -r 51157 51968 -f1 /flats/F1_064/stock_064_1.dat
```

DDL/GEN_WAREHOUSE_ALL.sh

```
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 1 1624 -f1 /flats/F1_001/warehouse_001_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 1625 3248 -f1 /flats/F1_002/warehouse_002_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 3249 4872 -f1 /flats/F1_003/warehouse_003_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 4873 6496 -f1 /flats/F1_004/warehouse_004_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 6497 8120 -f1 /flats/F1_005/warehouse_005_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 8121 9744 -f1 /flats/F1_006/warehouse_006_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 9745 11368 -f1 /flats/F1_007/warehouse_007_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 11369 12992 -f1 /flats/F1_008/warehouse_008_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 12993 14616 -f1 /flats/F1_009/warehouse_009_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 14617 16240 -f1 /flats/F1_010/warehouse_010_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 16241 17864 -f1 /flats/F1_011/warehouse_011_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 17865 19488 -f1 /flats/F1_012/warehouse_012_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 19489 21112 -f1 /flats/F1_013/warehouse_013_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 21113 22736 -f1 /flats/F1_014/warehouse_014_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 22737 24360 -f1 /flats/F1_015/warehouse_015_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 24361 25984 -f1 /flats/F1_016/warehouse_016_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 25985 27608 -f1 /flats/F1_017/warehouse_017_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 27609 29232 -f1 /flats/F1_018/warehouse_018_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 29233 30856 -f1 /flats/F1_019/warehouse_019_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 30857 32480 -f1 /flats/F1_020/warehouse_020_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 32481 34104 -f1 /flats/F1_021/warehouse_021_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 34105 35728 -f1 /flats/F1_022/warehouse_022_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 35729 37352 -f1 /flats/F1_023/warehouse_023_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 37353 38976 -f1 /flats/F1_024/warehouse_024_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 38977 40600 -f1 /flats/F1_025/warehouse_025_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 40601 42224 -f1 /flats/F1_026/warehouse_026_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 42225 43848 -f1 /flats/F1_027/warehouse_027_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 43849 45472 -f1 /flats/F1_028/warehouse_028_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 45473 47096 -f1 /flats/F1_029/warehouse_029_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 47097 48720 -f1 /flats/F1_030/warehouse_030_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 48721 50344 -f1 /flats/F1_031/warehouse_031_1.dat
/home/memmerto/eHV8/ddlgen/dbgen/gendata -t 3 -r 50345 51968 -f1 /flats/F1_032/warehouse_032_1.dat
```

DDL/LOAD_CUSTOMER_ALL.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER1 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_001/customer_001_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER1;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER2 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_002/customer_002_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER2;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER3 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_003/customer_003_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER3;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER4 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_004/customer_004_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER4;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER5 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_005/customer_005_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER5;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER6 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_006/customer_006_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER6;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER7 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_007/customer_007_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER7;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER8 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_008/customer_008_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER8;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER9 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_009/customer_009_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER9;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER10 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_010/customer_010_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER10;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER11 ACTIVATE NOT LOGGED INITIALLY;
```

```
IMPORT FROM /flats/F1_011/customer_011_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER11;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER12 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_012/customer_012_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER12;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER13 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_013/customer_013_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER13;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER14 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_014/customer_014_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER14;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER15 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_015/customer_015_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER15;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER16 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_016/customer_016_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER16;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER17 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_017/customer_017_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER17;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER18 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_018/customer_018_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER18;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER19 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_019/customer_019_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER19;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER20 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_020/customer_020_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER20;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER21 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_021/customer_021_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER21;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER22 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_022/customer_022_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER22;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER23 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_023/customer_023_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER23;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER24 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_024/customer_024_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER24;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER25 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_025/customer_025_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER25;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER26 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_026/customer_026_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER26;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER27 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_027/customer_027_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER27;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER28 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_028/customer_028_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER28;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER29 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_029/customer_029_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER29;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER30 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_030/customer_030_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER30;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER31 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_031/customer_031_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER31;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER32 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_032/customer_032_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER32;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER33 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_033/customer_033_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER33;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER34 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_034/customer_034_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER34;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER35 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_035/customer_035_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER35;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER36 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_036/customer_036_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER36;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER37 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_037/customer_037_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER37;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER38 ACTIVATE NOT LOGGED INITIALLY;
```

```
IMPORT FROM /flats/F1_038/customer_038_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER38;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER39 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_039/customer_039_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER39;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER40 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_040/customer_040_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER40;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER41 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_041/customer_041_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER41;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER42 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_042/customer_042_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER42;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER43 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_043/customer_043_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER43;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER44 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_044/customer_044_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER44;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER45 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_045/customer_045_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER45;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER46 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_046/customer_046_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER46;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER47 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_047/customer_047_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER47;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER48 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_048/customer_048_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER48;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER49 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_049/customer_049_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER49;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER50 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_050/customer_050_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER50;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER51 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_051/customer_051_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER51;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER52 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_052/customer_052_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER52;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER53 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_053/customer_053_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER53;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER54 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_054/customer_054_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER54;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER55 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_055/customer_055_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER55;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER56 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_056/customer_056_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER56;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER57 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_057/customer_057_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER57;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER58 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_058/customer_058_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER58;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER59 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_059/customer_059_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER59;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER60 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_060/customer_060_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER60;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER61 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_061/customer_061_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER61;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER62 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_062/customer_062_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER62;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER63 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_063/customer_063_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER63;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER64 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_064/customer_064_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO CUSTOMER64;
COMMIT WORK;
CONNECT RESET;
```

DDL/LOAD_DISTRICT_ALL.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_001/district_001_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT1;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_002/district_002_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT2;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_003/district_003_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT3;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_004/district_004_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT4;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_005/district_005_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT5;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_006/district_006_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT6;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_007/district_007_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT7;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_008/district_008_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT8;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_009/district_009_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT9;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_010/district_010_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT10;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_011/district_011_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT11;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_012/district_012_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT12;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_013/district_013_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT13;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_014/district_014_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT14;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_015/district_015_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT15;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_016/district_016_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT16;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_017/district_017_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT17;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_018/district_018_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT18;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_019/district_019_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT19;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_020/district_020_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT20;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_021/district_021_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT21;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_022/district_022_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT22;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_023/district_023_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT23;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_024/district_024_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT24;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_025/district_025_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT25;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_026/district_026_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT26;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_027/district_027_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT27;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_028/district_028_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT28;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_029/district_029_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT29;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_030/district_030_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT30;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_031/district_031_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT31;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_032/district_032_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO DISTRICT32;
COMMIT WORK;
CONNECT RESET;
```

DDL/LOAD_HISTORY_ALL.ddl

```
connect to TPCC in share mode;
LOAD FROM /flats/F1_001/history_001_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY1 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM /flats/F1_002/history_002_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY2 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
```

```
LOAD FROM /flats/F1_003/history_003_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY3 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM /flats/F1_004/history_004_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY4 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM /flats/F1_005/history_005_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY5 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM /flats/F1_006/history_006_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY6 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM /flats/F1_007/history_007_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY7 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM /flats/F1_008/history_008_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY8 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM /flats/F1_009/history_009_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY9 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM /flats/F1_010/history_010_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY10 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM /flats/F1_011/history_011_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY11 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM /flats/F1_012/history_012_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY12 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM /flats/F1_013/history_013_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY13 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM /flats/F1_014/history_014_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY14 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM /flats/F1_015/history_015_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY15 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM /flats/F1_016/history_016_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY16 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM /flats/F1_017/history_017_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY17 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM /flats/F1_018/history_018_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY18 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM /flats/F1_019/history_019_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY19 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM /flats/F1_020/history_020_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY20 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM /flats/F1_021/history_021_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY21 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM /flats/F1_022/history_022_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY22 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM /flats/F1_023/history_023_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY23 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM /flats/F1_024/history_024_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY24 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM /flats/F1_025/history_025_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY25 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM /flats/F1_026/history_026_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY26 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM /flats/F1_027/history_027_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY27 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM /flats/F1_028/history_028_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY28 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM /flats/F1_029/history_029_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY29 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM /flats/F1_030/history_030_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY30 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM /flats/F1_031/history_031_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY31 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM /flats/F1_032/history_032_1.dat OF DEL MODIFIED BY COLDEL| KEEPBLANKS
FASTPARSE REPLACE INTO HISTORY32 NONRECOVERABLE DATA BUFFER 16000
CPU_PARALLELISM 4 ;
connect reset;
```

DDL/LOAD_ITEM_ALL.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_001/item_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
1000 INSERT INTO ITEM;
COMMIT WORK;
CONNECT RESET;
```

DDL/LOAD_NEW_ORDER_ALL.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_001/neworder_001_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA1;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_002/neworder_002_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA2;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_003/neworder_003_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA3;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_004/neworder_004_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA4;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_005/neworder_005_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA5;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_006/neworder_006_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA6;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_007/neworder_007_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA7;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_008/neworder_008_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA8;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
```

```
IMPORT FROM /flats/F1_009/neworder_009_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA9;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_010/neworder_010_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA10;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_011/neworder_011_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA11;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_012/neworder_012_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA12;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_013/neworder_013_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA13;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_014/neworder_014_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA14;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_015/neworder_015_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA15;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_016/neworder_016_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA16;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_017/neworder_017_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA17;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_018/neworder_018_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA18;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_019/neworder_019_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA19;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_020/neworder_020_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA20;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_021/neworder_021_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA21;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_022/neworder_022_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA22;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_023/neworder_023_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA23;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_024/neworder_024_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA24;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_025/neworder_025_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA25;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_026/neworder_026_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA26;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_027/neworder_027_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA27;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_028/neworder_028_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA28;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_029/neworder_029_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA29;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_030/neworder_030_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA30;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_031/neworder_031_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA31;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_032/neworder_032_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 20000 INSERT INTO NEW_ORDERA32;
COMMIT WORK;
CONNECT RESET;
```

DDL/LOAD_ORDERS_ALL.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS1 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_001/orders_001_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS1;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS2 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_002/orders_002_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS2;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS3 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_003/orders_003_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS3;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS4 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_004/orders_004_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS4;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS5 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_005/orders_005_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS5;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS6 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_006/orders_006_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS6;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS7 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_007/orders_007_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS7;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS8 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_008/orders_008_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS8;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS9 ACTIVATE NOT LOGGED INITIALLY;
```

```
IMPORT FROM /flats/F1_009/orders_009_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS9;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS10 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_010/orders_010_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS10;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS11 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_011/orders_011_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS11;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS12 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_012/orders_012_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS12;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS13 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_013/orders_013_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS13;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS14 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_014/orders_014_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS14;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS15 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_015/orders_015_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS15;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS16 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_016/orders_016_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS16;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS17 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_017/orders_017_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS17;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS18 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_018/orders_018_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS18;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS19 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_019/orders_019_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS19;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS20 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_020/orders_020_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS20;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS21 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_021/orders_021_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS21;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS22 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_022/orders_022_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS22;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS23 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_023/orders_023_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS23;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS24 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_024/orders_024_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS24;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS25 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_025/orders_025_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS25;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS26 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_026/orders_026_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS26;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS27 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_027/orders_027_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS27;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS28 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_028/orders_028_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS28;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS29 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_029/orders_029_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS29;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS30 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_030/orders_030_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS30;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS31 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_031/orders_031_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS31;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS32 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_032/orders_032_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS32;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS33 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_033/orders_033_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS33;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS34 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_034/orders_034_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS34;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS35 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_035/orders_035_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS35;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS36 ACTIVATE NOT LOGGED INITIALLY;
```

```
IMPORT FROM /flats/F1_036/orders_036_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS36;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS37 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_037/orders_037_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS37;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS38 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_038/orders_038_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS38;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS39 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_039/orders_039_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS39;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS40 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_040/orders_040_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS40;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS41 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_041/orders_041_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS41;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS42 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_042/orders_042_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS42;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS43 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_043/orders_043_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS43;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS44 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_044/orders_044_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS44;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS45 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_045/orders_045_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS45;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS46 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_046/orders_046_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS46;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS47 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_047/orders_047_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS47;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS48 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_048/orders_048_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS48;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS49 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_049/orders_049_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS49;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS50 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_050/orders_050_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS50;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS51 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_051/orders_051_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS51;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS52 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_052/orders_052_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS52;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS53 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_053/orders_053_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS53;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS54 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_054/orders_054_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS54;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS55 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_055/orders_055_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS55;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS56 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_056/orders_056_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS56;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS57 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_057/orders_057_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS57;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS58 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_058/orders_058_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS58;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS59 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_059/orders_059_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS59;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS60 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_060/orders_060_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS60;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS61 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_061/orders_061_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS61;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS62 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_062/orders_062_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS62;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS63 ACTIVATE NOT LOGGED INITIALLY;
```

```
IMPORT FROM /flats/F1_063/orders_063_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS63;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS64 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_064/orders_064_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
24360000 INSERT INTO ORDERS64;
COMMIT WORK;
CONNECT RESET;
```

DDL/LOAD_ORDER_LINE_ALL.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE1 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_001/orderline_001_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE1;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE2 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_002/orderline_002_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE2;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE3 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_003/orderline_003_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE3;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE4 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_004/orderline_004_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE4;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE5 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_005/orderline_005_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE5;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE6 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_006/orderline_006_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE6;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE7 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_007/orderline_007_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE7;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE8 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_008/orderline_008_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE8;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE9 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_009/orderline_009_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE9;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE10 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_010/orderline_010_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE10;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE11 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_011/orderline_011_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE11;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE12 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_012/orderline_012_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE12;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE13 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_013/orderline_013_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE13;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE14 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_014/orderline_014_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE14;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE15 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_015/orderline_015_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE15;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE16 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_016/orderline_016_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE16;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE17 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_017/orderline_017_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE17;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE18 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_018/orderline_018_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE18;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE19 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_019/orderline_019_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE19;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE20 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_020/orderline_020_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE20;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE21 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_021/orderline_021_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE21;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE22 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_022/orderline_022_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE22;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE23 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_023/orderline_023_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE23;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE24 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_024/orderline_024_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE24;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE25 ACTIVATE NOT LOGGED INITIALLY;
```

```
IMPORT FROM /flats/F1_025/orderline_025_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE25;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE26 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_026/orderline_026_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE26;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE27 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_027/orderline_027_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE27;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE28 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_028/orderline_028_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE28;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE29 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_029/orderline_029_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE29;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE30 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_030/orderline_030_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE30;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE31 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_031/orderline_031_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE31;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE32 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_032/orderline_032_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE32;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE33 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_033/orderline_033_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE33;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE34 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_034/orderline_034_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE34;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE35 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_035/orderline_035_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE35;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE36 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_036/orderline_036_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE36;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE37 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_037/orderline_037_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE37;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE38 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_038/orderline_038_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE38;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE39 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_039/orderline_039_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE39;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE40 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_040/orderline_040_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE40;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE41 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_041/orderline_041_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE41;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE42 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_042/orderline_042_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE42;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE43 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_043/orderline_043_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE43;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE44 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_044/orderline_044_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE44;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE45 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_045/orderline_045_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE45;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE46 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_046/orderline_046_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE46;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE47 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_047/orderline_047_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE47;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE48 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_048/orderline_048_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE48;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE49 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_049/orderline_049_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE49;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE50 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_050/orderline_050_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE50;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE51 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_051/orderline_051_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE51;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE52 ACTIVATE NOT LOGGED INITIALLY;
```

```
IMPORT FROM /flats/F1_052/orderline_052_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE52;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE53 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_053/orderline_053_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE53;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE54 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_054/orderline_054_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE54;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE55 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_055/orderline_055_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE55;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE56 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_056/orderline_056_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE56;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE57 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_057/orderline_057_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE57;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE58 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_058/orderline_058_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE58;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE59 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_059/orderline_059_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE59;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE60 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_060/orderline_060_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE60;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE61 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_061/orderline_061_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE61;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE62 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_062/orderline_062_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE62;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE63 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_063/orderline_063_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE63;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE64 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_064/orderline_064_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
267960000 INSERT INTO ORDER_LINE64;
COMMIT WORK;
CONNECT RESET;
```

DDL/LOAD_STOCK_ALL.ddl

```
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK1 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_001/stock_001_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK1;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK2 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_002/stock_002_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK2;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK3 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_003/stock_003_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK3;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK4 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_004/stock_004_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK4;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK5 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_005/stock_005_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK5;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK6 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_006/stock_006_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK6;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK7 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_007/stock_007_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK7;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK8 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_008/stock_008_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK8;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK9 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_009/stock_009_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK9;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK10 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_010/stock_010_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK10;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK11 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_011/stock_011_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK11;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK12 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_012/stock_012_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK12;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK13 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_013/stock_013_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK13;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK14 ACTIVATE NOT LOGGED INITIALLY;
```

```
IMPORT FROM /flats/F1_014/stock_014_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK14;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK15 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_015/stock_015_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK15;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK16 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_016/stock_016_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK16;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK17 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_017/stock_017_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK17;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK18 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_018/stock_018_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK18;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK19 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_019/stock_019_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK19;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK20 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_020/stock_020_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK20;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK21 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_021/stock_021_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK21;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK22 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_022/stock_022_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK22;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK23 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_023/stock_023_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK23;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK24 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_024/stock_024_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK24;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK25 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_025/stock_025_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK25;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK26 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_026/stock_026_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK26;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK27 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_027/stock_027_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK27;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK28 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_028/stock_028_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK28;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK29 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_029/stock_029_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK29;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK30 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_030/stock_030_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK30;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK31 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_031/stock_031_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK31;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK32 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_032/stock_032_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK32;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK33 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_033/stock_033_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK33;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK34 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_034/stock_034_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK34;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK35 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_035/stock_035_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK35;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK36 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_036/stock_036_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK36;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK37 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_037/stock_037_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK37;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK38 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_038/stock_038_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK38;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK39 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_039/stock_039_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK39;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK40 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_040/stock_040_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK40;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK41 ACTIVATE NOT LOGGED INITIALLY;
```

```
IMPORT FROM /flats/F1_041/stock_041_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK41;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK42 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_042/stock_042_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK42;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK43 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_043/stock_043_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK43;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK44 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_044/stock_044_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK44;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK45 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_045/stock_045_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK45;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK46 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_046/stock_046_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK46;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK47 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_047/stock_047_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK47;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK48 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_048/stock_048_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK48;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK49 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_049/stock_049_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK49;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK50 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_050/stock_050_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK50;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK51 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_051/stock_051_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK51;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK52 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_052/stock_052_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK52;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK53 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_053/stock_053_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK53;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK54 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_054/stock_054_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK54;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK55 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_055/stock_055_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK55;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK56 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_056/stock_056_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK56;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK57 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_057/stock_057_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK57;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK58 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_058/stock_058_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK58;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK59 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_059/stock_059_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK59;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK60 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_060/stock_060_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK60;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK61 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_061/stock_061_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK61;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK62 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_062/stock_062_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK62;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK63 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_063/stock_063_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK63;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK64 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_064/stock_064_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50  COMMITCOUNT
81200000 INSERT INTO STOCK64;
COMMIT WORK;
CONNECT RESET;
```

DDL/LOAD_WAREHOUSE_ALL.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_001/warehouse_001_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE1;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_002/warehouse_002_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE2;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_003/warehouse_003_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE3;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
```

```
IMPORT FROM /flats/F1_004/warehouse_004_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE4;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_005/warehouse_005_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE5;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_006/warehouse_006_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE6;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_007/warehouse_007_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE7;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_008/warehouse_008_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE8;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_009/warehouse_009_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE9;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_010/warehouse_010_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE10;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_011/warehouse_011_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE11;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_012/warehouse_012_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE12;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_013/warehouse_013_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE13;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_014/warehouse_014_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE14;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_015/warehouse_015_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE15;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_016/warehouse_016_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE16;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_017/warehouse_017_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE17;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_018/warehouse_018_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE18;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_019/warehouse_019_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE19;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_020/warehouse_020_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE20;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_021/warehouse_021_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE21;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_022/warehouse_022_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE22;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_023/warehouse_023_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE23;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_024/warehouse_024_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE24;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_025/warehouse_025_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE25;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_026/warehouse_026_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE26;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_027/warehouse_027_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE27;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_028/warehouse_028_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE28;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_029/warehouse_029_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE29;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_030/warehouse_030_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE30;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_031/warehouse_031_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE31;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_032/warehouse_032_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE32;
COMMIT WORK;
CONNECT RESET;
```

DDL/RNST_CUSTOMER.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER1 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER2 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER3 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER4 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER5 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER6 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER7 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER8 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER9 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
```

```
RUNSTATS ON TABLE memmerto.CUSTOMER10 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER11 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER12 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER13 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER14 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER15 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER16 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER17 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER18 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER19 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER20 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER21 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER22 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER23 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER24 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER25 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER26 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER27 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER28 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER29 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER30 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER31 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER32 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER33 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER34 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER35 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER36 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER37 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER38 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER39 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER40 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER41 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER42 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER43 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER44 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER45 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER46 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER47 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER48 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER49 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER50 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER51 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER52 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER53 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER54 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER55 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER56 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER57 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER58 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER59 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER60 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER61 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER62 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER63 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.CUSTOMER64 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

DDL/RNST_DISTRICT.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT1 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT2 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT3 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT4 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT5 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT6 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT7 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT8 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT9 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT10 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT11 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT12 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT13 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT14 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT15 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT16 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT17 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT18 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT19 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT20 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT21 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT22 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT23 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT24 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT25 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT26 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT27 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT28 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT29 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT30 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT31 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.DISTRICT32 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

DDL/RNST_HISTORY.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY1 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY2 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY3 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY4 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY5 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY6 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY7 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY8 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY9 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY10 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY11 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY12 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY13 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY14 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY15 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY16 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY17 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY18 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY19 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
```

```
RUNSTATS ON TABLE memmerto.HISTORY20 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY21 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY22 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY23 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY24 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY25 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY26 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY27 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY28 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY29 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY30 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY31 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.HISTORY32 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

DDL/RNST_ITEM.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ITEM AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

DDL/RNST_NEW_ORDERA.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA1 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA2 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA3 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA4 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA5 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA6 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA7 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA8 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA9 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA10 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA11 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA12 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA13 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA14 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA15 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA16 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA17 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA18 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA19 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA20 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA21 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA22 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA23 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA24 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA25 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA26 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA27 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA28 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA29 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA30 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA31 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERA32 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

DDL/RNST_NEW_ORDERB.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB1 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB2 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB3 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB4 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

```
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB5 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB6 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB7 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB8 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB9 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB10 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB11 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB12 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB13 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB14 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB15 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB16 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB17 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB18 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB19 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB20 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB21 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB22 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB23 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB24 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB25 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB26 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB27 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB28 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB29 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB30 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB31 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.NEW_ORDERB32 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

DDL/RNST_ORDERS.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS1 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS2 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS3 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS4 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS5 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS6 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS7 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS8 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS9 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS10 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS11 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS12 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS13 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS14 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS15 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS16 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS17 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS18 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS19 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS20 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS21 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS22 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS23 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS24 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS25 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

```
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS26 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS27 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS28 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS29 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS30 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS31 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS32 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS33 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS34 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS35 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS36 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS37 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS38 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS39 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS40 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS41 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS42 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS43 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS44 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS45 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS46 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS47 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS48 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS49 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS50 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS51 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS52 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS53 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS54 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS55 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS56 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS57 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS58 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS59 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS60 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS61 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS62 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS63 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDERS64 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

DDL/RNST_ORDER_LINE.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE1 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE2 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE3 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE4 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE5 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE6 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE7 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE8 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE9 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE10 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE11 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE12 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE13 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE14 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

```
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE15 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE16 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE17 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE18 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE19 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE20 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE21 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE22 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE23 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE24 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE25 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE26 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE27 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE28 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE29 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE30 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE31 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE32 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE33 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE34 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE35 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE36 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE37 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE38 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE39 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE40 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE41 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE42 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE43 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE44 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE45 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE46 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE47 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE48 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE49 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE50 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE51 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE52 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE53 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE54 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE55 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE56 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE57 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE58 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE59 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE60 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE61 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE62 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE63 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.ORDER_LINE64 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

DDL/RNST_STOCK.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK1 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK2 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK3 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

```
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK4 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK5 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK6 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK7 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK8 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK9 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK10 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK11 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK12 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK13 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK14 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK15 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK16 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK17 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK18 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK19 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK20 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK21 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK22 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK23 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK24 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK25 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK26 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK27 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK28 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK29 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK30 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK31 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK32 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK33 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK34 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK35 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK36 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK37 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK38 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK39 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK40 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK41 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK42 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK43 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK44 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK45 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK46 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK47 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK48 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK49 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK50 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK51 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK52 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK53 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK54 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK55 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK56 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK57 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK58 AND INDEXES ALL;
COMMIT WORK;
```

```
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK59 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK60 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK61 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK62 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK63 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.STOCK64 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

_DDL/RNST_WAREHOUSE.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE1 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE2 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE3 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE4 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE5 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE6 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE7 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE8 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE9 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE10 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE11 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE12 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE13 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE14 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE15 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE16 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE17 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE18 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE19 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE20 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE21 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE22 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE23 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE24 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE25 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE26 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE27 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE28 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE29 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE30 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE31 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE memmerto.WAREHOUSE32 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

_bp/alter_bufferpool.ddl

```
connect to tpcc;
ALTER BUFFERPOOL WAR1 DEFERRED SIZE 380;
ALTER BUFFERPOOL WAR2 DEFERRED SIZE 380;
ALTER BUFFERPOOL WAR3 DEFERRED SIZE 380;
ALTER BUFFERPOOL WAR4 DEFERRED SIZE 380;
ALTER BUFFERPOOL DIS1 DEFERRED SIZE 4000;
ALTER BUFFERPOOL DIS2 DEFERRED SIZE 4000;
ALTER BUFFERPOOL DIS3 DEFERRED SIZE 4000;
ALTER BUFFERPOOL DIS4 DEFERRED SIZE 4000;
ALTER BUFFERPOOL ITM DEFERRED SIZE 1240;
ALTER BUFFERPOOL HST1 DEFERRED SIZE 678;
ALTER BUFFERPOOL HST2 DEFERRED SIZE 678;
ALTER BUFFERPOOL HST3 DEFERRED SIZE 678;
ALTER BUFFERPOOL HST4 DEFERRED SIZE 678;
ALTER BUFFERPOOL NEW1 DEFERRED SIZE 419850;
ALTER BUFFERPOOL NEW2 DEFERRED SIZE 419850;
ALTER BUFFERPOOL NEW3 DEFERRED SIZE 419850;
ALTER BUFFERPOOL NEW4 DEFERRED SIZE 419850;
ALTER BUFFERPOOL ORD1 DEFERRED SIZE 298470;
ALTER BUFFERPOOL ORD2 DEFERRED SIZE 298470;
ALTER BUFFERPOOL ORD3 DEFERRED SIZE 298470;
ALTER BUFFERPOOL ORD4 DEFERRED SIZE 298470;
ALTER BUFFERPOOL ORD_I1 DEFERRED SIZE 851376;
ALTER BUFFERPOOL ORD_I2 DEFERRED SIZE 851376;
ALTER BUFFERPOOL ORD_I3 DEFERRED SIZE 851376;
ALTER BUFFERPOOL ORD_I4 DEFERRED SIZE 851376;
ALTER BUFFERPOOL OLN1 DEFERRED SIZE 520080;
ALTER BUFFERPOOL OLN2 DEFERRED SIZE 520080;
ALTER BUFFERPOOL OLN3 DEFERRED SIZE 520080;
ALTER BUFFERPOOL OLN4 DEFERRED SIZE 520080;
ALTER BUFFERPOOL CST1 DEFERRED SIZE 110000;
ALTER BUFFERPOOL CST2 DEFERRED SIZE 110000;
ALTER BUFFERPOOL CST3 DEFERRED SIZE 110000;
ALTER BUFFERPOOL CST4 DEFERRED SIZE 110000;
ALTER BUFFERPOOL CST_I1 DEFERRED SIZE 468072;
ALTER BUFFERPOOL CST_I2 DEFERRED SIZE 468072;
ALTER BUFFERPOOL CST_I3 DEFERRED SIZE 468072;
ALTER BUFFERPOOL CST_I4 DEFERRED SIZE 468072;
ALTER BUFFERPOOL STK1 DEFERRED SIZE 10436503;
ALTER BUFFERPOOL STK2 DEFERRED SIZE 10436503;
ALTER BUFFERPOOL STK3 DEFERRED SIZE 10436503;
ALTER BUFFERPOOL STK4 DEFERRED SIZE 10436503;
ALTER BUFFERPOOL IBMDEFAULTBP DEFERRED SIZE 100;
connect reset;
```

_bp/alter_tablespace.ddl

```
connect to tpcc;
ALTER TABLESPACE ts_ware_001 BUFFERPOOL WAR1;
```

```
ALTER TABLESPACE ts_ware_002 BUFFERPOOL WAR1;
ALTER TABLESPACE ts_ware_003 BUFFERPOOL WAR1;
ALTER TABLESPACE ts_ware_004 BUFFERPOOL WAR1;
ALTER TABLESPACE ts_ware_005 BUFFERPOOL WAR1;
ALTER TABLESPACE ts_ware_006 BUFFERPOOL WAR1;
ALTER TABLESPACE ts_ware_007 BUFFERPOOL WAR1;
ALTER TABLESPACE ts_ware_008 BUFFERPOOL WAR1;
ALTER TABLESPACE ts_ware_009 BUFFERPOOL WAR2;
ALTER TABLESPACE ts_ware_010 BUFFERPOOL WAR2;
ALTER TABLESPACE ts_ware_011 BUFFERPOOL WAR2;
ALTER TABLESPACE ts_ware_012 BUFFERPOOL WAR2;
ALTER TABLESPACE ts_ware_013 BUFFERPOOL WAR2;
ALTER TABLESPACE ts_ware_014 BUFFERPOOL WAR2;
ALTER TABLESPACE ts_ware_015 BUFFERPOOL WAR2;
ALTER TABLESPACE ts_ware_016 BUFFERPOOL WAR2;
ALTER TABLESPACE ts_ware_017 BUFFERPOOL WAR3;
ALTER TABLESPACE ts_ware_018 BUFFERPOOL WAR3;
ALTER TABLESPACE ts_ware_019 BUFFERPOOL WAR3;
ALTER TABLESPACE ts_ware_020 BUFFERPOOL WAR3;
ALTER TABLESPACE ts_ware_021 BUFFERPOOL WAR3;
ALTER TABLESPACE ts_ware_022 BUFFERPOOL WAR3;
ALTER TABLESPACE ts_ware_023 BUFFERPOOL WAR3;
ALTER TABLESPACE ts_ware_024 BUFFERPOOL WAR3;
ALTER TABLESPACE ts_ware_025 BUFFERPOOL WAR4;
ALTER TABLESPACE ts_ware_026 BUFFERPOOL WAR4;
ALTER TABLESPACE ts_ware_027 BUFFERPOOL WAR4;
ALTER TABLESPACE ts_ware_028 BUFFERPOOL WAR4;
ALTER TABLESPACE ts_ware_029 BUFFERPOOL WAR4;
ALTER TABLESPACE ts_ware_030 BUFFERPOOL WAR4;
ALTER TABLESPACE ts_ware_031 BUFFERPOOL WAR4;
ALTER TABLESPACE ts_ware_032 BUFFERPOOL WAR4;
ALTER TABLESPACE ts_dist_001 BUFFERPOOL DIS1;
ALTER TABLESPACE ts_dist_002 BUFFERPOOL DIS1;
ALTER TABLESPACE ts_dist_003 BUFFERPOOL DIS1;
ALTER TABLESPACE ts_dist_004 BUFFERPOOL DIS1;
ALTER TABLESPACE ts_dist_005 BUFFERPOOL DIS1;
ALTER TABLESPACE ts_dist_006 BUFFERPOOL DIS1;
ALTER TABLESPACE ts_dist_007 BUFFERPOOL DIS1;
ALTER TABLESPACE ts_dist_008 BUFFERPOOL DIS1;
ALTER TABLESPACE ts_dist_009 BUFFERPOOL DIS2;
ALTER TABLESPACE ts_dist_010 BUFFERPOOL DIS2;
ALTER TABLESPACE ts_dist_011 BUFFERPOOL DIS2;
ALTER TABLESPACE ts_dist_012 BUFFERPOOL DIS2;
ALTER TABLESPACE ts_dist_013 BUFFERPOOL DIS2;
ALTER TABLESPACE ts_dist_014 BUFFERPOOL DIS2;
ALTER TABLESPACE ts_dist_015 BUFFERPOOL DIS2;
ALTER TABLESPACE ts_dist_016 BUFFERPOOL DIS2;
ALTER TABLESPACE ts_dist_017 BUFFERPOOL DIS3;
ALTER TABLESPACE ts_dist_018 BUFFERPOOL DIS3;
ALTER TABLESPACE ts_dist_019 BUFFERPOOL DIS3;
ALTER TABLESPACE ts_dist_020 BUFFERPOOL DIS3;
ALTER TABLESPACE ts_dist_021 BUFFERPOOL DIS3;
ALTER TABLESPACE ts_dist_022 BUFFERPOOL DIS3;
ALTER TABLESPACE ts_dist_023 BUFFERPOOL DIS3;
ALTER TABLESPACE ts_dist_024 BUFFERPOOL DIS3;
ALTER TABLESPACE ts_dist_025 BUFFERPOOL DIS4;
ALTER TABLESPACE ts_dist_026 BUFFERPOOL DIS4;
ALTER TABLESPACE ts_dist_027 BUFFERPOOL DIS4;
ALTER TABLESPACE ts_dist_028 BUFFERPOOL DIS4;
ALTER TABLESPACE ts_dist_029 BUFFERPOOL DIS4;
ALTER TABLESPACE ts_dist_030 BUFFERPOOL DIS4;
ALTER TABLESPACE ts_dist_031 BUFFERPOOL DIS4;
ALTER TABLESPACE ts_dist_032 BUFFERPOOL DIS4;
ALTER TABLESPACE ts_item_001 BUFFERPOOL ITM;
ALTER TABLESPACE ts_stock_001 BUFFERPOOL STK1;
ALTER TABLESPACE ts_stock_002 BUFFERPOOL STK1;
ALTER TABLESPACE ts_stock_003 BUFFERPOOL STK1;
ALTER TABLESPACE ts_stock_004 BUFFERPOOL STK1;
ALTER TABLESPACE ts_stock_005 BUFFERPOOL STK1;
ALTER TABLESPACE ts_stock_006 BUFFERPOOL STK1;
ALTER TABLESPACE ts_stock_007 BUFFERPOOL STK1;
ALTER TABLESPACE ts_stock_008 BUFFERPOOL STK1;
ALTER TABLESPACE ts_stock_009 BUFFERPOOL STK1;
ALTER TABLESPACE ts_stock_010 BUFFERPOOL STK1;
ALTER TABLESPACE ts_stock_011 BUFFERPOOL STK1;
ALTER TABLESPACE ts_stock_012 BUFFERPOOL STK1;
ALTER TABLESPACE ts_stock_013 BUFFERPOOL STK1;
ALTER TABLESPACE ts_stock_014 BUFFERPOOL STK1;
ALTER TABLESPACE ts_stock_015 BUFFERPOOL STK1;
ALTER TABLESPACE ts_stock_016 BUFFERPOOL STK1;
ALTER TABLESPACE ts_stock_017 BUFFERPOOL STK2;
ALTER TABLESPACE ts_stock_018 BUFFERPOOL STK2;
ALTER TABLESPACE ts_stock_019 BUFFERPOOL STK2;
ALTER TABLESPACE ts_stock_020 BUFFERPOOL STK2;
ALTER TABLESPACE ts_stock_021 BUFFERPOOL STK2;
ALTER TABLESPACE ts_stock_022 BUFFERPOOL STK2;
ALTER TABLESPACE ts_stock_023 BUFFERPOOL STK2;
ALTER TABLESPACE ts_stock_024 BUFFERPOOL STK2;
ALTER TABLESPACE ts_stock_025 BUFFERPOOL STK2;
ALTER TABLESPACE ts_stock_026 BUFFERPOOL STK2;
ALTER TABLESPACE ts_stock_027 BUFFERPOOL STK2;
ALTER TABLESPACE ts_stock_028 BUFFERPOOL STK2;
ALTER TABLESPACE ts_stock_029 BUFFERPOOL STK2;
ALTER TABLESPACE ts_stock_030 BUFFERPOOL STK2;
ALTER TABLESPACE ts_stock_031 BUFFERPOOL STK2;
ALTER TABLESPACE ts_stock_032 BUFFERPOOL STK2;
ALTER TABLESPACE ts_stock_033 BUFFERPOOL STK3;
ALTER TABLESPACE ts_stock_034 BUFFERPOOL STK3;
ALTER TABLESPACE ts_stock_035 BUFFERPOOL STK3;
ALTER TABLESPACE ts_stock_036 BUFFERPOOL STK3;
ALTER TABLESPACE ts_stock_037 BUFFERPOOL STK3;
ALTER TABLESPACE ts_stock_038 BUFFERPOOL STK3;
ALTER TABLESPACE ts_stock_039 BUFFERPOOL STK3;
ALTER TABLESPACE ts_stock_040 BUFFERPOOL STK3;
ALTER TABLESPACE ts_stock_041 BUFFERPOOL STK3;
ALTER TABLESPACE ts_stock_042 BUFFERPOOL STK3;
ALTER TABLESPACE ts_stock_043 BUFFERPOOL STK3;
ALTER TABLESPACE ts_stock_044 BUFFERPOOL STK3;
ALTER TABLESPACE ts_stock_045 BUFFERPOOL STK3;
ALTER TABLESPACE ts_stock_046 BUFFERPOOL STK3;
ALTER TABLESPACE ts_stock_047 BUFFERPOOL STK3;
ALTER TABLESPACE ts_stock_048 BUFFERPOOL STK3;
ALTER TABLESPACE ts_stock_049 BUFFERPOOL STK4;
ALTER TABLESPACE ts_stock_050 BUFFERPOOL STK4;
ALTER TABLESPACE ts_stock_051 BUFFERPOOL STK4;
ALTER TABLESPACE ts_stock_052 BUFFERPOOL STK4;
ALTER TABLESPACE ts_stock_053 BUFFERPOOL STK4;
ALTER TABLESPACE ts_stock_054 BUFFERPOOL STK4;
ALTER TABLESPACE ts_stock_055 BUFFERPOOL STK4;
ALTER TABLESPACE ts_stock_056 BUFFERPOOL STK4;
ALTER TABLESPACE ts_stock_057 BUFFERPOOL STK4;
ALTER TABLESPACE ts_stock_058 BUFFERPOOL STK4;
ALTER TABLESPACE ts_stock_059 BUFFERPOOL STK4;
ALTER TABLESPACE ts_stock_060 BUFFERPOOL STK4;
ALTER TABLESPACE ts_stock_061 BUFFERPOOL STK4;
ALTER TABLESPACE ts_stock_062 BUFFERPOOL STK4;
ALTER TABLESPACE ts_stock_063 BUFFERPOOL STK4;
ALTER TABLESPACE ts_stock_064 BUFFERPOOL STK4;
ALTER TABLESPACE ts_customer_001 BUFFERPOOL CST1;
ALTER TABLESPACE ts_customer_002 BUFFERPOOL CST1;
ALTER TABLESPACE ts_customer_003 BUFFERPOOL CST1;
ALTER TABLESPACE ts_customer_004 BUFFERPOOL CST1;
ALTER TABLESPACE ts_customer_005 BUFFERPOOL CST1;
ALTER TABLESPACE ts_customer_006 BUFFERPOOL CST1;
ALTER TABLESPACE ts_customer_007 BUFFERPOOL CST1;
ALTER TABLESPACE ts_customer_008 BUFFERPOOL CST1;
ALTER TABLESPACE ts_customer_009 BUFFERPOOL CST1;
ALTER TABLESPACE ts_customer_010 BUFFERPOOL CST1;
ALTER TABLESPACE ts_customer_011 BUFFERPOOL CST1;
ALTER TABLESPACE ts_customer_012 BUFFERPOOL CST1;
ALTER TABLESPACE ts_customer_013 BUFFERPOOL CST1;
ALTER TABLESPACE ts_customer_014 BUFFERPOOL CST1;
ALTER TABLESPACE ts_customer_015 BUFFERPOOL CST1;
ALTER TABLESPACE ts_customer_016 BUFFERPOOL CST1;
ALTER TABLESPACE ts_customer_017 BUFFERPOOL CST2;
ALTER TABLESPACE ts_customer_018 BUFFERPOOL CST2;
ALTER TABLESPACE ts_customer_019 BUFFERPOOL CST2;
ALTER TABLESPACE ts_customer_020 BUFFERPOOL CST2;
ALTER TABLESPACE ts_customer_021 BUFFERPOOL CST2;
ALTER TABLESPACE ts_customer_022 BUFFERPOOL CST2;
ALTER TABLESPACE ts_customer_023 BUFFERPOOL CST2;
ALTER TABLESPACE ts_customer_024 BUFFERPOOL CST2;
ALTER TABLESPACE ts_customer_025 BUFFERPOOL CST2;
ALTER TABLESPACE ts_customer_026 BUFFERPOOL CST2;
ALTER TABLESPACE ts_customer_027 BUFFERPOOL CST2;
ALTER TABLESPACE ts_customer_028 BUFFERPOOL CST2;
ALTER TABLESPACE ts_customer_029 BUFFERPOOL CST2;
ALTER TABLESPACE ts_customer_030 BUFFERPOOL CST2;
ALTER TABLESPACE ts_customer_031 BUFFERPOOL CST2;
ALTER TABLESPACE ts_customer_032 BUFFERPOOL CST2;
ALTER TABLESPACE ts_customer_033 BUFFERPOOL CST3;
ALTER TABLESPACE ts_customer_034 BUFFERPOOL CST3;
ALTER TABLESPACE ts_customer_035 BUFFERPOOL CST3;
ALTER TABLESPACE ts_customer_036 BUFFERPOOL CST3;
ALTER TABLESPACE ts_customer_037 BUFFERPOOL CST3;
ALTER TABLESPACE ts_customer_038 BUFFERPOOL CST3;
ALTER TABLESPACE ts_customer_039 BUFFERPOOL CST3;
ALTER TABLESPACE ts_customer_040 BUFFERPOOL CST3;
ALTER TABLESPACE ts_customer_041 BUFFERPOOL CST3;
ALTER TABLESPACE ts_customer_042 BUFFERPOOL CST3;
ALTER TABLESPACE ts_customer_043 BUFFERPOOL CST3;
ALTER TABLESPACE ts_customer_044 BUFFERPOOL CST3;
ALTER TABLESPACE ts_customer_045 BUFFERPOOL CST3;
ALTER TABLESPACE ts_customer_046 BUFFERPOOL CST3;
ALTER TABLESPACE ts_customer_047 BUFFERPOOL CST3;
ALTER TABLESPACE ts_customer_048 BUFFERPOOL CST3;
ALTER TABLESPACE ts_customer_049 BUFFERPOOL CST4;
ALTER TABLESPACE ts_customer_050 BUFFERPOOL CST4;
ALTER TABLESPACE ts_customer_051 BUFFERPOOL CST4;
ALTER TABLESPACE ts_customer_052 BUFFERPOOL CST4;
ALTER TABLESPACE ts_customer_053 BUFFERPOOL CST4;
ALTER TABLESPACE ts_customer_054 BUFFERPOOL CST4;
ALTER TABLESPACE ts_customer_055 BUFFERPOOL CST4;
ALTER TABLESPACE ts_customer_056 BUFFERPOOL CST4;
ALTER TABLESPACE ts_customer_057 BUFFERPOOL CST4;
ALTER TABLESPACE ts_customer_058 BUFFERPOOL CST4;
ALTER TABLESPACE ts_customer_059 BUFFERPOOL CST4;
ALTER TABLESPACE ts_customer_060 BUFFERPOOL CST4;
ALTER TABLESPACE ts_customer_061 BUFFERPOOL CST4;
ALTER TABLESPACE ts_customer_062 BUFFERPOOL CST4;
ALTER TABLESPACE ts_customer_063 BUFFERPOOL CST4;
ALTER TABLESPACE ts_customer_064 BUFFERPOOL CST4;
ALTER TABLESPACE is_customer_001 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_customer_002 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_customer_003 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_customer_004 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_customer_005 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_customer_006 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_customer_007 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_customer_008 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_customer_009 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_customer_010 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_customer_011 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_customer_012 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_customer_013 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_customer_014 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_customer_015 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_customer_016 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_customer_017 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_customer_018 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_customer_019 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_customer_020 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_customer_021 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_customer_022 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_customer_023 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_customer_024 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_customer_025 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_customer_026 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_customer_027 BUFFERPOOL CST_I2;
```

```
ALTER TABLESPACE is_customer_028 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_customer_029 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_customer_030 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_customer_031 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_customer_032 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_customer_033 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_customer_034 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_customer_035 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_customer_036 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_customer_037 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_customer_038 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_customer_039 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_customer_040 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_customer_041 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_customer_042 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_customer_043 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_customer_044 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_customer_045 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_customer_046 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_customer_047 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_customer_048 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_customer_049 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_customer_050 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_customer_051 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_customer_052 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_customer_053 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_customer_054 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_customer_055 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_customer_056 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_customer_057 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_customer_058 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_customer_059 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_customer_060 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_customer_061 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_customer_062 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_customer_063 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_customer_064 BUFFERPOOL CST_I4;
ALTER TABLESPACE ts_history_001 BUFFERPOOL HST1;
ALTER TABLESPACE ts_history_002 BUFFERPOOL HST1;
ALTER TABLESPACE ts_history_003 BUFFERPOOL HST1;
ALTER TABLESPACE ts_history_004 BUFFERPOOL HST1;
ALTER TABLESPACE ts_history_005 BUFFERPOOL HST1;
ALTER TABLESPACE ts_history_006 BUFFERPOOL HST1;
ALTER TABLESPACE ts_history_007 BUFFERPOOL HST1;
ALTER TABLESPACE ts_history_008 BUFFERPOOL HST1;
ALTER TABLESPACE ts_history_009 BUFFERPOOL HST2;
ALTER TABLESPACE ts_history_010 BUFFERPOOL HST2;
ALTER TABLESPACE ts_history_011 BUFFERPOOL HST2;
ALTER TABLESPACE ts_history_012 BUFFERPOOL HST2;
ALTER TABLESPACE ts_history_013 BUFFERPOOL HST2;
ALTER TABLESPACE ts_history_014 BUFFERPOOL HST2;
ALTER TABLESPACE ts_history_015 BUFFERPOOL HST2;
ALTER TABLESPACE ts_history_016 BUFFERPOOL HST2;
ALTER TABLESPACE ts_history_017 BUFFERPOOL HST3;
ALTER TABLESPACE ts_history_018 BUFFERPOOL HST3;
ALTER TABLESPACE ts_history_019 BUFFERPOOL HST3;
ALTER TABLESPACE ts_history_020 BUFFERPOOL HST3;
ALTER TABLESPACE ts_history_021 BUFFERPOOL HST3;
ALTER TABLESPACE ts_history_022 BUFFERPOOL HST3;
ALTER TABLESPACE ts_history_023 BUFFERPOOL HST3;
ALTER TABLESPACE ts_history_024 BUFFERPOOL HST3;
ALTER TABLESPACE ts_history_025 BUFFERPOOL HST4;
ALTER TABLESPACE ts_history_026 BUFFERPOOL HST4;
ALTER TABLESPACE ts_history_027 BUFFERPOOL HST4;
ALTER TABLESPACE ts_history_028 BUFFERPOOL HST4;
ALTER TABLESPACE ts_history_029 BUFFERPOOL HST4;
ALTER TABLESPACE ts_history_030 BUFFERPOOL HST4;
ALTER TABLESPACE ts_history_031 BUFFERPOOL HST4;
ALTER TABLESPACE ts_history_032 BUFFERPOOL HST4;
ALTER TABLESPACE ts_order_001 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_order_002 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_order_003 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_order_004 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_order_005 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_order_006 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_order_007 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_order_008 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_order_009 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_order_010 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_order_011 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_order_012 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_order_013 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_order_014 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_order_015 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_order_016 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_order_017 BUFFERPOOL ORD2;
ALTER TABLESPACE ts_order_018 BUFFERPOOL ORD2;
ALTER TABLESPACE ts_order_019 BUFFERPOOL ORD2;
ALTER TABLESPACE ts_order_020 BUFFERPOOL ORD2;
ALTER TABLESPACE ts_order_021 BUFFERPOOL ORD2;
ALTER TABLESPACE ts_order_022 BUFFERPOOL ORD2;
ALTER TABLESPACE ts_order_023 BUFFERPOOL ORD2;
ALTER TABLESPACE ts_order_024 BUFFERPOOL ORD2;
ALTER TABLESPACE ts_order_025 BUFFERPOOL ORD2;
ALTER TABLESPACE ts_order_026 BUFFERPOOL ORD2;
ALTER TABLESPACE ts_order_027 BUFFERPOOL ORD2;
ALTER TABLESPACE ts_order_028 BUFFERPOOL ORD2;
ALTER TABLESPACE ts_order_029 BUFFERPOOL ORD2;
ALTER TABLESPACE ts_order_030 BUFFERPOOL ORD2;
ALTER TABLESPACE ts_order_031 BUFFERPOOL ORD2;
ALTER TABLESPACE ts_order_032 BUFFERPOOL ORD2;
ALTER TABLESPACE ts_order_033 BUFFERPOOL ORD3;
ALTER TABLESPACE ts_order_034 BUFFERPOOL ORD3;
ALTER TABLESPACE ts_order_035 BUFFERPOOL ORD3;
ALTER TABLESPACE ts_order_036 BUFFERPOOL ORD3;
ALTER TABLESPACE ts_order_037 BUFFERPOOL ORD3;
ALTER TABLESPACE ts_order_038 BUFFERPOOL ORD3;
ALTER TABLESPACE ts_order_039 BUFFERPOOL ORD3;
ALTER TABLESPACE ts_order_040 BUFFERPOOL ORD3;
ALTER TABLESPACE ts_order_041 BUFFERPOOL ORD3;
ALTER TABLESPACE ts_order_042 BUFFERPOOL ORD3;
ALTER TABLESPACE ts_order_043 BUFFERPOOL ORD3;
ALTER TABLESPACE ts_order_044 BUFFERPOOL ORD3;
ALTER TABLESPACE ts_order_045 BUFFERPOOL ORD3;
ALTER TABLESPACE ts_order_046 BUFFERPOOL ORD3;
ALTER TABLESPACE ts_order_047 BUFFERPOOL ORD3;
ALTER TABLESPACE ts_order_048 BUFFERPOOL ORD3;
ALTER TABLESPACE ts_order_049 BUFFERPOOL ORD4;
ALTER TABLESPACE ts_order_050 BUFFERPOOL ORD4;
ALTER TABLESPACE ts_order_051 BUFFERPOOL ORD4;
ALTER TABLESPACE ts_order_052 BUFFERPOOL ORD4;
ALTER TABLESPACE ts_order_053 BUFFERPOOL ORD4;
ALTER TABLESPACE ts_order_054 BUFFERPOOL ORD4;
ALTER TABLESPACE ts_order_055 BUFFERPOOL ORD4;
ALTER TABLESPACE ts_order_056 BUFFERPOOL ORD4;
ALTER TABLESPACE ts_order_057 BUFFERPOOL ORD4;
ALTER TABLESPACE ts_order_058 BUFFERPOOL ORD4;
ALTER TABLESPACE ts_order_059 BUFFERPOOL ORD4;
ALTER TABLESPACE ts_order_060 BUFFERPOOL ORD4;
ALTER TABLESPACE ts_order_061 BUFFERPOOL ORD4;
ALTER TABLESPACE ts_order_062 BUFFERPOOL ORD4;
ALTER TABLESPACE ts_order_063 BUFFERPOOL ORD4;
ALTER TABLESPACE ts_order_064 BUFFERPOOL ORD4;
ALTER TABLESPACE is_order_001 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_order_002 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_order_003 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_order_004 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_order_005 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_order_006 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_order_007 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_order_008 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_order_009 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_order_010 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_order_011 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_order_012 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_order_013 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_order_014 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_order_015 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_order_016 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_order_017 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_order_018 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_order_019 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_order_020 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_order_021 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_order_022 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_order_023 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_order_024 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_order_025 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_order_026 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_order_027 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_order_028 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_order_029 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_order_030 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_order_031 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_order_032 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_order_033 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_order_034 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_order_035 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_order_036 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_order_037 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_order_038 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_order_039 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_order_040 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_order_041 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_order_042 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_order_043 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_order_044 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_order_045 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_order_046 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_order_047 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_order_048 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_order_049 BUFFERPOOL ORD_I4;
ALTER TABLESPACE is_order_050 BUFFERPOOL ORD_I4;
ALTER TABLESPACE is_order_051 BUFFERPOOL ORD_I4;
ALTER TABLESPACE is_order_052 BUFFERPOOL ORD_I4;
ALTER TABLESPACE is_order_053 BUFFERPOOL ORD_I4;
ALTER TABLESPACE is_order_054 BUFFERPOOL ORD_I4;
ALTER TABLESPACE is_order_055 BUFFERPOOL ORD_I4;
ALTER TABLESPACE is_order_056 BUFFERPOOL ORD_I4;
ALTER TABLESPACE is_order_057 BUFFERPOOL ORD_I4;
ALTER TABLESPACE is_order_058 BUFFERPOOL ORD_I4;
ALTER TABLESPACE is_order_059 BUFFERPOOL ORD_I4;
ALTER TABLESPACE is_order_060 BUFFERPOOL ORD_I4;
ALTER TABLESPACE is_order_061 BUFFERPOOL ORD_I4;
ALTER TABLESPACE is_order_062 BUFFERPOOL ORD_I4;
ALTER TABLESPACE is_order_063 BUFFERPOOL ORD_I4;
ALTER TABLESPACE is_order_064 BUFFERPOOL ORD_I4;
ALTER TABLESPACE ts_orderline_001 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_orderline_002 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_orderline_003 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_orderline_004 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_orderline_005 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_orderline_006 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_orderline_007 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_orderline_008 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_orderline_009 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_orderline_010 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_orderline_011 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_orderline_012 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_orderline_013 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_orderline_014 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_orderline_015 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_orderline_016 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_orderline_017 BUFFERPOOL OLN2;
ALTER TABLESPACE ts_orderline_018 BUFFERPOOL OLN2;
ALTER TABLESPACE ts_orderline_019 BUFFERPOOL OLN2;
ALTER TABLESPACE ts_orderline_020 BUFFERPOOL OLN2;
ALTER TABLESPACE ts_orderline_021 BUFFERPOOL OLN2;
ALTER TABLESPACE ts_orderline_022 BUFFERPOOL OLN2;
```

```
ALTER TABLESPACE ts_orderline_023 BUFFERPOOL OLN2;
ALTER TABLESPACE ts_orderline_024 BUFFERPOOL OLN2;
ALTER TABLESPACE ts_orderline_025 BUFFERPOOL OLN2;
ALTER TABLESPACE ts_orderline_026 BUFFERPOOL OLN2;
ALTER TABLESPACE ts_orderline_027 BUFFERPOOL OLN2;
ALTER TABLESPACE ts_orderline_028 BUFFERPOOL OLN2;
ALTER TABLESPACE ts_orderline_029 BUFFERPOOL OLN2;
ALTER TABLESPACE ts_orderline_030 BUFFERPOOL OLN2;
ALTER TABLESPACE ts_orderline_031 BUFFERPOOL OLN2;
ALTER TABLESPACE ts_orderline_032 BUFFERPOOL OLN2;
ALTER TABLESPACE ts_orderline_033 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_orderline_034 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_orderline_035 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_orderline_036 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_orderline_037 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_orderline_038 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_orderline_039 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_orderline_040 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_orderline_041 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_orderline_042 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_orderline_043 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_orderline_044 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_orderline_045 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_orderline_046 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_orderline_047 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_orderline_048 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_orderline_049 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_orderline_050 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_orderline_051 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_orderline_052 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_orderline_053 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_orderline_054 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_orderline_055 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_orderline_056 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_orderline_057 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_orderline_058 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_orderline_059 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_orderline_060 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_orderline_061 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_orderline_062 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_orderline_063 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_orderline_064 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_newordA_001 BUFFERPOOL NEW1;
ALTER TABLESPACE ts_newordA_002 BUFFERPOOL NEW1;
ALTER TABLESPACE ts_newordA_003 BUFFERPOOL NEW1;
ALTER TABLESPACE ts_newordA_004 BUFFERPOOL NEW1;
ALTER TABLESPACE ts_newordA_005 BUFFERPOOL NEW1;
ALTER TABLESPACE ts_newordA_006 BUFFERPOOL NEW1;
ALTER TABLESPACE ts_newordA_007 BUFFERPOOL NEW1;
ALTER TABLESPACE ts_newordA_008 BUFFERPOOL NEW1;
ALTER TABLESPACE ts_newordA_009 BUFFERPOOL NEW2;
ALTER TABLESPACE ts_newordA_010 BUFFERPOOL NEW2;
ALTER TABLESPACE ts_newordA_011 BUFFERPOOL NEW2;
ALTER TABLESPACE ts_newordA_012 BUFFERPOOL NEW2;
ALTER TABLESPACE ts_newordA_013 BUFFERPOOL NEW2;
ALTER TABLESPACE ts_newordA_014 BUFFERPOOL NEW2;
ALTER TABLESPACE ts_newordA_015 BUFFERPOOL NEW2;
ALTER TABLESPACE ts_newordA_016 BUFFERPOOL NEW2;
ALTER TABLESPACE ts_newordA_017 BUFFERPOOL NEW3;
ALTER TABLESPACE ts_newordA_018 BUFFERPOOL NEW3;
ALTER TABLESPACE ts_newordA_019 BUFFERPOOL NEW3;
ALTER TABLESPACE ts_newordA_020 BUFFERPOOL NEW3;
ALTER TABLESPACE ts_newordA_021 BUFFERPOOL NEW3;
ALTER TABLESPACE ts_newordA_022 BUFFERPOOL NEW3;
ALTER TABLESPACE ts_newordA_023 BUFFERPOOL NEW3;
ALTER TABLESPACE ts_newordA_024 BUFFERPOOL NEW3;
ALTER TABLESPACE ts_newordA_025 BUFFERPOOL NEW4;
ALTER TABLESPACE ts_newordA_026 BUFFERPOOL NEW4;
ALTER TABLESPACE ts_newordA_027 BUFFERPOOL NEW4;
ALTER TABLESPACE ts_newordA_028 BUFFERPOOL NEW4;
ALTER TABLESPACE ts_newordA_029 BUFFERPOOL NEW4;
ALTER TABLESPACE ts_newordA_030 BUFFERPOOL NEW4;
ALTER TABLESPACE ts_newordA_031 BUFFERPOOL NEW4;
ALTER TABLESPACE ts_newordA_032 BUFFERPOOL NEW4;
ALTER TABLESPACE ts_newordB_001 BUFFERPOOL NEW1;
ALTER TABLESPACE ts_newordB_002 BUFFERPOOL NEW1;
ALTER TABLESPACE ts_newordB_003 BUFFERPOOL NEW1;
ALTER TABLESPACE ts_newordB_004 BUFFERPOOL NEW1;
ALTER TABLESPACE ts_newordB_005 BUFFERPOOL NEW1;
ALTER TABLESPACE ts_newordB_006 BUFFERPOOL NEW1;
ALTER TABLESPACE ts_newordB_007 BUFFERPOOL NEW1;
ALTER TABLESPACE ts_newordB_008 BUFFERPOOL NEW1;
ALTER TABLESPACE ts_newordB_009 BUFFERPOOL NEW2;
ALTER TABLESPACE ts_newordB_010 BUFFERPOOL NEW2;
ALTER TABLESPACE ts_newordB_011 BUFFERPOOL NEW2;
ALTER TABLESPACE ts_newordB_012 BUFFERPOOL NEW2;
ALTER TABLESPACE ts_newordB_013 BUFFERPOOL NEW2;
ALTER TABLESPACE ts_newordB_014 BUFFERPOOL NEW2;
ALTER TABLESPACE ts_newordB_015 BUFFERPOOL NEW2;
ALTER TABLESPACE ts_newordB_016 BUFFERPOOL NEW2;
ALTER TABLESPACE ts_newordB_017 BUFFERPOOL NEW3;
ALTER TABLESPACE ts_newordB_018 BUFFERPOOL NEW3;
ALTER TABLESPACE ts_newordB_019 BUFFERPOOL NEW3;
ALTER TABLESPACE ts_newordB_020 BUFFERPOOL NEW3;
ALTER TABLESPACE ts_newordB_021 BUFFERPOOL NEW3;
ALTER TABLESPACE ts_newordB_022 BUFFERPOOL NEW3;
ALTER TABLESPACE ts_newordB_023 BUFFERPOOL NEW3;
ALTER TABLESPACE ts_newordB_024 BUFFERPOOL NEW3;
ALTER TABLESPACE ts_newordB_025 BUFFERPOOL NEW4;
ALTER TABLESPACE ts_newordB_026 BUFFERPOOL NEW4;
ALTER TABLESPACE ts_newordB_027 BUFFERPOOL NEW4;
ALTER TABLESPACE ts_newordB_028 BUFFERPOOL NEW4;
ALTER TABLESPACE ts_newordB_029 BUFFERPOOL NEW4;
ALTER TABLESPACE ts_newordB_030 BUFFERPOOL NEW4;
ALTER TABLESPACE ts_newordB_031 BUFFERPOOL NEW4;
ALTER TABLESPACE ts_newordB_032 BUFFERPOOL NEW4;
connect reset;
```

_bp/create_bufferpool.ddl

```
connect to tpcc;
CREATE BUFFERPOOL WAR1 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL WAR2 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL WAR3 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL WAR4 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL DIS1 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL DIS2 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL DIS3 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL DIS4 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL ITM SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL HST1 SIZE 100 PAGESIZE 16384;
CREATE BUFFERPOOL HST2 SIZE 100 PAGESIZE 16384;
CREATE BUFFERPOOL HST3 SIZE 100 PAGESIZE 16384;
CREATE BUFFERPOOL HST4 SIZE 100 PAGESIZE 16384;
CREATE BUFFERPOOL NEW1 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL NEW2 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL NEW3 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL NEW4 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL ORD1 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD2 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD3 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD4 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD_I1 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD_I2 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD_I3 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD_I4 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL OLN1 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL OLN2 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL OLN3 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL OLN4 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL CST1 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL CST2 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL CST3 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL CST4 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL CST_I1 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL CST_I2 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL CST_I3 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL CST_I4 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL STK1 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL STK2 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL STK3 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL STK4 SIZE 100 PAGESIZE 4096;
connect reset;
```

_db/create_database.ddl

```
-----------------------------------------------------------------------------
-- Licensed Materials - Property of IBM
--
-- Governed under the terms of the International
-- License Agreement for Non-Warranted Sample Code.
--
-- (C) COPYRIGHT International Business Machines Corp. 1996 - 2002
-- All Rights Reserved.
--
-- US Government Users Restricted Rights - Use, duplication or
-- disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
-----------------------------------------------------------------------------
drop database tpcc;
create database tpcc on /home/tpcc/db/tpccdb1 USING CODESET ISO8859-1 TERRITORY US COLLATE
USING IDENTITY
catalog tablespace
managed by system using ('/home/tpcc/db/db1catalog');
```

_ts/cris_customer.ddl

```
connect to tpcc;
-- now creating TS for is_customer_001 of D1

drop tablespace is_customer_001;
create regular tablespace is_customer_001 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F01V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_002 of D1

drop tablespace is_customer_002;
create regular tablespace is_customer_002 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F01V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_003 of D1
```

```
drop tablespace is_customer_003;
create regular tablespace is_customer_003 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F02V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_004 of D1

drop tablespace is_customer_004;
create regular tablespace is_customer_004 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F02V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_005 of D1

drop tablespace is_customer_005;
create regular tablespace is_customer_005 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F03V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_006 of D1

drop tablespace is_customer_006;
create regular tablespace is_customer_006 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F03V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_007 of D1

drop tablespace is_customer_007;
create regular tablespace is_customer_007 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F04V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_008 of D1

drop tablespace is_customer_008;
create regular tablespace is_customer_008 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F04V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_009 of D1

drop tablespace is_customer_009;
create regular tablespace is_customer_009 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F05V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_010 of D1

drop tablespace is_customer_010;
create regular tablespace is_customer_010 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F05V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_011 of D1

drop tablespace is_customer_011;
create regular tablespace is_customer_011 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F06V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_012 of D1

drop tablespace is_customer_012;
create regular tablespace is_customer_012 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F06V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_013 of D1

drop tablespace is_customer_013;
create regular tablespace is_customer_013 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F07V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_014 of D1

drop tablespace is_customer_014;
create regular tablespace is_customer_014 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F07V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_015 of D1

drop tablespace is_customer_015;
create regular tablespace is_customer_015 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F08V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_016 of D1

drop tablespace is_customer_016;
create regular tablespace is_customer_016 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F08V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_017 of D1

drop tablespace is_customer_017;
create regular tablespace is_customer_017 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F09V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_018 of D1

drop tablespace is_customer_018;
create regular tablespace is_customer_018 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F09V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
```

```
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_019 of D1

drop tablespace is_customer_019;
create regular tablespace is_customer_019 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F10V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_020 of D1

drop tablespace is_customer_020;
create regular tablespace is_customer_020 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F10V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_021 of D1

drop tablespace is_customer_021;
create regular tablespace is_customer_021 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F11V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_022 of D1

drop tablespace is_customer_022;
create regular tablespace is_customer_022 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F11V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_023 of D1

drop tablespace is_customer_023;
create regular tablespace is_customer_023 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F12V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_024 of D1

drop tablespace is_customer_024;
create regular tablespace is_customer_024 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F12V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_025 of D1

drop tablespace is_customer_025;
create regular tablespace is_customer_025 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F13V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_026 of D1

drop tablespace is_customer_026;
create regular tablespace is_customer_026 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F13V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_027 of D1

drop tablespace is_customer_027;
create regular tablespace is_customer_027 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F14V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_028 of D1

drop tablespace is_customer_028;
create regular tablespace is_customer_028 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F14V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_029 of D1

drop tablespace is_customer_029;
create regular tablespace is_customer_029 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F15V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_030 of D1

drop tablespace is_customer_030;
create regular tablespace is_customer_030 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F15V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_031 of D1

drop tablespace is_customer_031;
create regular tablespace is_customer_031 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F16V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_032 of D1

drop tablespace is_customer_032;
create regular tablespace is_customer_032 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F16V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_033 of D1

drop tablespace is_customer_033;
create regular tablespace is_customer_033 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F17V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_034 of D1

drop tablespace is_customer_034;
create regular tablespace is_customer_034 pagesize 8K
         managed by database
         using
```

```
        (
                device '/dev/rD1F17V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_035 of D1

drop tablespace is_customer_035;
create regular tablespace is_customer_035 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F18V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_036 of D1

drop tablespace is_customer_036;
create regular tablespace is_customer_036 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F18V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_037 of D1

drop tablespace is_customer_037;
create regular tablespace is_customer_037 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F19V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_038 of D1

drop tablespace is_customer_038;
create regular tablespace is_customer_038 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F19V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_039 of D1

drop tablespace is_customer_039;
create regular tablespace is_customer_039 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F20V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_040 of D1

drop tablespace is_customer_040;
create regular tablespace is_customer_040 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F20V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_041 of D1

drop tablespace is_customer_041;
create regular tablespace is_customer_041 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F21V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_042 of D1

drop tablespace is_customer_042;
create regular tablespace is_customer_042 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F21V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_043 of D1

drop tablespace is_customer_043;
create regular tablespace is_customer_043 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F22V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_044 of D1

drop tablespace is_customer_044;
create regular tablespace is_customer_044 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F22V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_045 of D1

drop tablespace is_customer_045;
create regular tablespace is_customer_045 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F23V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_046 of D1

drop tablespace is_customer_046;
create regular tablespace is_customer_046 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F23V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_047 of D1

drop tablespace is_customer_047;
create regular tablespace is_customer_047 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F24V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_048 of D1

drop tablespace is_customer_048;
create regular tablespace is_customer_048 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F24V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_049 of D1

drop tablespace is_customer_049;
create regular tablespace is_customer_049 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F25V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_050 of D1
```

```
drop tablespace is_customer_050;
create regular tablespace is_customer_050 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F25V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_051 of D1

drop tablespace is_customer_051;
create regular tablespace is_customer_051 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F26V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_052 of D1

drop tablespace is_customer_052;
create regular tablespace is_customer_052 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F26V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_053 of D1

drop tablespace is_customer_053;
create regular tablespace is_customer_053 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F27V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_054 of D1

drop tablespace is_customer_054;
create regular tablespace is_customer_054 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F27V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_055 of D1

drop tablespace is_customer_055;
create regular tablespace is_customer_055 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F28V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_056 of D1

drop tablespace is_customer_056;
create regular tablespace is_customer_056 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F28V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_057 of D1

drop tablespace is_customer_057;
create regular tablespace is_customer_057 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F29V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_058 of D1

drop tablespace is_customer_058;
create regular tablespace is_customer_058 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F29V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_059 of D1

drop tablespace is_customer_059;
create regular tablespace is_customer_059 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F30V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_060 of D1

drop tablespace is_customer_060;
create regular tablespace is_customer_060 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F30V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_061 of D1

drop tablespace is_customer_061;
create regular tablespace is_customer_061 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F31V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_062 of D1

drop tablespace is_customer_062;
create regular tablespace is_customer_062 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F31V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_063 of D1

drop tablespace is_customer_063;
create regular tablespace is_customer_063 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F32V1CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_064 of D1

drop tablespace is_customer_064;
create regular tablespace is_customer_064 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F32V2CSTI' 176128
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

connect reset;
```

ts/cris_order.ddl

```
connect to tpcc;
```

```
-- now creating TS for is_order_001 of D1

drop tablespace is_order_001;
create regular tablespace is_order_001 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F01V1ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_002 of D1

drop tablespace is_order_002;
create regular tablespace is_order_002 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F01V2ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_003 of D1

drop tablespace is_order_003;
create regular tablespace is_order_003 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F02V1ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_004 of D1

drop tablespace is_order_004;
create regular tablespace is_order_004 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F02V2ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_005 of D1

drop tablespace is_order_005;
create regular tablespace is_order_005 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F03V1ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_006 of D1

drop tablespace is_order_006;
create regular tablespace is_order_006 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F03V2ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_007 of D1

drop tablespace is_order_007;
create regular tablespace is_order_007 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F04V1ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_008 of D1

drop tablespace is_order_008;
create regular tablespace is_order_008 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F04V2ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_009 of D1

drop tablespace is_order_009;
create regular tablespace is_order_009 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F05V1ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_010 of D1

drop tablespace is_order_010;
create regular tablespace is_order_010 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F05V2ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_011 of D1

drop tablespace is_order_011;
create regular tablespace is_order_011 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F06V1ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_012 of D1

drop tablespace is_order_012;
create regular tablespace is_order_012 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F06V2ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_013 of D1

drop tablespace is_order_013;
create regular tablespace is_order_013 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F07V1ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_014 of D1

drop tablespace is_order_014;
create regular tablespace is_order_014 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F07V2ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_015 of D1

drop tablespace is_order_015;
create regular tablespace is_order_015 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F08V1ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_016 of D1

drop tablespace is_order_016;
create regular tablespace is_order_016 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F08V2ORDI' 114944
                )
```

```
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_017 of D1

drop tablespace is_order_017;
create regular tablespace is_order_017 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F09V1ORDI' 114944
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_018 of D1

drop tablespace is_order_018;
create regular tablespace is_order_018 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F09V2ORDI' 114944
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_019 of D1

drop tablespace is_order_019;
create regular tablespace is_order_019 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F10V1ORDI' 114944
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_020 of D1

drop tablespace is_order_020;
create regular tablespace is_order_020 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F10V2ORDI' 114944
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_021 of D1

drop tablespace is_order_021;
create regular tablespace is_order_021 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F11V1ORDI' 114944
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_022 of D1

drop tablespace is_order_022;
create regular tablespace is_order_022 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F11V2ORDI' 114944
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_023 of D1

drop tablespace is_order_023;
create regular tablespace is_order_023 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F12V1ORDI' 114944
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_024 of D1

drop tablespace is_order_024;
create regular tablespace is_order_024 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F12V2ORDI' 114944
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_025 of D1

drop tablespace is_order_025;
create regular tablespace is_order_025 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F13V1ORDI' 114944
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_026 of D1

drop tablespace is_order_026;
create regular tablespace is_order_026 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F13V2ORDI' 114944
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_027 of D1

drop tablespace is_order_027;
create regular tablespace is_order_027 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F14V1ORDI' 114944
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_028 of D1

drop tablespace is_order_028;
create regular tablespace is_order_028 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F14V2ORDI' 114944
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_029 of D1

drop tablespace is_order_029;
create regular tablespace is_order_029 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F15V1ORDI' 114944
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_030 of D1

drop tablespace is_order_030;
create regular tablespace is_order_030 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F15V2ORDI' 114944
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_031 of D1

drop tablespace is_order_031;
create regular tablespace is_order_031 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F16V1ORDI' 114944
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_032 of D1

drop tablespace is_order_032;
create regular tablespace is_order_032 pagesize 8K
```

```
        managed by database
        using
        (
                device '/dev/rD1F16V2ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_033 of D1

drop tablespace is_order_033;
create regular tablespace is_order_033 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F17V1ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_034 of D1

drop tablespace is_order_034;
create regular tablespace is_order_034 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F17V2ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_035 of D1

drop tablespace is_order_035;
create regular tablespace is_order_035 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F18V1ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_036 of D1

drop tablespace is_order_036;
create regular tablespace is_order_036 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F18V2ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_037 of D1

drop tablespace is_order_037;
create regular tablespace is_order_037 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F19V1ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_038 of D1

drop tablespace is_order_038;
create regular tablespace is_order_038 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F19V2ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_039 of D1

drop tablespace is_order_039;
create regular tablespace is_order_039 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F20V1ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_040 of D1

drop tablespace is_order_040;
create regular tablespace is_order_040 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F20V2ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_041 of D1

drop tablespace is_order_041;
create regular tablespace is_order_041 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F21V1ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_042 of D1

drop tablespace is_order_042;
create regular tablespace is_order_042 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F21V2ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_043 of D1

drop tablespace is_order_043;
create regular tablespace is_order_043 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F22V1ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_044 of D1

drop tablespace is_order_044;
create regular tablespace is_order_044 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F22V2ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_045 of D1

drop tablespace is_order_045;
create regular tablespace is_order_045 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F23V1ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_046 of D1

drop tablespace is_order_046;
create regular tablespace is_order_046 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F23V2ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_047 of D1

drop tablespace is_order_047;
create regular tablespace is_order_047 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F24V1ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;
```

```
-- now creating TS for is_order_048 of D1

drop tablespace is_order_048;
create regular tablespace is_order_048 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F24V2ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_049 of D1

drop tablespace is_order_049;
create regular tablespace is_order_049 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F25V1ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_050 of D1

drop tablespace is_order_050;
create regular tablespace is_order_050 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F25V2ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_051 of D1

drop tablespace is_order_051;
create regular tablespace is_order_051 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F26V1ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_052 of D1

drop tablespace is_order_052;
create regular tablespace is_order_052 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F26V2ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_053 of D1

drop tablespace is_order_053;
create regular tablespace is_order_053 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F27V1ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_054 of D1

drop tablespace is_order_054;
create regular tablespace is_order_054 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F27V2ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_055 of D1

drop tablespace is_order_055;
create regular tablespace is_order_055 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F28V1ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_056 of D1

drop tablespace is_order_056;
create regular tablespace is_order_056 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F28V2ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_057 of D1

drop tablespace is_order_057;
create regular tablespace is_order_057 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F29V1ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_058 of D1

drop tablespace is_order_058;
create regular tablespace is_order_058 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F29V2ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_059 of D1

drop tablespace is_order_059;
create regular tablespace is_order_059 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F30V1ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_060 of D1

drop tablespace is_order_060;
create regular tablespace is_order_060 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F30V2ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_061 of D1

drop tablespace is_order_061;
create regular tablespace is_order_061 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F31V1ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_062 of D1

drop tablespace is_order_062;
create regular tablespace is_order_062 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F31V2ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_063 of D1

drop tablespace is_order_063;
create regular tablespace is_order_063 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F32V1ORDI' 114944
```

```
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_064 of D1

drop tablespace is_order_064;
create regular tablespace is_order_064 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F32V2ORDI' 114944
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

connect reset;
```

_ts/crts_customer.ddl

```
connect to tpcc;
-- now creating TS for ts_customer_001 of D1

drop tablespace ts_customer_001;
create regular tablespace ts_customer_001 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F01V1CST' 5117184
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_002 of D1

drop tablespace ts_customer_002;
create regular tablespace ts_customer_002 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F01V2CST' 5117184
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_003 of D1

drop tablespace ts_customer_003;
create regular tablespace ts_customer_003 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F02V1CST' 5117184
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_004 of D1

drop tablespace ts_customer_004;
create regular tablespace ts_customer_004 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F02V2CST' 5117184
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_005 of D1

drop tablespace ts_customer_005;
create regular tablespace ts_customer_005 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F03V1CST' 5117184
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_006 of D1

drop tablespace ts_customer_006;
create regular tablespace ts_customer_006 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F03V2CST' 5117184
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_007 of D1

drop tablespace ts_customer_007;
create regular tablespace ts_customer_007 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F04V1CST' 5117184
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_008 of D1

drop tablespace ts_customer_008;
create regular tablespace ts_customer_008 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F04V2CST' 5117184
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_009 of D1

drop tablespace ts_customer_009;
create regular tablespace ts_customer_009 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F05V1CST' 5117184
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_010 of D1

drop tablespace ts_customer_010;
create regular tablespace ts_customer_010 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F05V2CST' 5117184
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_011 of D1

drop tablespace ts_customer_011;
create regular tablespace ts_customer_011 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F06V1CST' 5117184
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_012 of D1

drop tablespace ts_customer_012;
create regular tablespace ts_customer_012 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F06V2CST' 5117184
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_013 of D1

drop tablespace ts_customer_013;
create regular tablespace ts_customer_013 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F07V1CST' 5117184
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_014 of D1

drop tablespace ts_customer_014;
create regular tablespace ts_customer_014 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F07V2CST' 5117184
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_015 of D1

drop tablespace ts_customer_015;
create regular tablespace ts_customer_015 pagesize 4K
         managed by database
         using
         (
```

```
                    device '/dev/rD1F08V1CST' 5117184
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_customer_016 of D1

drop tablespace ts_customer_016;
create regular tablespace ts_customer_016 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F08V2CST' 5117184
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_customer_017 of D1

drop tablespace ts_customer_017;
create regular tablespace ts_customer_017 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F09V1CST' 5117184
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_customer_018 of D1

drop tablespace ts_customer_018;
create regular tablespace ts_customer_018 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F09V2CST' 5117184
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_customer_019 of D1

drop tablespace ts_customer_019;
create regular tablespace ts_customer_019 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F10V1CST' 5117184
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_customer_020 of D1

drop tablespace ts_customer_020;
create regular tablespace ts_customer_020 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F10V2CST' 5117184
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_customer_021 of D1

drop tablespace ts_customer_021;
create regular tablespace ts_customer_021 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F11V1CST' 5117184
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_customer_022 of D1

drop tablespace ts_customer_022;
create regular tablespace ts_customer_022 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F11V2CST' 5117184
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_customer_023 of D1

drop tablespace ts_customer_023;
create regular tablespace ts_customer_023 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F12V1CST' 5117184
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_customer_024 of D1

drop tablespace ts_customer_024;
create regular tablespace ts_customer_024 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F12V2CST' 5117184
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_customer_025 of D1

drop tablespace ts_customer_025;
create regular tablespace ts_customer_025 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F13V1CST' 5117184
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_customer_026 of D1

drop tablespace ts_customer_026;
create regular tablespace ts_customer_026 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F13V2CST' 5117184
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_customer_027 of D1

drop tablespace ts_customer_027;
create regular tablespace ts_customer_027 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F14V1CST' 5117184
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_customer_028 of D1

drop tablespace ts_customer_028;
create regular tablespace ts_customer_028 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F14V2CST' 5117184
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_customer_029 of D1

drop tablespace ts_customer_029;
create regular tablespace ts_customer_029 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F15V1CST' 5117184
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_customer_030 of D1

drop tablespace ts_customer_030;
create regular tablespace ts_customer_030 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F15V2CST' 5117184
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_customer_031 of D1

drop tablespace ts_customer_031;
create regular tablespace ts_customer_031 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F16V1CST' 5117184
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_customer_032 of D1

drop tablespace ts_customer_032;
create regular tablespace ts_customer_032 pagesize 4K
           managed by database
```

```
            using
            (
                        device '/dev/rD1F16V2CST' 5117184
                        )
            extentsize 256
            prefetchsize 4096;
commit;

-- now creating TS for ts_customer_033 of D1

drop tablespace ts_customer_033;
create regular tablespace ts_customer_033 pagesize 4K
            managed by database
            using
            (
                        device '/dev/rD1F17V1CST' 5117184
                        )
            extentsize 256
            prefetchsize 4096;
commit;

-- now creating TS for ts_customer_034 of D1

drop tablespace ts_customer_034;
create regular tablespace ts_customer_034 pagesize 4K
            managed by database
            using
            (
                        device '/dev/rD1F17V2CST' 5117184
                        )
            extentsize 256
            prefetchsize 4096;
commit;

-- now creating TS for ts_customer_035 of D1

drop tablespace ts_customer_035;
create regular tablespace ts_customer_035 pagesize 4K
            managed by database
            using
            (
                        device '/dev/rD1F18V1CST' 5117184
                        )
            extentsize 256
            prefetchsize 4096;
commit;

-- now creating TS for ts_customer_036 of D1

drop tablespace ts_customer_036;
create regular tablespace ts_customer_036 pagesize 4K
            managed by database
            using
            (
                        device '/dev/rD1F18V2CST' 5117184
                        )
            extentsize 256
            prefetchsize 4096;
commit;

-- now creating TS for ts_customer_037 of D1

drop tablespace ts_customer_037;
create regular tablespace ts_customer_037 pagesize 4K
            managed by database
            using
            (
                        device '/dev/rD1F19V1CST' 5117184
                        )
            extentsize 256
            prefetchsize 4096;
commit;

-- now creating TS for ts_customer_038 of D1

drop tablespace ts_customer_038;
create regular tablespace ts_customer_038 pagesize 4K
            managed by database
            using
            (
                        device '/dev/rD1F19V2CST' 5117184
                        )
            extentsize 256
            prefetchsize 4096;
commit;

-- now creating TS for ts_customer_039 of D1

drop tablespace ts_customer_039;
create regular tablespace ts_customer_039 pagesize 4K
            managed by database
            using
            (
                        device '/dev/rD1F20V1CST' 5117184
                        )
            extentsize 256
            prefetchsize 4096;
commit;

-- now creating TS for ts_customer_040 of D1

drop tablespace ts_customer_040;
create regular tablespace ts_customer_040 pagesize 4K
            managed by database
            using
            (
                        device '/dev/rD1F20V2CST' 5117184
                        )
            extentsize 256
            prefetchsize 4096;
commit;

-- now creating TS for ts_customer_041 of D1

drop tablespace ts_customer_041;
create regular tablespace ts_customer_041 pagesize 4K
            managed by database
            using
            (
                        device '/dev/rD1F21V1CST' 5117184
                        )
            extentsize 256
            prefetchsize 4096;
commit;

-- now creating TS for ts_customer_042 of D1

drop tablespace ts_customer_042;
create regular tablespace ts_customer_042 pagesize 4K
            managed by database
            using
            (
                        device '/dev/rD1F21V2CST' 5117184
                        )
            extentsize 256
            prefetchsize 4096;
commit;

-- now creating TS for ts_customer_043 of D1

drop tablespace ts_customer_043;
create regular tablespace ts_customer_043 pagesize 4K
            managed by database
            using
            (
                        device '/dev/rD1F22V1CST' 5117184
                        )
            extentsize 256
            prefetchsize 4096;
commit;

-- now creating TS for ts_customer_044 of D1

drop tablespace ts_customer_044;
create regular tablespace ts_customer_044 pagesize 4K
            managed by database
            using
            (
                        device '/dev/rD1F22V2CST' 5117184
                        )
            extentsize 256
            prefetchsize 4096;
commit;

-- now creating TS for ts_customer_045 of D1

drop tablespace ts_customer_045;
create regular tablespace ts_customer_045 pagesize 4K
            managed by database
            using
            (
                        device '/dev/rD1F23V1CST' 5117184
                        )
            extentsize 256
            prefetchsize 4096;
commit;

-- now creating TS for ts_customer_046 of D1

drop tablespace ts_customer_046;
create regular tablespace ts_customer_046 pagesize 4K
            managed by database
            using
            (
                        device '/dev/rD1F23V2CST' 5117184
                        )
            extentsize 256
            prefetchsize 4096;
commit;

-- now creating TS for ts_customer_047 of D1

drop tablespace ts_customer_047;
create regular tablespace ts_customer_047 pagesize 4K
            managed by database
            using
            (
                        device '/dev/rD1F24V1CST' 5117184
                        )
            extentsize 256
            prefetchsize 4096;
commit;

-- now creating TS for ts_customer_048 of D1

drop tablespace ts_customer_048;
create regular tablespace ts_customer_048 pagesize 4K
            managed by database
            using
            (
                        device '/dev/rD1F24V2CST' 5117184
                        )
            extentsize 256
            prefetchsize 4096;
commit;

-- now creating TS for ts_customer_049 of D1

drop tablespace ts_customer_049;
```

```
create regular tablespace ts_customer_049 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F25V1CST' 5117184
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_050 of D1

drop tablespace ts_customer_050;
create regular tablespace ts_customer_050 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F25V2CST' 5117184
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_051 of D1

drop tablespace ts_customer_051;
create regular tablespace ts_customer_051 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F26V1CST' 5117184
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_052 of D1

drop tablespace ts_customer_052;
create regular tablespace ts_customer_052 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F26V2CST' 5117184
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_053 of D1

drop tablespace ts_customer_053;
create regular tablespace ts_customer_053 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F27V1CST' 5117184
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_054 of D1

drop tablespace ts_customer_054;
create regular tablespace ts_customer_054 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F27V2CST' 5117184
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_055 of D1

drop tablespace ts_customer_055;
create regular tablespace ts_customer_055 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F28V1CST' 5117184
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_056 of D1

drop tablespace ts_customer_056;
create regular tablespace ts_customer_056 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F28V2CST' 5117184
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_057 of D1

drop tablespace ts_customer_057;
create regular tablespace ts_customer_057 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F29V1CST' 5117184
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_058 of D1

drop tablespace ts_customer_058;
create regular tablespace ts_customer_058 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F29V2CST' 5117184
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_059 of D1

drop tablespace ts_customer_059;
create regular tablespace ts_customer_059 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F30V1CST' 5117184
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_060 of D1

drop tablespace ts_customer_060;
create regular tablespace ts_customer_060 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F30V2CST' 5117184
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_061 of D1

drop tablespace ts_customer_061;
create regular tablespace ts_customer_061 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F31V1CST' 5117184
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_062 of D1

drop tablespace ts_customer_062;
create regular tablespace ts_customer_062 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F31V2CST' 5117184
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_063 of D1

drop tablespace ts_customer_063;
create regular tablespace ts_customer_063 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F32V1CST' 5117184
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_064 of D1

drop tablespace ts_customer_064;
create regular tablespace ts_customer_064 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F32V2CST' 5117184
                )
        extentsize 256
        prefetchsize 4096;
commit;

connect reset;
```

ts/crts_dist.ddl

```
connect to tpcc;
-- now creating TS for ts_dist_001 of D1

drop tablespace ts_dist_001;
create regular tablespace ts_dist_001 pagesize 4K
```

```
        managed by database
        using
        (
                device '/dev/rD1F01V1DIST' 512,
                device '/dev/rD1F01V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_002 of D1

drop tablespace ts_dist_002;
create regular tablespace ts_dist_002 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F02V1DIST' 512,
                device '/dev/rD1F02V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_003 of D1

drop tablespace ts_dist_003;
create regular tablespace ts_dist_003 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F03V1DIST' 512,
                device '/dev/rD1F03V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_004 of D1

drop tablespace ts_dist_004;
create regular tablespace ts_dist_004 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F04V1DIST' 512,
                device '/dev/rD1F04V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_005 of D1

drop tablespace ts_dist_005;
create regular tablespace ts_dist_005 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F05V1DIST' 512,
                device '/dev/rD1F05V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_006 of D1

drop tablespace ts_dist_006;
create regular tablespace ts_dist_006 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F06V1DIST' 512,
                device '/dev/rD1F06V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_007 of D1

drop tablespace ts_dist_007;
create regular tablespace ts_dist_007 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F07V1DIST' 512,
                device '/dev/rD1F07V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_008 of D1

drop tablespace ts_dist_008;
create regular tablespace ts_dist_008 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F08V1DIST' 512,
                device '/dev/rD1F08V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_009 of D1

drop tablespace ts_dist_009;
create regular tablespace ts_dist_009 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F09V1DIST' 512,
                device '/dev/rD1F09V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_010 of D1

drop tablespace ts_dist_010;
create regular tablespace ts_dist_010 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F10V1DIST' 512,
                device '/dev/rD1F10V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_011 of D1

drop tablespace ts_dist_011;
create regular tablespace ts_dist_011 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F11V1DIST' 512,
                device '/dev/rD1F11V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_012 of D1

drop tablespace ts_dist_012;
create regular tablespace ts_dist_012 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F12V1DIST' 512,
                device '/dev/rD1F12V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_013 of D1

drop tablespace ts_dist_013;
create regular tablespace ts_dist_013 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F13V1DIST' 512,
                device '/dev/rD1F13V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_014 of D1

drop tablespace ts_dist_014;
create regular tablespace ts_dist_014 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F14V1DIST' 512,
                device '/dev/rD1F14V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_015 of D1

drop tablespace ts_dist_015;
create regular tablespace ts_dist_015 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F15V1DIST' 512,
                device '/dev/rD1F15V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_016 of D1

drop tablespace ts_dist_016;
create regular tablespace ts_dist_016 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F16V1DIST' 512,
                device '/dev/rD1F16V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;
```

```
-- now creating TS for ts_dist_017 of D1

drop tablespace ts_dist_017;
create regular tablespace ts_dist_017 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F17V1DIST' 512,
                device '/dev/rD1F17V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_018 of D1

drop tablespace ts_dist_018;
create regular tablespace ts_dist_018 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F18V1DIST' 512,
                device '/dev/rD1F18V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_019 of D1

drop tablespace ts_dist_019;
create regular tablespace ts_dist_019 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F19V1DIST' 512,
                device '/dev/rD1F19V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_020 of D1

drop tablespace ts_dist_020;
create regular tablespace ts_dist_020 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F20V1DIST' 512,
                device '/dev/rD1F20V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_021 of D1

drop tablespace ts_dist_021;
create regular tablespace ts_dist_021 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F21V1DIST' 512,
                device '/dev/rD1F21V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_022 of D1

drop tablespace ts_dist_022;
create regular tablespace ts_dist_022 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F22V1DIST' 512,
                device '/dev/rD1F22V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_023 of D1

drop tablespace ts_dist_023;
create regular tablespace ts_dist_023 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F23V1DIST' 512,
                device '/dev/rD1F23V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_024 of D1

drop tablespace ts_dist_024;
create regular tablespace ts_dist_024 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F24V1DIST' 512,
                device '/dev/rD1F24V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_025 of D1

drop tablespace ts_dist_025;
create regular tablespace ts_dist_025 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F25V1DIST' 512,
                device '/dev/rD1F25V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_026 of D1

drop tablespace ts_dist_026;
create regular tablespace ts_dist_026 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F26V1DIST' 512,
                device '/dev/rD1F26V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_027 of D1

drop tablespace ts_dist_027;
create regular tablespace ts_dist_027 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F27V1DIST' 512,
                device '/dev/rD1F27V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_028 of D1

drop tablespace ts_dist_028;
create regular tablespace ts_dist_028 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F28V1DIST' 512,
                device '/dev/rD1F28V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_029 of D1

drop tablespace ts_dist_029;
create regular tablespace ts_dist_029 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F29V1DIST' 512,
                device '/dev/rD1F29V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_030 of D1

drop tablespace ts_dist_030;
create regular tablespace ts_dist_030 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F30V1DIST' 512,
                device '/dev/rD1F30V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_031 of D1

drop tablespace ts_dist_031;
create regular tablespace ts_dist_031 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F31V1DIST' 512,
                device '/dev/rD1F31V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_032 of D1

drop tablespace ts_dist_032;
create regular tablespace ts_dist_032 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F32V1DIST' 512,
```

```
                device '/dev/rD1F32V2DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

connect reset;
```

_ts/crts_history.ddl

```
connect to tpcc;
-- now creating TS for ts_history_001 of D1

drop tablespace ts_history_001;
create regular tablespace ts_history_001 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F01V1HIST' 122880,
                device '/dev/rD1F01V2HIST' 122880
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_002 of D1

drop tablespace ts_history_002;
create regular tablespace ts_history_002 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F02V1HIST' 122880,
                device '/dev/rD1F02V2HIST' 122880
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_003 of D1

drop tablespace ts_history_003;
create regular tablespace ts_history_003 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F03V1HIST' 122880,
                device '/dev/rD1F03V2HIST' 122880
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_004 of D1

drop tablespace ts_history_004;
create regular tablespace ts_history_004 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F04V1HIST' 122880,
                device '/dev/rD1F04V2HIST' 122880
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_005 of D1

drop tablespace ts_history_005;
create regular tablespace ts_history_005 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F05V1HIST' 122880,
                device '/dev/rD1F05V2HIST' 122880
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_006 of D1

drop tablespace ts_history_006;
create regular tablespace ts_history_006 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F06V1HIST' 122880,
                device '/dev/rD1F06V2HIST' 122880
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_007 of D1

drop tablespace ts_history_007;
create regular tablespace ts_history_007 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F07V1HIST' 122880,
                device '/dev/rD1F07V2HIST' 122880
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_008 of D1

drop tablespace ts_history_008;
create regular tablespace ts_history_008 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F08V1HIST' 122880,
                device '/dev/rD1F08V2HIST' 122880
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_009 of D1

drop tablespace ts_history_009;
create regular tablespace ts_history_009 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F09V1HIST' 122880,
                device '/dev/rD1F09V2HIST' 122880
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_010 of D1

drop tablespace ts_history_010;
create regular tablespace ts_history_010 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F10V1HIST' 122880,
                device '/dev/rD1F10V2HIST' 122880
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_011 of D1

drop tablespace ts_history_011;
create regular tablespace ts_history_011 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F11V1HIST' 122880,
                device '/dev/rD1F11V2HIST' 122880
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_012 of D1

drop tablespace ts_history_012;
create regular tablespace ts_history_012 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F12V1HIST' 122880,
                device '/dev/rD1F12V2HIST' 122880
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_013 of D1

drop tablespace ts_history_013;
create regular tablespace ts_history_013 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F13V1HIST' 122880,
                device '/dev/rD1F13V2HIST' 122880
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_014 of D1

drop tablespace ts_history_014;
create regular tablespace ts_history_014 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F14V1HIST' 122880,
```

```
                    device '/dev/rD1F14V2HIST' 122880
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_015 of D1

drop tablespace ts_history_015;
create regular tablespace ts_history_015 pagesize 16K
           managed by database
           using
           (
                    device '/dev/rD1F15V1HIST' 122880,
                    device '/dev/rD1F15V2HIST' 122880
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_016 of D1

drop tablespace ts_history_016;
create regular tablespace ts_history_016 pagesize 16K
           managed by database
           using
           (
                    device '/dev/rD1F16V1HIST' 122880,
                    device '/dev/rD1F16V2HIST' 122880
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_017 of D1

drop tablespace ts_history_017;
create regular tablespace ts_history_017 pagesize 16K
           managed by database
           using
           (
                    device '/dev/rD1F17V1HIST' 122880,
                    device '/dev/rD1F17V2HIST' 122880
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_018 of D1

drop tablespace ts_history_018;
create regular tablespace ts_history_018 pagesize 16K
           managed by database
           using
           (
                    device '/dev/rD1F18V1HIST' 122880,
                    device '/dev/rD1F18V2HIST' 122880
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_019 of D1

drop tablespace ts_history_019;
create regular tablespace ts_history_019 pagesize 16K
           managed by database
           using
           (
                    device '/dev/rD1F19V1HIST' 122880,
                    device '/dev/rD1F19V2HIST' 122880
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_020 of D1

drop tablespace ts_history_020;
create regular tablespace ts_history_020 pagesize 16K
           managed by database
           using
           (
                    device '/dev/rD1F20V1HIST' 122880,
                    device '/dev/rD1F20V2HIST' 122880
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_021 of D1

drop tablespace ts_history_021;
create regular tablespace ts_history_021 pagesize 16K
           managed by database
           using
           (
                    device '/dev/rD1F21V1HIST' 122880,
                    device '/dev/rD1F21V2HIST' 122880
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_022 of D1

drop tablespace ts_history_022;
create regular tablespace ts_history_022 pagesize 16K
           managed by database
           using
           (
                    device '/dev/rD1F22V1HIST' 122880,
                    device '/dev/rD1F22V2HIST' 122880
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_023 of D1

drop tablespace ts_history_023;
create regular tablespace ts_history_023 pagesize 16K
           managed by database
           using
           (
                    device '/dev/rD1F23V1HIST' 122880,
                    device '/dev/rD1F23V2HIST' 122880
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_024 of D1

drop tablespace ts_history_024;
create regular tablespace ts_history_024 pagesize 16K
           managed by database
           using
           (
                    device '/dev/rD1F24V1HIST' 122880,
                    device '/dev/rD1F24V2HIST' 122880
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_025 of D1

drop tablespace ts_history_025;
create regular tablespace ts_history_025 pagesize 16K
           managed by database
           using
           (
                    device '/dev/rD1F25V1HIST' 122880,
                    device '/dev/rD1F25V2HIST' 122880
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_026 of D1

drop tablespace ts_history_026;
create regular tablespace ts_history_026 pagesize 16K
           managed by database
           using
           (
                    device '/dev/rD1F26V1HIST' 122880,
                    device '/dev/rD1F26V2HIST' 122880
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_027 of D1

drop tablespace ts_history_027;
create regular tablespace ts_history_027 pagesize 16K
           managed by database
           using
           (
                    device '/dev/rD1F27V1HIST' 122880,
                    device '/dev/rD1F27V2HIST' 122880
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_028 of D1

drop tablespace ts_history_028;
create regular tablespace ts_history_028 pagesize 16K
           managed by database
           using
           (
                    device '/dev/rD1F28V1HIST' 122880,
                    device '/dev/rD1F28V2HIST' 122880
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_029 of D1
```

```
drop tablespace ts_history_029;
create regular tablespace ts_history_029 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F29V1HIST' 122880,
                device '/dev/rD1F29V2HIST' 122880
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_030 of D1

drop tablespace ts_history_030;
create regular tablespace ts_history_030 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F30V1HIST' 122880,
                device '/dev/rD1F30V2HIST' 122880
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_031 of D1

drop tablespace ts_history_031;
create regular tablespace ts_history_031 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F31V1HIST' 122880,
                device '/dev/rD1F31V2HIST' 122880
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_032 of D1

drop tablespace ts_history_032;
create regular tablespace ts_history_032 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F32V1HIST' 122880,
                device '/dev/rD1F32V2HIST' 122880
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

connect reset;
```

ts/crts_item.ddl

```
connect to tpcc;
-- now creating TS for ts_item_001 of D1

drop tablespace ts_item_001;
create regular tablespace ts_item_001 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F01V1ITEM' 1408,
                device '/dev/rD1F01V2ITEM' 1408,
                device '/dev/rD1F02V1ITEM' 1408,
                device '/dev/rD1F02V2ITEM' 1408,
                device '/dev/rD1F03V1ITEM' 1408,
                device '/dev/rD1F03V2ITEM' 1408,
                device '/dev/rD1F04V1ITEM' 1408,
                device '/dev/rD1F04V2ITEM' 1408,
                device '/dev/rD1F05V1ITEM' 1408,
                device '/dev/rD1F05V2ITEM' 1408,
                device '/dev/rD1F06V1ITEM' 1408,
                device '/dev/rD1F06V2ITEM' 1408,
                device '/dev/rD1F07V1ITEM' 1408,
                device '/dev/rD1F07V2ITEM' 1408,
                device '/dev/rD1F08V1ITEM' 1408,
                device '/dev/rD1F08V2ITEM' 1408,
                device '/dev/rD1F09V1ITEM' 1408,
                device '/dev/rD1F09V2ITEM' 1408,
                device '/dev/rD1F10V1ITEM' 1408,
                device '/dev/rD1F10V2ITEM' 1408,
                device '/dev/rD1F11V1ITEM' 1408,
                device '/dev/rD1F11V2ITEM' 1408,
                device '/dev/rD1F12V1ITEM' 1408,
                device '/dev/rD1F12V2ITEM' 1408,
                device '/dev/rD1F13V1ITEM' 1408,
                device '/dev/rD1F13V2ITEM' 1408,
                device '/dev/rD1F14V1ITEM' 1408,
                device '/dev/rD1F14V2ITEM' 1408,
                device '/dev/rD1F15V1ITEM' 1408,
                device '/dev/rD1F15V2ITEM' 1408,
                device '/dev/rD1F16V1ITEM' 1408,
                device '/dev/rD1F16V2ITEM' 1408,
                device '/dev/rD1F17V1ITEM' 1408,
                device '/dev/rD1F17V2ITEM' 1408,
                device '/dev/rD1F18V1ITEM' 1408,
                device '/dev/rD1F18V2ITEM' 1408,
                device '/dev/rD1F19V1ITEM' 1408,
                device '/dev/rD1F19V2ITEM' 1408,
                device '/dev/rD1F20V1ITEM' 1408,
                device '/dev/rD1F20V2ITEM' 1408,
                device '/dev/rD1F21V1ITEM' 1408,
                device '/dev/rD1F21V2ITEM' 1408,
                device '/dev/rD1F22V1ITEM' 1408,
                device '/dev/rD1F22V2ITEM' 1408,
                device '/dev/rD1F23V1ITEM' 1408,
                device '/dev/rD1F23V2ITEM' 1408,
                device '/dev/rD1F24V1ITEM' 1408,
                device '/dev/rD1F24V2ITEM' 1408,
                device '/dev/rD1F25V1ITEM' 1408,
                device '/dev/rD1F25V2ITEM' 1408,
                device '/dev/rD1F26V1ITEM' 1408,
                device '/dev/rD1F26V2ITEM' 1408,
                device '/dev/rD1F27V1ITEM' 1408,
                device '/dev/rD1F27V2ITEM' 1408,
                device '/dev/rD1F28V1ITEM' 1408,
                device '/dev/rD1F28V2ITEM' 1408,
                device '/dev/rD1F29V1ITEM' 1408,
                device '/dev/rD1F29V2ITEM' 1408,
                device '/dev/rD1F30V1ITEM' 1408,
                device '/dev/rD1F30V2ITEM' 1408,
                device '/dev/rD1F31V1ITEM' 1408,
                device '/dev/rD1F31V2ITEM' 1408,
                device '/dev/rD1F32V1ITEM' 1408,
                device '/dev/rD1F32V2ITEM' 1408
                )
        extentsize 16
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

connect reset;
```

ts/crts_neworda.ddl

```
connect to tpcc;
-- now creating TS for ts_neworda_001 of D1

drop tablespace ts_neworda_001;
create regular tablespace ts_neworda_001 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F01V1NORA' 153856,
                device '/dev/rD1F01V2NORA' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_002 of D1

drop tablespace ts_neworda_002;
create regular tablespace ts_neworda_002 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F02V1NORA' 153856,
                device '/dev/rD1F02V2NORA' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_003 of D1

drop tablespace ts_neworda_003;
create regular tablespace ts_neworda_003 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F03V1NORA' 153856,
                device '/dev/rD1F03V2NORA' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_004 of D1

drop tablespace ts_neworda_004;
create regular tablespace ts_neworda_004 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F04V1NORA' 153856,
                device '/dev/rD1F04V2NORA' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_005 of D1

drop tablespace ts_neworda_005;
create regular tablespace ts_neworda_005 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F05V1NORA' 153856,
```

```
                    device '/dev/rD1F05V2NORA' 153856
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_006 of D1

drop tablespace ts_neworda_006;
create regular tablespace ts_neworda_006 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F06V1NORA' 153856,
                    device '/dev/rD1F06V2NORA' 153856
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_007 of D1

drop tablespace ts_neworda_007;
create regular tablespace ts_neworda_007 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F07V1NORA' 153856,
                    device '/dev/rD1F07V2NORA' 153856
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_008 of D1

drop tablespace ts_neworda_008;
create regular tablespace ts_neworda_008 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F08V1NORA' 153856,
                    device '/dev/rD1F08V2NORA' 153856
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_009 of D1

drop tablespace ts_neworda_009;
create regular tablespace ts_neworda_009 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F09V1NORA' 153856,
                    device '/dev/rD1F09V2NORA' 153856
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_010 of D1

drop tablespace ts_neworda_010;
create regular tablespace ts_neworda_010 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F10V1NORA' 153856,
                    device '/dev/rD1F10V2NORA' 153856
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_011 of D1

drop tablespace ts_neworda_011;
create regular tablespace ts_neworda_011 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F11V1NORA' 153856,
                    device '/dev/rD1F11V2NORA' 153856
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_012 of D1

drop tablespace ts_neworda_012;
create regular tablespace ts_neworda_012 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F12V1NORA' 153856,
                    device '/dev/rD1F12V2NORA' 153856
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_013 of D1

drop tablespace ts_neworda_013;
create regular tablespace ts_neworda_013 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F13V1NORA' 153856,
                    device '/dev/rD1F13V2NORA' 153856
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_014 of D1

drop tablespace ts_neworda_014;
create regular tablespace ts_neworda_014 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F14V1NORA' 153856,
                    device '/dev/rD1F14V2NORA' 153856
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_015 of D1

drop tablespace ts_neworda_015;
create regular tablespace ts_neworda_015 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F15V1NORA' 153856,
                    device '/dev/rD1F15V2NORA' 153856
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_016 of D1

drop tablespace ts_neworda_016;
create regular tablespace ts_neworda_016 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F16V1NORA' 153856,
                    device '/dev/rD1F16V2NORA' 153856
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_017 of D1

drop tablespace ts_neworda_017;
create regular tablespace ts_neworda_017 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F17V1NORA' 153856,
                    device '/dev/rD1F17V2NORA' 153856
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_018 of D1

drop tablespace ts_neworda_018;
create regular tablespace ts_neworda_018 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F18V1NORA' 153856,
                    device '/dev/rD1F18V2NORA' 153856
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_019 of D1

drop tablespace ts_neworda_019;
create regular tablespace ts_neworda_019 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F19V1NORA' 153856,
                    device '/dev/rD1F19V2NORA' 153856
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_020 of D1

drop tablespace ts_neworda_020;
create regular tablespace ts_neworda_020 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F20V1NORA' 153856,
                    device '/dev/rD1F20V2NORA' 153856
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_021 of D1

drop tablespace ts_neworda_021;
```

```
create regular tablespace ts_neworda_021 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F21V1NORA' 153856,
                device '/dev/rD1F21V2NORA' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_022 of D1

drop tablespace ts_neworda_022;
create regular tablespace ts_neworda_022 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F22V1NORA' 153856,
                device '/dev/rD1F22V2NORA' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_023 of D1

drop tablespace ts_neworda_023;
create regular tablespace ts_neworda_023 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F23V1NORA' 153856,
                device '/dev/rD1F23V2NORA' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_024 of D1

drop tablespace ts_neworda_024;
create regular tablespace ts_neworda_024 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F24V1NORA' 153856,
                device '/dev/rD1F24V2NORA' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_025 of D1

drop tablespace ts_neworda_025;
create regular tablespace ts_neworda_025 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F25V1NORA' 153856,
                device '/dev/rD1F25V2NORA' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_026 of D1

drop tablespace ts_neworda_026;
create regular tablespace ts_neworda_026 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F26V1NORA' 153856,
                device '/dev/rD1F26V2NORA' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_027 of D1

drop tablespace ts_neworda_027;
create regular tablespace ts_neworda_027 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F27V1NORA' 153856,
                device '/dev/rD1F27V2NORA' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_028 of D1

drop tablespace ts_neworda_028;
create regular tablespace ts_neworda_028 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F28V1NORA' 153856,
                device '/dev/rD1F28V2NORA' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_029 of D1

drop tablespace ts_neworda_029;
create regular tablespace ts_neworda_029 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F29V1NORA' 153856,
                device '/dev/rD1F29V2NORA' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_030 of D1

drop tablespace ts_neworda_030;
create regular tablespace ts_neworda_030 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F30V1NORA' 153856,
                device '/dev/rD1F30V2NORA' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_031 of D1

drop tablespace ts_neworda_031;
create regular tablespace ts_neworda_031 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F31V1NORA' 153856,
                device '/dev/rD1F31V2NORA' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_032 of D1

drop tablespace ts_neworda_032;
create regular tablespace ts_neworda_032 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F32V1NORA' 153856,
                device '/dev/rD1F32V2NORA' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

connect reset;
```

ts/crts_newordb.ddl

```
connect to tpcc;
-- now creating TS for ts_newordb_001 of D1

drop tablespace ts_newordb_001;
create regular tablespace ts_newordb_001 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F01V1NORB' 153856,
                device '/dev/rD1F01V2NORB' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_002 of D1

drop tablespace ts_newordb_002;
create regular tablespace ts_newordb_002 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F02V1NORB' 153856,
                device '/dev/rD1F02V2NORB' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_003 of D1

drop tablespace ts_newordb_003;
create regular tablespace ts_newordb_003 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F03V1NORB' 153856,
                device '/dev/rD1F03V2NORB' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_004 of D1
```

```
drop tablespace ts_newordb_004;
create regular tablespace ts_newordb_004 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F04V1NORB' 153856,
                device '/dev/rD1F04V2NORB' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_005 of D1

drop tablespace ts_newordb_005;
create regular tablespace ts_newordb_005 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F05V1NORB' 153856,
                device '/dev/rD1F05V2NORB' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_006 of D1

drop tablespace ts_newordb_006;
create regular tablespace ts_newordb_006 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F06V1NORB' 153856,
                device '/dev/rD1F06V2NORB' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_007 of D1

drop tablespace ts_newordb_007;
create regular tablespace ts_newordb_007 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F07V1NORB' 153856,
                device '/dev/rD1F07V2NORB' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_008 of D1

drop tablespace ts_newordb_008;
create regular tablespace ts_newordb_008 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F08V1NORB' 153856,
                device '/dev/rD1F08V2NORB' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_009 of D1

drop tablespace ts_newordb_009;
create regular tablespace ts_newordb_009 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F09V1NORB' 153856,
                device '/dev/rD1F09V2NORB' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_010 of D1

drop tablespace ts_newordb_010;
create regular tablespace ts_newordb_010 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F10V1NORB' 153856,
                device '/dev/rD1F10V2NORB' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_011 of D1

drop tablespace ts_newordb_011;
create regular tablespace ts_newordb_011 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F11V1NORB' 153856,
                device '/dev/rD1F11V2NORB' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_012 of D1

drop tablespace ts_newordb_012;
create regular tablespace ts_newordb_012 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F12V1NORB' 153856,
                device '/dev/rD1F12V2NORB' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_013 of D1

drop tablespace ts_newordb_013;
create regular tablespace ts_newordb_013 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F13V1NORB' 153856,
                device '/dev/rD1F13V2NORB' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_014 of D1

drop tablespace ts_newordb_014;
create regular tablespace ts_newordb_014 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F14V1NORB' 153856,
                device '/dev/rD1F14V2NORB' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_015 of D1

drop tablespace ts_newordb_015;
create regular tablespace ts_newordb_015 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F15V1NORB' 153856,
                device '/dev/rD1F15V2NORB' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_016 of D1

drop tablespace ts_newordb_016;
create regular tablespace ts_newordb_016 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F16V1NORB' 153856,
                device '/dev/rD1F16V2NORB' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_017 of D1

drop tablespace ts_newordb_017;
create regular tablespace ts_newordb_017 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F17V1NORB' 153856,
                device '/dev/rD1F17V2NORB' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_018 of D1

drop tablespace ts_newordb_018;
create regular tablespace ts_newordb_018 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F18V1NORB' 153856,
                device '/dev/rD1F18V2NORB' 153856
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_019 of D1

drop tablespace ts_newordb_019;
create regular tablespace ts_newordb_019 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F19V1NORB' 153856,
                device '/dev/rD1F19V2NORB' 153856
                )
        extentsize 256
```

```
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_020 of D1

drop tablespace ts_newordb_020;
create regular tablespace ts_newordb_020 pagesize 4K
         managed by database
         using
         (
                  device '/dev/rD1F20V1NORB' 153856,
                  device '/dev/rD1F20V2NORB' 153856
                 )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_021 of D1

drop tablespace ts_newordb_021;
create regular tablespace ts_newordb_021 pagesize 4K
         managed by database
         using
         (
                  device '/dev/rD1F21V1NORB' 153856,
                  device '/dev/rD1F21V2NORB' 153856
                 )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_022 of D1

drop tablespace ts_newordb_022;
create regular tablespace ts_newordb_022 pagesize 4K
         managed by database
         using
         (
                  device '/dev/rD1F22V1NORB' 153856,
                  device '/dev/rD1F22V2NORB' 153856
                 )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_023 of D1

drop tablespace ts_newordb_023;
create regular tablespace ts_newordb_023 pagesize 4K
         managed by database
         using
         (
                  device '/dev/rD1F23V1NORB' 153856,
                  device '/dev/rD1F23V2NORB' 153856
                 )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_024 of D1

drop tablespace ts_newordb_024;
create regular tablespace ts_newordb_024 pagesize 4K
         managed by database
         using
         (
                  device '/dev/rD1F24V1NORB' 153856,
                  device '/dev/rD1F24V2NORB' 153856
                 )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_025 of D1

drop tablespace ts_newordb_025;
create regular tablespace ts_newordb_025 pagesize 4K
         managed by database
         using
         (
                  device '/dev/rD1F25V1NORB' 153856,
                  device '/dev/rD1F25V2NORB' 153856
                 )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_026 of D1

drop tablespace ts_newordb_026;
create regular tablespace ts_newordb_026 pagesize 4K
         managed by database
         using
         (
                  device '/dev/rD1F26V1NORB' 153856,
                  device '/dev/rD1F26V2NORB' 153856
                 )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_027 of D1

drop tablespace ts_newordb_027;
create regular tablespace ts_newordb_027 pagesize 4K
         managed by database
         using
         (
                  device '/dev/rD1F27V1NORB' 153856,
                  device '/dev/rD1F27V2NORB' 153856
                 )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_028 of D1

drop tablespace ts_newordb_028;
create regular tablespace ts_newordb_028 pagesize 4K
         managed by database
         using
         (
                  device '/dev/rD1F28V1NORB' 153856,
                  device '/dev/rD1F28V2NORB' 153856
                 )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_029 of D1

drop tablespace ts_newordb_029;
create regular tablespace ts_newordb_029 pagesize 4K
         managed by database
         using
         (
                  device '/dev/rD1F29V1NORB' 153856,
                  device '/dev/rD1F29V2NORB' 153856
                 )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_030 of D1

drop tablespace ts_newordb_030;
create regular tablespace ts_newordb_030 pagesize 4K
         managed by database
         using
         (
                  device '/dev/rD1F30V1NORB' 153856,
                  device '/dev/rD1F30V2NORB' 153856
                 )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_031 of D1

drop tablespace ts_newordb_031;
create regular tablespace ts_newordb_031 pagesize 4K
         managed by database
         using
         (
                  device '/dev/rD1F31V1NORB' 153856,
                  device '/dev/rD1F31V2NORB' 153856
                 )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_032 of D1

drop tablespace ts_newordb_032;
create regular tablespace ts_newordb_032 pagesize 4K
         managed by database
         using
         (
                  device '/dev/rD1F32V1NORB' 153856,
                  device '/dev/rD1F32V2NORB' 153856
                 )
        extentsize 256
        prefetchsize 4096;
commit;

connect reset;
```

_ts/crts_order.ddl

```
connect to tpcc;
-- now creating TS for ts_order_001 of D1

drop tablespace ts_order_001;
create regular tablespace ts_order_001 pagesize 8K
         managed by database
         using
         (
                  device '/dev/rD1F01V1ORD' 156928
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_002 of D1

drop tablespace ts_order_002;
create regular tablespace ts_order_002 pagesize 8K
         managed by database
         using
         (
                  device '/dev/rD1F01V2ORD' 156928
                 )
        extentsize 256
```

```
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_003 of D1

drop tablespace ts_order_003;
create regular tablespace ts_order_003 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F02V1ORD' 156928
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_004 of D1

drop tablespace ts_order_004;
create regular tablespace ts_order_004 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F02V2ORD' 156928
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_005 of D1

drop tablespace ts_order_005;
create regular tablespace ts_order_005 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F03V1ORD' 156928
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_006 of D1

drop tablespace ts_order_006;
create regular tablespace ts_order_006 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F03V2ORD' 156928
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_007 of D1

drop tablespace ts_order_007;
create regular tablespace ts_order_007 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F04V1ORD' 156928
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_008 of D1

drop tablespace ts_order_008;
create regular tablespace ts_order_008 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F04V2ORD' 156928
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_009 of D1

drop tablespace ts_order_009;
create regular tablespace ts_order_009 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F05V1ORD' 156928
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_010 of D1

drop tablespace ts_order_010;
create regular tablespace ts_order_010 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F05V2ORD' 156928
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_011 of D1

drop tablespace ts_order_011;
create regular tablespace ts_order_011 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F06V1ORD' 156928
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_012 of D1

drop tablespace ts_order_012;
create regular tablespace ts_order_012 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F06V2ORD' 156928
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_013 of D1

drop tablespace ts_order_013;
create regular tablespace ts_order_013 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F07V1ORD' 156928
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_014 of D1

drop tablespace ts_order_014;
create regular tablespace ts_order_014 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F07V2ORD' 156928
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_015 of D1

drop tablespace ts_order_015;
create regular tablespace ts_order_015 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F08V1ORD' 156928
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_016 of D1

drop tablespace ts_order_016;
create regular tablespace ts_order_016 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F08V2ORD' 156928
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_017 of D1

drop tablespace ts_order_017;
create regular tablespace ts_order_017 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F09V1ORD' 156928
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_018 of D1

drop tablespace ts_order_018;
create regular tablespace ts_order_018 pagesize 8K
         managed by database
```

```
            using
            (
                        device '/dev/rD1F09V2ORD' 156928
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_019 of D1

drop tablespace ts_order_019;
create regular tablespace ts_order_019 pagesize 8K
            managed by database
            using
            (
                        device '/dev/rD1F10V1ORD' 156928
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_020 of D1

drop tablespace ts_order_020;
create regular tablespace ts_order_020 pagesize 8K
            managed by database
            using
            (
                        device '/dev/rD1F10V2ORD' 156928
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_021 of D1

drop tablespace ts_order_021;
create regular tablespace ts_order_021 pagesize 8K
            managed by database
            using
            (
                        device '/dev/rD1F11V1ORD' 156928
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_022 of D1

drop tablespace ts_order_022;
create regular tablespace ts_order_022 pagesize 8K
            managed by database
            using
            (
                        device '/dev/rD1F11V2ORD' 156928
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_023 of D1

drop tablespace ts_order_023;
create regular tablespace ts_order_023 pagesize 8K
            managed by database
            using
            (
                        device '/dev/rD1F12V1ORD' 156928
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_024 of D1

drop tablespace ts_order_024;
create regular tablespace ts_order_024 pagesize 8K
            managed by database
            using
            (
                        device '/dev/rD1F12V2ORD' 156928
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_025 of D1

drop tablespace ts_order_025;
create regular tablespace ts_order_025 pagesize 8K
            managed by database
            using
            (
                        device '/dev/rD1F13V1ORD' 156928
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_026 of D1

drop tablespace ts_order_026;
create regular tablespace ts_order_026 pagesize 8K
            managed by database
            using
            (
                        device '/dev/rD1F13V2ORD' 156928
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_027 of D1

drop tablespace ts_order_027;
create regular tablespace ts_order_027 pagesize 8K
            managed by database
            using
            (
                        device '/dev/rD1F14V1ORD' 156928
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_028 of D1

drop tablespace ts_order_028;
create regular tablespace ts_order_028 pagesize 8K
            managed by database
            using
            (
                        device '/dev/rD1F14V2ORD' 156928
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_029 of D1

drop tablespace ts_order_029;
create regular tablespace ts_order_029 pagesize 8K
            managed by database
            using
            (
                        device '/dev/rD1F15V1ORD' 156928
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_030 of D1

drop tablespace ts_order_030;
create regular tablespace ts_order_030 pagesize 8K
            managed by database
            using
            (
                        device '/dev/rD1F15V2ORD' 156928
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_031 of D1

drop tablespace ts_order_031;
create regular tablespace ts_order_031 pagesize 8K
            managed by database
            using
            (
                        device '/dev/rD1F16V1ORD' 156928
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_032 of D1

drop tablespace ts_order_032;
create regular tablespace ts_order_032 pagesize 8K
            managed by database
            using
            (
                        device '/dev/rD1F16V2ORD' 156928
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_033 of D1

drop tablespace ts_order_033;
create regular tablespace ts_order_033 pagesize 8K
            managed by database
            using
            (
                        device '/dev/rD1F17V1ORD' 156928
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;
```

```
-- now creating TS for ts_order_034 of D1

drop tablespace ts_order_034;
create regular tablespace ts_order_034 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F17V2ORD' 156928
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_035 of D1

drop tablespace ts_order_035;
create regular tablespace ts_order_035 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F18V1ORD' 156928
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_036 of D1

drop tablespace ts_order_036;
create regular tablespace ts_order_036 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F18V2ORD' 156928
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_037 of D1

drop tablespace ts_order_037;
create regular tablespace ts_order_037 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F19V1ORD' 156928
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_038 of D1

drop tablespace ts_order_038;
create regular tablespace ts_order_038 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F19V2ORD' 156928
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_039 of D1

drop tablespace ts_order_039;
create regular tablespace ts_order_039 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F20V1ORD' 156928
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_040 of D1

drop tablespace ts_order_040;
create regular tablespace ts_order_040 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F20V2ORD' 156928
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_041 of D1

drop tablespace ts_order_041;
create regular tablespace ts_order_041 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F21V1ORD' 156928
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_042 of D1

drop tablespace ts_order_042;
create regular tablespace ts_order_042 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F21V2ORD' 156928
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_043 of D1

drop tablespace ts_order_043;
create regular tablespace ts_order_043 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F22V1ORD' 156928
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_044 of D1

drop tablespace ts_order_044;
create regular tablespace ts_order_044 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F22V2ORD' 156928
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_045 of D1

drop tablespace ts_order_045;
create regular tablespace ts_order_045 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F23V1ORD' 156928
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_046 of D1

drop tablespace ts_order_046;
create regular tablespace ts_order_046 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F23V2ORD' 156928
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_047 of D1

drop tablespace ts_order_047;
create regular tablespace ts_order_047 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F24V1ORD' 156928
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_048 of D1

drop tablespace ts_order_048;
create regular tablespace ts_order_048 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F24V2ORD' 156928
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_049 of D1

drop tablespace ts_order_049;
create regular tablespace ts_order_049 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F25V1ORD' 156928
                )
```

```
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_050 of D1

drop tablespace ts_order_050;
create regular tablespace ts_order_050 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F25V2ORD' 156928
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_051 of D1

drop tablespace ts_order_051;
create regular tablespace ts_order_051 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F26V1ORD' 156928
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_052 of D1

drop tablespace ts_order_052;
create regular tablespace ts_order_052 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F26V2ORD' 156928
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_053 of D1

drop tablespace ts_order_053;
create regular tablespace ts_order_053 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F27V1ORD' 156928
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_054 of D1

drop tablespace ts_order_054;
create regular tablespace ts_order_054 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F27V2ORD' 156928
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_055 of D1

drop tablespace ts_order_055;
create regular tablespace ts_order_055 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F28V1ORD' 156928
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_056 of D1

drop tablespace ts_order_056;
create regular tablespace ts_order_056 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F28V2ORD' 156928
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_057 of D1

drop tablespace ts_order_057;
create regular tablespace ts_order_057 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F29V1ORD' 156928
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_058 of D1

drop tablespace ts_order_058;
create regular tablespace ts_order_058 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F29V2ORD' 156928
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_059 of D1

drop tablespace ts_order_059;
create regular tablespace ts_order_059 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F30V1ORD' 156928
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_060 of D1

drop tablespace ts_order_060;
create regular tablespace ts_order_060 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F30V2ORD' 156928
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_061 of D1

drop tablespace ts_order_061;
create regular tablespace ts_order_061 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F31V1ORD' 156928
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_062 of D1

drop tablespace ts_order_062;
create regular tablespace ts_order_062 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F31V2ORD' 156928
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_063 of D1

drop tablespace ts_order_063;
create regular tablespace ts_order_063 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F32V1ORD' 156928
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_064 of D1

drop tablespace ts_order_064;
create regular tablespace ts_order_064 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F32V2ORD' 156928
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

connect reset;
```

ts/crts_orderline.ddl

```
connect to tpcc;
-- now creating TS for ts_orderline_001 of D1

drop tablespace ts_orderline_001;
create regular tablespace ts_orderline_001 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F01V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_002 of D1

drop tablespace ts_orderline_002;
create regular tablespace ts_orderline_002 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F01V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_003 of D1

drop tablespace ts_orderline_003;
create regular tablespace ts_orderline_003 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F02V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_004 of D1

drop tablespace ts_orderline_004;
create regular tablespace ts_orderline_004 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F02V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_005 of D1

drop tablespace ts_orderline_005;
create regular tablespace ts_orderline_005 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F03V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_006 of D1

drop tablespace ts_orderline_006;
create regular tablespace ts_orderline_006 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F03V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_007 of D1

drop tablespace ts_orderline_007;
create regular tablespace ts_orderline_007 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F04V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_008 of D1

drop tablespace ts_orderline_008;
create regular tablespace ts_orderline_008 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F04V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_009 of D1

drop tablespace ts_orderline_009;
create regular tablespace ts_orderline_009 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F05V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_010 of D1

drop tablespace ts_orderline_010;
create regular tablespace ts_orderline_010 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F05V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_011 of D1

drop tablespace ts_orderline_011;
create regular tablespace ts_orderline_011 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F06V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_012 of D1

drop tablespace ts_orderline_012;
create regular tablespace ts_orderline_012 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F06V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_013 of D1

drop tablespace ts_orderline_013;
create regular tablespace ts_orderline_013 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F07V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_014 of D1

drop tablespace ts_orderline_014;
create regular tablespace ts_orderline_014 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F07V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_015 of D1

drop tablespace ts_orderline_015;
create regular tablespace ts_orderline_015 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F08V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_016 of D1

drop tablespace ts_orderline_016;
create regular tablespace ts_orderline_016 pagesize 8K
```

```
        managed by database
        using
        (
                device '/dev/rD1F08V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_017 of D1

drop tablespace ts_orderline_017;
create regular tablespace ts_orderline_017 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F09V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_018 of D1

drop tablespace ts_orderline_018;
create regular tablespace ts_orderline_018 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F09V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_019 of D1

drop tablespace ts_orderline_019;
create regular tablespace ts_orderline_019 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F10V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_020 of D1

drop tablespace ts_orderline_020;
create regular tablespace ts_orderline_020 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F10V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_021 of D1

drop tablespace ts_orderline_021;
create regular tablespace ts_orderline_021 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F11V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_022 of D1

drop tablespace ts_orderline_022;
create regular tablespace ts_orderline_022 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F11V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_023 of D1

drop tablespace ts_orderline_023;
create regular tablespace ts_orderline_023 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F12V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_024 of D1

drop tablespace ts_orderline_024;
create regular tablespace ts_orderline_024 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F12V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_025 of D1

drop tablespace ts_orderline_025;
create regular tablespace ts_orderline_025 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F13V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_026 of D1

drop tablespace ts_orderline_026;
create regular tablespace ts_orderline_026 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F13V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_027 of D1

drop tablespace ts_orderline_027;
create regular tablespace ts_orderline_027 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F14V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_028 of D1

drop tablespace ts_orderline_028;
create regular tablespace ts_orderline_028 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F14V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_029 of D1

drop tablespace ts_orderline_029;
create regular tablespace ts_orderline_029 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F15V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_030 of D1

drop tablespace ts_orderline_030;
create regular tablespace ts_orderline_030 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F15V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_031 of D1

drop tablespace ts_orderline_031;
create regular tablespace ts_orderline_031 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F16V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;
```

```
-- now creating TS for ts_orderline_032 of D1

drop tablespace ts_orderline_032;
create regular tablespace ts_orderline_032 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F16V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_033 of D1

drop tablespace ts_orderline_033;
create regular tablespace ts_orderline_033 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F17V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_034 of D1

drop tablespace ts_orderline_034;
create regular tablespace ts_orderline_034 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F17V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_035 of D1

drop tablespace ts_orderline_035;
create regular tablespace ts_orderline_035 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F18V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_036 of D1

drop tablespace ts_orderline_036;
create regular tablespace ts_orderline_036 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F18V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_037 of D1

drop tablespace ts_orderline_037;
create regular tablespace ts_orderline_037 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F19V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_038 of D1

drop tablespace ts_orderline_038;
create regular tablespace ts_orderline_038 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F19V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_039 of D1

drop tablespace ts_orderline_039;
create regular tablespace ts_orderline_039 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F20V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_040 of D1

drop tablespace ts_orderline_040;
create regular tablespace ts_orderline_040 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F20V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_041 of D1

drop tablespace ts_orderline_041;
create regular tablespace ts_orderline_041 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F21V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_042 of D1

drop tablespace ts_orderline_042;
create regular tablespace ts_orderline_042 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F21V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_043 of D1

drop tablespace ts_orderline_043;
create regular tablespace ts_orderline_043 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F22V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_044 of D1

drop tablespace ts_orderline_044;
create regular tablespace ts_orderline_044 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F22V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_045 of D1

drop tablespace ts_orderline_045;
create regular tablespace ts_orderline_045 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F23V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_046 of D1

drop tablespace ts_orderline_046;
create regular tablespace ts_orderline_046 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F23V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_047 of D1

drop tablespace ts_orderline_047;
create regular tablespace ts_orderline_047 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F24V1ORL' 4074752
```

```
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_048 of D1

drop tablespace ts_orderline_048;
create regular tablespace ts_orderline_048 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F24V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_049 of D1

drop tablespace ts_orderline_049;
create regular tablespace ts_orderline_049 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F25V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_050 of D1

drop tablespace ts_orderline_050;
create regular tablespace ts_orderline_050 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F25V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_051 of D1

drop tablespace ts_orderline_051;
create regular tablespace ts_orderline_051 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F26V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_052 of D1

drop tablespace ts_orderline_052;
create regular tablespace ts_orderline_052 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F26V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_053 of D1

drop tablespace ts_orderline_053;
create regular tablespace ts_orderline_053 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F27V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_054 of D1

drop tablespace ts_orderline_054;
create regular tablespace ts_orderline_054 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F27V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_055 of D1

drop tablespace ts_orderline_055;
create regular tablespace ts_orderline_055 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F28V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_056 of D1

drop tablespace ts_orderline_056;
create regular tablespace ts_orderline_056 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F28V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_057 of D1

drop tablespace ts_orderline_057;
create regular tablespace ts_orderline_057 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F29V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_058 of D1

drop tablespace ts_orderline_058;
create regular tablespace ts_orderline_058 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F29V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_059 of D1

drop tablespace ts_orderline_059;
create regular tablespace ts_orderline_059 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F30V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_060 of D1

drop tablespace ts_orderline_060;
create regular tablespace ts_orderline_060 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F30V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_061 of D1

drop tablespace ts_orderline_061;
create regular tablespace ts_orderline_061 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F31V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_062 of D1

drop tablespace ts_orderline_062;
create regular tablespace ts_orderline_062 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F31V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_063 of D1

drop tablespace ts_orderline_063;
```

```
create regular tablespace ts_orderline_063 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F32V1ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_064 of D1

drop tablespace ts_orderline_064;
create regular tablespace ts_orderline_064 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F32V2ORL' 4074752
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

connect reset;
```

_ts/crts_stock.ddl

```
connect to tpcc;
-- now creating TS for ts_stock_001 of D1

drop tablespace ts_stock_001;
create regular tablespace ts_stock_001 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F01V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_002 of D1

drop tablespace ts_stock_002;
create regular tablespace ts_stock_002 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F01V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_003 of D1

drop tablespace ts_stock_003;
create regular tablespace ts_stock_003 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F02V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_004 of D1

drop tablespace ts_stock_004;
create regular tablespace ts_stock_004 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F02V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_005 of D1

drop tablespace ts_stock_005;
create regular tablespace ts_stock_005 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F03V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_006 of D1

drop tablespace ts_stock_006;
create regular tablespace ts_stock_006 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F03V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_007 of D1

drop tablespace ts_stock_007;
create regular tablespace ts_stock_007 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F04V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_008 of D1

drop tablespace ts_stock_008;
create regular tablespace ts_stock_008 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F04V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_009 of D1

drop tablespace ts_stock_009;
create regular tablespace ts_stock_009 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F05V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_010 of D1

drop tablespace ts_stock_010;
create regular tablespace ts_stock_010 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F05V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_011 of D1

drop tablespace ts_stock_011;
create regular tablespace ts_stock_011 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F06V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_012 of D1

drop tablespace ts_stock_012;
create regular tablespace ts_stock_012 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F06V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_013 of D1

drop tablespace ts_stock_013;
create regular tablespace ts_stock_013 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F07V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_014 of D1

drop tablespace ts_stock_014;
create regular tablespace ts_stock_014 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F07V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_015 of D1
```

```
drop tablespace ts_stock_015;
create regular tablespace ts_stock_015 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F08V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_016 of D1

drop tablespace ts_stock_016;
create regular tablespace ts_stock_016 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F08V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_017 of D1

drop tablespace ts_stock_017;
create regular tablespace ts_stock_017 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F09V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_018 of D1

drop tablespace ts_stock_018;
create regular tablespace ts_stock_018 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F09V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_019 of D1

drop tablespace ts_stock_019;
create regular tablespace ts_stock_019 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F10V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_020 of D1

drop tablespace ts_stock_020;
create regular tablespace ts_stock_020 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F10V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_021 of D1

drop tablespace ts_stock_021;
create regular tablespace ts_stock_021 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F11V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_022 of D1

drop tablespace ts_stock_022;
create regular tablespace ts_stock_022 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F11V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_023 of D1

drop tablespace ts_stock_023;
create regular tablespace ts_stock_023 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F12V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_024 of D1

drop tablespace ts_stock_024;
create regular tablespace ts_stock_024 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F12V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_025 of D1

drop tablespace ts_stock_025;
create regular tablespace ts_stock_025 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F13V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_026 of D1

drop tablespace ts_stock_026;
create regular tablespace ts_stock_026 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F13V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_027 of D1

drop tablespace ts_stock_027;
create regular tablespace ts_stock_027 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F14V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_028 of D1

drop tablespace ts_stock_028;
create regular tablespace ts_stock_028 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F14V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_029 of D1

drop tablespace ts_stock_029;
create regular tablespace ts_stock_029 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F15V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_030 of D1

drop tablespace ts_stock_030;
create regular tablespace ts_stock_030 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F15V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_031 of D1

drop tablespace ts_stock_031;
create regular tablespace ts_stock_031 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F16V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;
```

```
-- now creating TS for ts_stock_032 of D1

drop tablespace ts_stock_032;
create regular tablespace ts_stock_032 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F16V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_033 of D1

drop tablespace ts_stock_033;
create regular tablespace ts_stock_033 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F17V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_034 of D1

drop tablespace ts_stock_034;
create regular tablespace ts_stock_034 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F17V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_035 of D1

drop tablespace ts_stock_035;
create regular tablespace ts_stock_035 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F18V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_036 of D1

drop tablespace ts_stock_036;
create regular tablespace ts_stock_036 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F18V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_037 of D1

drop tablespace ts_stock_037;
create regular tablespace ts_stock_037 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F19V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_038 of D1

drop tablespace ts_stock_038;
create regular tablespace ts_stock_038 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F19V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_039 of D1

drop tablespace ts_stock_039;
create regular tablespace ts_stock_039 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F20V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_040 of D1

drop tablespace ts_stock_040;
create regular tablespace ts_stock_040 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F20V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_041 of D1

drop tablespace ts_stock_041;
create regular tablespace ts_stock_041 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F21V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_042 of D1

drop tablespace ts_stock_042;
create regular tablespace ts_stock_042 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F21V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_043 of D1

drop tablespace ts_stock_043;
create regular tablespace ts_stock_043 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F22V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_044 of D1

drop tablespace ts_stock_044;
create regular tablespace ts_stock_044 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F22V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_045 of D1

drop tablespace ts_stock_045;
create regular tablespace ts_stock_045 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F23V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_046 of D1

drop tablespace ts_stock_046;
create regular tablespace ts_stock_046 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F23V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_047 of D1

drop tablespace ts_stock_047;
create regular tablespace ts_stock_047 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F24V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_048 of D1

drop tablespace ts_stock_048;
create regular tablespace ts_stock_048 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F24V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
```

```
commit;

-- now creating TS for ts_stock_049 of D1

drop tablespace ts_stock_049;
create regular tablespace ts_stock_049 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F25V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_050 of D1

drop tablespace ts_stock_050;
create regular tablespace ts_stock_050 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F25V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_051 of D1

drop tablespace ts_stock_051;
create regular tablespace ts_stock_051 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F26V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_052 of D1

drop tablespace ts_stock_052;
create regular tablespace ts_stock_052 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F26V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_053 of D1

drop tablespace ts_stock_053;
create regular tablespace ts_stock_053 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F27V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_054 of D1

drop tablespace ts_stock_054;
create regular tablespace ts_stock_054 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F27V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_055 of D1

drop tablespace ts_stock_055;
create regular tablespace ts_stock_055 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F28V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_056 of D1

drop tablespace ts_stock_056;
create regular tablespace ts_stock_056 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F28V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_057 of D1

drop tablespace ts_stock_057;
create regular tablespace ts_stock_057 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F29V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_058 of D1

drop tablespace ts_stock_058;
create regular tablespace ts_stock_058 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F29V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_059 of D1

drop tablespace ts_stock_059;
create regular tablespace ts_stock_059 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F30V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_060 of D1

drop tablespace ts_stock_060;
create regular tablespace ts_stock_060 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F30V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_061 of D1

drop tablespace ts_stock_061;
create regular tablespace ts_stock_061 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F31V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_062 of D1

drop tablespace ts_stock_062;
create regular tablespace ts_stock_062 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F31V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_063 of D1

drop tablespace ts_stock_063;
create regular tablespace ts_stock_063 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F32V1STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_064 of D1

drop tablespace ts_stock_064;
create regular tablespace ts_stock_064 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F32V2STK' 7106560
                )
        extentsize 256
        prefetchsize 4096;
commit;

connect reset;
```

_ts/crts_ware.ddl

```
connect to tpcc;
-- now creating TS for ts_ware_001 of D1

drop tablespace ts_ware_001;
create regular tablespace ts_ware_001 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F01V1WARE' 160,
                device '/dev/rD1F01V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_002 of D1

drop tablespace ts_ware_002;
create regular tablespace ts_ware_002 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F02V1WARE' 160,
                device '/dev/rD1F02V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_003 of D1

drop tablespace ts_ware_003;
create regular tablespace ts_ware_003 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F03V1WARE' 160,
                device '/dev/rD1F03V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_004 of D1

drop tablespace ts_ware_004;
create regular tablespace ts_ware_004 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F04V1WARE' 160,
                device '/dev/rD1F04V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_005 of D1

drop tablespace ts_ware_005;
create regular tablespace ts_ware_005 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F05V1WARE' 160,
                device '/dev/rD1F05V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_006 of D1

drop tablespace ts_ware_006;
create regular tablespace ts_ware_006 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F06V1WARE' 160,
                device '/dev/rD1F06V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_007 of D1

drop tablespace ts_ware_007;
create regular tablespace ts_ware_007 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F07V1WARE' 160,
                device '/dev/rD1F07V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_008 of D1

drop tablespace ts_ware_008;
create regular tablespace ts_ware_008 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F08V1WARE' 160,
                device '/dev/rD1F08V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_009 of D1

drop tablespace ts_ware_009;
create regular tablespace ts_ware_009 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F09V1WARE' 160,
                device '/dev/rD1F09V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_010 of D1

drop tablespace ts_ware_010;
create regular tablespace ts_ware_010 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F10V1WARE' 160,
                device '/dev/rD1F10V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_011 of D1

drop tablespace ts_ware_011;
create regular tablespace ts_ware_011 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F11V1WARE' 160,
                device '/dev/rD1F11V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_012 of D1

drop tablespace ts_ware_012;
create regular tablespace ts_ware_012 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F12V1WARE' 160,
                device '/dev/rD1F12V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_013 of D1

drop tablespace ts_ware_013;
create regular tablespace ts_ware_013 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F13V1WARE' 160,
                device '/dev/rD1F13V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_014 of D1

drop tablespace ts_ware_014;
create regular tablespace ts_ware_014 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F14V1WARE' 160,
                device '/dev/rD1F14V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_015 of D1

drop tablespace ts_ware_015;
create regular tablespace ts_ware_015 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F15V1WARE' 160,
                device '/dev/rD1F15V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_016 of D1

drop tablespace ts_ware_016;
create regular tablespace ts_ware_016 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F16V1WARE' 160,
```

```
                device '/dev/rD1F16V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_017 of D1

drop tablespace ts_ware_017;
create regular tablespace ts_ware_017 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F17V1WARE' 160,
                device '/dev/rD1F17V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_018 of D1

drop tablespace ts_ware_018;
create regular tablespace ts_ware_018 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F18V1WARE' 160,
                device '/dev/rD1F18V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_019 of D1

drop tablespace ts_ware_019;
create regular tablespace ts_ware_019 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F19V1WARE' 160,
                device '/dev/rD1F19V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_020 of D1

drop tablespace ts_ware_020;
create regular tablespace ts_ware_020 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F20V1WARE' 160,
                device '/dev/rD1F20V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_021 of D1

drop tablespace ts_ware_021;
create regular tablespace ts_ware_021 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F21V1WARE' 160,
                device '/dev/rD1F21V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_022 of D1

drop tablespace ts_ware_022;
create regular tablespace ts_ware_022 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F22V1WARE' 160,
                device '/dev/rD1F22V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_023 of D1

drop tablespace ts_ware_023;
create regular tablespace ts_ware_023 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F23V1WARE' 160,
                device '/dev/rD1F23V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_024 of D1

drop tablespace ts_ware_024;
create regular tablespace ts_ware_024 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F24V1WARE' 160,
                device '/dev/rD1F24V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_025 of D1

drop tablespace ts_ware_025;
create regular tablespace ts_ware_025 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F25V1WARE' 160,
                device '/dev/rD1F25V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_026 of D1

drop tablespace ts_ware_026;
create regular tablespace ts_ware_026 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F26V1WARE' 160,
                device '/dev/rD1F26V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_027 of D1

drop tablespace ts_ware_027;
create regular tablespace ts_ware_027 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F27V1WARE' 160,
                device '/dev/rD1F27V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_028 of D1

drop tablespace ts_ware_028;
create regular tablespace ts_ware_028 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F28V1WARE' 160,
                device '/dev/rD1F28V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_029 of D1

drop tablespace ts_ware_029;
create regular tablespace ts_ware_029 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F29V1WARE' 160,
                device '/dev/rD1F29V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_030 of D1

drop tablespace ts_ware_030;
create regular tablespace ts_ware_030 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F30V1WARE' 160,
                device '/dev/rD1F30V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_031 of D1

drop tablespace ts_ware_031;
create regular tablespace ts_ware_031 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F31V1WARE' 160,
                device '/dev/rD1F31V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_032 of D1

drop tablespace ts_ware_032;
```

```
create regular tablespace ts_ware_032 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F32V1WARE' 160,
                device '/dev/rD1F32V2WARE' 160
                )
        extentsize 32
        prefetchsize 4096;
commit;

connect reset;


FERPOOL WAR3 DEFERRED
```

bp/alter_bufferpool.ddl

```
connect to tpcc;
ALTER BUFFERPOOL WAR1 DEFERRED SIZE 516;
ALTER BUFFERPOOL WAR2 DEFERRED SIZE 516;
ALTER BUFFERPOOL WAR3 DEFERRED SIZE 516;
ALTER BUFFERPOOL WAR4 DEFERRED SIZE 516;
ALTER BUFFERPOOL WAR5 DEFERRED SIZE 516;
ALTER BUFFERPOOL WAR6 DEFERRED SIZE 516;
ALTER BUFFERPOOL WAR7 DEFERRED SIZE 516;
ALTER BUFFERPOOL WAR8 DEFERRED SIZE 516;
ALTER BUFFERPOOL DIS1 DEFERRED SIZE 4713;
ALTER BUFFERPOOL DIS2 DEFERRED SIZE 4713;
ALTER BUFFERPOOL DIS3 DEFERRED SIZE 4713;
ALTER BUFFERPOOL DIS4 DEFERRED SIZE 4713;
ALTER BUFFERPOOL DIS5 DEFERRED SIZE 4713;
ALTER BUFFERPOOL DIS6 DEFERRED SIZE 4713;
ALTER BUFFERPOOL DIS7 DEFERRED SIZE 4713;
ALTER BUFFERPOOL DIS8 DEFERRED SIZE 4713;
ALTER BUFFERPOOL ITM DEFERRED SIZE 1240;
ALTER BUFFERPOOL HST1 DEFERRED SIZE 875;
ALTER BUFFERPOOL HST2 DEFERRED SIZE 875;
ALTER BUFFERPOOL HST3 DEFERRED SIZE 875;
ALTER BUFFERPOOL HST4 DEFERRED SIZE 875;
ALTER BUFFERPOOL HST5 DEFERRED SIZE 875;
ALTER BUFFERPOOL HST6 DEFERRED SIZE 875;
ALTER BUFFERPOOL HST7 DEFERRED SIZE 875;
ALTER BUFFERPOOL HST8 DEFERRED SIZE 875;
ALTER BUFFERPOOL NEW1 DEFERRED SIZE 343312;
ALTER BUFFERPOOL NEW2 DEFERRED SIZE 343312;
ALTER BUFFERPOOL NEW3 DEFERRED SIZE 343312;
ALTER BUFFERPOOL NEW4 DEFERRED SIZE 343312;
ALTER BUFFERPOOL NEW5 DEFERRED SIZE 343312;
ALTER BUFFERPOOL NEW6 DEFERRED SIZE 343312;
ALTER BUFFERPOOL NEW7 DEFERRED SIZE 343312;
ALTER BUFFERPOOL NEW8 DEFERRED SIZE 343312;
ALTER BUFFERPOOL ORD1 DEFERRED SIZE 355020;
ALTER BUFFERPOOL ORD2 DEFERRED SIZE 355020;
ALTER BUFFERPOOL ORD3 DEFERRED SIZE 355020;
ALTER BUFFERPOOL ORD4 DEFERRED SIZE 355020;
ALTER BUFFERPOOL ORD5 DEFERRED SIZE 355020;
ALTER BUFFERPOOL ORD6 DEFERRED SIZE 355020;
ALTER BUFFERPOOL ORD7 DEFERRED SIZE 355020;
ALTER BUFFERPOOL ORD8 DEFERRED SIZE 355020;
ALTER BUFFERPOOL ORD_I1 DEFERRED SIZE 1062569;
ALTER BUFFERPOOL ORD_I2 DEFERRED SIZE 1062569;
ALTER BUFFERPOOL ORD_I3 DEFERRED SIZE 1062569;
ALTER BUFFERPOOL ORD_I4 DEFERRED SIZE 1062569;
ALTER BUFFERPOOL ORD_I5 DEFERRED SIZE 1062569;
ALTER BUFFERPOOL ORD_I6 DEFERRED SIZE 1062569;
ALTER BUFFERPOOL ORD_I7 DEFERRED SIZE 1062569;
ALTER BUFFERPOOL ORD_I8 DEFERRED SIZE 1062569;
ALTER BUFFERPOOL OLN1 DEFERRED SIZE 651852;
ALTER BUFFERPOOL OLN2 DEFERRED SIZE 651852;
ALTER BUFFERPOOL OLN3 DEFERRED SIZE 651852;
ALTER BUFFERPOOL OLN4 DEFERRED SIZE 651852;
ALTER BUFFERPOOL OLN5 DEFERRED SIZE 651852;
ALTER BUFFERPOOL OLN6 DEFERRED SIZE 651852;
ALTER BUFFERPOOL OLN7 DEFERRED SIZE 651852;
ALTER BUFFERPOOL OLN8 DEFERRED SIZE 651852;
ALTER BUFFERPOOL CST1 DEFERRED SIZE 127227;
ALTER BUFFERPOOL CST2 DEFERRED SIZE 127227;
ALTER BUFFERPOOL CST3 DEFERRED SIZE 305227;
#ALTER BUFFERPOOL CST3 DEFERRED SIZE 180227;
ALTER BUFFERPOOL CST4 DEFERRED SIZE 305227;
ALTER BUFFERPOOL CST5 DEFERRED SIZE 305227;
ALTER BUFFERPOOL CST6 DEFERRED SIZE 305227;
ALTER BUFFERPOOL CST7 DEFERRED SIZE 305227;
ALTER BUFFERPOOL CST8 DEFERRED SIZE 305227;
ALTER BUFFERPOOL CST_I1 DEFERRED SIZE 543511;
ALTER BUFFERPOOL CST_I2 DEFERRED SIZE 543511;
ALTER BUFFERPOOL CST_I3 DEFERRED SIZE 543511;
ALTER BUFFERPOOL CST_I4 DEFERRED SIZE 543511;
ALTER BUFFERPOOL CST_I5 DEFERRED SIZE 543511;
ALTER BUFFERPOOL CST_I6 DEFERRED SIZE 543511;
ALTER BUFFERPOOL CST_I7 DEFERRED SIZE 543511;
ALTER BUFFERPOOL CST_I8 DEFERRED SIZE 543511;
ALTER BUFFERPOOL STK1 DEFERRED SIZE 17292728;
ALTER BUFFERPOOL STK2 DEFERRED SIZE 17292728;
ALTER BUFFERPOOL STK3 DEFERRED SIZE 17467728;
ALTER BUFFERPOOL STK4 DEFERRED SIZE 17467728;
ALTER BUFFERPOOL STK5 DEFERRED SIZE 17467728;
ALTER BUFFERPOOL STK6 DEFERRED SIZE 17467728;
ALTER BUFFERPOOL STK7 DEFERRED SIZE 17467728;
ALTER BUFFERPOOL STK8 DEFERRED SIZE 17467728;
connect reset;
```

bp/alter_tablespace.ddl

```
connect to tpcc;
ALTER TABLESPACE ts_ware_001 BUFFERPOOL WAR1;
ALTER TABLESPACE ts_ware_002 BUFFERPOOL WAR1;
ALTER TABLESPACE ts_ware_003 BUFFERPOOL WAR2;
ALTER TABLESPACE ts_ware_004 BUFFERPOOL WAR2;
ALTER TABLESPACE ts_ware_005 BUFFERPOOL WAR3;
ALTER TABLESPACE ts_ware_006 BUFFERPOOL WAR3;
ALTER TABLESPACE ts_ware_007 BUFFERPOOL WAR4;
ALTER TABLESPACE ts_ware_008 BUFFERPOOL WAR4;
ALTER TABLESPACE ts_ware_009 BUFFERPOOL WAR5;
ALTER TABLESPACE ts_ware_010 BUFFERPOOL WAR5;
ALTER TABLESPACE ts_ware_011 BUFFERPOOL WAR6;
ALTER TABLESPACE ts_ware_012 BUFFERPOOL WAR6;
ALTER TABLESPACE ts_ware_013 BUFFERPOOL WAR7;
ALTER TABLESPACE ts_ware_014 BUFFERPOOL WAR7;
ALTER TABLESPACE ts_ware_015 BUFFERPOOL WAR8;
ALTER TABLESPACE ts_ware_016 BUFFERPOOL WAR8;
ALTER TABLESPACE ts_dist_001 BUFFERPOOL DIS1;
ALTER TABLESPACE ts_dist_002 BUFFERPOOL DIS1;
ALTER TABLESPACE ts_dist_003 BUFFERPOOL DIS2;
ALTER TABLESPACE ts_dist_004 BUFFERPOOL DIS2;
ALTER TABLESPACE ts_dist_005 BUFFERPOOL DIS3;
ALTER TABLESPACE ts_dist_006 BUFFERPOOL DIS3;
ALTER TABLESPACE ts_dist_007 BUFFERPOOL DIS4;
ALTER TABLESPACE ts_dist_008 BUFFERPOOL DIS4;
ALTER TABLESPACE ts_dist_009 BUFFERPOOL DIS5;
ALTER TABLESPACE ts_dist_010 BUFFERPOOL DIS5;
ALTER TABLESPACE ts_dist_011 BUFFERPOOL DIS6;
ALTER TABLESPACE ts_dist_012 BUFFERPOOL DIS6;
ALTER TABLESPACE ts_dist_013 BUFFERPOOL DIS7;
ALTER TABLESPACE ts_dist_014 BUFFERPOOL DIS7;
ALTER TABLESPACE ts_dist_015 BUFFERPOOL DIS8;
ALTER TABLESPACE ts_dist_016 BUFFERPOOL DIS8;
ALTER TABLESPACE ts_item_001 BUFFERPOOL ITM;
ALTER TABLESPACE ts_stock_001 BUFFERPOOL STK1;
ALTER TABLESPACE ts_stock_002 BUFFERPOOL STK1;
ALTER TABLESPACE ts_stock_003 BUFFERPOOL STK1;
ALTER TABLESPACE ts_stock_004 BUFFERPOOL STK1;
ALTER TABLESPACE ts_stock_005 BUFFERPOOL STK1;
ALTER TABLESPACE ts_stock_006 BUFFERPOOL STK1;
ALTER TABLESPACE ts_stock_007 BUFFERPOOL STK1;
ALTER TABLESPACE ts_stock_008 BUFFERPOOL STK1;
ALTER TABLESPACE ts_stock_009 BUFFERPOOL STK1;
ALTER TABLESPACE ts_stock_010 BUFFERPOOL STK1;
ALTER TABLESPACE ts_stock_011 BUFFERPOOL STK1;
ALTER TABLESPACE ts_stock_012 BUFFERPOOL STK1;
ALTER TABLESPACE ts_stock_013 BUFFERPOOL STK2;
ALTER TABLESPACE ts_stock_014 BUFFERPOOL STK2;
ALTER TABLESPACE ts_stock_015 BUFFERPOOL STK2;
ALTER TABLESPACE ts_stock_016 BUFFERPOOL STK2;
ALTER TABLESPACE ts_stock_017 BUFFERPOOL STK2;
ALTER TABLESPACE ts_stock_018 BUFFERPOOL STK2;
ALTER TABLESPACE ts_stock_019 BUFFERPOOL STK2;
ALTER TABLESPACE ts_stock_020 BUFFERPOOL STK2;
ALTER TABLESPACE ts_stock_021 BUFFERPOOL STK2;
ALTER TABLESPACE ts_stock_022 BUFFERPOOL STK2;
ALTER TABLESPACE ts_stock_023 BUFFERPOOL STK2;
ALTER TABLESPACE ts_stock_024 BUFFERPOOL STK2;
ALTER TABLESPACE ts_stock_025 BUFFERPOOL STK3;
ALTER TABLESPACE ts_stock_026 BUFFERPOOL STK3;
ALTER TABLESPACE ts_stock_027 BUFFERPOOL STK3;
ALTER TABLESPACE ts_stock_028 BUFFERPOOL STK3;
ALTER TABLESPACE ts_stock_029 BUFFERPOOL STK3;
ALTER TABLESPACE ts_stock_030 BUFFERPOOL STK3;
ALTER TABLESPACE ts_stock_031 BUFFERPOOL STK3;
ALTER TABLESPACE ts_stock_032 BUFFERPOOL STK3;
ALTER TABLESPACE ts_stock_033 BUFFERPOOL STK3;
ALTER TABLESPACE ts_stock_034 BUFFERPOOL STK3;
ALTER TABLESPACE ts_stock_035 BUFFERPOOL STK3;
ALTER TABLESPACE ts_stock_036 BUFFERPOOL STK3;
ALTER TABLESPACE ts_stock_037 BUFFERPOOL STK4;
ALTER TABLESPACE ts_stock_038 BUFFERPOOL STK4;
ALTER TABLESPACE ts_stock_039 BUFFERPOOL STK4;
ALTER TABLESPACE ts_stock_040 BUFFERPOOL STK4;
ALTER TABLESPACE ts_stock_041 BUFFERPOOL STK4;
ALTER TABLESPACE ts_stock_042 BUFFERPOOL STK4;
ALTER TABLESPACE ts_stock_043 BUFFERPOOL STK4;
ALTER TABLESPACE ts_stock_044 BUFFERPOOL STK4;
ALTER TABLESPACE ts_stock_045 BUFFERPOOL STK4;
ALTER TABLESPACE ts_stock_046 BUFFERPOOL STK4;
ALTER TABLESPACE ts_stock_047 BUFFERPOOL STK4;
ALTER TABLESPACE ts_stock_048 BUFFERPOOL STK4;
ALTER TABLESPACE ts_stock_049 BUFFERPOOL STK5;
ALTER TABLESPACE ts_stock_050 BUFFERPOOL STK5;
ALTER TABLESPACE ts_stock_051 BUFFERPOOL STK5;
ALTER TABLESPACE ts_stock_052 BUFFERPOOL STK5;
ALTER TABLESPACE ts_stock_053 BUFFERPOOL STK5;
ALTER TABLESPACE ts_stock_054 BUFFERPOOL STK5;
ALTER TABLESPACE ts_stock_055 BUFFERPOOL STK5;
ALTER TABLESPACE ts_stock_056 BUFFERPOOL STK5;
ALTER TABLESPACE ts_stock_057 BUFFERPOOL STK5;
ALTER TABLESPACE ts_stock_058 BUFFERPOOL STK5;
ALTER TABLESPACE ts_stock_059 BUFFERPOOL STK5;
ALTER TABLESPACE ts_stock_060 BUFFERPOOL STK5;
ALTER TABLESPACE ts_stock_061 BUFFERPOOL STK6;
ALTER TABLESPACE ts_stock_062 BUFFERPOOL STK6;
ALTER TABLESPACE ts_stock_063 BUFFERPOOL STK6;
ALTER TABLESPACE ts_stock_064 BUFFERPOOL STK6;
ALTER TABLESPACE ts_stock_065 BUFFERPOOL STK6;
ALTER TABLESPACE ts_stock_066 BUFFERPOOL STK6;
ALTER TABLESPACE ts_stock_067 BUFFERPOOL STK6;
ALTER TABLESPACE ts_stock_068 BUFFERPOOL STK6;
ALTER TABLESPACE ts_stock_069 BUFFERPOOL STK6;
ALTER TABLESPACE ts_stock_070 BUFFERPOOL STK6;
ALTER TABLESPACE ts_stock_071 BUFFERPOOL STK6;
ALTER TABLESPACE ts_stock_072 BUFFERPOOL STK6;
```

```
ALTER TABLESPACE ts_stock_073 BUFFERPOOL STK7;
ALTER TABLESPACE ts_stock_074 BUFFERPOOL STK7;
ALTER TABLESPACE ts_stock_075 BUFFERPOOL STK7;
ALTER TABLESPACE ts_stock_076 BUFFERPOOL STK7;
ALTER TABLESPACE ts_stock_077 BUFFERPOOL STK7;
ALTER TABLESPACE ts_stock_078 BUFFERPOOL STK7;
ALTER TABLESPACE ts_stock_079 BUFFERPOOL STK7;
ALTER TABLESPACE ts_stock_080 BUFFERPOOL STK7;
ALTER TABLESPACE ts_stock_081 BUFFERPOOL STK7;
ALTER TABLESPACE ts_stock_082 BUFFERPOOL STK7;
ALTER TABLESPACE ts_stock_083 BUFFERPOOL STK7;
ALTER TABLESPACE ts_stock_084 BUFFERPOOL STK7;
ALTER TABLESPACE ts_stock_085 BUFFERPOOL STK8;
ALTER TABLESPACE ts_stock_086 BUFFERPOOL STK8;
ALTER TABLESPACE ts_stock_087 BUFFERPOOL STK8;
ALTER TABLESPACE ts_stock_088 BUFFERPOOL STK8;
ALTER TABLESPACE ts_stock_089 BUFFERPOOL STK8;
ALTER TABLESPACE ts_stock_090 BUFFERPOOL STK8;
ALTER TABLESPACE ts_stock_091 BUFFERPOOL STK8;
ALTER TABLESPACE ts_stock_092 BUFFERPOOL STK8;
ALTER TABLESPACE ts_stock_093 BUFFERPOOL STK8;
ALTER TABLESPACE ts_stock_094 BUFFERPOOL STK8;
ALTER TABLESPACE ts_stock_095 BUFFERPOOL STK8;
ALTER TABLESPACE ts_stock_096 BUFFERPOOL STK8;
ALTER TABLESPACE ts_customer_001 BUFFERPOOL CST1;
ALTER TABLESPACE ts_customer_002 BUFFERPOOL CST1;
ALTER TABLESPACE ts_customer_003 BUFFERPOOL CST1;
ALTER TABLESPACE ts_customer_004 BUFFERPOOL CST1;
ALTER TABLESPACE ts_customer_005 BUFFERPOOL CST1;
ALTER TABLESPACE ts_customer_006 BUFFERPOOL CST1;
ALTER TABLESPACE ts_customer_007 BUFFERPOOL CST1;
ALTER TABLESPACE ts_customer_008 BUFFERPOOL CST1;
ALTER TABLESPACE ts_customer_009 BUFFERPOOL CST1;
ALTER TABLESPACE ts_customer_010 BUFFERPOOL CST1;
ALTER TABLESPACE ts_customer_011 BUFFERPOOL CST1;
ALTER TABLESPACE ts_customer_012 BUFFERPOOL CST1;
ALTER TABLESPACE ts_customer_013 BUFFERPOOL CST2;
ALTER TABLESPACE ts_customer_014 BUFFERPOOL CST2;
ALTER TABLESPACE ts_customer_015 BUFFERPOOL CST2;
ALTER TABLESPACE ts_customer_016 BUFFERPOOL CST2;
ALTER TABLESPACE ts_customer_017 BUFFERPOOL CST2;
ALTER TABLESPACE ts_customer_018 BUFFERPOOL CST2;
ALTER TABLESPACE ts_customer_019 BUFFERPOOL CST2;
ALTER TABLESPACE ts_customer_020 BUFFERPOOL CST2;
ALTER TABLESPACE ts_customer_021 BUFFERPOOL CST2;
ALTER TABLESPACE ts_customer_022 BUFFERPOOL CST2;
ALTER TABLESPACE ts_customer_023 BUFFERPOOL CST2;
ALTER TABLESPACE ts_customer_024 BUFFERPOOL CST2;
ALTER TABLESPACE ts_customer_025 BUFFERPOOL CST3;
ALTER TABLESPACE ts_customer_026 BUFFERPOOL CST3;
ALTER TABLESPACE ts_customer_027 BUFFERPOOL CST3;
ALTER TABLESPACE ts_customer_028 BUFFERPOOL CST3;
ALTER TABLESPACE ts_customer_029 BUFFERPOOL CST3;
ALTER TABLESPACE ts_customer_030 BUFFERPOOL CST3;
ALTER TABLESPACE ts_customer_031 BUFFERPOOL CST3;
ALTER TABLESPACE ts_customer_032 BUFFERPOOL CST3;
ALTER TABLESPACE ts_customer_033 BUFFERPOOL CST3;
ALTER TABLESPACE ts_customer_034 BUFFERPOOL CST3;
ALTER TABLESPACE ts_customer_035 BUFFERPOOL CST3;
ALTER TABLESPACE ts_customer_036 BUFFERPOOL CST3;
ALTER TABLESPACE ts_customer_037 BUFFERPOOL CST4;
ALTER TABLESPACE ts_customer_038 BUFFERPOOL CST4;
ALTER TABLESPACE ts_customer_039 BUFFERPOOL CST4;
ALTER TABLESPACE ts_customer_040 BUFFERPOOL CST4;
ALTER TABLESPACE ts_customer_041 BUFFERPOOL CST4;
ALTER TABLESPACE ts_customer_042 BUFFERPOOL CST4;
ALTER TABLESPACE ts_customer_043 BUFFERPOOL CST4;
ALTER TABLESPACE ts_customer_044 BUFFERPOOL CST4;
ALTER TABLESPACE ts_customer_045 BUFFERPOOL CST4;
ALTER TABLESPACE ts_customer_046 BUFFERPOOL CST4;
ALTER TABLESPACE ts_customer_047 BUFFERPOOL CST4;
ALTER TABLESPACE ts_customer_048 BUFFERPOOL CST4;
ALTER TABLESPACE ts_customer_049 BUFFERPOOL CST5;
ALTER TABLESPACE ts_customer_050 BUFFERPOOL CST5;
ALTER TABLESPACE ts_customer_051 BUFFERPOOL CST5;
ALTER TABLESPACE ts_customer_052 BUFFERPOOL CST5;
ALTER TABLESPACE ts_customer_053 BUFFERPOOL CST5;
ALTER TABLESPACE ts_customer_054 BUFFERPOOL CST5;
ALTER TABLESPACE ts_customer_055 BUFFERPOOL CST5;
ALTER TABLESPACE ts_customer_056 BUFFERPOOL CST5;
ALTER TABLESPACE ts_customer_057 BUFFERPOOL CST5;
ALTER TABLESPACE ts_customer_058 BUFFERPOOL CST5;
ALTER TABLESPACE ts_customer_059 BUFFERPOOL CST5;
ALTER TABLESPACE ts_customer_060 BUFFERPOOL CST5;
ALTER TABLESPACE ts_customer_061 BUFFERPOOL CST6;
ALTER TABLESPACE ts_customer_062 BUFFERPOOL CST6;
ALTER TABLESPACE ts_customer_063 BUFFERPOOL CST6;
ALTER TABLESPACE ts_customer_064 BUFFERPOOL CST6;
ALTER TABLESPACE ts_customer_065 BUFFERPOOL CST6;
ALTER TABLESPACE ts_customer_066 BUFFERPOOL CST6;
ALTER TABLESPACE ts_customer_067 BUFFERPOOL CST6;
ALTER TABLESPACE ts_customer_068 BUFFERPOOL CST6;
ALTER TABLESPACE ts_customer_069 BUFFERPOOL CST6;
ALTER TABLESPACE ts_customer_070 BUFFERPOOL CST6;
ALTER TABLESPACE ts_customer_071 BUFFERPOOL CST6;
ALTER TABLESPACE ts_customer_072 BUFFERPOOL CST6;
ALTER TABLESPACE ts_customer_073 BUFFERPOOL CST7;
ALTER TABLESPACE ts_customer_074 BUFFERPOOL CST7;
ALTER TABLESPACE ts_customer_075 BUFFERPOOL CST7;
ALTER TABLESPACE ts_customer_076 BUFFERPOOL CST7;
ALTER TABLESPACE ts_customer_077 BUFFERPOOL CST7;
ALTER TABLESPACE ts_customer_078 BUFFERPOOL CST7;
ALTER TABLESPACE ts_customer_079 BUFFERPOOL CST7;
ALTER TABLESPACE ts_customer_080 BUFFERPOOL CST7;
ALTER TABLESPACE ts_customer_081 BUFFERPOOL CST7;
ALTER TABLESPACE ts_customer_082 BUFFERPOOL CST7;
ALTER TABLESPACE ts_customer_083 BUFFERPOOL CST7;
ALTER TABLESPACE ts_customer_084 BUFFERPOOL CST7;
ALTER TABLESPACE ts_customer_085 BUFFERPOOL CST8;
ALTER TABLESPACE ts_customer_086 BUFFERPOOL CST8;
ALTER TABLESPACE ts_customer_087 BUFFERPOOL CST8;
ALTER TABLESPACE ts_customer_088 BUFFERPOOL CST8;
ALTER TABLESPACE ts_customer_089 BUFFERPOOL CST8;
ALTER TABLESPACE ts_customer_090 BUFFERPOOL CST8;
ALTER TABLESPACE ts_customer_091 BUFFERPOOL CST8;
ALTER TABLESPACE ts_customer_092 BUFFERPOOL CST8;
ALTER TABLESPACE ts_customer_093 BUFFERPOOL CST8;
ALTER TABLESPACE ts_customer_094 BUFFERPOOL CST8;
ALTER TABLESPACE ts_customer_095 BUFFERPOOL CST8;
ALTER TABLESPACE ts_customer_096 BUFFERPOOL CST8;
ALTER TABLESPACE is_customer_001 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_customer_002 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_customer_003 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_customer_004 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_customer_005 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_customer_006 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_customer_007 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_customer_008 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_customer_009 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_customer_010 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_customer_011 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_customer_012 BUFFERPOOL CST_I1;
ALTER TABLESPACE is_customer_013 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_customer_014 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_customer_015 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_customer_016 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_customer_017 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_customer_018 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_customer_019 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_customer_020 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_customer_021 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_customer_022 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_customer_023 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_customer_024 BUFFERPOOL CST_I2;
ALTER TABLESPACE is_customer_025 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_customer_026 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_customer_027 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_customer_028 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_customer_029 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_customer_030 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_customer_031 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_customer_032 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_customer_033 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_customer_034 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_customer_035 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_customer_036 BUFFERPOOL CST_I3;
ALTER TABLESPACE is_customer_037 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_customer_038 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_customer_039 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_customer_040 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_customer_041 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_customer_042 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_customer_043 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_customer_044 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_customer_045 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_customer_046 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_customer_047 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_customer_048 BUFFERPOOL CST_I4;
ALTER TABLESPACE is_customer_049 BUFFERPOOL CST_I5;
ALTER TABLESPACE is_customer_050 BUFFERPOOL CST_I5;
ALTER TABLESPACE is_customer_051 BUFFERPOOL CST_I5;
ALTER TABLESPACE is_customer_052 BUFFERPOOL CST_I5;
ALTER TABLESPACE is_customer_053 BUFFERPOOL CST_I5;
ALTER TABLESPACE is_customer_054 BUFFERPOOL CST_I5;
ALTER TABLESPACE is_customer_055 BUFFERPOOL CST_I5;
ALTER TABLESPACE is_customer_056 BUFFERPOOL CST_I5;
ALTER TABLESPACE is_customer_057 BUFFERPOOL CST_I5;
ALTER TABLESPACE is_customer_058 BUFFERPOOL CST_I5;
ALTER TABLESPACE is_customer_059 BUFFERPOOL CST_I5;
ALTER TABLESPACE is_customer_060 BUFFERPOOL CST_I5;
ALTER TABLESPACE is_customer_061 BUFFERPOOL CST_I6;
ALTER TABLESPACE is_customer_062 BUFFERPOOL CST_I6;
ALTER TABLESPACE is_customer_063 BUFFERPOOL CST_I6;
ALTER TABLESPACE is_customer_064 BUFFERPOOL CST_I6;
ALTER TABLESPACE is_customer_065 BUFFERPOOL CST_I6;
ALTER TABLESPACE is_customer_066 BUFFERPOOL CST_I6;
ALTER TABLESPACE is_customer_067 BUFFERPOOL CST_I6;
ALTER TABLESPACE is_customer_068 BUFFERPOOL CST_I6;
ALTER TABLESPACE is_customer_069 BUFFERPOOL CST_I6;
ALTER TABLESPACE is_customer_070 BUFFERPOOL CST_I6;
ALTER TABLESPACE is_customer_071 BUFFERPOOL CST_I6;
ALTER TABLESPACE is_customer_072 BUFFERPOOL CST_I6;
ALTER TABLESPACE is_customer_073 BUFFERPOOL CST_I7;
ALTER TABLESPACE is_customer_074 BUFFERPOOL CST_I7;
ALTER TABLESPACE is_customer_075 BUFFERPOOL CST_I7;
ALTER TABLESPACE is_customer_076 BUFFERPOOL CST_I7;
ALTER TABLESPACE is_customer_077 BUFFERPOOL CST_I7;
ALTER TABLESPACE is_customer_078 BUFFERPOOL CST_I7;
ALTER TABLESPACE is_customer_079 BUFFERPOOL CST_I7;
ALTER TABLESPACE is_customer_080 BUFFERPOOL CST_I7;
ALTER TABLESPACE is_customer_081 BUFFERPOOL CST_I7;
ALTER TABLESPACE is_customer_082 BUFFERPOOL CST_I7;
ALTER TABLESPACE is_customer_083 BUFFERPOOL CST_I7;
ALTER TABLESPACE is_customer_084 BUFFERPOOL CST_I7;
ALTER TABLESPACE is_customer_085 BUFFERPOOL CST_I8;
ALTER TABLESPACE is_customer_086 BUFFERPOOL CST_I8;
ALTER TABLESPACE is_customer_087 BUFFERPOOL CST_I8;
ALTER TABLESPACE is_customer_088 BUFFERPOOL CST_I8;
ALTER TABLESPACE is_customer_089 BUFFERPOOL CST_I8;
ALTER TABLESPACE is_customer_090 BUFFERPOOL CST_I8;
ALTER TABLESPACE is_customer_091 BUFFERPOOL CST_I8;
ALTER TABLESPACE is_customer_092 BUFFERPOOL CST_I8;
ALTER TABLESPACE is_customer_093 BUFFERPOOL CST_I8;
ALTER TABLESPACE is_customer_094 BUFFERPOOL CST_I8;
ALTER TABLESPACE is_customer_095 BUFFERPOOL CST_I8;
ALTER TABLESPACE is_customer_096 BUFFERPOOL CST_I8;
ALTER TABLESPACE ts_history_001 BUFFERPOOL HST1;
ALTER TABLESPACE ts_history_002 BUFFERPOOL HST1;
ALTER TABLESPACE ts_history_003 BUFFERPOOL HST2;
```

```
ALTER TABLESPACE ts_history_004 BUFFERPOOL HST2;
ALTER TABLESPACE ts_history_005 BUFFERPOOL HST3;
ALTER TABLESPACE ts_history_006 BUFFERPOOL HST3;
ALTER TABLESPACE ts_history_007 BUFFERPOOL HST4;
ALTER TABLESPACE ts_history_008 BUFFERPOOL HST4;
ALTER TABLESPACE ts_history_009 BUFFERPOOL HST5;
ALTER TABLESPACE ts_history_010 BUFFERPOOL HST5;
ALTER TABLESPACE ts_history_011 BUFFERPOOL HST6;
ALTER TABLESPACE ts_history_012 BUFFERPOOL HST6;
ALTER TABLESPACE ts_history_013 BUFFERPOOL HST7;
ALTER TABLESPACE ts_history_014 BUFFERPOOL HST7;
ALTER TABLESPACE ts_history_015 BUFFERPOOL HST8;
ALTER TABLESPACE ts_history_016 BUFFERPOOL HST8;
ALTER TABLESPACE ts_order_001 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_order_002 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_order_003 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_order_004 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_order_005 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_order_006 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_order_007 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_order_008 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_order_009 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_order_010 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_order_011 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_order_012 BUFFERPOOL ORD1;
ALTER TABLESPACE ts_order_013 BUFFERPOOL ORD2;
ALTER TABLESPACE ts_order_014 BUFFERPOOL ORD2;
ALTER TABLESPACE ts_order_015 BUFFERPOOL ORD2;
ALTER TABLESPACE ts_order_016 BUFFERPOOL ORD2;
ALTER TABLESPACE ts_order_017 BUFFERPOOL ORD2;
ALTER TABLESPACE ts_order_018 BUFFERPOOL ORD2;
ALTER TABLESPACE ts_order_019 BUFFERPOOL ORD2;
ALTER TABLESPACE ts_order_020 BUFFERPOOL ORD2;
ALTER TABLESPACE ts_order_021 BUFFERPOOL ORD2;
ALTER TABLESPACE ts_order_022 BUFFERPOOL ORD2;
ALTER TABLESPACE ts_order_023 BUFFERPOOL ORD2;
ALTER TABLESPACE ts_order_024 BUFFERPOOL ORD2;
ALTER TABLESPACE ts_order_025 BUFFERPOOL ORD3;
ALTER TABLESPACE ts_order_026 BUFFERPOOL ORD3;
ALTER TABLESPACE ts_order_027 BUFFERPOOL ORD3;
ALTER TABLESPACE ts_order_028 BUFFERPOOL ORD3;
ALTER TABLESPACE ts_order_029 BUFFERPOOL ORD3;
ALTER TABLESPACE ts_order_030 BUFFERPOOL ORD3;
ALTER TABLESPACE ts_order_031 BUFFERPOOL ORD3;
ALTER TABLESPACE ts_order_032 BUFFERPOOL ORD3;
ALTER TABLESPACE ts_order_033 BUFFERPOOL ORD3;
ALTER TABLESPACE ts_order_034 BUFFERPOOL ORD3;
ALTER TABLESPACE ts_order_035 BUFFERPOOL ORD3;
ALTER TABLESPACE ts_order_036 BUFFERPOOL ORD3;
ALTER TABLESPACE ts_order_037 BUFFERPOOL ORD4;
ALTER TABLESPACE ts_order_038 BUFFERPOOL ORD4;
ALTER TABLESPACE ts_order_039 BUFFERPOOL ORD4;
ALTER TABLESPACE ts_order_040 BUFFERPOOL ORD4;
ALTER TABLESPACE ts_order_041 BUFFERPOOL ORD4;
ALTER TABLESPACE ts_order_042 BUFFERPOOL ORD4;
ALTER TABLESPACE ts_order_043 BUFFERPOOL ORD4;
ALTER TABLESPACE ts_order_044 BUFFERPOOL ORD4;
ALTER TABLESPACE ts_order_045 BUFFERPOOL ORD4;
ALTER TABLESPACE ts_order_046 BUFFERPOOL ORD4;
ALTER TABLESPACE ts_order_047 BUFFERPOOL ORD4;
ALTER TABLESPACE ts_order_048 BUFFERPOOL ORD4;
ALTER TABLESPACE ts_order_049 BUFFERPOOL ORD5;
ALTER TABLESPACE ts_order_050 BUFFERPOOL ORD5;
ALTER TABLESPACE ts_order_051 BUFFERPOOL ORD5;
ALTER TABLESPACE ts_order_052 BUFFERPOOL ORD5;
ALTER TABLESPACE ts_order_053 BUFFERPOOL ORD5;
ALTER TABLESPACE ts_order_054 BUFFERPOOL ORD5;
ALTER TABLESPACE ts_order_055 BUFFERPOOL ORD5;
ALTER TABLESPACE ts_order_056 BUFFERPOOL ORD5;
ALTER TABLESPACE ts_order_057 BUFFERPOOL ORD5;
ALTER TABLESPACE ts_order_058 BUFFERPOOL ORD5;
ALTER TABLESPACE ts_order_059 BUFFERPOOL ORD5;
ALTER TABLESPACE ts_order_060 BUFFERPOOL ORD5;
ALTER TABLESPACE ts_order_061 BUFFERPOOL ORD6;
ALTER TABLESPACE ts_order_062 BUFFERPOOL ORD6;
ALTER TABLESPACE ts_order_063 BUFFERPOOL ORD6;
ALTER TABLESPACE ts_order_064 BUFFERPOOL ORD6;
ALTER TABLESPACE ts_order_065 BUFFERPOOL ORD6;
ALTER TABLESPACE ts_order_066 BUFFERPOOL ORD6;
ALTER TABLESPACE ts_order_067 BUFFERPOOL ORD6;
ALTER TABLESPACE ts_order_068 BUFFERPOOL ORD6;
ALTER TABLESPACE ts_order_069 BUFFERPOOL ORD6;
ALTER TABLESPACE ts_order_070 BUFFERPOOL ORD6;
ALTER TABLESPACE ts_order_071 BUFFERPOOL ORD6;
ALTER TABLESPACE ts_order_072 BUFFERPOOL ORD6;
ALTER TABLESPACE ts_order_073 BUFFERPOOL ORD7;
ALTER TABLESPACE ts_order_074 BUFFERPOOL ORD7;
ALTER TABLESPACE ts_order_075 BUFFERPOOL ORD7;
ALTER TABLESPACE ts_order_076 BUFFERPOOL ORD7;
ALTER TABLESPACE ts_order_077 BUFFERPOOL ORD7;
ALTER TABLESPACE ts_order_078 BUFFERPOOL ORD7;
ALTER TABLESPACE ts_order_079 BUFFERPOOL ORD7;
ALTER TABLESPACE ts_order_080 BUFFERPOOL ORD7;
ALTER TABLESPACE ts_order_081 BUFFERPOOL ORD7;
ALTER TABLESPACE ts_order_082 BUFFERPOOL ORD7;
ALTER TABLESPACE ts_order_083 BUFFERPOOL ORD7;
ALTER TABLESPACE ts_order_084 BUFFERPOOL ORD7;
ALTER TABLESPACE ts_order_085 BUFFERPOOL ORD8;
ALTER TABLESPACE ts_order_086 BUFFERPOOL ORD8;
ALTER TABLESPACE ts_order_087 BUFFERPOOL ORD8;
ALTER TABLESPACE ts_order_088 BUFFERPOOL ORD8;
ALTER TABLESPACE ts_order_089 BUFFERPOOL ORD8;
ALTER TABLESPACE ts_order_090 BUFFERPOOL ORD8;
ALTER TABLESPACE ts_order_091 BUFFERPOOL ORD8;
ALTER TABLESPACE ts_order_092 BUFFERPOOL ORD8;
ALTER TABLESPACE ts_order_093 BUFFERPOOL ORD8;
ALTER TABLESPACE ts_order_094 BUFFERPOOL ORD8;
ALTER TABLESPACE ts_order_095 BUFFERPOOL ORD8;
ALTER TABLESPACE ts_order_096 BUFFERPOOL ORD8;
ALTER TABLESPACE is_order_001 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_order_002 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_order_003 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_order_004 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_order_005 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_order_006 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_order_007 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_order_008 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_order_009 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_order_010 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_order_011 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_order_012 BUFFERPOOL ORD_I1;
ALTER TABLESPACE is_order_013 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_order_014 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_order_015 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_order_016 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_order_017 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_order_018 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_order_019 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_order_020 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_order_021 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_order_022 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_order_023 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_order_024 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_order_025 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_order_026 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_order_027 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_order_028 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_order_029 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_order_030 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_order_031 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_order_032 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_order_033 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_order_034 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_order_035 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_order_036 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_order_037 BUFFERPOOL ORD_I4;
ALTER TABLESPACE is_order_038 BUFFERPOOL ORD_I4;
ALTER TABLESPACE is_order_039 BUFFERPOOL ORD_I4;
ALTER TABLESPACE is_order_040 BUFFERPOOL ORD_I4;
ALTER TABLESPACE is_order_041 BUFFERPOOL ORD_I4;
ALTER TABLESPACE is_order_042 BUFFERPOOL ORD_I4;
ALTER TABLESPACE is_order_043 BUFFERPOOL ORD_I4;
ALTER TABLESPACE is_order_044 BUFFERPOOL ORD_I4;
ALTER TABLESPACE is_order_045 BUFFERPOOL ORD_I4;
ALTER TABLESPACE is_order_046 BUFFERPOOL ORD_I4;
ALTER TABLESPACE is_order_047 BUFFERPOOL ORD_I4;
ALTER TABLESPACE is_order_048 BUFFERPOOL ORD_I4;
ALTER TABLESPACE is_order_049 BUFFERPOOL ORD_I5;
ALTER TABLESPACE is_order_050 BUFFERPOOL ORD_I5;
ALTER TABLESPACE is_order_051 BUFFERPOOL ORD_I5;
ALTER TABLESPACE is_order_052 BUFFERPOOL ORD_I5;
ALTER TABLESPACE is_order_053 BUFFERPOOL ORD_I5;
ALTER TABLESPACE is_order_054 BUFFERPOOL ORD_I5;
ALTER TABLESPACE is_order_055 BUFFERPOOL ORD_I5;
ALTER TABLESPACE is_order_056 BUFFERPOOL ORD_I5;
ALTER TABLESPACE is_order_057 BUFFERPOOL ORD_I5;
ALTER TABLESPACE is_order_058 BUFFERPOOL ORD_I5;
ALTER TABLESPACE is_order_059 BUFFERPOOL ORD_I5;
ALTER TABLESPACE is_order_060 BUFFERPOOL ORD_I5;
ALTER TABLESPACE is_order_061 BUFFERPOOL ORD_I6;
ALTER TABLESPACE is_order_062 BUFFERPOOL ORD_I6;
ALTER TABLESPACE is_order_063 BUFFERPOOL ORD_I6;
ALTER TABLESPACE is_order_064 BUFFERPOOL ORD_I6;
ALTER TABLESPACE is_order_065 BUFFERPOOL ORD_I6;
ALTER TABLESPACE is_order_066 BUFFERPOOL ORD_I6;
ALTER TABLESPACE is_order_067 BUFFERPOOL ORD_I6;
ALTER TABLESPACE is_order_068 BUFFERPOOL ORD_I6;
ALTER TABLESPACE is_order_069 BUFFERPOOL ORD_I6;
ALTER TABLESPACE is_order_070 BUFFERPOOL ORD_I6;
ALTER TABLESPACE is_order_071 BUFFERPOOL ORD_I6;
ALTER TABLESPACE is_order_072 BUFFERPOOL ORD_I6;
ALTER TABLESPACE is_order_073 BUFFERPOOL ORD_I7;
ALTER TABLESPACE is_order_074 BUFFERPOOL ORD_I7;
ALTER TABLESPACE is_order_075 BUFFERPOOL ORD_I7;
ALTER TABLESPACE is_order_076 BUFFERPOOL ORD_I7;
ALTER TABLESPACE is_order_077 BUFFERPOOL ORD_I7;
ALTER TABLESPACE is_order_078 BUFFERPOOL ORD_I7;
ALTER TABLESPACE is_order_079 BUFFERPOOL ORD_I7;
ALTER TABLESPACE is_order_080 BUFFERPOOL ORD_I7;
ALTER TABLESPACE is_order_081 BUFFERPOOL ORD_I7;
ALTER TABLESPACE is_order_082 BUFFERPOOL ORD_I7;
ALTER TABLESPACE is_order_083 BUFFERPOOL ORD_I7;
ALTER TABLESPACE is_order_084 BUFFERPOOL ORD_I7;
ALTER TABLESPACE is_order_085 BUFFERPOOL ORD_I8;
ALTER TABLESPACE is_order_086 BUFFERPOOL ORD_I8;
ALTER TABLESPACE is_order_087 BUFFERPOOL ORD_I8;
ALTER TABLESPACE is_order_088 BUFFERPOOL ORD_I8;
ALTER TABLESPACE is_order_089 BUFFERPOOL ORD_I8;
ALTER TABLESPACE is_order_090 BUFFERPOOL ORD_I8;
ALTER TABLESPACE is_order_091 BUFFERPOOL ORD_I8;
ALTER TABLESPACE is_order_092 BUFFERPOOL ORD_I8;
ALTER TABLESPACE is_order_093 BUFFERPOOL ORD_I8;
ALTER TABLESPACE is_order_094 BUFFERPOOL ORD_I8;
ALTER TABLESPACE is_order_095 BUFFERPOOL ORD_I8;
ALTER TABLESPACE is_order_096 BUFFERPOOL ORD_I8;
ALTER TABLESPACE ts_orderline_001 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_orderline_002 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_orderline_003 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_orderline_004 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_orderline_005 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_orderline_006 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_orderline_007 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_orderline_008 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_orderline_009 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_orderline_010 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_orderline_011 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_orderline_012 BUFFERPOOL OLN1;
ALTER TABLESPACE ts_orderline_013 BUFFERPOOL OLN2;
ALTER TABLESPACE ts_orderline_014 BUFFERPOOL OLN2;
```

```
ALTER TABLESPACE ts_orderline_015 BUFFERPOOL OLN2;
ALTER TABLESPACE ts_orderline_016 BUFFERPOOL OLN2;
ALTER TABLESPACE ts_orderline_017 BUFFERPOOL OLN2;
ALTER TABLESPACE ts_orderline_018 BUFFERPOOL OLN2;
ALTER TABLESPACE ts_orderline_019 BUFFERPOOL OLN2;
ALTER TABLESPACE ts_orderline_020 BUFFERPOOL OLN2;
ALTER TABLESPACE ts_orderline_021 BUFFERPOOL OLN2;
ALTER TABLESPACE ts_orderline_022 BUFFERPOOL OLN2;
ALTER TABLESPACE ts_orderline_023 BUFFERPOOL OLN2;
ALTER TABLESPACE ts_orderline_024 BUFFERPOOL OLN2;
ALTER TABLESPACE ts_orderline_025 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_orderline_026 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_orderline_027 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_orderline_028 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_orderline_029 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_orderline_030 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_orderline_031 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_orderline_032 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_orderline_033 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_orderline_034 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_orderline_035 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_orderline_036 BUFFERPOOL OLN3;
ALTER TABLESPACE ts_orderline_037 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_orderline_038 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_orderline_039 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_orderline_040 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_orderline_041 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_orderline_042 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_orderline_043 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_orderline_044 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_orderline_045 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_orderline_046 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_orderline_047 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_orderline_048 BUFFERPOOL OLN4;
ALTER TABLESPACE ts_orderline_049 BUFFERPOOL OLN5;
ALTER TABLESPACE ts_orderline_050 BUFFERPOOL OLN5;
ALTER TABLESPACE ts_orderline_051 BUFFERPOOL OLN5;
ALTER TABLESPACE ts_orderline_052 BUFFERPOOL OLN5;
ALTER TABLESPACE ts_orderline_053 BUFFERPOOL OLN5;
ALTER TABLESPACE ts_orderline_054 BUFFERPOOL OLN5;
ALTER TABLESPACE ts_orderline_055 BUFFERPOOL OLN5;
ALTER TABLESPACE ts_orderline_056 BUFFERPOOL OLN5;
ALTER TABLESPACE ts_orderline_057 BUFFERPOOL OLN5;
ALTER TABLESPACE ts_orderline_058 BUFFERPOOL OLN5;
ALTER TABLESPACE ts_orderline_059 BUFFERPOOL OLN5;
ALTER TABLESPACE ts_orderline_060 BUFFERPOOL OLN5;
ALTER TABLESPACE ts_orderline_061 BUFFERPOOL OLN6;
ALTER TABLESPACE ts_orderline_062 BUFFERPOOL OLN6;
ALTER TABLESPACE ts_orderline_063 BUFFERPOOL OLN6;
ALTER TABLESPACE ts_orderline_064 BUFFERPOOL OLN6;
ALTER TABLESPACE ts_orderline_065 BUFFERPOOL OLN6;
ALTER TABLESPACE ts_orderline_066 BUFFERPOOL OLN6;
ALTER TABLESPACE ts_orderline_067 BUFFERPOOL OLN6;
ALTER TABLESPACE ts_orderline_068 BUFFERPOOL OLN6;
ALTER TABLESPACE ts_orderline_069 BUFFERPOOL OLN6;
ALTER TABLESPACE ts_orderline_070 BUFFERPOOL OLN6;
ALTER TABLESPACE ts_orderline_071 BUFFERPOOL OLN6;
ALTER TABLESPACE ts_orderline_072 BUFFERPOOL OLN6;
ALTER TABLESPACE ts_orderline_073 BUFFERPOOL OLN7;
ALTER TABLESPACE ts_orderline_074 BUFFERPOOL OLN7;
ALTER TABLESPACE ts_orderline_075 BUFFERPOOL OLN7;
ALTER TABLESPACE ts_orderline_076 BUFFERPOOL OLN7;
ALTER TABLESPACE ts_orderline_077 BUFFERPOOL OLN7;
ALTER TABLESPACE ts_orderline_078 BUFFERPOOL OLN7;
ALTER TABLESPACE ts_orderline_079 BUFFERPOOL OLN7;
ALTER TABLESPACE ts_orderline_080 BUFFERPOOL OLN7;
ALTER TABLESPACE ts_orderline_081 BUFFERPOOL OLN7;
ALTER TABLESPACE ts_orderline_082 BUFFERPOOL OLN7;
ALTER TABLESPACE ts_orderline_083 BUFFERPOOL OLN7;
ALTER TABLESPACE ts_orderline_084 BUFFERPOOL OLN7;
ALTER TABLESPACE ts_orderline_085 BUFFERPOOL OLN8;
ALTER TABLESPACE ts_orderline_086 BUFFERPOOL OLN8;
ALTER TABLESPACE ts_orderline_087 BUFFERPOOL OLN8;
ALTER TABLESPACE ts_orderline_088 BUFFERPOOL OLN8;
ALTER TABLESPACE ts_orderline_089 BUFFERPOOL OLN8;
ALTER TABLESPACE ts_orderline_090 BUFFERPOOL OLN8;
ALTER TABLESPACE ts_orderline_091 BUFFERPOOL OLN8;
ALTER TABLESPACE ts_orderline_092 BUFFERPOOL OLN8;
ALTER TABLESPACE ts_orderline_093 BUFFERPOOL OLN8;
ALTER TABLESPACE ts_orderline_094 BUFFERPOOL OLN8;
ALTER TABLESPACE ts_orderline_095 BUFFERPOOL OLN8;
ALTER TABLESPACE ts_orderline_096 BUFFERPOOL OLN8;
ALTER TABLESPACE ts_newordA_001 BUFFERPOOL NEW1;
ALTER TABLESPACE ts_newordA_002 BUFFERPOOL NEW1;
ALTER TABLESPACE ts_newordA_003 BUFFERPOOL NEW2;
ALTER TABLESPACE ts_newordA_004 BUFFERPOOL NEW2;
ALTER TABLESPACE ts_newordA_005 BUFFERPOOL NEW3;
ALTER TABLESPACE ts_newordA_006 BUFFERPOOL NEW3;
ALTER TABLESPACE ts_newordA_007 BUFFERPOOL NEW4;
ALTER TABLESPACE ts_newordA_008 BUFFERPOOL NEW4;
ALTER TABLESPACE ts_newordA_009 BUFFERPOOL NEW5;
ALTER TABLESPACE ts_newordA_010 BUFFERPOOL NEW5;
ALTER TABLESPACE ts_newordA_011 BUFFERPOOL NEW6;
ALTER TABLESPACE ts_newordA_012 BUFFERPOOL NEW6;
ALTER TABLESPACE ts_newordA_013 BUFFERPOOL NEW7;
ALTER TABLESPACE ts_newordA_014 BUFFERPOOL NEW7;
ALTER TABLESPACE ts_newordA_015 BUFFERPOOL NEW8;
ALTER TABLESPACE ts_newordA_016 BUFFERPOOL NEW8;
ALTER TABLESPACE ts_newordB_001 BUFFERPOOL NEW1;
ALTER TABLESPACE ts_newordB_002 BUFFERPOOL NEW1;
ALTER TABLESPACE ts_newordB_003 BUFFERPOOL NEW2;
ALTER TABLESPACE ts_newordB_004 BUFFERPOOL NEW2;
ALTER TABLESPACE ts_newordB_005 BUFFERPOOL NEW3;
ALTER TABLESPACE ts_newordB_006 BUFFERPOOL NEW3;
ALTER TABLESPACE ts_newordB_007 BUFFERPOOL NEW4;
ALTER TABLESPACE ts_newordB_008 BUFFERPOOL NEW4;
ALTER TABLESPACE ts_newordB_009 BUFFERPOOL NEW5;
ALTER TABLESPACE ts_newordB_010 BUFFERPOOL NEW5;
ALTER TABLESPACE ts_newordB_011 BUFFERPOOL NEW6;
ALTER TABLESPACE ts_newordB_012 BUFFERPOOL NEW6;
ALTER TABLESPACE ts_newordB_013 BUFFERPOOL NEW7;
ALTER TABLESPACE ts_newordB_014 BUFFERPOOL NEW7;
ALTER TABLESPACE ts_newordB_015 BUFFERPOOL NEW8;
ALTER TABLESPACE ts_newordB_016 BUFFERPOOL NEW8;
connect reset;
```

bp/create_bufferpool.ddl

```
connect to tpcc;
CREATE BUFFERPOOL WAR1 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL WAR2 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL WAR3 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL WAR4 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL WAR5 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL WAR6 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL WAR7 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL WAR8 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL DIS1 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL DIS2 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL DIS3 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL DIS4 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL DIS5 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL DIS6 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL DIS7 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL DIS8 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL ITM SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL HST1 SIZE 100 PAGESIZE 16384;
CREATE BUFFERPOOL HST2 SIZE 100 PAGESIZE 16384;
CREATE BUFFERPOOL HST3 SIZE 100 PAGESIZE 16384;
CREATE BUFFERPOOL HST4 SIZE 100 PAGESIZE 16384;
CREATE BUFFERPOOL HST5 SIZE 100 PAGESIZE 16384;
CREATE BUFFERPOOL HST6 SIZE 100 PAGESIZE 16384;
CREATE BUFFERPOOL HST7 SIZE 100 PAGESIZE 16384;
CREATE BUFFERPOOL HST8 SIZE 100 PAGESIZE 16384;
CREATE BUFFERPOOL NEW1 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL NEW2 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL NEW3 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL NEW4 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL NEW5 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL NEW6 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL NEW7 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL NEW8 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL ORD1 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD2 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD3 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD4 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD5 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD6 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD7 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD8 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD_I1 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD_I2 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD_I3 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD_I4 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD_I5 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD_I6 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD_I7 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL ORD_I8 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL OLN1 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL OLN2 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL OLN3 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL OLN4 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL OLN5 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL OLN6 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL OLN7 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL OLN8 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL CST1 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL CST2 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL CST3 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL CST4 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL CST5 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL CST6 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL CST7 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL CST8 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL CST_I1 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL CST_I2 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL CST_I3 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL CST_I4 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL CST_I5 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL CST_I6 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL CST_I7 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL CST_I8 SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL STK1 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL STK2 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL STK3 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL STK4 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL STK5 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL STK6 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL STK7 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL STK8 SIZE 100 PAGESIZE 4096;
CREATE BUFFERPOOL IBMDEFAULT8K SIZE 100 PAGESIZE 8192;
CREATE BUFFERPOOL IBMDEFAULT16K SIZE 100 PAGESIZE 16384;
connect reset;
```

db/create_database.ddl

```
-----------------------------------------------------------------------------
-- Licensed Materials - Property of IBM
--
-- Governed under the terms of the International
-- License Agreement for Non-Warranted Sample Code.
--
-- (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
-- All Rights Reserved.
--
```

```
-- US Government Users Restricted Rights - Use, duplication or
-- disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
-----------------------------------------------------------------------------
drop database tpcc;
create database tpcc on /home/tpcc/db/tpccdb1 collate using identity
catalog tablespace
managed by system using ('/home/tpcc/db/db1catalog');
```

ts/cris_customer.ddl

```
connect to tpcc;
-- now creating TS for is_customer_001 of D1

drop tablespace is_customer_001;
create regular tablespace is_customer_001 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F01V1CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_002 of D1

drop tablespace is_customer_002;
create regular tablespace is_customer_002 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F01V2CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_003 of D1

drop tablespace is_customer_003;
create regular tablespace is_customer_003 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F01V3CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_004 of D1

drop tablespace is_customer_004;
create regular tablespace is_customer_004 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F01V4CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_005 of D1

drop tablespace is_customer_005;
create regular tablespace is_customer_005 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F01V5CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_006 of D1

drop tablespace is_customer_006;
create regular tablespace is_customer_006 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F01V6CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_007 of D1

drop tablespace is_customer_007;
create regular tablespace is_customer_007 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F02V1CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_008 of D1

drop tablespace is_customer_008;
create regular tablespace is_customer_008 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F02V2CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_009 of D1

drop tablespace is_customer_009;
create regular tablespace is_customer_009 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F02V3CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_010 of D1

drop tablespace is_customer_010;
create regular tablespace is_customer_010 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F02V4CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_011 of D1

drop tablespace is_customer_011;
create regular tablespace is_customer_011 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F02V5CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_012 of D1

drop tablespace is_customer_012;
create regular tablespace is_customer_012 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F02V6CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_013 of D1

drop tablespace is_customer_013;
create regular tablespace is_customer_013 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F03V1CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_014 of D1

drop tablespace is_customer_014;
create regular tablespace is_customer_014 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F03V2CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_015 of D1

drop tablespace is_customer_015;
create regular tablespace is_customer_015 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F03V3CSTI' 222720
                )
        extentsize 256
```

```
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_016 of D1

drop tablespace is_customer_016;
create regular tablespace is_customer_016 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F03V4CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_017 of D1

drop tablespace is_customer_017;
create regular tablespace is_customer_017 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F03V5CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_018 of D1

drop tablespace is_customer_018;
create regular tablespace is_customer_018 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F03V6CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_019 of D1

drop tablespace is_customer_019;
create regular tablespace is_customer_019 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F04V1CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_020 of D1

drop tablespace is_customer_020;
create regular tablespace is_customer_020 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F04V2CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_021 of D1

drop tablespace is_customer_021;
create regular tablespace is_customer_021 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F04V3CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_022 of D1

drop tablespace is_customer_022;
create regular tablespace is_customer_022 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F04V4CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_023 of D1

drop tablespace is_customer_023;
create regular tablespace is_customer_023 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F04V5CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_024 of D1

drop tablespace is_customer_024;
create regular tablespace is_customer_024 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F04V6CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_025 of D1

drop tablespace is_customer_025;
create regular tablespace is_customer_025 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F05V1CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_026 of D1

drop tablespace is_customer_026;
create regular tablespace is_customer_026 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F05V2CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_027 of D1

drop tablespace is_customer_027;
create regular tablespace is_customer_027 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F05V3CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_028 of D1

drop tablespace is_customer_028;
create regular tablespace is_customer_028 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F05V4CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_029 of D1

drop tablespace is_customer_029;
create regular tablespace is_customer_029 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F05V5CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_030 of D1

drop tablespace is_customer_030;
create regular tablespace is_customer_030 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F05V6CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_031 of D1

drop tablespace is_customer_031;
create regular tablespace is_customer_031 pagesize 8K
         managed by database
```

```
        using
        (
                device '/dev/rD1F06V1CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_032 of D1

drop tablespace is_customer_032;
create regular tablespace is_customer_032 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F06V2CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_033 of D1

drop tablespace is_customer_033;
create regular tablespace is_customer_033 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F06V3CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_034 of D1

drop tablespace is_customer_034;
create regular tablespace is_customer_034 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F06V4CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_035 of D1

drop tablespace is_customer_035;
create regular tablespace is_customer_035 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F06V5CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_036 of D1

drop tablespace is_customer_036;
create regular tablespace is_customer_036 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F06V6CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_037 of D1

drop tablespace is_customer_037;
create regular tablespace is_customer_037 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F07V1CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_038 of D1

drop tablespace is_customer_038;
create regular tablespace is_customer_038 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F07V2CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_039 of D1

drop tablespace is_customer_039;
create regular tablespace is_customer_039 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F07V3CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_040 of D1

drop tablespace is_customer_040;
create regular tablespace is_customer_040 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F07V4CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_041 of D1

drop tablespace is_customer_041;
create regular tablespace is_customer_041 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F07V5CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_042 of D1

drop tablespace is_customer_042;
create regular tablespace is_customer_042 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F07V6CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_043 of D1

drop tablespace is_customer_043;
create regular tablespace is_customer_043 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F08V1CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_044 of D1

drop tablespace is_customer_044;
create regular tablespace is_customer_044 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F08V2CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_045 of D1

drop tablespace is_customer_045;
create regular tablespace is_customer_045 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F08V3CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_046 of D1

drop tablespace is_customer_046;
create regular tablespace is_customer_046 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F08V4CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;
```

```
-- now creating TS for is_customer_047 of D1

drop tablespace is_customer_047;
create regular tablespace is_customer_047 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F08V5CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_048 of D1

drop tablespace is_customer_048;
create regular tablespace is_customer_048 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F08V6CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_049 of D1

drop tablespace is_customer_049;
create regular tablespace is_customer_049 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F09V1CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_050 of D1

drop tablespace is_customer_050;
create regular tablespace is_customer_050 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F09V2CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_051 of D1

drop tablespace is_customer_051;
create regular tablespace is_customer_051 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F09V3CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_052 of D1

drop tablespace is_customer_052;
create regular tablespace is_customer_052 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F09V4CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_053 of D1

drop tablespace is_customer_053;
create regular tablespace is_customer_053 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F09V5CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_054 of D1

drop tablespace is_customer_054;
create regular tablespace is_customer_054 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F09V6CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_055 of D1

drop tablespace is_customer_055;
create regular tablespace is_customer_055 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F10V1CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_056 of D1

drop tablespace is_customer_056;
create regular tablespace is_customer_056 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F10V2CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_057 of D1

drop tablespace is_customer_057;
create regular tablespace is_customer_057 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F10V3CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_058 of D1

drop tablespace is_customer_058;
create regular tablespace is_customer_058 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F10V4CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_059 of D1

drop tablespace is_customer_059;
create regular tablespace is_customer_059 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F10V5CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_060 of D1

drop tablespace is_customer_060;
create regular tablespace is_customer_060 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F10V6CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_061 of D1

drop tablespace is_customer_061;
create regular tablespace is_customer_061 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F11V1CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_062 of D1

drop tablespace is_customer_062;
create regular tablespace is_customer_062 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F11V2CSTI' 222720
                )
```

```
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_063 of D1

drop tablespace is_customer_063;
create regular tablespace is_customer_063 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F11V3CSTI' 222720
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_064 of D1

drop tablespace is_customer_064;
create regular tablespace is_customer_064 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F11V4CSTI' 222720
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_065 of D1

drop tablespace is_customer_065;
create regular tablespace is_customer_065 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F11V5CSTI' 222720
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_066 of D1

drop tablespace is_customer_066;
create regular tablespace is_customer_066 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F11V6CSTI' 222720
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_067 of D1

drop tablespace is_customer_067;
create regular tablespace is_customer_067 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F12V1CSTI' 222720
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_068 of D1

drop tablespace is_customer_068;
create regular tablespace is_customer_068 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F12V2CSTI' 222720
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_069 of D1

drop tablespace is_customer_069;
create regular tablespace is_customer_069 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F12V3CSTI' 222720
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_070 of D1

drop tablespace is_customer_070;
create regular tablespace is_customer_070 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F12V4CSTI' 222720
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_071 of D1

drop tablespace is_customer_071;
create regular tablespace is_customer_071 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F12V5CSTI' 222720
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_072 of D1

drop tablespace is_customer_072;
create regular tablespace is_customer_072 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F12V6CSTI' 222720
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_073 of D1

drop tablespace is_customer_073;
create regular tablespace is_customer_073 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F13V1CSTI' 222720
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_074 of D1

drop tablespace is_customer_074;
create regular tablespace is_customer_074 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F13V2CSTI' 222720
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_075 of D1

drop tablespace is_customer_075;
create regular tablespace is_customer_075 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F13V3CSTI' 222720
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_076 of D1

drop tablespace is_customer_076;
create regular tablespace is_customer_076 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F13V4CSTI' 222720
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_077 of D1

drop tablespace is_customer_077;
create regular tablespace is_customer_077 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F13V5CSTI' 222720
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_078 of D1

drop tablespace is_customer_078;
create regular tablespace is_customer_078 pagesize 8K
```

```
            managed by database
            using
            (
                    device '/dev/rD1F13V6CSTI' 222720
                    )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_079 of D1

drop tablespace is_customer_079;
create regular tablespace is_customer_079 pagesize 8K
            managed by database
            using
            (
                    device '/dev/rD1F14V1CSTI' 222720
                    )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_080 of D1

drop tablespace is_customer_080;
create regular tablespace is_customer_080 pagesize 8K
            managed by database
            using
            (
                    device '/dev/rD1F14V2CSTI' 222720
                    )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_081 of D1

drop tablespace is_customer_081;
create regular tablespace is_customer_081 pagesize 8K
            managed by database
            using
            (
                    device '/dev/rD1F14V3CSTI' 222720
                    )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_082 of D1

drop tablespace is_customer_082;
create regular tablespace is_customer_082 pagesize 8K
            managed by database
            using
            (
                    device '/dev/rD1F14V4CSTI' 222720
                    )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_083 of D1

drop tablespace is_customer_083;
create regular tablespace is_customer_083 pagesize 8K
            managed by database
            using
            (
                    device '/dev/rD1F14V5CSTI' 222720
                    )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_084 of D1

drop tablespace is_customer_084;
create regular tablespace is_customer_084 pagesize 8K
            managed by database
            using
            (
                    device '/dev/rD1F14V6CSTI' 222720
                    )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_085 of D1

drop tablespace is_customer_085;
create regular tablespace is_customer_085 pagesize 8K
            managed by database
            using
            (
                    device '/dev/rD1F15V1CSTI' 222720
                    )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_086 of D1

drop tablespace is_customer_086;
create regular tablespace is_customer_086 pagesize 8K
            managed by database
            using
            (
                    device '/dev/rD1F15V2CSTI' 222720
                    )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_087 of D1

drop tablespace is_customer_087;
create regular tablespace is_customer_087 pagesize 8K
            managed by database
            using
            (
                    device '/dev/rD1F15V3CSTI' 222720
                    )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_088 of D1

drop tablespace is_customer_088;
create regular tablespace is_customer_088 pagesize 8K
            managed by database
            using
            (
                    device '/dev/rD1F15V4CSTI' 222720
                    )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_089 of D1

drop tablespace is_customer_089;
create regular tablespace is_customer_089 pagesize 8K
            managed by database
            using
            (
                    device '/dev/rD1F15V5CSTI' 222720
                    )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_090 of D1

drop tablespace is_customer_090;
create regular tablespace is_customer_090 pagesize 8K
            managed by database
            using
            (
                    device '/dev/rD1F15V6CSTI' 222720
                    )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_091 of D1

drop tablespace is_customer_091;
create regular tablespace is_customer_091 pagesize 8K
            managed by database
            using
            (
                    device '/dev/rD1F16V1CSTI' 222720
                    )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_092 of D1

drop tablespace is_customer_092;
create regular tablespace is_customer_092 pagesize 8K
            managed by database
            using
            (
                    device '/dev/rD1F16V2CSTI' 222720
                    )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_093 of D1

drop tablespace is_customer_093;
create regular tablespace is_customer_093 pagesize 8K
            managed by database
            using
            (
                    device '/dev/rD1F16V3CSTI' 222720
                    )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;
```

```
-- now creating TS for is_customer_094 of D1

drop tablespace is_customer_094;
create regular tablespace is_customer_094 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F16V4CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_095 of D1

drop tablespace is_customer_095;
create regular tablespace is_customer_095 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F16V5CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_096 of D1

drop tablespace is_customer_096;
create regular tablespace is_customer_096 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F16V6CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_097 of D1

drop tablespace is_customer_097;
create regular tablespace is_customer_097 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F17V1CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_098 of D1

drop tablespace is_customer_098;
create regular tablespace is_customer_098 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F17V2CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_099 of D1

drop tablespace is_customer_099;
create regular tablespace is_customer_099 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F17V3CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_100 of D1

drop tablespace is_customer_100;
create regular tablespace is_customer_100 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F17V4CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_101 of D1

drop tablespace is_customer_101;
create regular tablespace is_customer_101 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F17V5CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_102 of D1

drop tablespace is_customer_102;
create regular tablespace is_customer_102 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F17V6CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_103 of D1

drop tablespace is_customer_103;
create regular tablespace is_customer_103 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F18V1CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_104 of D1

drop tablespace is_customer_104;
create regular tablespace is_customer_104 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F18V2CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_105 of D1

drop tablespace is_customer_105;
create regular tablespace is_customer_105 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F18V3CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_106 of D1

drop tablespace is_customer_106;
create regular tablespace is_customer_106 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F18V4CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_107 of D1

drop tablespace is_customer_107;
create regular tablespace is_customer_107 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F18V5CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_108 of D1

drop tablespace is_customer_108;
create regular tablespace is_customer_108 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F18V6CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_109 of D1

drop tablespace is_customer_109;
create regular tablespace is_customer_109 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F19V1CSTI' 222720
```

```
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_110 of D1

drop tablespace is_customer_110;
create regular tablespace is_customer_110 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F19V2CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_111 of D1

drop tablespace is_customer_111;
create regular tablespace is_customer_111 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F19V3CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_112 of D1

drop tablespace is_customer_112;
create regular tablespace is_customer_112 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F19V4CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_113 of D1

drop tablespace is_customer_113;
create regular tablespace is_customer_113 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F19V5CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_114 of D1

drop tablespace is_customer_114;
create regular tablespace is_customer_114 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F19V6CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_115 of D1

drop tablespace is_customer_115;
create regular tablespace is_customer_115 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F20V1CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_116 of D1

drop tablespace is_customer_116;
create regular tablespace is_customer_116 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F20V2CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_117 of D1

drop tablespace is_customer_117;
create regular tablespace is_customer_117 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F20V3CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_118 of D1

drop tablespace is_customer_118;
create regular tablespace is_customer_118 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F20V4CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_119 of D1

drop tablespace is_customer_119;
create regular tablespace is_customer_119 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F20V5CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_120 of D1

drop tablespace is_customer_120;
create regular tablespace is_customer_120 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F20V6CSTI' 222720
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

connect reset;
```

ts/cris_order.ddl

```
connect to tpcc;
-- now creating TS for is_order_001 of D1

drop tablespace is_order_001;
create regular tablespace is_order_001 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F01V1ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_002 of D1

drop tablespace is_order_002;
create regular tablespace is_order_002 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F01V2ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_003 of D1

drop tablespace is_order_003;
create regular tablespace is_order_003 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F01V3ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_004 of D1

drop tablespace is_order_004;
create regular tablespace is_order_004 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F01V4ORDI' 191744
                )
```

```
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_005 of D1

drop tablespace is_order_005;
create regular tablespace is_order_005 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F01V5ORDI' 191744
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_006 of D1

drop tablespace is_order_006;
create regular tablespace is_order_006 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F01V6ORDI' 191744
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_007 of D1

drop tablespace is_order_007;
create regular tablespace is_order_007 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F02V1ORDI' 191744
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_008 of D1

drop tablespace is_order_008;
create regular tablespace is_order_008 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F02V2ORDI' 191744
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_009 of D1

drop tablespace is_order_009;
create regular tablespace is_order_009 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F02V3ORDI' 191744
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_010 of D1

drop tablespace is_order_010;
create regular tablespace is_order_010 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F02V4ORDI' 191744
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_011 of D1

drop tablespace is_order_011;
create regular tablespace is_order_011 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F02V5ORDI' 191744
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_012 of D1

drop tablespace is_order_012;
create regular tablespace is_order_012 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F02V6ORDI' 191744
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_013 of D1

drop tablespace is_order_013;
create regular tablespace is_order_013 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F03V1ORDI' 191744
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_014 of D1

drop tablespace is_order_014;
create regular tablespace is_order_014 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F03V2ORDI' 191744
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_015 of D1

drop tablespace is_order_015;
create regular tablespace is_order_015 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F03V3ORDI' 191744
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_016 of D1

drop tablespace is_order_016;
create regular tablespace is_order_016 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F03V4ORDI' 191744
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_017 of D1

drop tablespace is_order_017;
create regular tablespace is_order_017 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F03V5ORDI' 191744
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_018 of D1

drop tablespace is_order_018;
create regular tablespace is_order_018 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F03V6ORDI' 191744
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_019 of D1

drop tablespace is_order_019;
create regular tablespace is_order_019 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F04V1ORDI' 191744
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_020 of D1

drop tablespace is_order_020;
create regular tablespace is_order_020 pagesize 8K
```

```
        managed by database
        using
        (
                device '/dev/rD1F04V2ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_021 of D1

drop tablespace is_order_021;
create regular tablespace is_order_021 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F04V3ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_022 of D1

drop tablespace is_order_022;
create regular tablespace is_order_022 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F04V4ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_023 of D1

drop tablespace is_order_023;
create regular tablespace is_order_023 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F04V5ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_024 of D1

drop tablespace is_order_024;
create regular tablespace is_order_024 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F04V6ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_025 of D1

drop tablespace is_order_025;
create regular tablespace is_order_025 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F05V1ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_026 of D1

drop tablespace is_order_026;
create regular tablespace is_order_026 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F05V2ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_027 of D1

drop tablespace is_order_027;
create regular tablespace is_order_027 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F05V3ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_028 of D1

drop tablespace is_order_028;
create regular tablespace is_order_028 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F05V4ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_029 of D1

drop tablespace is_order_029;
create regular tablespace is_order_029 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F05V5ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_030 of D1

drop tablespace is_order_030;
create regular tablespace is_order_030 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F05V6ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_031 of D1

drop tablespace is_order_031;
create regular tablespace is_order_031 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F06V1ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_032 of D1

drop tablespace is_order_032;
create regular tablespace is_order_032 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F06V2ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_033 of D1

drop tablespace is_order_033;
create regular tablespace is_order_033 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F06V3ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_034 of D1

drop tablespace is_order_034;
create regular tablespace is_order_034 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F06V4ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_035 of D1

drop tablespace is_order_035;
create regular tablespace is_order_035 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F06V5ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;
```

```
-- now creating TS for is_order_036 of D1

drop tablespace is_order_036;
create regular tablespace is_order_036 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F06V6ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_037 of D1

drop tablespace is_order_037;
create regular tablespace is_order_037 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F07V1ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_038 of D1

drop tablespace is_order_038;
create regular tablespace is_order_038 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F07V2ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_039 of D1

drop tablespace is_order_039;
create regular tablespace is_order_039 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F07V3ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_040 of D1

drop tablespace is_order_040;
create regular tablespace is_order_040 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F07V4ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_041 of D1

drop tablespace is_order_041;
create regular tablespace is_order_041 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F07V5ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_042 of D1

drop tablespace is_order_042;
create regular tablespace is_order_042 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F07V6ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_043 of D1

drop tablespace is_order_043;
create regular tablespace is_order_043 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F08V1ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_044 of D1

drop tablespace is_order_044;
create regular tablespace is_order_044 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F08V2ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_045 of D1

drop tablespace is_order_045;
create regular tablespace is_order_045 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F08V3ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_046 of D1

drop tablespace is_order_046;
create regular tablespace is_order_046 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F08V4ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_047 of D1

drop tablespace is_order_047;
create regular tablespace is_order_047 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F08V5ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_048 of D1

drop tablespace is_order_048;
create regular tablespace is_order_048 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F08V6ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_049 of D1

drop tablespace is_order_049;
create regular tablespace is_order_049 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F09V1ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_050 of D1

drop tablespace is_order_050;
create regular tablespace is_order_050 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F09V2ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_051 of D1

drop tablespace is_order_051;
create regular tablespace is_order_051 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F09V3ORDI' 191744
```

```
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_052 of D1

drop tablespace is_order_052;
create regular tablespace is_order_052 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F09V4ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_053 of D1

drop tablespace is_order_053;
create regular tablespace is_order_053 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F09V5ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_054 of D1

drop tablespace is_order_054;
create regular tablespace is_order_054 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F09V6ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_055 of D1

drop tablespace is_order_055;
create regular tablespace is_order_055 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F10V1ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_056 of D1

drop tablespace is_order_056;
create regular tablespace is_order_056 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F10V2ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_057 of D1

drop tablespace is_order_057;
create regular tablespace is_order_057 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F10V3ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_058 of D1

drop tablespace is_order_058;
create regular tablespace is_order_058 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F10V4ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_059 of D1

drop tablespace is_order_059;
create regular tablespace is_order_059 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F10V5ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_060 of D1

drop tablespace is_order_060;
create regular tablespace is_order_060 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F10V6ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_061 of D1

drop tablespace is_order_061;
create regular tablespace is_order_061 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F11V1ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_062 of D1

drop tablespace is_order_062;
create regular tablespace is_order_062 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F11V2ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_063 of D1

drop tablespace is_order_063;
create regular tablespace is_order_063 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F11V3ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_064 of D1

drop tablespace is_order_064;
create regular tablespace is_order_064 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F11V4ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_065 of D1

drop tablespace is_order_065;
create regular tablespace is_order_065 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F11V5ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_066 of D1

drop tablespace is_order_066;
create regular tablespace is_order_066 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F11V6ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_067 of D1

drop tablespace is_order_067;
```

```
create regular tablespace is_order_067 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F12V1ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_068 of D1

drop tablespace is_order_068;
create regular tablespace is_order_068 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F12V2ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_069 of D1

drop tablespace is_order_069;
create regular tablespace is_order_069 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F12V3ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_070 of D1

drop tablespace is_order_070;
create regular tablespace is_order_070 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F12V4ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_071 of D1

drop tablespace is_order_071;
create regular tablespace is_order_071 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F12V5ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_072 of D1

drop tablespace is_order_072;
create regular tablespace is_order_072 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F12V6ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_073 of D1

drop tablespace is_order_073;
create regular tablespace is_order_073 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F13V1ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_074 of D1

drop tablespace is_order_074;
create regular tablespace is_order_074 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F13V2ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_075 of D1

drop tablespace is_order_075;
create regular tablespace is_order_075 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F13V3ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_076 of D1

drop tablespace is_order_076;
create regular tablespace is_order_076 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F13V4ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_077 of D1

drop tablespace is_order_077;
create regular tablespace is_order_077 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F13V5ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_078 of D1

drop tablespace is_order_078;
create regular tablespace is_order_078 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F13V6ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_079 of D1

drop tablespace is_order_079;
create regular tablespace is_order_079 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F14V1ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_080 of D1

drop tablespace is_order_080;
create regular tablespace is_order_080 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F14V2ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_081 of D1

drop tablespace is_order_081;
create regular tablespace is_order_081 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F14V3ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_082 of D1

drop tablespace is_order_082;
create regular tablespace is_order_082 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F14V4ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
```

```
commit;

-- now creating TS for is_order_083 of D1

drop tablespace is_order_083;
create regular tablespace is_order_083 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F14V5ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_084 of D1

drop tablespace is_order_084;
create regular tablespace is_order_084 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F14V6ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_085 of D1

drop tablespace is_order_085;
create regular tablespace is_order_085 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F15V1ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_086 of D1

drop tablespace is_order_086;
create regular tablespace is_order_086 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F15V2ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_087 of D1

drop tablespace is_order_087;
create regular tablespace is_order_087 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F15V3ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_088 of D1

drop tablespace is_order_088;
create regular tablespace is_order_088 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F15V4ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_089 of D1

drop tablespace is_order_089;
create regular tablespace is_order_089 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F15V5ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_090 of D1

drop tablespace is_order_090;
create regular tablespace is_order_090 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F15V6ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_091 of D1

drop tablespace is_order_091;
create regular tablespace is_order_091 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F16V1ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_092 of D1

drop tablespace is_order_092;
create regular tablespace is_order_092 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F16V2ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_093 of D1

drop tablespace is_order_093;
create regular tablespace is_order_093 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F16V3ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_094 of D1

drop tablespace is_order_094;
create regular tablespace is_order_094 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F16V4ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_095 of D1

drop tablespace is_order_095;
create regular tablespace is_order_095 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F16V5ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_096 of D1

drop tablespace is_order_096;
create regular tablespace is_order_096 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F16V6ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_097 of D1

drop tablespace is_order_097;
create regular tablespace is_order_097 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F17V1ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_098 of D1

drop tablespace is_order_098;
create regular tablespace is_order_098 pagesize 8K
        managed by database
        using
        (
```

```
                device '/dev/rD1F17V2ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_099 of D1

drop tablespace is_order_099;
create regular tablespace is_order_099 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F17V3ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_100 of D1

drop tablespace is_order_100;
create regular tablespace is_order_100 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F17V4ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_101 of D1

drop tablespace is_order_101;
create regular tablespace is_order_101 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F17V5ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_102 of D1

drop tablespace is_order_102;
create regular tablespace is_order_102 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F17V6ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_103 of D1

drop tablespace is_order_103;
create regular tablespace is_order_103 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F18V1ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_104 of D1

drop tablespace is_order_104;
create regular tablespace is_order_104 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F18V2ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_105 of D1

drop tablespace is_order_105;
create regular tablespace is_order_105 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F18V3ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_106 of D1

drop tablespace is_order_106;
create regular tablespace is_order_106 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F18V4ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_107 of D1

drop tablespace is_order_107;
create regular tablespace is_order_107 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F18V5ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_108 of D1

drop tablespace is_order_108;
create regular tablespace is_order_108 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F18V6ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_109 of D1

drop tablespace is_order_109;
create regular tablespace is_order_109 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F19V1ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_110 of D1

drop tablespace is_order_110;
create regular tablespace is_order_110 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F19V2ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_111 of D1

drop tablespace is_order_111;
create regular tablespace is_order_111 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F19V3ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_112 of D1

drop tablespace is_order_112;
create regular tablespace is_order_112 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F19V4ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_113 of D1

drop tablespace is_order_113;
create regular tablespace is_order_113 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F19V5ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_114 of D1
```

```
drop tablespace is_order_114;
create regular tablespace is_order_114 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F19V6ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_115 of D1

drop tablespace is_order_115;
create regular tablespace is_order_115 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F20V1ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_116 of D1

drop tablespace is_order_116;
create regular tablespace is_order_116 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F20V2ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_117 of D1

drop tablespace is_order_117;
create regular tablespace is_order_117 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F20V3ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_118 of D1

drop tablespace is_order_118;
create regular tablespace is_order_118 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F20V4ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_119 of D1

drop tablespace is_order_119;
create regular tablespace is_order_119 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F20V5ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_order_120 of D1

drop tablespace is_order_120;
create regular tablespace is_order_120 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F20V6ORDI' 191744
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

connect reset;
```

ts/crts_customer.ddl

```
connect to tpcc;
-- now creating TS for ts_customer_001 of D1

drop tablespace ts_customer_001;
create regular tablespace ts_customer_001 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F01V1CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_002 of D1

drop tablespace ts_customer_002;
create regular tablespace ts_customer_002 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F01V2CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_003 of D1

drop tablespace ts_customer_003;
create regular tablespace ts_customer_003 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F01V3CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_004 of D1

drop tablespace ts_customer_004;
create regular tablespace ts_customer_004 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F01V4CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_005 of D1

drop tablespace ts_customer_005;
create regular tablespace ts_customer_005 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F01V5CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_006 of D1

drop tablespace ts_customer_006;
create regular tablespace ts_customer_006 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F01V6CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_007 of D1

drop tablespace ts_customer_007;
create regular tablespace ts_customer_007 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F02V1CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_008 of D1

drop tablespace ts_customer_008;
create regular tablespace ts_customer_008 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F02V2CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_009 of D1

drop tablespace ts_customer_009;
create regular tablespace ts_customer_009 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F02V3CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
```

```
commit;

-- now creating TS for ts_customer_010 of D1

drop tablespace ts_customer_010;
create regular tablespace ts_customer_010 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F02V4CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_011 of D1

drop tablespace ts_customer_011;
create regular tablespace ts_customer_011 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F02V5CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_012 of D1

drop tablespace ts_customer_012;
create regular tablespace ts_customer_012 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F02V6CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_013 of D1

drop tablespace ts_customer_013;
create regular tablespace ts_customer_013 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F03V1CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_014 of D1

drop tablespace ts_customer_014;
create regular tablespace ts_customer_014 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F03V2CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_015 of D1

drop tablespace ts_customer_015;
create regular tablespace ts_customer_015 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F03V3CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_016 of D1

drop tablespace ts_customer_016;
create regular tablespace ts_customer_016 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F03V4CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_017 of D1

drop tablespace ts_customer_017;
create regular tablespace ts_customer_017 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F03V5CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_018 of D1

drop tablespace ts_customer_018;
create regular tablespace ts_customer_018 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F03V6CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_019 of D1

drop tablespace ts_customer_019;
create regular tablespace ts_customer_019 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F04V1CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_020 of D1

drop tablespace ts_customer_020;
create regular tablespace ts_customer_020 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F04V2CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_021 of D1

drop tablespace ts_customer_021;
create regular tablespace ts_customer_021 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F04V3CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_022 of D1

drop tablespace ts_customer_022;
create regular tablespace ts_customer_022 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F04V4CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_023 of D1

drop tablespace ts_customer_023;
create regular tablespace ts_customer_023 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F04V5CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_024 of D1

drop tablespace ts_customer_024;
create regular tablespace ts_customer_024 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F04V6CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_025 of D1

drop tablespace ts_customer_025;
create regular tablespace ts_customer_025 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F05V1CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_026 of D1

drop tablespace ts_customer_026;
create regular tablespace ts_customer_026 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F05V2CST' 6717440
                )
```

```
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_027 of D1

drop tablespace ts_customer_027;
create regular tablespace ts_customer_027 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F05V3CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_028 of D1

drop tablespace ts_customer_028;
create regular tablespace ts_customer_028 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F05V4CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_029 of D1

drop tablespace ts_customer_029;
create regular tablespace ts_customer_029 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F05V5CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_030 of D1

drop tablespace ts_customer_030;
create regular tablespace ts_customer_030 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F05V6CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_031 of D1

drop tablespace ts_customer_031;
create regular tablespace ts_customer_031 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F06V1CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_032 of D1

drop tablespace ts_customer_032;
create regular tablespace ts_customer_032 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F06V2CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_033 of D1

drop tablespace ts_customer_033;
create regular tablespace ts_customer_033 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F06V3CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_034 of D1

drop tablespace ts_customer_034;
create regular tablespace ts_customer_034 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F06V4CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_035 of D1

drop tablespace ts_customer_035;
create regular tablespace ts_customer_035 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F06V5CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_036 of D1

drop tablespace ts_customer_036;
create regular tablespace ts_customer_036 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F06V6CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_037 of D1

drop tablespace ts_customer_037;
create regular tablespace ts_customer_037 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F07V1CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_038 of D1

drop tablespace ts_customer_038;
create regular tablespace ts_customer_038 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F07V2CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_039 of D1

drop tablespace ts_customer_039;
create regular tablespace ts_customer_039 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F07V3CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_040 of D1

drop tablespace ts_customer_040;
create regular tablespace ts_customer_040 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F07V4CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_041 of D1

drop tablespace ts_customer_041;
create regular tablespace ts_customer_041 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F07V5CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_042 of D1

drop tablespace ts_customer_042;
create regular tablespace ts_customer_042 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F07V6CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_043 of D1

drop tablespace ts_customer_043;
create regular tablespace ts_customer_043 pagesize 4K
         managed by database
         using
         (
```

```
                device '/dev/rD1F08V1CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_044 of D1

drop tablespace ts_customer_044;
create regular tablespace ts_customer_044 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F08V2CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_045 of D1

drop tablespace ts_customer_045;
create regular tablespace ts_customer_045 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F08V3CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_046 of D1

drop tablespace ts_customer_046;
create regular tablespace ts_customer_046 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F08V4CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_047 of D1

drop tablespace ts_customer_047;
create regular tablespace ts_customer_047 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F08V5CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_048 of D1

drop tablespace ts_customer_048;
create regular tablespace ts_customer_048 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F08V6CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_049 of D1

drop tablespace ts_customer_049;
create regular tablespace ts_customer_049 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F09V1CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_050 of D1

drop tablespace ts_customer_050;
create regular tablespace ts_customer_050 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F09V2CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_051 of D1

drop tablespace ts_customer_051;
create regular tablespace ts_customer_051 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F09V3CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_052 of D1

drop tablespace ts_customer_052;
create regular tablespace ts_customer_052 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F09V4CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_053 of D1

drop tablespace ts_customer_053;
create regular tablespace ts_customer_053 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F09V5CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_054 of D1

drop tablespace ts_customer_054;
create regular tablespace ts_customer_054 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F09V6CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_055 of D1

drop tablespace ts_customer_055;
create regular tablespace ts_customer_055 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F10V1CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_056 of D1

drop tablespace ts_customer_056;
create regular tablespace ts_customer_056 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F10V2CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_057 of D1

drop tablespace ts_customer_057;
create regular tablespace ts_customer_057 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F10V3CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_058 of D1

drop tablespace ts_customer_058;
create regular tablespace ts_customer_058 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F10V4CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_059 of D1

drop tablespace ts_customer_059;
create regular tablespace ts_customer_059 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F10V5CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_060 of D1

drop tablespace ts_customer_060;
create regular tablespace ts_customer_060 pagesize 4K
        managed by database
```

```
        using
        (
                device '/dev/rD1F10V6CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_061 of D1

drop tablespace ts_customer_061;
create regular tablespace ts_customer_061 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F11V1CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_062 of D1

drop tablespace ts_customer_062;
create regular tablespace ts_customer_062 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F11V2CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_063 of D1

drop tablespace ts_customer_063;
create regular tablespace ts_customer_063 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F11V3CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_064 of D1

drop tablespace ts_customer_064;
create regular tablespace ts_customer_064 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F11V4CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_065 of D1

drop tablespace ts_customer_065;
create regular tablespace ts_customer_065 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F11V5CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_066 of D1

drop tablespace ts_customer_066;
create regular tablespace ts_customer_066 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F11V6CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_067 of D1

drop tablespace ts_customer_067;
create regular tablespace ts_customer_067 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F12V1CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_068 of D1

drop tablespace ts_customer_068;
create regular tablespace ts_customer_068 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F12V2CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_069 of D1

drop tablespace ts_customer_069;
create regular tablespace ts_customer_069 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F12V3CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_070 of D1

drop tablespace ts_customer_070;
create regular tablespace ts_customer_070 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F12V4CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_071 of D1

drop tablespace ts_customer_071;
create regular tablespace ts_customer_071 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F12V5CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_072 of D1

drop tablespace ts_customer_072;
create regular tablespace ts_customer_072 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F12V6CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_073 of D1

drop tablespace ts_customer_073;
create regular tablespace ts_customer_073 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F13V1CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_074 of D1

drop tablespace ts_customer_074;
create regular tablespace ts_customer_074 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F13V2CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_075 of D1

drop tablespace ts_customer_075;
create regular tablespace ts_customer_075 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F13V3CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_076 of D1

drop tablespace ts_customer_076;
create regular tablespace ts_customer_076 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F13V4CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_077 of D1

drop tablespace ts_customer_077;
```

```
create regular tablespace ts_customer_077 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F13V5CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_078 of D1

drop tablespace ts_customer_078;
create regular tablespace ts_customer_078 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F13V6CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_079 of D1

drop tablespace ts_customer_079;
create regular tablespace ts_customer_079 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F14V1CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_080 of D1

drop tablespace ts_customer_080;
create regular tablespace ts_customer_080 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F14V2CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_081 of D1

drop tablespace ts_customer_081;
create regular tablespace ts_customer_081 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F14V3CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_082 of D1

drop tablespace ts_customer_082;
create regular tablespace ts_customer_082 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F14V4CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_083 of D1

drop tablespace ts_customer_083;
create regular tablespace ts_customer_083 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F14V5CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_084 of D1

drop tablespace ts_customer_084;
create regular tablespace ts_customer_084 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F14V6CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_085 of D1

drop tablespace ts_customer_085;
create regular tablespace ts_customer_085 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F15V1CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_086 of D1

drop tablespace ts_customer_086;
create regular tablespace ts_customer_086 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F15V2CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_087 of D1

drop tablespace ts_customer_087;
create regular tablespace ts_customer_087 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F15V3CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_088 of D1

drop tablespace ts_customer_088;
create regular tablespace ts_customer_088 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F15V4CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_089 of D1

drop tablespace ts_customer_089;
create regular tablespace ts_customer_089 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F15V5CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_090 of D1

drop tablespace ts_customer_090;
create regular tablespace ts_customer_090 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F15V6CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_091 of D1

drop tablespace ts_customer_091;
create regular tablespace ts_customer_091 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F16V1CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_092 of D1

drop tablespace ts_customer_092;
create regular tablespace ts_customer_092 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F16V2CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_093 of D1

drop tablespace ts_customer_093;
create regular tablespace ts_customer_093 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F16V3CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_094 of D1
```

```
drop tablespace ts_customer_094;
create regular tablespace ts_customer_094 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F16V4CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_095 of D1

drop tablespace ts_customer_095;
create regular tablespace ts_customer_095 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F16V5CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_096 of D1

drop tablespace ts_customer_096;
create regular tablespace ts_customer_096 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F16V6CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_097 of D1

drop tablespace ts_customer_097;
create regular tablespace ts_customer_097 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F17V1CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_098 of D1

drop tablespace ts_customer_098;
create regular tablespace ts_customer_098 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F17V2CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_099 of D1

drop tablespace ts_customer_099;
create regular tablespace ts_customer_099 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F17V3CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_100 of D1

drop tablespace ts_customer_100;
create regular tablespace ts_customer_100 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F17V4CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_101 of D1

drop tablespace ts_customer_101;
create regular tablespace ts_customer_101 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F17V5CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_102 of D1

drop tablespace ts_customer_102;
create regular tablespace ts_customer_102 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F17V6CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_103 of D1

drop tablespace ts_customer_103;
create regular tablespace ts_customer_103 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F18V1CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_104 of D1

drop tablespace ts_customer_104;
create regular tablespace ts_customer_104 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F18V2CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_105 of D1

drop tablespace ts_customer_105;
create regular tablespace ts_customer_105 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F18V3CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_106 of D1

drop tablespace ts_customer_106;
create regular tablespace ts_customer_106 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F18V4CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_107 of D1

drop tablespace ts_customer_107;
create regular tablespace ts_customer_107 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F18V5CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_108 of D1

drop tablespace ts_customer_108;
create regular tablespace ts_customer_108 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F18V6CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_109 of D1

drop tablespace ts_customer_109;
create regular tablespace ts_customer_109 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F19V1CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_110 of D1

drop tablespace ts_customer_110;
create regular tablespace ts_customer_110 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F19V2CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;
```

```
-- now creating TS for ts_customer_111 of D1

drop tablespace ts_customer_111;
create regular tablespace ts_customer_111 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F19V3CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_112 of D1

drop tablespace ts_customer_112;
create regular tablespace ts_customer_112 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F19V4CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_113 of D1

drop tablespace ts_customer_113;
create regular tablespace ts_customer_113 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F19V5CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_114 of D1

drop tablespace ts_customer_114;
create regular tablespace ts_customer_114 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F19V6CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_115 of D1

drop tablespace ts_customer_115;
create regular tablespace ts_customer_115 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F20V1CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_116 of D1

drop tablespace ts_customer_116;
create regular tablespace ts_customer_116 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F20V2CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_117 of D1

drop tablespace ts_customer_117;
create regular tablespace ts_customer_117 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F20V3CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_118 of D1

drop tablespace ts_customer_118;
create regular tablespace ts_customer_118 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F20V4CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_119 of D1

drop tablespace ts_customer_119;
create regular tablespace ts_customer_119 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F20V5CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_customer_120 of D1

drop tablespace ts_customer_120;
create regular tablespace ts_customer_120 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F20V6CST' 6717440
                )
        extentsize 256
        prefetchsize 4096;
commit;

connect reset;
```

ts/crts_dist.ddl

```
connect to tpcc;
-- now creating TS for ts_dist_001 of D1

drop tablespace ts_dist_001;
create regular tablespace ts_dist_001 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F01V1DIST' 512,
                device '/dev/rD1F01V2DIST' 512,
                device '/dev/rD1F01V3DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_002 of D1

drop tablespace ts_dist_002;
create regular tablespace ts_dist_002 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F01V4DIST' 512,
                device '/dev/rD1F01V5DIST' 512,
                device '/dev/rD1F01V6DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_003 of D1

drop tablespace ts_dist_003;
create regular tablespace ts_dist_003 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F02V1DIST' 512,
                device '/dev/rD1F02V2DIST' 512,
                device '/dev/rD1F02V3DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_004 of D1

drop tablespace ts_dist_004;
create regular tablespace ts_dist_004 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F02V4DIST' 512,
                device '/dev/rD1F02V5DIST' 512,
                device '/dev/rD1F02V6DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_005 of D1

drop tablespace ts_dist_005;
create regular tablespace ts_dist_005 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F03V1DIST' 512,
                device '/dev/rD1F03V2DIST' 512,
                device '/dev/rD1F03V3DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_006 of D1

drop tablespace ts_dist_006;
create regular tablespace ts_dist_006 pagesize 4K
        managed by database
```

```
        using
        (
                device '/dev/rD1F03V4DIST' 512,
                device '/dev/rD1F03V5DIST' 512,
                device '/dev/rD1F03V6DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_007 of D1

drop tablespace ts_dist_007;
create regular tablespace ts_dist_007 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F04V1DIST' 512,
                device '/dev/rD1F04V2DIST' 512,
                device '/dev/rD1F04V3DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_008 of D1

drop tablespace ts_dist_008;
create regular tablespace ts_dist_008 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F04V4DIST' 512,
                device '/dev/rD1F04V5DIST' 512,
                device '/dev/rD1F04V6DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_009 of D1

drop tablespace ts_dist_009;
create regular tablespace ts_dist_009 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F05V1DIST' 512,
                device '/dev/rD1F05V2DIST' 512,
                device '/dev/rD1F05V3DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_010 of D1

drop tablespace ts_dist_010;
create regular tablespace ts_dist_010 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F05V4DIST' 512,
                device '/dev/rD1F05V5DIST' 512,
                device '/dev/rD1F05V6DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_011 of D1

drop tablespace ts_dist_011;
create regular tablespace ts_dist_011 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F06V1DIST' 512,
                device '/dev/rD1F06V2DIST' 512,
                device '/dev/rD1F06V3DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_012 of D1

drop tablespace ts_dist_012;
create regular tablespace ts_dist_012 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F06V4DIST' 512,
                device '/dev/rD1F06V5DIST' 512,
                device '/dev/rD1F06V6DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_013 of D1

drop tablespace ts_dist_013;
create regular tablespace ts_dist_013 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F07V1DIST' 512,
                device '/dev/rD1F07V2DIST' 512,
                device '/dev/rD1F07V3DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_014 of D1

drop tablespace ts_dist_014;
create regular tablespace ts_dist_014 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F07V4DIST' 512,
                device '/dev/rD1F07V5DIST' 512,
                device '/dev/rD1F07V6DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_015 of D1

drop tablespace ts_dist_015;
create regular tablespace ts_dist_015 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F08V1DIST' 512,
                device '/dev/rD1F08V2DIST' 512,
                device '/dev/rD1F08V3DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_016 of D1

drop tablespace ts_dist_016;
create regular tablespace ts_dist_016 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F08V4DIST' 512,
                device '/dev/rD1F08V5DIST' 512,
                device '/dev/rD1F08V6DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_017 of D1

drop tablespace ts_dist_017;
create regular tablespace ts_dist_017 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F09V1DIST' 512,
                device '/dev/rD1F09V2DIST' 512,
                device '/dev/rD1F09V3DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_018 of D1

drop tablespace ts_dist_018;
create regular tablespace ts_dist_018 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F09V4DIST' 512,
                device '/dev/rD1F09V5DIST' 512,
                device '/dev/rD1F09V6DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_019 of D1

drop tablespace ts_dist_019;
create regular tablespace ts_dist_019 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F10V1DIST' 512,
                device '/dev/rD1F10V2DIST' 512,
                device '/dev/rD1F10V3DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_020 of D1

drop tablespace ts_dist_020;
create regular tablespace ts_dist_020 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F10V4DIST' 512,
                device '/dev/rD1F10V5DIST' 512,
                device '/dev/rD1F10V6DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;
```

```
-- now creating TS for ts_dist_021 of D1

drop tablespace ts_dist_021;
create regular tablespace ts_dist_021 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F11V1DIST' 512,
                device '/dev/rD1F11V2DIST' 512,
                device '/dev/rD1F11V3DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_022 of D1

drop tablespace ts_dist_022;
create regular tablespace ts_dist_022 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F11V4DIST' 512,
                device '/dev/rD1F11V5DIST' 512,
                device '/dev/rD1F11V6DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_023 of D1

drop tablespace ts_dist_023;
create regular tablespace ts_dist_023 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F12V1DIST' 512,
                device '/dev/rD1F12V2DIST' 512,
                device '/dev/rD1F12V3DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_024 of D1

drop tablespace ts_dist_024;
create regular tablespace ts_dist_024 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F12V4DIST' 512,
                device '/dev/rD1F12V5DIST' 512,
                device '/dev/rD1F12V6DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_025 of D1

drop tablespace ts_dist_025;
create regular tablespace ts_dist_025 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F13V1DIST' 512,
                device '/dev/rD1F13V2DIST' 512,
                device '/dev/rD1F13V3DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_026 of D1

drop tablespace ts_dist_026;
create regular tablespace ts_dist_026 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F13V4DIST' 512,
                device '/dev/rD1F13V5DIST' 512,
                device '/dev/rD1F13V6DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_027 of D1

drop tablespace ts_dist_027;
create regular tablespace ts_dist_027 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F14V1DIST' 512,
                device '/dev/rD1F14V2DIST' 512,
                device '/dev/rD1F14V3DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_028 of D1

drop tablespace ts_dist_028;
create regular tablespace ts_dist_028 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F14V4DIST' 512,
                device '/dev/rD1F14V5DIST' 512,
                device '/dev/rD1F14V6DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_029 of D1

drop tablespace ts_dist_029;
create regular tablespace ts_dist_029 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F15V1DIST' 512,
                device '/dev/rD1F15V2DIST' 512,
                device '/dev/rD1F15V3DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_030 of D1

drop tablespace ts_dist_030;
create regular tablespace ts_dist_030 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F15V4DIST' 512,
                device '/dev/rD1F15V5DIST' 512,
                device '/dev/rD1F15V6DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_031 of D1

drop tablespace ts_dist_031;
create regular tablespace ts_dist_031 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F16V1DIST' 512,
                device '/dev/rD1F16V2DIST' 512,
                device '/dev/rD1F16V3DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_032 of D1

drop tablespace ts_dist_032;
create regular tablespace ts_dist_032 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F16V4DIST' 512,
                device '/dev/rD1F16V5DIST' 512,
                device '/dev/rD1F16V6DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_033 of D1

drop tablespace ts_dist_033;
create regular tablespace ts_dist_033 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F17V1DIST' 512,
                device '/dev/rD1F17V2DIST' 512,
                device '/dev/rD1F17V3DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_034 of D1

drop tablespace ts_dist_034;
create regular tablespace ts_dist_034 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F17V4DIST' 512,
                device '/dev/rD1F17V5DIST' 512,
                device '/dev/rD1F17V6DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_035 of D1

drop tablespace ts_dist_035;
create regular tablespace ts_dist_035 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F18V1DIST' 512,
```

```
                device '/dev/rD1F18V2DIST' 512,
                device '/dev/rD1F18V3DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_036 of D1

drop tablespace ts_dist_036;
create regular tablespace ts_dist_036 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F18V4DIST' 512,
                device '/dev/rD1F18V5DIST' 512,
                device '/dev/rD1F18V6DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_037 of D1

drop tablespace ts_dist_037;
create regular tablespace ts_dist_037 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F19V1DIST' 512,
                device '/dev/rD1F19V2DIST' 512,
                device '/dev/rD1F19V3DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_038 of D1

drop tablespace ts_dist_038;
create regular tablespace ts_dist_038 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F19V4DIST' 512,
                device '/dev/rD1F19V5DIST' 512,
                device '/dev/rD1F19V6DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_039 of D1

drop tablespace ts_dist_039;
create regular tablespace ts_dist_039 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F20V1DIST' 512,
                device '/dev/rD1F20V2DIST' 512,
                device '/dev/rD1F20V3DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

-- now creating TS for ts_dist_040 of D1

drop tablespace ts_dist_040;
create regular tablespace ts_dist_040 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F20V4DIST' 512,
                device '/dev/rD1F20V5DIST' 512,
                device '/dev/rD1F20V6DIST' 512
                )
        extentsize 64
        prefetchsize 4096;
commit;

connect reset;
```

ts/crts_history.ddl

```
connect to tpcc;
-- now creating TS for ts_history_001 of D1

drop tablespace ts_history_001;
create regular tablespace ts_history_001 pagesize 16K
         managed by database
         using
         (
                device '/dev/rD1F01V1HIST' 160768,
                device '/dev/rD1F01V2HIST' 160768,
                device '/dev/rD1F01V3HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_002 of D1

drop tablespace ts_history_002;
create regular tablespace ts_history_002 pagesize 16K
         managed by database
         using
         (
                device '/dev/rD1F01V4HIST' 160768,
                device '/dev/rD1F01V5HIST' 160768,
                device '/dev/rD1F01V6HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_003 of D1

drop tablespace ts_history_003;
create regular tablespace ts_history_003 pagesize 16K
         managed by database
         using
         (
                device '/dev/rD1F02V1HIST' 160768,
                device '/dev/rD1F02V2HIST' 160768,
                device '/dev/rD1F02V3HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_004 of D1

drop tablespace ts_history_004;
create regular tablespace ts_history_004 pagesize 16K
         managed by database
         using
         (
                device '/dev/rD1F02V4HIST' 160768,
                device '/dev/rD1F02V5HIST' 160768,
                device '/dev/rD1F02V6HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_005 of D1

drop tablespace ts_history_005;
create regular tablespace ts_history_005 pagesize 16K
         managed by database
         using
         (
                device '/dev/rD1F03V1HIST' 160768,
                device '/dev/rD1F03V2HIST' 160768,
                device '/dev/rD1F03V3HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_006 of D1

drop tablespace ts_history_006;
create regular tablespace ts_history_006 pagesize 16K
         managed by database
         using
         (
                device '/dev/rD1F03V4HIST' 160768,
                device '/dev/rD1F03V5HIST' 160768,
                device '/dev/rD1F03V6HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_007 of D1

drop tablespace ts_history_007;
create regular tablespace ts_history_007 pagesize 16K
         managed by database
         using
         (
                device '/dev/rD1F04V1HIST' 160768,
                device '/dev/rD1F04V2HIST' 160768,
                device '/dev/rD1F04V3HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_008 of D1

drop tablespace ts_history_008;
create regular tablespace ts_history_008 pagesize 16K
         managed by database
         using
         (
                device '/dev/rD1F04V4HIST' 160768,
                device '/dev/rD1F04V5HIST' 160768,
                device '/dev/rD1F04V6HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_009 of D1
```

```
drop tablespace ts_history_009;
create regular tablespace ts_history_009 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F05V1HIST' 160768,
                device '/dev/rD1F05V2HIST' 160768,
                device '/dev/rD1F05V3HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_010 of D1

drop tablespace ts_history_010;
create regular tablespace ts_history_010 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F05V4HIST' 160768,
                device '/dev/rD1F05V5HIST' 160768,
                device '/dev/rD1F05V6HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_011 of D1

drop tablespace ts_history_011;
create regular tablespace ts_history_011 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F06V1HIST' 160768,
                device '/dev/rD1F06V2HIST' 160768,
                device '/dev/rD1F06V3HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_012 of D1

drop tablespace ts_history_012;
create regular tablespace ts_history_012 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F06V4HIST' 160768,
                device '/dev/rD1F06V5HIST' 160768,
                device '/dev/rD1F06V6HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_013 of D1

drop tablespace ts_history_013;
create regular tablespace ts_history_013 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F07V1HIST' 160768,
                device '/dev/rD1F07V2HIST' 160768,
                device '/dev/rD1F07V3HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_014 of D1

drop tablespace ts_history_014;
create regular tablespace ts_history_014 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F07V4HIST' 160768,
                device '/dev/rD1F07V5HIST' 160768,
                device '/dev/rD1F07V6HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_015 of D1

drop tablespace ts_history_015;
create regular tablespace ts_history_015 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F08V1HIST' 160768,
                device '/dev/rD1F08V2HIST' 160768,
                device '/dev/rD1F08V3HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_016 of D1

drop tablespace ts_history_016;
create regular tablespace ts_history_016 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F08V4HIST' 160768,
                device '/dev/rD1F08V5HIST' 160768,
                device '/dev/rD1F08V6HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_017 of D1

drop tablespace ts_history_017;
create regular tablespace ts_history_017 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F09V1HIST' 160768,
                device '/dev/rD1F09V2HIST' 160768,
                device '/dev/rD1F09V3HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_018 of D1

drop tablespace ts_history_018;
create regular tablespace ts_history_018 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F09V4HIST' 160768,
                device '/dev/rD1F09V5HIST' 160768,
                device '/dev/rD1F09V6HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_019 of D1

drop tablespace ts_history_019;
create regular tablespace ts_history_019 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F10V1HIST' 160768,
                device '/dev/rD1F10V2HIST' 160768,
                device '/dev/rD1F10V3HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_020 of D1

drop tablespace ts_history_020;
create regular tablespace ts_history_020 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F10V4HIST' 160768,
                device '/dev/rD1F10V5HIST' 160768,
                device '/dev/rD1F10V6HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_021 of D1

drop tablespace ts_history_021;
create regular tablespace ts_history_021 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F11V1HIST' 160768,
                device '/dev/rD1F11V2HIST' 160768,
                device '/dev/rD1F11V3HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_022 of D1

drop tablespace ts_history_022;
create regular tablespace ts_history_022 pagesize 16K
        managed by database
        using
        (
                device '/dev/rD1F11V4HIST' 160768,
                device '/dev/rD1F11V5HIST' 160768,
                device '/dev/rD1F11V6HIST' 160768
                )
        extentsize 256
```

```
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_023 of D1

drop tablespace ts_history_023;
create regular tablespace ts_history_023 pagesize 16K
         managed by database
         using
         (
                 device '/dev/rD1F12V1HIST' 160768,
                 device '/dev/rD1F12V2HIST' 160768,
                 device '/dev/rD1F12V3HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_024 of D1

drop tablespace ts_history_024;
create regular tablespace ts_history_024 pagesize 16K
         managed by database
         using
         (
                 device '/dev/rD1F12V4HIST' 160768,
                 device '/dev/rD1F12V5HIST' 160768,
                 device '/dev/rD1F12V6HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_025 of D1

drop tablespace ts_history_025;
create regular tablespace ts_history_025 pagesize 16K
         managed by database
         using
         (
                 device '/dev/rD1F13V1HIST' 160768,
                 device '/dev/rD1F13V2HIST' 160768,
                 device '/dev/rD1F13V3HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_026 of D1

drop tablespace ts_history_026;
create regular tablespace ts_history_026 pagesize 16K
         managed by database
         using
         (
                 device '/dev/rD1F13V4HIST' 160768,
                 device '/dev/rD1F13V5HIST' 160768,
                 device '/dev/rD1F13V6HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_027 of D1

drop tablespace ts_history_027;
create regular tablespace ts_history_027 pagesize 16K
         managed by database
         using
         (
                 device '/dev/rD1F14V1HIST' 160768,
                 device '/dev/rD1F14V2HIST' 160768,
                 device '/dev/rD1F14V3HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_028 of D1

drop tablespace ts_history_028;
create regular tablespace ts_history_028 pagesize 16K
         managed by database
         using
         (
                 device '/dev/rD1F14V4HIST' 160768,
                 device '/dev/rD1F14V5HIST' 160768,
                 device '/dev/rD1F14V6HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_029 of D1

drop tablespace ts_history_029;
create regular tablespace ts_history_029 pagesize 16K
         managed by database
         using
         (
                 device '/dev/rD1F15V1HIST' 160768,
                 device '/dev/rD1F15V2HIST' 160768,
                 device '/dev/rD1F15V3HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_030 of D1

drop tablespace ts_history_030;
create regular tablespace ts_history_030 pagesize 16K
         managed by database
         using
         (
                 device '/dev/rD1F15V4HIST' 160768,
                 device '/dev/rD1F15V5HIST' 160768,
                 device '/dev/rD1F15V6HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_031 of D1

drop tablespace ts_history_031;
create regular tablespace ts_history_031 pagesize 16K
         managed by database
         using
         (
                 device '/dev/rD1F16V1HIST' 160768,
                 device '/dev/rD1F16V2HIST' 160768,
                 device '/dev/rD1F16V3HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_032 of D1

drop tablespace ts_history_032;
create regular tablespace ts_history_032 pagesize 16K
         managed by database
         using
         (
                 device '/dev/rD1F16V4HIST' 160768,
                 device '/dev/rD1F16V5HIST' 160768,
                 device '/dev/rD1F16V6HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_033 of D1

drop tablespace ts_history_033;
create regular tablespace ts_history_033 pagesize 16K
         managed by database
         using
         (
                 device '/dev/rD1F17V1HIST' 160768,
                 device '/dev/rD1F17V2HIST' 160768,
                 device '/dev/rD1F17V3HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_034 of D1

drop tablespace ts_history_034;
create regular tablespace ts_history_034 pagesize 16K
         managed by database
         using
         (
                 device '/dev/rD1F17V4HIST' 160768,
                 device '/dev/rD1F17V5HIST' 160768,
                 device '/dev/rD1F17V6HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_035 of D1

drop tablespace ts_history_035;
create regular tablespace ts_history_035 pagesize 16K
         managed by database
         using
         (
                 device '/dev/rD1F18V1HIST' 160768,
                 device '/dev/rD1F18V2HIST' 160768,
                 device '/dev/rD1F18V3HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_036 of D1

drop tablespace ts_history_036;
create regular tablespace ts_history_036 pagesize 16K
         managed by database
         using
         (
```

```
                device '/dev/rD1F18V4HIST' 160768,
                device '/dev/rD1F18V5HIST' 160768,
                device '/dev/rD1F18V6HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_037 of D1

drop tablespace ts_history_037;
create regular tablespace ts_history_037 pagesize 16K
         managed by database
         using
         (
                device '/dev/rD1F19V1HIST' 160768,
                device '/dev/rD1F19V2HIST' 160768,
                device '/dev/rD1F19V3HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_038 of D1

drop tablespace ts_history_038;
create regular tablespace ts_history_038 pagesize 16K
         managed by database
         using
         (
                device '/dev/rD1F19V4HIST' 160768,
                device '/dev/rD1F19V5HIST' 160768,
                device '/dev/rD1F19V6HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_039 of D1

drop tablespace ts_history_039;
create regular tablespace ts_history_039 pagesize 16K
         managed by database
         using
         (
                device '/dev/rD1F20V1HIST' 160768,
                device '/dev/rD1F20V2HIST' 160768,
                device '/dev/rD1F20V3HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

-- now creating TS for ts_history_040 of D1

drop tablespace ts_history_040;
create regular tablespace ts_history_040 pagesize 16K
         managed by database
         using
         (
                device '/dev/rD1F20V4HIST' 160768,
                device '/dev/rD1F20V5HIST' 160768,
                device '/dev/rD1F20V6HIST' 160768
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp16K;
commit;

connect reset;
```

ts/crts_item.ddl

```
connect to tpcc;
-- now creating TS for ts_item_001 of D1

drop tablespace ts_item_001;
create regular tablespace ts_item_001 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F01V1ITEM' 1408,
                device '/dev/rD1F01V2ITEM' 1408,
                device '/dev/rD1F01V3ITEM' 1408,
                device '/dev/rD1F01V4ITEM' 1408,
                device '/dev/rD1F01V5ITEM' 1408,
                device '/dev/rD1F01V6ITEM' 1408,
                device '/dev/rD1F02V1ITEM' 1408,
                device '/dev/rD1F02V2ITEM' 1408,
                device '/dev/rD1F02V3ITEM' 1408,
                device '/dev/rD1F02V4ITEM' 1408,
                device '/dev/rD1F02V5ITEM' 1408,
                device '/dev/rD1F02V6ITEM' 1408,
                device '/dev/rD1F03V1ITEM' 1408,
                device '/dev/rD1F03V2ITEM' 1408,
                device '/dev/rD1F03V3ITEM' 1408,
                device '/dev/rD1F03V4ITEM' 1408,
                device '/dev/rD1F03V5ITEM' 1408,
                device '/dev/rD1F03V6ITEM' 1408,
                device '/dev/rD1F04V1ITEM' 1408,
                device '/dev/rD1F04V2ITEM' 1408,
                device '/dev/rD1F04V3ITEM' 1408,
                device '/dev/rD1F04V4ITEM' 1408,
                device '/dev/rD1F04V5ITEM' 1408,
                device '/dev/rD1F04V6ITEM' 1408,
                device '/dev/rD1F05V1ITEM' 1408,
                device '/dev/rD1F05V2ITEM' 1408,
                device '/dev/rD1F05V3ITEM' 1408,
                device '/dev/rD1F05V4ITEM' 1408,
                device '/dev/rD1F05V5ITEM' 1408,
                device '/dev/rD1F05V6ITEM' 1408,
                device '/dev/rD1F06V1ITEM' 1408,
                device '/dev/rD1F06V2ITEM' 1408,
                device '/dev/rD1F06V3ITEM' 1408,
                device '/dev/rD1F06V4ITEM' 1408,
                device '/dev/rD1F06V5ITEM' 1408,
                device '/dev/rD1F06V6ITEM' 1408,
                device '/dev/rD1F07V1ITEM' 1408,
                device '/dev/rD1F07V2ITEM' 1408,
                device '/dev/rD1F07V3ITEM' 1408,
                device '/dev/rD1F07V4ITEM' 1408,
                device '/dev/rD1F07V5ITEM' 1408,
                device '/dev/rD1F07V6ITEM' 1408,
                device '/dev/rD1F08V1ITEM' 1408,
                device '/dev/rD1F08V2ITEM' 1408,
                device '/dev/rD1F08V3ITEM' 1408,
                device '/dev/rD1F08V4ITEM' 1408,
                device '/dev/rD1F08V5ITEM' 1408,
                device '/dev/rD1F08V6ITEM' 1408,
                device '/dev/rD1F09V1ITEM' 1408,
                device '/dev/rD1F09V2ITEM' 1408,
                device '/dev/rD1F09V3ITEM' 1408,
                device '/dev/rD1F09V4ITEM' 1408,
                device '/dev/rD1F09V5ITEM' 1408,
                device '/dev/rD1F09V6ITEM' 1408,
                device '/dev/rD1F10V1ITEM' 1408,
                device '/dev/rD1F10V2ITEM' 1408,
                device '/dev/rD1F10V3ITEM' 1408,
                device '/dev/rD1F10V4ITEM' 1408,
                device '/dev/rD1F10V5ITEM' 1408,
                device '/dev/rD1F10V6ITEM' 1408,
                device '/dev/rD1F11V1ITEM' 1408,
                device '/dev/rD1F11V2ITEM' 1408,
                device '/dev/rD1F11V3ITEM' 1408,
                device '/dev/rD1F11V4ITEM' 1408,
                device '/dev/rD1F11V5ITEM' 1408,
                device '/dev/rD1F11V6ITEM' 1408,
                device '/dev/rD1F12V1ITEM' 1408,
                device '/dev/rD1F12V2ITEM' 1408,
                device '/dev/rD1F12V3ITEM' 1408,
                device '/dev/rD1F12V4ITEM' 1408,
                device '/dev/rD1F12V5ITEM' 1408,
                device '/dev/rD1F12V6ITEM' 1408,
                device '/dev/rD1F13V1ITEM' 1408,
                device '/dev/rD1F13V2ITEM' 1408,
                device '/dev/rD1F13V3ITEM' 1408,
                device '/dev/rD1F13V4ITEM' 1408,
                device '/dev/rD1F13V5ITEM' 1408,
                device '/dev/rD1F13V6ITEM' 1408,
                device '/dev/rD1F14V1ITEM' 1408,
                device '/dev/rD1F14V2ITEM' 1408,
                device '/dev/rD1F14V3ITEM' 1408,
                device '/dev/rD1F14V4ITEM' 1408,
                device '/dev/rD1F14V5ITEM' 1408,
                device '/dev/rD1F14V6ITEM' 1408,
                device '/dev/rD1F15V1ITEM' 1408,
                device '/dev/rD1F15V2ITEM' 1408,
                device '/dev/rD1F15V3ITEM' 1408,
                device '/dev/rD1F15V4ITEM' 1408,
                device '/dev/rD1F15V5ITEM' 1408,
                device '/dev/rD1F15V6ITEM' 1408,
                device '/dev/rD1F16V1ITEM' 1408,
                device '/dev/rD1F16V2ITEM' 1408,
                device '/dev/rD1F16V3ITEM' 1408,
                device '/dev/rD1F16V4ITEM' 1408,
                device '/dev/rD1F16V5ITEM' 1408,
                device '/dev/rD1F16V6ITEM' 1408,
                device '/dev/rD1F17V1ITEM' 1408,
                device '/dev/rD1F17V2ITEM' 1408,
                device '/dev/rD1F17V3ITEM' 1408,
                device '/dev/rD1F17V4ITEM' 1408,
                device '/dev/rD1F17V5ITEM' 1408,
                device '/dev/rD1F17V6ITEM' 1408,
                device '/dev/rD1F18V1ITEM' 1408,
                device '/dev/rD1F18V2ITEM' 1408,
                device '/dev/rD1F18V3ITEM' 1408,
                device '/dev/rD1F18V4ITEM' 1408,
                device '/dev/rD1F18V5ITEM' 1408,
                device '/dev/rD1F18V6ITEM' 1408,
                device '/dev/rD1F19V1ITEM' 1408,
                device '/dev/rD1F19V2ITEM' 1408,
                device '/dev/rD1F19V3ITEM' 1408,
                device '/dev/rD1F19V4ITEM' 1408,
                device '/dev/rD1F19V5ITEM' 1408,
                device '/dev/rD1F19V6ITEM' 1408,
                device '/dev/rD1F20V1ITEM' 1408,
                device '/dev/rD1F20V2ITEM' 1408,
                device '/dev/rD1F20V3ITEM' 1408,
                device '/dev/rD1F20V4ITEM' 1408,
                device '/dev/rD1F20V5ITEM' 1408,
                device '/dev/rD1F20V6ITEM' 1408
                )
        extentsize 16
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

connect reset;
```

ts/crts_neworda.ddl

```
connect to tpcc;
-- now creating TS for ts_neworda_001 of D1

drop tablespace ts_neworda_001;
create regular tablespace ts_neworda_001 pagesize 4K
	managed by database
	using
	(
		device '/dev/rD1F01V1NORA' 201472,
		device '/dev/rD1F01V2NORA' 201472,
		device '/dev/rD1F01V3NORA' 201472
		)
	extentsize 256
	prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_002 of D1

drop tablespace ts_neworda_002;
create regular tablespace ts_neworda_002 pagesize 4K
	managed by database
	using
	(
		device '/dev/rD1F01V4NORA' 201472,
		device '/dev/rD1F01V5NORA' 201472,
		device '/dev/rD1F01V6NORA' 201472
		)
	extentsize 256
	prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_003 of D1

drop tablespace ts_neworda_003;
create regular tablespace ts_neworda_003 pagesize 4K
	managed by database
	using
	(
		device '/dev/rD1F02V1NORA' 201472,
		device '/dev/rD1F02V2NORA' 201472,
		device '/dev/rD1F02V3NORA' 201472
		)
	extentsize 256
	prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_004 of D1

drop tablespace ts_neworda_004;
create regular tablespace ts_neworda_004 pagesize 4K
	managed by database
	using
	(
		device '/dev/rD1F02V4NORA' 201472,
		device '/dev/rD1F02V5NORA' 201472,
		device '/dev/rD1F02V6NORA' 201472
		)
	extentsize 256
	prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_005 of D1

drop tablespace ts_neworda_005;
create regular tablespace ts_neworda_005 pagesize 4K
	managed by database
	using
	(
		device '/dev/rD1F03V1NORA' 201472,
		device '/dev/rD1F03V2NORA' 201472,
		device '/dev/rD1F03V3NORA' 201472
		)
	extentsize 256
	prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_006 of D1

drop tablespace ts_neworda_006;
create regular tablespace ts_neworda_006 pagesize 4K
	managed by database
	using
	(
		device '/dev/rD1F03V4NORA' 201472,
		device '/dev/rD1F03V5NORA' 201472,
		device '/dev/rD1F03V6NORA' 201472
		)
	extentsize 256
	prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_007 of D1

drop tablespace ts_neworda_007;
create regular tablespace ts_neworda_007 pagesize 4K
	managed by database
	using
	(
		device '/dev/rD1F04V1NORA' 201472,
		device '/dev/rD1F04V2NORA' 201472,
		device '/dev/rD1F04V3NORA' 201472
		)
	extentsize 256
	prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_008 of D1

drop tablespace ts_neworda_008;
create regular tablespace ts_neworda_008 pagesize 4K
	managed by database
	using
	(
		device '/dev/rD1F04V4NORA' 201472,
		device '/dev/rD1F04V5NORA' 201472,
		device '/dev/rD1F04V6NORA' 201472
		)
	extentsize 256
	prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_009 of D1

drop tablespace ts_neworda_009;
create regular tablespace ts_neworda_009 pagesize 4K
	managed by database
	using
	(
		device '/dev/rD1F05V1NORA' 201472,
		device '/dev/rD1F05V2NORA' 201472,
		device '/dev/rD1F05V3NORA' 201472
		)
	extentsize 256
	prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_010 of D1

drop tablespace ts_neworda_010;
create regular tablespace ts_neworda_010 pagesize 4K
	managed by database
	using
	(
		device '/dev/rD1F05V4NORA' 201472,
		device '/dev/rD1F05V5NORA' 201472,
		device '/dev/rD1F05V6NORA' 201472
		)
	extentsize 256
	prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_011 of D1

drop tablespace ts_neworda_011;
create regular tablespace ts_neworda_011 pagesize 4K
	managed by database
	using
	(
		device '/dev/rD1F06V1NORA' 201472,
		device '/dev/rD1F06V2NORA' 201472,
		device '/dev/rD1F06V3NORA' 201472
		)
	extentsize 256
	prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_012 of D1

drop tablespace ts_neworda_012;
create regular tablespace ts_neworda_012 pagesize 4K
	managed by database
	using
	(
		device '/dev/rD1F06V4NORA' 201472,
		device '/dev/rD1F06V5NORA' 201472,
		device '/dev/rD1F06V6NORA' 201472
		)
	extentsize 256
	prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_013 of D1

drop tablespace ts_neworda_013;
create regular tablespace ts_neworda_013 pagesize 4K
	managed by database
	using
	(
		device '/dev/rD1F07V1NORA' 201472,
		device '/dev/rD1F07V2NORA' 201472,
		device '/dev/rD1F07V3NORA' 201472
		)
	extentsize 256
	prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_014 of D1

drop tablespace ts_neworda_014;
create regular tablespace ts_neworda_014 pagesize 4K
	managed by database
	using
	(
		device '/dev/rD1F07V4NORA' 201472,
		device '/dev/rD1F07V5NORA' 201472,
		device '/dev/rD1F07V6NORA' 201472
		)
	extentsize 256
	prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_015 of D1

drop tablespace ts_neworda_015;
create regular tablespace ts_neworda_015 pagesize 4K
	managed by database
```

```
 using
 (
        device '/dev/rD1F08V1NORA' 201472,
        device '/dev/rD1F08V2NORA' 201472,
        device '/dev/rD1F08V3NORA' 201472
       )
 extentsize 256
 prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_016 of D1

drop tablespace ts_neworda_016;
create regular tablespace ts_neworda_016 pagesize 4K
 managed by database
 using
 (
        device '/dev/rD1F08V4NORA' 201472,
        device '/dev/rD1F08V5NORA' 201472,
        device '/dev/rD1F08V6NORA' 201472
       )
 extentsize 256
 prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_017 of D1

drop tablespace ts_neworda_017;
create regular tablespace ts_neworda_017 pagesize 4K
 managed by database
 using
 (
        device '/dev/rD1F09V1NORA' 201472,
        device '/dev/rD1F09V2NORA' 201472,
        device '/dev/rD1F09V3NORA' 201472
       )
 extentsize 256
 prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_018 of D1

drop tablespace ts_neworda_018;
create regular tablespace ts_neworda_018 pagesize 4K
 managed by database
 using
 (
        device '/dev/rD1F09V4NORA' 201472,
        device '/dev/rD1F09V5NORA' 201472,
        device '/dev/rD1F09V6NORA' 201472
       )
 extentsize 256
 prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_019 of D1

drop tablespace ts_neworda_019;
create regular tablespace ts_neworda_019 pagesize 4K
 managed by database
 using
 (
        device '/dev/rD1F10V1NORA' 201472,
        device '/dev/rD1F10V2NORA' 201472,
        device '/dev/rD1F10V3NORA' 201472
       )
 extentsize 256
 prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_020 of D1

drop tablespace ts_neworda_020;
create regular tablespace ts_neworda_020 pagesize 4K
 managed by database
 using
 (
        device '/dev/rD1F10V4NORA' 201472,
        device '/dev/rD1F10V5NORA' 201472,
        device '/dev/rD1F10V6NORA' 201472
       )
 extentsize 256
 prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_021 of D1

drop tablespace ts_neworda_021;
create regular tablespace ts_neworda_021 pagesize 4K
 managed by database
 using
 (
        device '/dev/rD1F11V1NORA' 201472,
        device '/dev/rD1F11V2NORA' 201472,
        device '/dev/rD1F11V3NORA' 201472
       )
 extentsize 256
 prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_022 of D1

drop tablespace ts_neworda_022;
create regular tablespace ts_neworda_022 pagesize 4K
 managed by database
 using
 (
        device '/dev/rD1F11V4NORA' 201472,
        device '/dev/rD1F11V5NORA' 201472,
        device '/dev/rD1F11V6NORA' 201472
       )
 extentsize 256
 prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_023 of D1

drop tablespace ts_neworda_023;
create regular tablespace ts_neworda_023 pagesize 4K
 managed by database
 using
 (
        device '/dev/rD1F12V1NORA' 201472,
        device '/dev/rD1F12V2NORA' 201472,
        device '/dev/rD1F12V3NORA' 201472
       )
 extentsize 256
 prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_024 of D1

drop tablespace ts_neworda_024;
create regular tablespace ts_neworda_024 pagesize 4K
 managed by database
 using
 (
        device '/dev/rD1F12V4NORA' 201472,
        device '/dev/rD1F12V5NORA' 201472,
        device '/dev/rD1F12V6NORA' 201472
       )
 extentsize 256
 prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_025 of D1

drop tablespace ts_neworda_025;
create regular tablespace ts_neworda_025 pagesize 4K
 managed by database
 using
 (
        device '/dev/rD1F13V1NORA' 201472,
        device '/dev/rD1F13V2NORA' 201472,
        device '/dev/rD1F13V3NORA' 201472
       )
 extentsize 256
 prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_026 of D1

drop tablespace ts_neworda_026;
create regular tablespace ts_neworda_026 pagesize 4K
 managed by database
 using
 (
        device '/dev/rD1F13V4NORA' 201472,
        device '/dev/rD1F13V5NORA' 201472,
        device '/dev/rD1F13V6NORA' 201472
       )
 extentsize 256
 prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_027 of D1

drop tablespace ts_neworda_027;
create regular tablespace ts_neworda_027 pagesize 4K
 managed by database
 using
 (
        device '/dev/rD1F14V1NORA' 201472,
        device '/dev/rD1F14V2NORA' 201472,
        device '/dev/rD1F14V3NORA' 201472
       )
 extentsize 256
 prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_028 of D1

drop tablespace ts_neworda_028;
create regular tablespace ts_neworda_028 pagesize 4K
 managed by database
 using
 (
        device '/dev/rD1F14V4NORA' 201472,
        device '/dev/rD1F14V5NORA' 201472,
        device '/dev/rD1F14V6NORA' 201472
       )
 extentsize 256
 prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_029 of D1

drop tablespace ts_neworda_029;
create regular tablespace ts_neworda_029 pagesize 4K
 managed by database
 using
 (
        device '/dev/rD1F15V1NORA' 201472,
        device '/dev/rD1F15V2NORA' 201472,
        device '/dev/rD1F15V3NORA' 201472
       )
 extentsize 256
 prefetchsize 4096;
commit;
```

```
-- now creating TS for ts_neworda_030 of D1

drop tablespace ts_neworda_030;
create regular tablespace ts_neworda_030 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F15V4NORA' 201472,
                device '/dev/rD1F15V5NORA' 201472,
                device '/dev/rD1F15V6NORA' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_031 of D1

drop tablespace ts_neworda_031;
create regular tablespace ts_neworda_031 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F16V1NORA' 201472,
                device '/dev/rD1F16V2NORA' 201472,
                device '/dev/rD1F16V3NORA' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_032 of D1

drop tablespace ts_neworda_032;
create regular tablespace ts_neworda_032 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F16V4NORA' 201472,
                device '/dev/rD1F16V5NORA' 201472,
                device '/dev/rD1F16V6NORA' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_033 of D1

drop tablespace ts_neworda_033;
create regular tablespace ts_neworda_033 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F17V1NORA' 201472,
                device '/dev/rD1F17V2NORA' 201472,
                device '/dev/rD1F17V3NORA' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_034 of D1

drop tablespace ts_neworda_034;
create regular tablespace ts_neworda_034 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F17V4NORA' 201472,
                device '/dev/rD1F17V5NORA' 201472,
                device '/dev/rD1F17V6NORA' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_035 of D1

drop tablespace ts_neworda_035;
create regular tablespace ts_neworda_035 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F18V1NORA' 201472,
                device '/dev/rD1F18V2NORA' 201472,
                device '/dev/rD1F18V3NORA' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_036 of D1

drop tablespace ts_neworda_036;
create regular tablespace ts_neworda_036 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F18V4NORA' 201472,
                device '/dev/rD1F18V5NORA' 201472,
                device '/dev/rD1F18V6NORA' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_037 of D1

drop tablespace ts_neworda_037;
create regular tablespace ts_neworda_037 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F19V1NORA' 201472,
                device '/dev/rD1F19V2NORA' 201472,
                device '/dev/rD1F19V3NORA' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_038 of D1

drop tablespace ts_neworda_038;
create regular tablespace ts_neworda_038 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F19V4NORA' 201472,
                device '/dev/rD1F19V5NORA' 201472,
                device '/dev/rD1F19V6NORA' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_039 of D1

drop tablespace ts_neworda_039;
create regular tablespace ts_neworda_039 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F20V1NORA' 201472,
                device '/dev/rD1F20V2NORA' 201472,
                device '/dev/rD1F20V3NORA' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_neworda_040 of D1

drop tablespace ts_neworda_040;
create regular tablespace ts_neworda_040 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F20V4NORA' 201472,
                device '/dev/rD1F20V5NORA' 201472,
                device '/dev/rD1F20V6NORA' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

connect reset;
```

ts/crts_newordb.ddl

```
connect to tpcc;
-- now creating TS for ts_newordb_001 of D1

drop tablespace ts_newordb_001;
create regular tablespace ts_newordb_001 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F01V1NORB' 201472,
                device '/dev/rD1F01V2NORB' 201472,
                device '/dev/rD1F01V3NORB' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_002 of D1

drop tablespace ts_newordb_002;
create regular tablespace ts_newordb_002 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F01V4NORB' 201472,
                device '/dev/rD1F01V5NORB' 201472,
                device '/dev/rD1F01V6NORB' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_003 of D1

drop tablespace ts_newordb_003;
create regular tablespace ts_newordb_003 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F02V1NORB' 201472,
                device '/dev/rD1F02V2NORB' 201472,
                device '/dev/rD1F02V3NORB' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;
```

```
-- now creating TS for ts_newordb_004 of D1

drop tablespace ts_newordb_004;
create regular tablespace ts_newordb_004 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F02V4NORB' 201472,
                device '/dev/rD1F02V5NORB' 201472,
                device '/dev/rD1F02V6NORB' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_005 of D1

drop tablespace ts_newordb_005;
create regular tablespace ts_newordb_005 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F03V1NORB' 201472,
                device '/dev/rD1F03V2NORB' 201472,
                device '/dev/rD1F03V3NORB' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_006 of D1

drop tablespace ts_newordb_006;
create regular tablespace ts_newordb_006 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F03V4NORB' 201472,
                device '/dev/rD1F03V5NORB' 201472,
                device '/dev/rD1F03V6NORB' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_007 of D1

drop tablespace ts_newordb_007;
create regular tablespace ts_newordb_007 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F04V1NORB' 201472,
                device '/dev/rD1F04V2NORB' 201472,
                device '/dev/rD1F04V3NORB' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_008 of D1

drop tablespace ts_newordb_008;
create regular tablespace ts_newordb_008 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F04V4NORB' 201472,
                device '/dev/rD1F04V5NORB' 201472,
                device '/dev/rD1F04V6NORB' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_009 of D1

drop tablespace ts_newordb_009;
create regular tablespace ts_newordb_009 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F05V1NORB' 201472,
                device '/dev/rD1F05V2NORB' 201472,
                device '/dev/rD1F05V3NORB' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_010 of D1

drop tablespace ts_newordb_010;
create regular tablespace ts_newordb_010 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F05V4NORB' 201472,
                device '/dev/rD1F05V5NORB' 201472,
                device '/dev/rD1F05V6NORB' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_011 of D1

drop tablespace ts_newordb_011;
create regular tablespace ts_newordb_011 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F06V1NORB' 201472,
                device '/dev/rD1F06V2NORB' 201472,
                device '/dev/rD1F06V3NORB' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_012 of D1

drop tablespace ts_newordb_012;
create regular tablespace ts_newordb_012 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F06V4NORB' 201472,
                device '/dev/rD1F06V5NORB' 201472,
                device '/dev/rD1F06V6NORB' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_013 of D1

drop tablespace ts_newordb_013;
create regular tablespace ts_newordb_013 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F07V1NORB' 201472,
                device '/dev/rD1F07V2NORB' 201472,
                device '/dev/rD1F07V3NORB' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_014 of D1

drop tablespace ts_newordb_014;
create regular tablespace ts_newordb_014 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F07V4NORB' 201472,
                device '/dev/rD1F07V5NORB' 201472,
                device '/dev/rD1F07V6NORB' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_015 of D1

drop tablespace ts_newordb_015;
create regular tablespace ts_newordb_015 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F08V1NORB' 201472,
                device '/dev/rD1F08V2NORB' 201472,
                device '/dev/rD1F08V3NORB' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_016 of D1

drop tablespace ts_newordb_016;
create regular tablespace ts_newordb_016 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F08V4NORB' 201472,
                device '/dev/rD1F08V5NORB' 201472,
                device '/dev/rD1F08V6NORB' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_017 of D1

drop tablespace ts_newordb_017;
create regular tablespace ts_newordb_017 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F09V1NORB' 201472,
                device '/dev/rD1F09V2NORB' 201472,
                device '/dev/rD1F09V3NORB' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_018 of D1

drop tablespace ts_newordb_018;
create regular tablespace ts_newordb_018 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F09V4NORB' 201472,
                device '/dev/rD1F09V5NORB' 201472,
```

```
                    device '/dev/rD1F09V6NORB' 201472
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_019 of D1

drop tablespace ts_newordb_019;
create regular tablespace ts_newordb_019 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F10V1NORB' 201472,
                    device '/dev/rD1F10V2NORB' 201472,
                    device '/dev/rD1F10V3NORB' 201472
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_020 of D1

drop tablespace ts_newordb_020;
create regular tablespace ts_newordb_020 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F10V4NORB' 201472,
                    device '/dev/rD1F10V5NORB' 201472,
                    device '/dev/rD1F10V6NORB' 201472
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_021 of D1

drop tablespace ts_newordb_021;
create regular tablespace ts_newordb_021 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F11V1NORB' 201472,
                    device '/dev/rD1F11V2NORB' 201472,
                    device '/dev/rD1F11V3NORB' 201472
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_022 of D1

drop tablespace ts_newordb_022;
create regular tablespace ts_newordb_022 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F11V4NORB' 201472,
                    device '/dev/rD1F11V5NORB' 201472,
                    device '/dev/rD1F11V6NORB' 201472
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_023 of D1

drop tablespace ts_newordb_023;
create regular tablespace ts_newordb_023 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F12V1NORB' 201472,
                    device '/dev/rD1F12V2NORB' 201472,
                    device '/dev/rD1F12V3NORB' 201472
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_024 of D1

drop tablespace ts_newordb_024;
create regular tablespace ts_newordb_024 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F12V4NORB' 201472,
                    device '/dev/rD1F12V5NORB' 201472,
                    device '/dev/rD1F12V6NORB' 201472
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_025 of D1

drop tablespace ts_newordb_025;
create regular tablespace ts_newordb_025 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F13V1NORB' 201472,
                    device '/dev/rD1F13V2NORB' 201472,
                    device '/dev/rD1F13V3NORB' 201472
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_026 of D1

drop tablespace ts_newordb_026;
create regular tablespace ts_newordb_026 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F13V4NORB' 201472,
                    device '/dev/rD1F13V5NORB' 201472,
                    device '/dev/rD1F13V6NORB' 201472
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_027 of D1

drop tablespace ts_newordb_027;
create regular tablespace ts_newordb_027 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F14V1NORB' 201472,
                    device '/dev/rD1F14V2NORB' 201472,
                    device '/dev/rD1F14V3NORB' 201472
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_028 of D1

drop tablespace ts_newordb_028;
create regular tablespace ts_newordb_028 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F14V4NORB' 201472,
                    device '/dev/rD1F14V5NORB' 201472,
                    device '/dev/rD1F14V6NORB' 201472
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_029 of D1

drop tablespace ts_newordb_029;
create regular tablespace ts_newordb_029 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F15V1NORB' 201472,
                    device '/dev/rD1F15V2NORB' 201472,
                    device '/dev/rD1F15V3NORB' 201472
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_030 of D1

drop tablespace ts_newordb_030;
create regular tablespace ts_newordb_030 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F15V4NORB' 201472,
                    device '/dev/rD1F15V5NORB' 201472,
                    device '/dev/rD1F15V6NORB' 201472
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_031 of D1

drop tablespace ts_newordb_031;
create regular tablespace ts_newordb_031 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F16V1NORB' 201472,
                    device '/dev/rD1F16V2NORB' 201472,
                    device '/dev/rD1F16V3NORB' 201472
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_032 of D1

drop tablespace ts_newordb_032;
create regular tablespace ts_newordb_032 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F16V4NORB' 201472,
                    device '/dev/rD1F16V5NORB' 201472,
                    device '/dev/rD1F16V6NORB' 201472
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_033 of D1

drop tablespace ts_newordb_033;
```

```
create regular tablespace ts_newordb_033 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F17V1NORB' 201472,
                device '/dev/rD1F17V2NORB' 201472,
                device '/dev/rD1F17V3NORB' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_034 of D1

drop tablespace ts_newordb_034;
create regular tablespace ts_newordb_034 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F17V4NORB' 201472,
                device '/dev/rD1F17V5NORB' 201472,
                device '/dev/rD1F17V6NORB' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_035 of D1

drop tablespace ts_newordb_035;
create regular tablespace ts_newordb_035 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F18V1NORB' 201472,
                device '/dev/rD1F18V2NORB' 201472,
                device '/dev/rD1F18V3NORB' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_036 of D1

drop tablespace ts_newordb_036;
create regular tablespace ts_newordb_036 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F18V4NORB' 201472,
                device '/dev/rD1F18V5NORB' 201472,
                device '/dev/rD1F18V6NORB' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_037 of D1

drop tablespace ts_newordb_037;
create regular tablespace ts_newordb_037 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F19V1NORB' 201472,
                device '/dev/rD1F19V2NORB' 201472,
                device '/dev/rD1F19V3NORB' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_038 of D1

drop tablespace ts_newordb_038;
create regular tablespace ts_newordb_038 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F19V4NORB' 201472,
                device '/dev/rD1F19V5NORB' 201472,
                device '/dev/rD1F19V6NORB' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_039 of D1

drop tablespace ts_newordb_039;
create regular tablespace ts_newordb_039 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F20V1NORB' 201472,
                device '/dev/rD1F20V2NORB' 201472,
                device '/dev/rD1F20V3NORB' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_newordb_040 of D1

drop tablespace ts_newordb_040;
create regular tablespace ts_newordb_040 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F20V4NORB' 201472,
                device '/dev/rD1F20V5NORB' 201472,
                device '/dev/rD1F20V6NORB' 201472
                )
        extentsize 256
        prefetchsize 4096;
commit;

connect reset;
```

ts/crts_order.ddl

```
connect to tpcc;
-- now creating TS for ts_order_001 of D1

drop tablespace ts_order_001;
create regular tablespace ts_order_001 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F01V1ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_002 of D1

drop tablespace ts_order_002;
create regular tablespace ts_order_002 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F01V2ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_003 of D1

drop tablespace ts_order_003;
create regular tablespace ts_order_003 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F01V3ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_004 of D1

drop tablespace ts_order_004;
create regular tablespace ts_order_004 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F01V4ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_005 of D1

drop tablespace ts_order_005;
create regular tablespace ts_order_005 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F01V5ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_006 of D1

drop tablespace ts_order_006;
create regular tablespace ts_order_006 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F01V6ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_007 of D1

drop tablespace ts_order_007;
create regular tablespace ts_order_007 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F02V1ORD' 205312
                )
        extentsize 256
```

```
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_008 of D1

drop tablespace ts_order_008;
create regular tablespace ts_order_008 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F02V2ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_009 of D1

drop tablespace ts_order_009;
create regular tablespace ts_order_009 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F02V3ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_010 of D1

drop tablespace ts_order_010;
create regular tablespace ts_order_010 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F02V4ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_011 of D1

drop tablespace ts_order_011;
create regular tablespace ts_order_011 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F02V5ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_012 of D1

drop tablespace ts_order_012;
create regular tablespace ts_order_012 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F02V6ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_013 of D1

drop tablespace ts_order_013;
create regular tablespace ts_order_013 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F03V1ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_014 of D1

drop tablespace ts_order_014;
create regular tablespace ts_order_014 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F03V2ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_015 of D1

drop tablespace ts_order_015;
create regular tablespace ts_order_015 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F03V3ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_016 of D1

drop tablespace ts_order_016;
create regular tablespace ts_order_016 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F03V4ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_017 of D1

drop tablespace ts_order_017;
create regular tablespace ts_order_017 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F03V5ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_018 of D1

drop tablespace ts_order_018;
create regular tablespace ts_order_018 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F03V6ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_019 of D1

drop tablespace ts_order_019;
create regular tablespace ts_order_019 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F04V1ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_020 of D1

drop tablespace ts_order_020;
create regular tablespace ts_order_020 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F04V2ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_021 of D1

drop tablespace ts_order_021;
create regular tablespace ts_order_021 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F04V3ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_022 of D1

drop tablespace ts_order_022;
create regular tablespace ts_order_022 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F04V4ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_023 of D1

drop tablespace ts_order_023;
create regular tablespace ts_order_023 pagesize 8K
         managed by database
```

```
            using
            (
                        device '/dev/rD1F04V5ORD' 205312
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_024 of D1

drop tablespace ts_order_024;
create regular tablespace ts_order_024 pagesize 8K
            managed by database
            using
            (
                        device '/dev/rD1F04V6ORD' 205312
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_025 of D1

drop tablespace ts_order_025;
create regular tablespace ts_order_025 pagesize 8K
            managed by database
            using
            (
                        device '/dev/rD1F05V1ORD' 205312
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_026 of D1

drop tablespace ts_order_026;
create regular tablespace ts_order_026 pagesize 8K
            managed by database
            using
            (
                        device '/dev/rD1F05V2ORD' 205312
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_027 of D1

drop tablespace ts_order_027;
create regular tablespace ts_order_027 pagesize 8K
            managed by database
            using
            (
                        device '/dev/rD1F05V3ORD' 205312
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_028 of D1

drop tablespace ts_order_028;
create regular tablespace ts_order_028 pagesize 8K
            managed by database
            using
            (
                        device '/dev/rD1F05V4ORD' 205312
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_029 of D1

drop tablespace ts_order_029;
create regular tablespace ts_order_029 pagesize 8K
            managed by database
            using
            (
                        device '/dev/rD1F05V5ORD' 205312
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_030 of D1

drop tablespace ts_order_030;
create regular tablespace ts_order_030 pagesize 8K
            managed by database
            using
            (
                        device '/dev/rD1F05V6ORD' 205312
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_031 of D1

drop tablespace ts_order_031;
create regular tablespace ts_order_031 pagesize 8K
            managed by database
            using
            (
                        device '/dev/rD1F06V1ORD' 205312
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_032 of D1

drop tablespace ts_order_032;
create regular tablespace ts_order_032 pagesize 8K
            managed by database
            using
            (
                        device '/dev/rD1F06V2ORD' 205312
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_033 of D1

drop tablespace ts_order_033;
create regular tablespace ts_order_033 pagesize 8K
            managed by database
            using
            (
                        device '/dev/rD1F06V3ORD' 205312
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_034 of D1

drop tablespace ts_order_034;
create regular tablespace ts_order_034 pagesize 8K
            managed by database
            using
            (
                        device '/dev/rD1F06V4ORD' 205312
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_035 of D1

drop tablespace ts_order_035;
create regular tablespace ts_order_035 pagesize 8K
            managed by database
            using
            (
                        device '/dev/rD1F06V5ORD' 205312
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_036 of D1

drop tablespace ts_order_036;
create regular tablespace ts_order_036 pagesize 8K
            managed by database
            using
            (
                        device '/dev/rD1F06V6ORD' 205312
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_037 of D1

drop tablespace ts_order_037;
create regular tablespace ts_order_037 pagesize 8K
            managed by database
            using
            (
                        device '/dev/rD1F07V1ORD' 205312
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_038 of D1

drop tablespace ts_order_038;
create regular tablespace ts_order_038 pagesize 8K
            managed by database
            using
            (
                        device '/dev/rD1F07V2ORD' 205312
                        )
            extentsize 256
            prefetchsize 4096
            bufferpool ibmdefaultbp8K;
commit;
```

```
-- now creating TS for ts_order_039 of D1

drop tablespace ts_order_039;
create regular tablespace ts_order_039 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F07V3ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_040 of D1

drop tablespace ts_order_040;
create regular tablespace ts_order_040 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F07V4ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_041 of D1

drop tablespace ts_order_041;
create regular tablespace ts_order_041 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F07V5ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_042 of D1

drop tablespace ts_order_042;
create regular tablespace ts_order_042 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F07V6ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_043 of D1

drop tablespace ts_order_043;
create regular tablespace ts_order_043 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F08V1ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_044 of D1

drop tablespace ts_order_044;
create regular tablespace ts_order_044 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F08V2ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_045 of D1

drop tablespace ts_order_045;
create regular tablespace ts_order_045 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F08V3ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_046 of D1

drop tablespace ts_order_046;
create regular tablespace ts_order_046 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F08V4ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_047 of D1

drop tablespace ts_order_047;
create regular tablespace ts_order_047 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F08V5ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_048 of D1

drop tablespace ts_order_048;
create regular tablespace ts_order_048 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F08V6ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_049 of D1

drop tablespace ts_order_049;
create regular tablespace ts_order_049 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F09V1ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_050 of D1

drop tablespace ts_order_050;
create regular tablespace ts_order_050 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F09V2ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_051 of D1

drop tablespace ts_order_051;
create regular tablespace ts_order_051 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F09V3ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_052 of D1

drop tablespace ts_order_052;
create regular tablespace ts_order_052 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F09V4ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_053 of D1

drop tablespace ts_order_053;
create regular tablespace ts_order_053 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F09V5ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_054 of D1

drop tablespace ts_order_054;
create regular tablespace ts_order_054 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F09V6ORD' 205312
                )
```

```
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_055 of D1

drop tablespace ts_order_055;
create regular tablespace ts_order_055 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F10V1ORD' 205312
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_056 of D1

drop tablespace ts_order_056;
create regular tablespace ts_order_056 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F10V2ORD' 205312
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_057 of D1

drop tablespace ts_order_057;
create regular tablespace ts_order_057 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F10V3ORD' 205312
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_058 of D1

drop tablespace ts_order_058;
create regular tablespace ts_order_058 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F10V4ORD' 205312
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_059 of D1

drop tablespace ts_order_059;
create regular tablespace ts_order_059 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F10V5ORD' 205312
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_060 of D1

drop tablespace ts_order_060;
create regular tablespace ts_order_060 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F10V6ORD' 205312
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_061 of D1

drop tablespace ts_order_061;
create regular tablespace ts_order_061 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F11V1ORD' 205312
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_062 of D1

drop tablespace ts_order_062;
create regular tablespace ts_order_062 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F11V2ORD' 205312
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_063 of D1

drop tablespace ts_order_063;
create regular tablespace ts_order_063 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F11V3ORD' 205312
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_064 of D1

drop tablespace ts_order_064;
create regular tablespace ts_order_064 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F11V4ORD' 205312
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_065 of D1

drop tablespace ts_order_065;
create regular tablespace ts_order_065 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F11V5ORD' 205312
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_066 of D1

drop tablespace ts_order_066;
create regular tablespace ts_order_066 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F11V6ORD' 205312
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_067 of D1

drop tablespace ts_order_067;
create regular tablespace ts_order_067 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F12V1ORD' 205312
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_068 of D1

drop tablespace ts_order_068;
create regular tablespace ts_order_068 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F12V2ORD' 205312
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_069 of D1

drop tablespace ts_order_069;
create regular tablespace ts_order_069 pagesize 8K
         managed by database
         using
         (
                 device '/dev/rD1F12V3ORD' 205312
                 )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_070 of D1

drop tablespace ts_order_070;
create regular tablespace ts_order_070 pagesize 8K
```

```
        managed by database
        using
        (
                device '/dev/rD1F12V4ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_071 of D1

drop tablespace ts_order_071;
create regular tablespace ts_order_071 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F12V5ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_072 of D1

drop tablespace ts_order_072;
create regular tablespace ts_order_072 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F12V6ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_073 of D1

drop tablespace ts_order_073;
create regular tablespace ts_order_073 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F13V1ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_074 of D1

drop tablespace ts_order_074;
create regular tablespace ts_order_074 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F13V2ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_075 of D1

drop tablespace ts_order_075;
create regular tablespace ts_order_075 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F13V3ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_076 of D1

drop tablespace ts_order_076;
create regular tablespace ts_order_076 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F13V4ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_077 of D1

drop tablespace ts_order_077;
create regular tablespace ts_order_077 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F13V5ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_078 of D1

drop tablespace ts_order_078;
create regular tablespace ts_order_078 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F13V6ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_079 of D1

drop tablespace ts_order_079;
create regular tablespace ts_order_079 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F14V1ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_080 of D1

drop tablespace ts_order_080;
create regular tablespace ts_order_080 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F14V2ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_081 of D1

drop tablespace ts_order_081;
create regular tablespace ts_order_081 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F14V3ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_082 of D1

drop tablespace ts_order_082;
create regular tablespace ts_order_082 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F14V4ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_083 of D1

drop tablespace ts_order_083;
create regular tablespace ts_order_083 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F14V5ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_084 of D1

drop tablespace ts_order_084;
create regular tablespace ts_order_084 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F14V6ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_085 of D1

drop tablespace ts_order_085;
create regular tablespace ts_order_085 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F15V1ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;
```

```
-- now creating TS for ts_order_086 of D1

drop tablespace ts_order_086;
create regular tablespace ts_order_086 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F15V2ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_087 of D1

drop tablespace ts_order_087;
create regular tablespace ts_order_087 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F15V3ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_088 of D1

drop tablespace ts_order_088;
create regular tablespace ts_order_088 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F15V4ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_089 of D1

drop tablespace ts_order_089;
create regular tablespace ts_order_089 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F15V5ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_090 of D1

drop tablespace ts_order_090;
create regular tablespace ts_order_090 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F15V6ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_091 of D1

drop tablespace ts_order_091;
create regular tablespace ts_order_091 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F16V1ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_092 of D1

drop tablespace ts_order_092;
create regular tablespace ts_order_092 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F16V2ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_093 of D1

drop tablespace ts_order_093;
create regular tablespace ts_order_093 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F16V3ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_094 of D1

drop tablespace ts_order_094;
create regular tablespace ts_order_094 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F16V4ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_095 of D1

drop tablespace ts_order_095;
create regular tablespace ts_order_095 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F16V5ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_096 of D1

drop tablespace ts_order_096;
create regular tablespace ts_order_096 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F16V6ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_097 of D1

drop tablespace ts_order_097;
create regular tablespace ts_order_097 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F17V1ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_098 of D1

drop tablespace ts_order_098;
create regular tablespace ts_order_098 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F17V2ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_099 of D1

drop tablespace ts_order_099;
create regular tablespace ts_order_099 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F17V3ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_100 of D1

drop tablespace ts_order_100;
create regular tablespace ts_order_100 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F17V4ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_101 of D1

drop tablespace ts_order_101;
create regular tablespace ts_order_101 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F17V5ORD' 205312
```

```
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_102 of D1

drop tablespace ts_order_102;
create regular tablespace ts_order_102 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F17V6ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_103 of D1

drop tablespace ts_order_103;
create regular tablespace ts_order_103 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F18V1ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_104 of D1

drop tablespace ts_order_104;
create regular tablespace ts_order_104 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F18V2ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_105 of D1

drop tablespace ts_order_105;
create regular tablespace ts_order_105 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F18V3ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_106 of D1

drop tablespace ts_order_106;
create regular tablespace ts_order_106 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F18V4ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_107 of D1

drop tablespace ts_order_107;
create regular tablespace ts_order_107 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F18V5ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_108 of D1

drop tablespace ts_order_108;
create regular tablespace ts_order_108 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F18V6ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_109 of D1

drop tablespace ts_order_109;
create regular tablespace ts_order_109 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F19V1ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_110 of D1

drop tablespace ts_order_110;
create regular tablespace ts_order_110 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F19V2ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_111 of D1

drop tablespace ts_order_111;
create regular tablespace ts_order_111 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F19V3ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_112 of D1

drop tablespace ts_order_112;
create regular tablespace ts_order_112 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F19V4ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_113 of D1

drop tablespace ts_order_113;
create regular tablespace ts_order_113 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F19V5ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_114 of D1

drop tablespace ts_order_114;
create regular tablespace ts_order_114 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F19V6ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_115 of D1

drop tablespace ts_order_115;
create regular tablespace ts_order_115 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F20V1ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_116 of D1

drop tablespace ts_order_116;
create regular tablespace ts_order_116 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F20V2ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_117 of D1

drop tablespace ts_order_117;
```

```
create regular tablespace ts_order_117 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F20V3ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_118 of D1

drop tablespace ts_order_118;
create regular tablespace ts_order_118 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F20V4ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_119 of D1

drop tablespace ts_order_119;
create regular tablespace ts_order_119 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F20V5ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_order_120 of D1

drop tablespace ts_order_120;
create regular tablespace ts_order_120 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F20V6ORD' 205312
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

connect reset;
```

ts/crts_orderline.ddl

```
connect to tpcc;
-- now creating TS for ts_orderline_001 of D1

drop tablespace ts_orderline_001;
create regular tablespace ts_orderline_001 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F01V1ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_002 of D1

drop tablespace ts_orderline_002;
create regular tablespace ts_orderline_002 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F01V2ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_003 of D1

drop tablespace ts_orderline_003;
create regular tablespace ts_orderline_003 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F01V3ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_004 of D1

drop tablespace ts_orderline_004;
create regular tablespace ts_orderline_004 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F01V4ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_005 of D1

drop tablespace ts_orderline_005;
create regular tablespace ts_orderline_005 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F01V5ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_006 of D1

drop tablespace ts_orderline_006;
create regular tablespace ts_orderline_006 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F01V6ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_007 of D1

drop tablespace ts_orderline_007;
create regular tablespace ts_orderline_007 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F02V1ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_008 of D1

drop tablespace ts_orderline_008;
create regular tablespace ts_orderline_008 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F02V2ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_009 of D1

drop tablespace ts_orderline_009;
create regular tablespace ts_orderline_009 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F02V3ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_010 of D1

drop tablespace ts_orderline_010;
create regular tablespace ts_orderline_010 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F02V4ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_011 of D1

drop tablespace ts_orderline_011;
create regular tablespace ts_orderline_011 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F02V5ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_012 of D1

drop tablespace ts_orderline_012;
create regular tablespace ts_orderline_012 pagesize 8K
```

```
        managed by database
        using
        (
                device '/dev/rD1F02V6ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_013 of D1

drop tablespace ts_orderline_013;
create regular tablespace ts_orderline_013 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F03V1ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_014 of D1

drop tablespace ts_orderline_014;
create regular tablespace ts_orderline_014 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F03V2ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_015 of D1

drop tablespace ts_orderline_015;
create regular tablespace ts_orderline_015 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F03V3ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_016 of D1

drop tablespace ts_orderline_016;
create regular tablespace ts_orderline_016 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F03V4ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_017 of D1

drop tablespace ts_orderline_017;
create regular tablespace ts_orderline_017 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F03V5ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_018 of D1

drop tablespace ts_orderline_018;
create regular tablespace ts_orderline_018 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F03V6ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_019 of D1

drop tablespace ts_orderline_019;
create regular tablespace ts_orderline_019 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F04V1ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_020 of D1

drop tablespace ts_orderline_020;
create regular tablespace ts_orderline_020 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F04V2ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_021 of D1

drop tablespace ts_orderline_021;
create regular tablespace ts_orderline_021 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F04V3ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_022 of D1

drop tablespace ts_orderline_022;
create regular tablespace ts_orderline_022 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F04V4ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_023 of D1

drop tablespace ts_orderline_023;
create regular tablespace ts_orderline_023 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F04V5ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_024 of D1

drop tablespace ts_orderline_024;
create regular tablespace ts_orderline_024 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F04V6ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_025 of D1

drop tablespace ts_orderline_025;
create regular tablespace ts_orderline_025 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F05V1ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_026 of D1

drop tablespace ts_orderline_026;
create regular tablespace ts_orderline_026 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F05V2ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_027 of D1

drop tablespace ts_orderline_027;
create regular tablespace ts_orderline_027 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F05V3ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;
```

```
-- now creating TS for ts_orderline_028 of D1

drop tablespace ts_orderline_028;
create regular tablespace ts_orderline_028 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F05V4ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_029 of D1

drop tablespace ts_orderline_029;
create regular tablespace ts_orderline_029 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F05V5ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_030 of D1

drop tablespace ts_orderline_030;
create regular tablespace ts_orderline_030 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F05V6ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_031 of D1

drop tablespace ts_orderline_031;
create regular tablespace ts_orderline_031 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F06V1ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_032 of D1

drop tablespace ts_orderline_032;
create regular tablespace ts_orderline_032 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F06V2ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_033 of D1

drop tablespace ts_orderline_033;
create regular tablespace ts_orderline_033 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F06V3ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_034 of D1

drop tablespace ts_orderline_034;
create regular tablespace ts_orderline_034 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F06V4ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_035 of D1

drop tablespace ts_orderline_035;
create regular tablespace ts_orderline_035 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F06V5ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_036 of D1

drop tablespace ts_orderline_036;
create regular tablespace ts_orderline_036 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F06V6ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_037 of D1

drop tablespace ts_orderline_037;
create regular tablespace ts_orderline_037 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F07V1ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_038 of D1

drop tablespace ts_orderline_038;
create regular tablespace ts_orderline_038 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F07V2ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_039 of D1

drop tablespace ts_orderline_039;
create regular tablespace ts_orderline_039 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F07V3ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_040 of D1

drop tablespace ts_orderline_040;
create regular tablespace ts_orderline_040 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F07V4ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_041 of D1

drop tablespace ts_orderline_041;
create regular tablespace ts_orderline_041 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F07V5ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_042 of D1

drop tablespace ts_orderline_042;
create regular tablespace ts_orderline_042 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F07V6ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_043 of D1

drop tablespace ts_orderline_043;
create regular tablespace ts_orderline_043 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F08V1ORL' 5348864
```

```
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_044 of D1

drop tablespace ts_orderline_044;
create regular tablespace ts_orderline_044 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F08V2ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_045 of D1

drop tablespace ts_orderline_045;
create regular tablespace ts_orderline_045 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F08V3ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_046 of D1

drop tablespace ts_orderline_046;
create regular tablespace ts_orderline_046 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F08V4ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_047 of D1

drop tablespace ts_orderline_047;
create regular tablespace ts_orderline_047 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F08V5ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_048 of D1

drop tablespace ts_orderline_048;
create regular tablespace ts_orderline_048 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F08V6ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_049 of D1

drop tablespace ts_orderline_049;
create regular tablespace ts_orderline_049 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F09V1ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_050 of D1

drop tablespace ts_orderline_050;
create regular tablespace ts_orderline_050 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F09V2ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_051 of D1

drop tablespace ts_orderline_051;
create regular tablespace ts_orderline_051 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F09V3ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_052 of D1

drop tablespace ts_orderline_052;
create regular tablespace ts_orderline_052 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F09V4ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_053 of D1

drop tablespace ts_orderline_053;
create regular tablespace ts_orderline_053 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F09V5ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_054 of D1

drop tablespace ts_orderline_054;
create regular tablespace ts_orderline_054 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F09V6ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_055 of D1

drop tablespace ts_orderline_055;
create regular tablespace ts_orderline_055 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F10V1ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_056 of D1

drop tablespace ts_orderline_056;
create regular tablespace ts_orderline_056 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F10V2ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_057 of D1

drop tablespace ts_orderline_057;
create regular tablespace ts_orderline_057 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F10V3ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_058 of D1

drop tablespace ts_orderline_058;
create regular tablespace ts_orderline_058 pagesize 8K
         managed by database
         using
         (
                device '/dev/rD1F10V4ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_059 of D1

drop tablespace ts_orderline_059;
```

```
create regular tablespace ts_orderline_059 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F10V5ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_060 of D1

drop tablespace ts_orderline_060;
create regular tablespace ts_orderline_060 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F10V6ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_061 of D1

drop tablespace ts_orderline_061;
create regular tablespace ts_orderline_061 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F11V1ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_062 of D1

drop tablespace ts_orderline_062;
create regular tablespace ts_orderline_062 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F11V2ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_063 of D1

drop tablespace ts_orderline_063;
create regular tablespace ts_orderline_063 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F11V3ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_064 of D1

drop tablespace ts_orderline_064;
create regular tablespace ts_orderline_064 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F11V4ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_065 of D1

drop tablespace ts_orderline_065;
create regular tablespace ts_orderline_065 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F11V5ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_066 of D1

drop tablespace ts_orderline_066;
create regular tablespace ts_orderline_066 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F11V6ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_067 of D1

drop tablespace ts_orderline_067;
create regular tablespace ts_orderline_067 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F12V1ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_068 of D1

drop tablespace ts_orderline_068;
create regular tablespace ts_orderline_068 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F12V2ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_069 of D1

drop tablespace ts_orderline_069;
create regular tablespace ts_orderline_069 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F12V3ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_070 of D1

drop tablespace ts_orderline_070;
create regular tablespace ts_orderline_070 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F12V4ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_071 of D1

drop tablespace ts_orderline_071;
create regular tablespace ts_orderline_071 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F12V5ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_072 of D1

drop tablespace ts_orderline_072;
create regular tablespace ts_orderline_072 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F12V6ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_073 of D1

drop tablespace ts_orderline_073;
create regular tablespace ts_orderline_073 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F13V1ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_074 of D1

drop tablespace ts_orderline_074;
create regular tablespace ts_orderline_074 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F13V2ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
```

```
commit;

-- now creating TS for ts_orderline_075 of D1

drop tablespace ts_orderline_075;
create regular tablespace ts_orderline_075 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F13V3ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_076 of D1

drop tablespace ts_orderline_076;
create regular tablespace ts_orderline_076 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F13V4ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_077 of D1

drop tablespace ts_orderline_077;
create regular tablespace ts_orderline_077 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F13V5ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_078 of D1

drop tablespace ts_orderline_078;
create regular tablespace ts_orderline_078 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F13V6ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_079 of D1

drop tablespace ts_orderline_079;
create regular tablespace ts_orderline_079 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F14V1ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_080 of D1

drop tablespace ts_orderline_080;
create regular tablespace ts_orderline_080 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F14V2ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_081 of D1

drop tablespace ts_orderline_081;
create regular tablespace ts_orderline_081 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F14V3ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_082 of D1

drop tablespace ts_orderline_082;
create regular tablespace ts_orderline_082 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F14V4ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_083 of D1

drop tablespace ts_orderline_083;
create regular tablespace ts_orderline_083 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F14V5ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_084 of D1

drop tablespace ts_orderline_084;
create regular tablespace ts_orderline_084 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F14V6ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_085 of D1

drop tablespace ts_orderline_085;
create regular tablespace ts_orderline_085 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F15V1ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_086 of D1

drop tablespace ts_orderline_086;
create regular tablespace ts_orderline_086 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F15V2ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_087 of D1

drop tablespace ts_orderline_087;
create regular tablespace ts_orderline_087 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F15V3ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_088 of D1

drop tablespace ts_orderline_088;
create regular tablespace ts_orderline_088 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F15V4ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_089 of D1

drop tablespace ts_orderline_089;
create regular tablespace ts_orderline_089 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F15V5ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_090 of D1

drop tablespace ts_orderline_090;
create regular tablespace ts_orderline_090 pagesize 8K
        managed by database
        using
        (
```

```
                    device '/dev/rD1F15V6ORL' 5348864
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_091 of D1

drop tablespace ts_orderline_091;
create regular tablespace ts_orderline_091 pagesize 8K
           managed by database
           using
           (
                    device '/dev/rD1F16V1ORL' 5348864
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_092 of D1

drop tablespace ts_orderline_092;
create regular tablespace ts_orderline_092 pagesize 8K
           managed by database
           using
           (
                    device '/dev/rD1F16V2ORL' 5348864
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_093 of D1

drop tablespace ts_orderline_093;
create regular tablespace ts_orderline_093 pagesize 8K
           managed by database
           using
           (
                    device '/dev/rD1F16V3ORL' 5348864
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_094 of D1

drop tablespace ts_orderline_094;
create regular tablespace ts_orderline_094 pagesize 8K
           managed by database
           using
           (
                    device '/dev/rD1F16V4ORL' 5348864
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_095 of D1

drop tablespace ts_orderline_095;
create regular tablespace ts_orderline_095 pagesize 8K
           managed by database
           using
           (
                    device '/dev/rD1F16V5ORL' 5348864
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_096 of D1

drop tablespace ts_orderline_096;
create regular tablespace ts_orderline_096 pagesize 8K
           managed by database
           using
           (
                    device '/dev/rD1F16V6ORL' 5348864
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_097 of D1

drop tablespace ts_orderline_097;
create regular tablespace ts_orderline_097 pagesize 8K
           managed by database
           using
           (
                    device '/dev/rD1F17V1ORL' 5348864
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_098 of D1

drop tablespace ts_orderline_098;
create regular tablespace ts_orderline_098 pagesize 8K
           managed by database
           using
           (
                    device '/dev/rD1F17V2ORL' 5348864
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_099 of D1

drop tablespace ts_orderline_099;
create regular tablespace ts_orderline_099 pagesize 8K
           managed by database
           using
           (
                    device '/dev/rD1F17V3ORL' 5348864
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_100 of D1

drop tablespace ts_orderline_100;
create regular tablespace ts_orderline_100 pagesize 8K
           managed by database
           using
           (
                    device '/dev/rD1F17V4ORL' 5348864
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_101 of D1

drop tablespace ts_orderline_101;
create regular tablespace ts_orderline_101 pagesize 8K
           managed by database
           using
           (
                    device '/dev/rD1F17V5ORL' 5348864
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_102 of D1

drop tablespace ts_orderline_102;
create regular tablespace ts_orderline_102 pagesize 8K
           managed by database
           using
           (
                    device '/dev/rD1F17V6ORL' 5348864
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_103 of D1

drop tablespace ts_orderline_103;
create regular tablespace ts_orderline_103 pagesize 8K
           managed by database
           using
           (
                    device '/dev/rD1F18V1ORL' 5348864
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_104 of D1

drop tablespace ts_orderline_104;
create regular tablespace ts_orderline_104 pagesize 8K
           managed by database
           using
           (
                    device '/dev/rD1F18V2ORL' 5348864
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_105 of D1

drop tablespace ts_orderline_105;
create regular tablespace ts_orderline_105 pagesize 8K
           managed by database
           using
           (
                    device '/dev/rD1F18V3ORL' 5348864
                    )
          extentsize 256
          prefetchsize 4096
          bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_106 of D1
```

```
drop tablespace ts_orderline_106;
create regular tablespace ts_orderline_106 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F18V4ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_107 of D1

drop tablespace ts_orderline_107;
create regular tablespace ts_orderline_107 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F18V5ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_108 of D1

drop tablespace ts_orderline_108;
create regular tablespace ts_orderline_108 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F18V6ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_109 of D1

drop tablespace ts_orderline_109;
create regular tablespace ts_orderline_109 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F19V1ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_110 of D1

drop tablespace ts_orderline_110;
create regular tablespace ts_orderline_110 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F19V2ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_111 of D1

drop tablespace ts_orderline_111;
create regular tablespace ts_orderline_111 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F19V3ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_112 of D1

drop tablespace ts_orderline_112;
create regular tablespace ts_orderline_112 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F19V4ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_113 of D1

drop tablespace ts_orderline_113;
create regular tablespace ts_orderline_113 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F19V5ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_114 of D1

drop tablespace ts_orderline_114;
create regular tablespace ts_orderline_114 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F19V6ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_115 of D1

drop tablespace ts_orderline_115;
create regular tablespace ts_orderline_115 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F20V1ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_116 of D1

drop tablespace ts_orderline_116;
create regular tablespace ts_orderline_116 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F20V2ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_117 of D1

drop tablespace ts_orderline_117;
create regular tablespace ts_orderline_117 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F20V3ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_118 of D1

drop tablespace ts_orderline_118;
create regular tablespace ts_orderline_118 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F20V4ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_119 of D1

drop tablespace ts_orderline_119;
create regular tablespace ts_orderline_119 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F20V5ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_120 of D1

drop tablespace ts_orderline_120;
create regular tablespace ts_orderline_120 pagesize 8K
        managed by database
        using
        (
                device '/dev/rD1F20V6ORL' 5348864
                )
        extentsize 256
        prefetchsize 4096
        bufferpool ibmdefaultbp8K;
commit;

connect reset;
```

ts/crts_stock.ddl

```
connect to tpcc;
-- now creating TS for ts_stock_001 of D1

drop tablespace ts_stock_001;
```

```
create regular tablespace ts_stock_001 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F01V1STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_002 of D1

drop tablespace ts_stock_002;
create regular tablespace ts_stock_002 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F01V2STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_003 of D1

drop tablespace ts_stock_003;
create regular tablespace ts_stock_003 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F01V3STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_004 of D1

drop tablespace ts_stock_004;
create regular tablespace ts_stock_004 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F01V4STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_005 of D1

drop tablespace ts_stock_005;
create regular tablespace ts_stock_005 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F01V5STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_006 of D1

drop tablespace ts_stock_006;
create regular tablespace ts_stock_006 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F01V6STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_007 of D1

drop tablespace ts_stock_007;
create regular tablespace ts_stock_007 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F02V1STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_008 of D1

drop tablespace ts_stock_008;
create regular tablespace ts_stock_008 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F02V2STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_009 of D1

drop tablespace ts_stock_009;
create regular tablespace ts_stock_009 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F02V3STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_010 of D1

drop tablespace ts_stock_010;
create regular tablespace ts_stock_010 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F02V4STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_011 of D1

drop tablespace ts_stock_011;
create regular tablespace ts_stock_011 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F02V5STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_012 of D1

drop tablespace ts_stock_012;
create regular tablespace ts_stock_012 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F02V6STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_013 of D1

drop tablespace ts_stock_013;
create regular tablespace ts_stock_013 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F03V1STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_014 of D1

drop tablespace ts_stock_014;
create regular tablespace ts_stock_014 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F03V2STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_015 of D1

drop tablespace ts_stock_015;
create regular tablespace ts_stock_015 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F03V3STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_016 of D1

drop tablespace ts_stock_016;
create regular tablespace ts_stock_016 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F03V4STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_017 of D1

drop tablespace ts_stock_017;
create regular tablespace ts_stock_017 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F03V5STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_018 of D1
```

```
drop tablespace ts_stock_018;
create regular tablespace ts_stock_018 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F03V6STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_019 of D1

drop tablespace ts_stock_019;
create regular tablespace ts_stock_019 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F04V1STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_020 of D1

drop tablespace ts_stock_020;
create regular tablespace ts_stock_020 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F04V2STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_021 of D1

drop tablespace ts_stock_021;
create regular tablespace ts_stock_021 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F04V3STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_022 of D1

drop tablespace ts_stock_022;
create regular tablespace ts_stock_022 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F04V4STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_023 of D1

drop tablespace ts_stock_023;
create regular tablespace ts_stock_023 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F04V5STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_024 of D1

drop tablespace ts_stock_024;
create regular tablespace ts_stock_024 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F04V6STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_025 of D1

drop tablespace ts_stock_025;
create regular tablespace ts_stock_025 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F05V1STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_026 of D1

drop tablespace ts_stock_026;
create regular tablespace ts_stock_026 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F05V2STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_027 of D1

drop tablespace ts_stock_027;
create regular tablespace ts_stock_027 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F05V3STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_028 of D1

drop tablespace ts_stock_028;
create regular tablespace ts_stock_028 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F05V4STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_029 of D1

drop tablespace ts_stock_029;
create regular tablespace ts_stock_029 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F05V5STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_030 of D1

drop tablespace ts_stock_030;
create regular tablespace ts_stock_030 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F05V6STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_031 of D1

drop tablespace ts_stock_031;
create regular tablespace ts_stock_031 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F06V1STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_032 of D1

drop tablespace ts_stock_032;
create regular tablespace ts_stock_032 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F06V2STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_033 of D1

drop tablespace ts_stock_033;
create regular tablespace ts_stock_033 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F06V3STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_034 of D1

drop tablespace ts_stock_034;
create regular tablespace ts_stock_034 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F06V4STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;
```

```
-- now creating TS for ts_stock_035 of D1

drop tablespace ts_stock_035;
create regular tablespace ts_stock_035 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F06V5STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_036 of D1

drop tablespace ts_stock_036;
create regular tablespace ts_stock_036 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F06V6STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_037 of D1

drop tablespace ts_stock_037;
create regular tablespace ts_stock_037 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F07V1STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_038 of D1

drop tablespace ts_stock_038;
create regular tablespace ts_stock_038 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F07V2STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_039 of D1

drop tablespace ts_stock_039;
create regular tablespace ts_stock_039 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F07V3STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_040 of D1

drop tablespace ts_stock_040;
create regular tablespace ts_stock_040 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F07V4STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_041 of D1

drop tablespace ts_stock_041;
create regular tablespace ts_stock_041 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F07V5STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_042 of D1

drop tablespace ts_stock_042;
create regular tablespace ts_stock_042 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F07V6STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_043 of D1

drop tablespace ts_stock_043;
create regular tablespace ts_stock_043 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F08V1STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_044 of D1

drop tablespace ts_stock_044;
create regular tablespace ts_stock_044 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F08V2STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_045 of D1

drop tablespace ts_stock_045;
create regular tablespace ts_stock_045 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F08V3STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_046 of D1

drop tablespace ts_stock_046;
create regular tablespace ts_stock_046 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F08V4STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_047 of D1

drop tablespace ts_stock_047;
create regular tablespace ts_stock_047 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F08V5STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_048 of D1

drop tablespace ts_stock_048;
create regular tablespace ts_stock_048 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F08V6STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_049 of D1

drop tablespace ts_stock_049;
create regular tablespace ts_stock_049 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F09V1STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_050 of D1

drop tablespace ts_stock_050;
create regular tablespace ts_stock_050 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F09V2STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_051 of D1

drop tablespace ts_stock_051;
create regular tablespace ts_stock_051 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F09V3STK' 9329152
                )
        extentsize 256
```

```
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_052 of D1

drop tablespace ts_stock_052;
create regular tablespace ts_stock_052 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F09V4STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_053 of D1

drop tablespace ts_stock_053;
create regular tablespace ts_stock_053 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F09V5STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_054 of D1

drop tablespace ts_stock_054;
create regular tablespace ts_stock_054 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F09V6STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_055 of D1

drop tablespace ts_stock_055;
create regular tablespace ts_stock_055 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F10V1STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_056 of D1

drop tablespace ts_stock_056;
create regular tablespace ts_stock_056 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F10V2STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_057 of D1

drop tablespace ts_stock_057;
create regular tablespace ts_stock_057 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F10V3STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_058 of D1

drop tablespace ts_stock_058;
create regular tablespace ts_stock_058 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F10V4STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_059 of D1

drop tablespace ts_stock_059;
create regular tablespace ts_stock_059 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F10V5STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_060 of D1

drop tablespace ts_stock_060;
create regular tablespace ts_stock_060 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F10V6STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_061 of D1

drop tablespace ts_stock_061;
create regular tablespace ts_stock_061 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F11V1STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_062 of D1

drop tablespace ts_stock_062;
create regular tablespace ts_stock_062 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F11V2STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_063 of D1

drop tablespace ts_stock_063;
create regular tablespace ts_stock_063 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F11V3STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_064 of D1

drop tablespace ts_stock_064;
create regular tablespace ts_stock_064 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F11V4STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_065 of D1

drop tablespace ts_stock_065;
create regular tablespace ts_stock_065 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F11V5STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_066 of D1

drop tablespace ts_stock_066;
create regular tablespace ts_stock_066 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F11V6STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_067 of D1

drop tablespace ts_stock_067;
create regular tablespace ts_stock_067 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F12V1STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_068 of D1

drop tablespace ts_stock_068;
create regular tablespace ts_stock_068 pagesize 4K
         managed by database
         using
         (
                 device '/dev/rD1F12V2STK' 9329152
```

```
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_069 of D1

drop tablespace ts_stock_069;
create regular tablespace ts_stock_069 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F12V3STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_070 of D1

drop tablespace ts_stock_070;
create regular tablespace ts_stock_070 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F12V4STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_071 of D1

drop tablespace ts_stock_071;
create regular tablespace ts_stock_071 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F12V5STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_072 of D1

drop tablespace ts_stock_072;
create regular tablespace ts_stock_072 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F12V6STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_073 of D1

drop tablespace ts_stock_073;
create regular tablespace ts_stock_073 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F13V1STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_074 of D1

drop tablespace ts_stock_074;
create regular tablespace ts_stock_074 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F13V2STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_075 of D1

drop tablespace ts_stock_075;
create regular tablespace ts_stock_075 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F13V3STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_076 of D1

drop tablespace ts_stock_076;
create regular tablespace ts_stock_076 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F13V4STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_077 of D1

drop tablespace ts_stock_077;
create regular tablespace ts_stock_077 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F13V5STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_078 of D1

drop tablespace ts_stock_078;
create regular tablespace ts_stock_078 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F13V6STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_079 of D1

drop tablespace ts_stock_079;
create regular tablespace ts_stock_079 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F14V1STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_080 of D1

drop tablespace ts_stock_080;
create regular tablespace ts_stock_080 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F14V2STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_081 of D1

drop tablespace ts_stock_081;
create regular tablespace ts_stock_081 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F14V3STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_082 of D1

drop tablespace ts_stock_082;
create regular tablespace ts_stock_082 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F14V4STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_083 of D1

drop tablespace ts_stock_083;
create regular tablespace ts_stock_083 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F14V5STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_084 of D1

drop tablespace ts_stock_084;
create regular tablespace ts_stock_084 pagesize 4K
         managed by database
         using
         (
                device '/dev/rD1F14V6STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_085 of D1

drop tablespace ts_stock_085;
create regular tablespace ts_stock_085 pagesize 4K
         managed by database
         using
```

```
        (
                device '/dev/rD1F15V1STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_086 of D1

drop tablespace ts_stock_086;
create regular tablespace ts_stock_086 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F15V2STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_087 of D1

drop tablespace ts_stock_087;
create regular tablespace ts_stock_087 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F15V3STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_088 of D1

drop tablespace ts_stock_088;
create regular tablespace ts_stock_088 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F15V4STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_089 of D1

drop tablespace ts_stock_089;
create regular tablespace ts_stock_089 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F15V5STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_090 of D1

drop tablespace ts_stock_090;
create regular tablespace ts_stock_090 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F15V6STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_091 of D1

drop tablespace ts_stock_091;
create regular tablespace ts_stock_091 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F16V1STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_092 of D1

drop tablespace ts_stock_092;
create regular tablespace ts_stock_092 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F16V2STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_093 of D1

drop tablespace ts_stock_093;
create regular tablespace ts_stock_093 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F16V3STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_094 of D1

drop tablespace ts_stock_094;
create regular tablespace ts_stock_094 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F16V4STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_095 of D1

drop tablespace ts_stock_095;
create regular tablespace ts_stock_095 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F16V5STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_096 of D1

drop tablespace ts_stock_096;
create regular tablespace ts_stock_096 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F16V6STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_097 of D1

drop tablespace ts_stock_097;
create regular tablespace ts_stock_097 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F17V1STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_098 of D1

drop tablespace ts_stock_098;
create regular tablespace ts_stock_098 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F17V2STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_099 of D1

drop tablespace ts_stock_099;
create regular tablespace ts_stock_099 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F17V3STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_100 of D1

drop tablespace ts_stock_100;
create regular tablespace ts_stock_100 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F17V4STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_101 of D1

drop tablespace ts_stock_101;
create regular tablespace ts_stock_101 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F17V5STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_102 of D1

drop tablespace ts_stock_102;
create regular tablespace ts_stock_102 pagesize 4K
```

```
            managed by database
            using
            (
                    device '/dev/rD1F17V6STK' 9329152
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_stock_103 of D1

drop tablespace ts_stock_103;
create regular tablespace ts_stock_103 pagesize 4K
            managed by database
            using
            (
                    device '/dev/rD1F18V1STK' 9329152
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_stock_104 of D1

drop tablespace ts_stock_104;
create regular tablespace ts_stock_104 pagesize 4K
            managed by database
            using
            (
                    device '/dev/rD1F18V2STK' 9329152
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_stock_105 of D1

drop tablespace ts_stock_105;
create regular tablespace ts_stock_105 pagesize 4K
            managed by database
            using
            (
                    device '/dev/rD1F18V3STK' 9329152
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_stock_106 of D1

drop tablespace ts_stock_106;
create regular tablespace ts_stock_106 pagesize 4K
            managed by database
            using
            (
                    device '/dev/rD1F18V4STK' 9329152
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_stock_107 of D1

drop tablespace ts_stock_107;
create regular tablespace ts_stock_107 pagesize 4K
            managed by database
            using
            (
                    device '/dev/rD1F18V5STK' 9329152
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_stock_108 of D1

drop tablespace ts_stock_108;
create regular tablespace ts_stock_108 pagesize 4K
            managed by database
            using
            (
                    device '/dev/rD1F18V6STK' 9329152
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_stock_109 of D1

drop tablespace ts_stock_109;
create regular tablespace ts_stock_109 pagesize 4K
            managed by database
            using
            (
                    device '/dev/rD1F19V1STK' 9329152
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_stock_110 of D1

drop tablespace ts_stock_110;
create regular tablespace ts_stock_110 pagesize 4K
            managed by database
            using
            (
                    device '/dev/rD1F19V2STK' 9329152
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_stock_111 of D1

drop tablespace ts_stock_111;
create regular tablespace ts_stock_111 pagesize 4K
            managed by database
            using
            (
                    device '/dev/rD1F19V3STK' 9329152
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_stock_112 of D1

drop tablespace ts_stock_112;
create regular tablespace ts_stock_112 pagesize 4K
            managed by database
            using
            (
                    device '/dev/rD1F19V4STK' 9329152
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_stock_113 of D1

drop tablespace ts_stock_113;
create regular tablespace ts_stock_113 pagesize 4K
            managed by database
            using
            (
                    device '/dev/rD1F19V5STK' 9329152
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_stock_114 of D1

drop tablespace ts_stock_114;
create regular tablespace ts_stock_114 pagesize 4K
            managed by database
            using
            (
                    device '/dev/rD1F19V6STK' 9329152
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_stock_115 of D1

drop tablespace ts_stock_115;
create regular tablespace ts_stock_115 pagesize 4K
            managed by database
            using
            (
                    device '/dev/rD1F20V1STK' 9329152
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_stock_116 of D1

drop tablespace ts_stock_116;
create regular tablespace ts_stock_116 pagesize 4K
            managed by database
            using
            (
                    device '/dev/rD1F20V2STK' 9329152
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_stock_117 of D1

drop tablespace ts_stock_117;
create regular tablespace ts_stock_117 pagesize 4K
            managed by database
            using
            (
                    device '/dev/rD1F20V3STK' 9329152
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_stock_118 of D1

drop tablespace ts_stock_118;
create regular tablespace ts_stock_118 pagesize 4K
            managed by database
            using
            (
                    device '/dev/rD1F20V4STK' 9329152
                    )
          extentsize 256
          prefetchsize 4096;
commit;

-- now creating TS for ts_stock_119 of D1
```

```
drop tablespace ts_stock_119;
create regular tablespace ts_stock_119 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F20V5STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

-- now creating TS for ts_stock_120 of D1

drop tablespace ts_stock_120;
create regular tablespace ts_stock_120 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F20V6STK' 9329152
                )
        extentsize 256
        prefetchsize 4096;
commit;

connect reset;
```

ts/crts_ware.ddl

```
connect to tpcc;
-- now creating TS for ts_ware_001 of D1

drop tablespace ts_ware_001;
create regular tablespace ts_ware_001 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F01V1WARE' 128,
                device '/dev/rD1F01V2WARE' 128,
                device '/dev/rD1F01V3WARE' 128
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_002 of D1

drop tablespace ts_ware_002;
create regular tablespace ts_ware_002 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F01V4WARE' 128,
                device '/dev/rD1F01V5WARE' 128,
                device '/dev/rD1F01V6WARE' 128
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_003 of D1

drop tablespace ts_ware_003;
create regular tablespace ts_ware_003 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F02V1WARE' 128,
                device '/dev/rD1F02V2WARE' 128,
                device '/dev/rD1F02V3WARE' 128
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_004 of D1

drop tablespace ts_ware_004;
create regular tablespace ts_ware_004 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F02V4WARE' 128,
                device '/dev/rD1F02V5WARE' 128,
                device '/dev/rD1F02V6WARE' 128
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_005 of D1

drop tablespace ts_ware_005;
create regular tablespace ts_ware_005 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F03V1WARE' 128,
                device '/dev/rD1F03V2WARE' 128,
                device '/dev/rD1F03V3WARE' 128
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_006 of D1

drop tablespace ts_ware_006;
create regular tablespace ts_ware_006 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F03V4WARE' 128,
                device '/dev/rD1F03V5WARE' 128,
                device '/dev/rD1F03V6WARE' 128
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_007 of D1

drop tablespace ts_ware_007;
create regular tablespace ts_ware_007 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F04V1WARE' 128,
                device '/dev/rD1F04V2WARE' 128,
                device '/dev/rD1F04V3WARE' 128
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_008 of D1

drop tablespace ts_ware_008;
create regular tablespace ts_ware_008 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F04V4WARE' 128,
                device '/dev/rD1F04V5WARE' 128,
                device '/dev/rD1F04V6WARE' 128
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_009 of D1

drop tablespace ts_ware_009;
create regular tablespace ts_ware_009 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F05V1WARE' 128,
                device '/dev/rD1F05V2WARE' 128,
                device '/dev/rD1F05V3WARE' 128
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_010 of D1

drop tablespace ts_ware_010;
create regular tablespace ts_ware_010 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F05V4WARE' 128,
                device '/dev/rD1F05V5WARE' 128,
                device '/dev/rD1F05V6WARE' 128
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_011 of D1

drop tablespace ts_ware_011;
create regular tablespace ts_ware_011 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F06V1WARE' 128,
                device '/dev/rD1F06V2WARE' 128,
                device '/dev/rD1F06V3WARE' 128
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_012 of D1

drop tablespace ts_ware_012;
create regular tablespace ts_ware_012 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F06V4WARE' 128,
                device '/dev/rD1F06V5WARE' 128,
                device '/dev/rD1F06V6WARE' 128
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_013 of D1

drop tablespace ts_ware_013;
create regular tablespace ts_ware_013 pagesize 4K
        managed by database
        using
        (
```

```
                    device '/dev/rD1F07V1WARE' 128,
                    device '/dev/rD1F07V2WARE' 128,
                    device '/dev/rD1F07V3WARE' 128
                   )
          extentsize 32
          prefetchsize 4096;
commit;

-- now creating TS for ts_ware_014 of D1

drop tablespace ts_ware_014;
create regular tablespace ts_ware_014 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F07V4WARE' 128,
                    device '/dev/rD1F07V5WARE' 128,
                    device '/dev/rD1F07V6WARE' 128
                   )
          extentsize 32
          prefetchsize 4096;
commit;

-- now creating TS for ts_ware_015 of D1

drop tablespace ts_ware_015;
create regular tablespace ts_ware_015 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F08V1WARE' 128,
                    device '/dev/rD1F08V2WARE' 128,
                    device '/dev/rD1F08V3WARE' 128
                   )
          extentsize 32
          prefetchsize 4096;
commit;

-- now creating TS for ts_ware_016 of D1

drop tablespace ts_ware_016;
create regular tablespace ts_ware_016 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F08V4WARE' 128,
                    device '/dev/rD1F08V5WARE' 128,
                    device '/dev/rD1F08V6WARE' 128
                   )
          extentsize 32
          prefetchsize 4096;
commit;

-- now creating TS for ts_ware_017 of D1

drop tablespace ts_ware_017;
create regular tablespace ts_ware_017 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F09V1WARE' 128,
                    device '/dev/rD1F09V2WARE' 128,
                    device '/dev/rD1F09V3WARE' 128
                   )
          extentsize 32
          prefetchsize 4096;
commit;

-- now creating TS for ts_ware_018 of D1

drop tablespace ts_ware_018;
create regular tablespace ts_ware_018 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F09V4WARE' 128,
                    device '/dev/rD1F09V5WARE' 128,
                    device '/dev/rD1F09V6WARE' 128
                   )
          extentsize 32
          prefetchsize 4096;
commit;

-- now creating TS for ts_ware_019 of D1

drop tablespace ts_ware_019;
create regular tablespace ts_ware_019 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F10V1WARE' 128,
                    device '/dev/rD1F10V2WARE' 128,
                    device '/dev/rD1F10V3WARE' 128
                   )
          extentsize 32
          prefetchsize 4096;
commit;

-- now creating TS for ts_ware_020 of D1

drop tablespace ts_ware_020;
create regular tablespace ts_ware_020 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F10V4WARE' 128,
                    device '/dev/rD1F10V5WARE' 128,
                    device '/dev/rD1F10V6WARE' 128
                   )
          extentsize 32
          prefetchsize 4096;
commit;

-- now creating TS for ts_ware_021 of D1

drop tablespace ts_ware_021;
create regular tablespace ts_ware_021 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F11V1WARE' 128,
                    device '/dev/rD1F11V2WARE' 128,
                    device '/dev/rD1F11V3WARE' 128
                   )
          extentsize 32
          prefetchsize 4096;
commit;

-- now creating TS for ts_ware_022 of D1

drop tablespace ts_ware_022;
create regular tablespace ts_ware_022 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F11V4WARE' 128,
                    device '/dev/rD1F11V5WARE' 128,
                    device '/dev/rD1F11V6WARE' 128
                   )
          extentsize 32
          prefetchsize 4096;
commit;

-- now creating TS for ts_ware_023 of D1

drop tablespace ts_ware_023;
create regular tablespace ts_ware_023 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F12V1WARE' 128,
                    device '/dev/rD1F12V2WARE' 128,
                    device '/dev/rD1F12V3WARE' 128
                   )
          extentsize 32
          prefetchsize 4096;
commit;

-- now creating TS for ts_ware_024 of D1

drop tablespace ts_ware_024;
create regular tablespace ts_ware_024 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F12V4WARE' 128,
                    device '/dev/rD1F12V5WARE' 128,
                    device '/dev/rD1F12V6WARE' 128
                   )
          extentsize 32
          prefetchsize 4096;
commit;

-- now creating TS for ts_ware_025 of D1

drop tablespace ts_ware_025;
create regular tablespace ts_ware_025 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F13V1WARE' 128,
                    device '/dev/rD1F13V2WARE' 128,
                    device '/dev/rD1F13V3WARE' 128
                   )
          extentsize 32
          prefetchsize 4096;
commit;

-- now creating TS for ts_ware_026 of D1

drop tablespace ts_ware_026;
create regular tablespace ts_ware_026 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F13V4WARE' 128,
                    device '/dev/rD1F13V5WARE' 128,
                    device '/dev/rD1F13V6WARE' 128
                   )
          extentsize 32
          prefetchsize 4096;
commit;

-- now creating TS for ts_ware_027 of D1

drop tablespace ts_ware_027;
create regular tablespace ts_ware_027 pagesize 4K
           managed by database
           using
           (
                    device '/dev/rD1F14V1WARE' 128,
                    device '/dev/rD1F14V2WARE' 128,
                    device '/dev/rD1F14V3WARE' 128
                   )
          extentsize 32
          prefetchsize 4096;
commit;

-- now creating TS for ts_ware_028 of D1
```

```
drop tablespace ts_ware_028;
create regular tablespace ts_ware_028 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F14V4WARE' 128,
                device '/dev/rD1F14V5WARE' 128,
                device '/dev/rD1F14V6WARE' 128
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_029 of D1

drop tablespace ts_ware_029;
create regular tablespace ts_ware_029 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F15V1WARE' 128,
                device '/dev/rD1F15V2WARE' 128,
                device '/dev/rD1F15V3WARE' 128
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_030 of D1

drop tablespace ts_ware_030;
create regular tablespace ts_ware_030 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F15V4WARE' 128,
                device '/dev/rD1F15V5WARE' 128,
                device '/dev/rD1F15V6WARE' 128
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_031 of D1

drop tablespace ts_ware_031;
create regular tablespace ts_ware_031 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F16V1WARE' 128,
                device '/dev/rD1F16V2WARE' 128,
                device '/dev/rD1F16V3WARE' 128
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_032 of D1

drop tablespace ts_ware_032;
create regular tablespace ts_ware_032 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F16V4WARE' 128,
                device '/dev/rD1F16V5WARE' 128,
                device '/dev/rD1F16V6WARE' 128
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_033 of D1

drop tablespace ts_ware_033;
create regular tablespace ts_ware_033 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F17V1WARE' 128,
                device '/dev/rD1F17V2WARE' 128,
                device '/dev/rD1F17V3WARE' 128
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_034 of D1

drop tablespace ts_ware_034;
create regular tablespace ts_ware_034 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F17V4WARE' 128,
                device '/dev/rD1F17V5WARE' 128,
                device '/dev/rD1F17V6WARE' 128
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_035 of D1

drop tablespace ts_ware_035;
create regular tablespace ts_ware_035 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F18V1WARE' 128,
                device '/dev/rD1F18V2WARE' 128,
                device '/dev/rD1F18V3WARE' 128
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_036 of D1

drop tablespace ts_ware_036;
create regular tablespace ts_ware_036 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F18V4WARE' 128,
                device '/dev/rD1F18V5WARE' 128,
                device '/dev/rD1F18V6WARE' 128
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_037 of D1

drop tablespace ts_ware_037;
create regular tablespace ts_ware_037 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F19V1WARE' 128,
                device '/dev/rD1F19V2WARE' 128,
                device '/dev/rD1F19V3WARE' 128
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_038 of D1

drop tablespace ts_ware_038;
create regular tablespace ts_ware_038 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F19V4WARE' 128,
                device '/dev/rD1F19V5WARE' 128,
                device '/dev/rD1F19V6WARE' 128
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_039 of D1

drop tablespace ts_ware_039;
create regular tablespace ts_ware_039 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F20V1WARE' 128,
                device '/dev/rD1F20V2WARE' 128,
                device '/dev/rD1F20V3WARE' 128
                )
        extentsize 32
        prefetchsize 4096;
commit;

-- now creating TS for ts_ware_040 of D1

drop tablespace ts_ware_040;
create regular tablespace ts_ware_040 pagesize 4K
        managed by database
        using
        (
                device '/dev/rD1F20V4WARE' 128,
                device '/dev/rD1F20V5WARE' 128,
                device '/dev/rD1F20V6WARE' 128
                )
        extentsize 32
        prefetchsize 4096;
commit;

connect reset;
```

C.2 Data Generation Code

Makefile.config

```
##########################################################################
## Licensed Materials - Property of IBM
##
## Governed under the terms of the International
## License Agreement for Non-Warranted Sample Code.
##
## (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
## All Rights Reserved.
##
```

```
## US Government Users Restricted Rights - Use, duplication or
## disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
################################################################################

#
# Makefile.config - AIX 64-bit
#
#

# Make Configuration
MAKE=make

# Compiler Configuration.
# CFLAGS_DEBUG may be set to "-g", "-DDEBUGIT" "-g -DDEBUGIT" or left blank
CC=xlc
CFLAGS_OS=-qflag=i:i -qlanglvl=ansi -qcpluscmt -DSQLUNIX -DSQLAIX -q64 -O3 -D_LARGE_FILES
CFLAGS_OUT=-o
CFLAGS_DEBUG=

# Linker Configuration
LD_EXEC=xlc
LD_STORP=xlc
LDFLAGS_EXEC=-lm -q64
LDFLAGS_SHLIB=-qmkshrobj
LDFLAGS_STORP=$(LDFLAGS_SHLIB) -bE:$@.exp -lc -b64
LDFLAGS_LIB=-L$(TPCC_SQLLIB)/lib -ldb2
LDFLAGS_OUT=-o

# Library Configuration
AR=ar
ARFLAGS=-r -v -X64
ARFLAGS_LIB=
ARFLAGS_OUT=

# OS Commands
ERASE=rm -f
ERASEDIR=$(ERASE) -R
MOVE=mv
COPY=cp

# OS File Extensions & Path Separators
OBJEXT=.o
LIBEXT=.a
SHLIBEXT=.a
BINEXT=
SLASH=/
CMDSEP=;
```

Src.Common/Makefile

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

#
# Makefile - Makefile for Src.Common
#
#

include $(TPCC_ROOT)/Makefile.config

# ##############################################################################
# Preprocessor, Compiler and LInker Flags
# ##############################################################################

BND_OPTS =   GRANT PUBLIC \
             MESSAGES $*.bnd.msg
PRP_OPTS =   BINDFILE \
             OPTLEVEL 1 \
             ISOLATION RR \
             MESSAGES $*.prep.msg \
             LEVEL $(TPCC_VERSION) \
             NOLINEMACRO

INCLUDE =    -I$(TPCC_SQLLIB)/include -I$(TPCC_ROOT)/include

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

UTIL_OBJ_DBG =      tpccdbg$(OBJEXT)
UTIL_OBJ_GEN =      tpccmisc$(OBJEXT)
UTIL_OBJ_DB2 =      tpccctx$(OBJEXT)

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

all:     $(UTIL_OBJ_DBG) $(UTIL_OBJ_GEN) connect $(UTIL_OBJ_DB2) disconnect

dbgen:  $(UTIL_OBJ_GEN)

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

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

connect:
         - db2 connect to $(TPCC_DBNAME)

disconnect:
         - db2 connect reset
         - db2 terminate

rebind:  connect
         db2 bind tpccctx.bnd $(BND_OPTS)

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

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

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

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

# Source
tpccdbg$(OBJEXT):     tpccdbg.c
tpccctx$(OBJEXT):     tpccctx.c
tpccmisc$(OBJEXT):    tpccmisc.c

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

Src.Common/tpccmisc.c

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

/*
 *
 * tpccmisc.c - Miscellaneous routines
 *
 */

#include <stdlib.h>
#include <sys/types.h>
#include <sys/time.h>

double current_time_ms(void);
double current_time(void);


/* Current time in SECONDS, precision SECONDS */
double current_time(void)
{
  /* use time() to get seconds */
  return(time(NULL));
}

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

dbgen/Makefile

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

# Makefile - Build gendata tool
#
```

```
include $(TPCC_ROOT)/Makefile.config

# ##########################################################################
# Preprocessor, Compiler and Linker Flags
# ##########################################################################

INCLUDE =       -I$(TPCC_SQLLIB)/include -I$(TPCC_ROOT)/include

CFLAGS =        $(INCLUDE) $(CFLAGS_OS) -DLINT_ARGS -DSQLA_NOLINES \
                -D$(DB2EDITION) -D$(DB2VERSION) $(CFLAGS_DEBUG)

LDFLAGS =       $(LDFLAGS_EXEC) $(LDFLAGS_LIB)

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

OBJS =          tpccrnd$(OBJEXT) \
                $(TPCC_ROOT)/Src.Common/tpccmisc$(OBJEXT)
OBJ_EEE =       $(TPCC_ROOT)/Src.Common/tpcclwh$(OBJEXT)

EXEC =          gendata$(BINEXT)

# ##########################################################################
# End-User Targets
# ##########################################################################

all:    $(EXEC)

clean:
        - $(ERASE) *$(OBJEXT) $(EXEC)

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

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

# We use $@$(OBJEXT) here so that the UNIX makefiles work with both
# 'traditional' make and GNU make
$(EXEC):
        $(LD_EXEC) $(LDFLAGS) $(OBJS) $@$(OBJEXT) $(LDFLAGS_OUT)$@

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

# Link Dependencies
gendata$(BINEXT):       $(OBJS) gendata$(OBJEXT)

# Build Dependencies
# Source
gendata$(OBJEXT):       gendata.c

# Headers
gendata.c:      $(TPCC_ROOT)/include/tpccmd.h $(TPCC_ROOT)/include/lval.h
```

dbgen/gendata.c

```
/*****************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****************************************************************************/

/*
 * gendata.c - Generate data for TPC-C database
 *
 */

#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#include <sqlutil.h>
/* UNIX named pipe support */
#include <sys/stat.h>
#include <errno.h>
#include <fcntl.h>
#include <time.h>

#include "platform.h"
#include "db2tpcc.h"
#include "tpccrnd.h"
#include "tpccmisc.h"
#include "lval.h"


/*  PROTOTYPES.  */
void gen_dist_tbl( void );
void gen_cust_tbl( void );
void gen_hist_tbl( void );
void gen_nu_ord_tbl( void );
void gen_ordr_tbl( void );
void gen_item_tbl( void );
void gen_stock_tbl( void );
void gen_ware_tbl( void );

int i, j;
double timestamp1, timestamp2, elapse;
int rc, rc1, rc2;

int using_range = 0;
int using_npipe = 0;
int using_rctload = 0;
int quiet_mode = 0;
sqlint32 ware_start=-1, ware_end=-1;

char fmtWare[]    = "%s|%s|%s|%s|%s|%s|%04.4f|%.2f|%d\n";
char fmtDist[]    = "%d|%04.4f|%.2f|%s|%s|%s|%s|%s|%d|%d\n";
char fmtItem[]    = "%s|%.2f|%s|%d|%d\n";
char fmtStock[]   = "%d|%d|%d|%d|%s|%s|%s|%s|%s|%s|%s|%s|%s|%s|%d|%d\n";
char fmtCust[]    =
"%d|%s|%s|%s|%s|%.2f|%s|%s|%04.4f|%s|%s|%s|%s|%s|%s|%d|%d|%d|%.2f|%.2f|%d\n";
char fmtHist[]    = "%d|%d|%d|%d|%d|%s|%.2f|%s\n";
char fmtOrdr[]    = "%d|%s|%d|%d|%d|%d|%d|%d\n";
char fmtOLine[]   = "%s|%.2f|%d|%d|%d|%s|%d|%d|%d|%d\n";
char fmtNewOrd[]  = "%d|%d|%d\n";
void InitFormatStrings(char delim);
void ScalingReport(void);

int outtype1 = 0;
int outtype2 = 0;
char *outname1 = NULL;
char *outname2 = NULL;

/*-------------------------------------------------------------------*/
/*      main                                                         */
/*-------------------------------------------------------------------*/
int main (int argc, char *argv[])
{
  int option = -1;
  char *delim = NULL;

  /* ********************************************************** */
  /* Compute Warehouse Ranges                                   */
  /* ********************************************************** */
  ware_start = 1;
  ware_end = WAREHOUSES;

  /* ********************************************************** */
  /* Process Command Line Arguments                             */
  /* ********************************************************** */

  /* Valid Command Line Options
   * -------------------------
   * Table Option:          -t <table>        (-t 3 for warehouse)
   * Output Column Delimiter:  -d <char>          (-d ' ', -d '|', etc)
   * Output to File:        -f[n] <file>      (-f customer.dat)
   * Output to Pipe:        -p[n] <pipename>   (-p tpccpipe.000)
   * Warehouse Range:         -r <start> <end>   (-r 1 100)
   * Scaling Report:          -s
   * Quiet Mode:              -q
   *
   * The -f[n] and/or -p[n] options are required.
   * The -t, -d, -r, -s and -q options are optional.
   *
   * If -d is omitted, the vertical bar (pipe) symbol ('|') will be used.
   * If -r is omitted, the range [1..WAREHOUSES] will be used.
   *
   * Due to the TPC-C spec requiring that orders and orderline be
   * generated at the same time, there is an extension to the -f and -p
   * options to specify one of the two output streams for each argument.
   *
   * -f1 orders.dat -f2 orderline.dat will output to two files
   * -f1 orders.dat -p2 tpccpipe.000 will output to a file and a pipe
   *
   * -f1/-p1 specifies the destination for the orders table
   * -f2/-p2 specifies the destination for the orderline table
   *
   */

  /* Read Arguments */
  for (i=1; i<argc; i++)
  {
    if (strcmp(argv[i], "-t") == 0) {
      option = atoi(argv[i+1]);
      i++;
    } else if (strcmp(argv[i], "-r") == 0) {
      ware_start = atoi(argv[i+1]);
      ware_end = atoi(argv[i+2]);
      i += 2;
    } else if (strcmp(argv[i], "-d") == 0) {
      delim = argv[i+1];
      i++;
    } else if ((strcmp(argv[i], "-f")  == 0) ||
               (strcmp(argv[i], "-f1") == 0)) {
      outtype1 = IOH_FILE;
      outname1 = argv[i+1];
      i++;
    } else if (strcmp(argv[i], "-f2") == 0) {
      outtype2 = IOH_FILE;
      outname2 = argv[i+1];
      i++;
    } else if ((strcmp(argv[i], "-p")  == 0) ||
               (strcmp(argv[i], "-p1") == 0)) {
      outtype1 = IOH_PIPE;
      outname1 = argv[i+1];
      i++;
    } else if (strcmp(argv[i], "-p2") == 0) {
      outtype2 = IOH_PIPE;
      outname2 = argv[i+1];
      i++;
    } else if (strcmp(argv[i], "-s") == 0) {
      ScalingReport();
      exit(0);
    } else if (strcmp(argv[i], "-q") == 0) {
      quiet_mode = 1;
```

```
    } else {
      fprintf(stderr, "gendata: Don't understand argument: %s\n",argv[i]);
      exit(-1);
    }
  }

  /* ************************************************************** */
  /* Validate Command Line Arguments                                */
  /* ************************************************************** */

  /* Validate Table Argument */
  if (option < 3 || option > 11 || option == 10)
  {
    fprintf(stderr,"gendata: Invalid table selected: %d\n",option);
    exit(-1);
  }

  /* Validate Delimiter Argument */
  if (delim == NULL) {
    // default delimiter is used for IMPORT & LOAD, no changes neccessary
    using_rctload = 0;
  } else if (strlen(delim) == 1 && !isalnum(delim[0]) &&
             delim[0] != '.' && delim[0] != '%')
  {
    // user-supplied delimiter used for rctload
    InitFormatStrings(delim[0]);
    using_rctload = 1;
  } else {
    fprintf(stderr,"gendata: Invalid delimiter specified: %s\n",delim);
    exit(-1);
  }

  /* Validate File/Pipe Arguments */
  if (option != 9 && outtype1 > 0 && outtype2 > 0)
  {
    fprintf(stderr,"gendata: Specifying two output file/pipes allowed only when
generating\norders/orderline.\n");
    exit(-1);
  }
  if (option == 9 && ((outtype1 == 0) || (outtype2 == 0)))
  {
    fprintf(stderr,"gendata: Must specify two output file/pipes when generating orders/orderline.\n");
    exit(-1);
  }
  if (outtype1 == 0 || outname1 == NULL || strcmp(outname1,"") == 0)
  {
    fprintf(stderr,"gendata: Invalid 1st output file/pipe specified.\n");
    exit(-1);
  }
  if (option == 9 && (outtype2 == 0 || outname2 == NULL || strcmp(outname2,"") == 0))
  {
    fprintf(stderr,"gendata: Invalid 2nd output file/pipe specified.\n");
    exit(-1);
  }
  /* Ensure O/OL flat files are opened in append mode.  This is required */
  /* because we generate O/OL concurrently.  See comments in genload.pl  */
  /* for further details on why this is neccessary.                      */
  if (option == 9)
  {
    if (outtype1 == IOH_FILE) outtype1 = IOH_FILE_APPEND;
    if (outtype2 == IOH_FILE) outtype2 = IOH_FILE_APPEND;
  }

  /* Validate Range Arguments */
  if (ware_start <= 0 || ware_start > WAREHOUSES) {
    fprintf(stderr,"gendata: Invalid range starting value: %d\n",ware_start);
     exit(-1);
  }
  if (ware_end <= 0 || ware_end > WAREHOUSES || ware_end < ware_start) {
    fprintf(stderr,"gendata: Invalid range ending value: %d\n",ware_end);
    exit(-1);
  }

  initialize_random();

  /* ************************************************************** */
  /* Generate Data                                                  */
  /* ************************************************************** */
  switch (option) {
   case 3: /* WAREHOUSE */
     gen_ware_tbl();
     break;
   case 4: /* DISTRICT  */
     gen_dist_tbl();
     break;
   case 5: /* ITEM  */
     gen_item_tbl();
     break;
   case 6: /* STOCK     */
     gen_stock_tbl();
     break;
   case 7: /* CUSTOMER  */
     gen_cust_tbl();
     break;
   case 8: /* HISTORY   */
     gen_hist_tbl();
     break;
   case 9: /* ORDERS + ORDER_LINE */
     gen_ordr_tbl();
     break;
   case 11: /* NEW_ORDER */
     gen_nu_ord_tbl();
     break;
   case 2:
   case 10:
   default:
     fprintf(stderr, "Error: invalid option = %d \n",(option));
     break;
  }
  return 0;
}

/*----------------------------------------------------------------*/
/* generate item table                                            */
/*----------------------------------------------------------------*/

void gen_item_tbl( void )
{
  sqlint32  item_num = 0 ;
  sqlint32  item_im_id ;
  char  item_name[25] ;
  double  item_price ;
  char  item_data[51] ;

  IOH_NUM numBytes;
  ioHandle hnd;
  char Buffer[1024];

  timestamp1 = current_time();

  rc = GenericOpen(&hnd, outtype1, outname1);
  if (rc != 0) { goto item_done; }

  for(item_num = 1; item_num <= ITEMS; item_num++)
  {
    /* create image id field */
    item_im_id = rand_integer( 1, 10000 ) ;
    /* create name field */
    create_random_a_string( item_name, 14, 24);
    /* create price field */
    item_price = rand_decimal( 100, 10000, 2 ) ;
    /* create ORIGINAL field */
    create_a_string_with_original( item_data, 26, 50, 10) ;

    numBytes = sprintf(Buffer, fmtItem,
                       item_name,
                       item_price,
                       item_data,
                       item_im_id,
                       item_num);

    rc = GenericWrite(&hnd, Buffer, numBytes);
    if (rc != 0) { goto item_done; }

  } /* end for... */

  rc = GenericClose(&hnd);

item_done:

  timestamp2 = current_time();
  elapse = timestamp2 - timestamp1;
  if (rc == 0) {
    if (!quiet_mode) {
      fprintf(stdout,"\nITEM table generated in %8.2f seconds.\n\n",elapse);
      fflush(stdout);
    }
  } else {
    fprintf(stderr,"\nITEM table FAILED at (l %d) after %8.2f seconds.\n\n",item_num,elapse);
    fflush(stderr);
  }
}

/*----------------------------------------------------------------*/
/* generate stock table                                           */
/*----------------------------------------------------------------*/
void gen_stock_tbl( void )
{
  sqlint32 ware_num = 0 ;
  sqlint32 stock_num = 0 ;
  sqlint32 stock_quant;
  sqlint32 s_ytd;
  sqlint32 s_order_cnt, s_remote_cnt;
  char  stock_dist_01[25] ;
  char  stock_dist_02[25] ;
  char  stock_dist_03[25] ;
  char  stock_dist_04[25] ;
  char  stock_dist_05[25] ;
  char  stock_dist_06[25] ;
  char  stock_dist_07[25] ;
  char  stock_dist_08[25] ;
  char  stock_dist_09[25] ;
  char  stock_dist_10[25] ;
  char  stock_data[51] ;

  IOH_NUM numBytes;
  ioHandle hnd;
  char Buffer[1024];

  timestamp1 = current_time();

  rc = GenericOpen(&hnd, outtype1, outname1);
  if (rc != 0) { goto stock_done; }

  for (stock_num = 1; stock_num <= STOCK_PER_WAREHOUSE; stock_num++)
  {
    if (!quiet_mode && (stock_num%500 == 0))
    {
      fprintf(stdout, "STOCK for Item #%d\n",stock_num);
      fflush(stdout);
    }
    for (ware_num = ware_start; ware_num <= ware_end; ware_num++)
    {
      stock_quant = rand_integer( 10, 100 ) ;
      create_random_a_string( stock_dist_01, 24, 24);
      create_random_a_string( stock_dist_02, 24, 24);
      create_random_a_string( stock_dist_03, 24, 24);
      create_random_a_string( stock_dist_04, 24, 24);
      create_random_a_string( stock_dist_05, 24, 24);
```

```
        create_random_a_string( stock_dist_06, 24, 24);
        create_random_a_string( stock_dist_07, 24, 24);
        create_random_a_string( stock_dist_08, 24, 24);
        create_random_a_string( stock_dist_09, 24, 24);
        create_random_a_string( stock_dist_10, 24, 24);

        /* create ORIGINAL field */
        create_a_string_with_original( stock_data, 26, 50, 10 );
        s_ytd = s_order_cnt = s_remote_cnt = 0;

        numBytes = sprintf(Buffer, fmtStock,
                        s_remote_cnt,
                        stock_quant,
                        s_order_cnt,
                        s_ytd,
                        stock_data,
                        stock_dist_01,
                        stock_dist_02,
                        stock_dist_03,
                        stock_dist_04,
                        stock_dist_05,
                        stock_dist_06,
                        stock_dist_07,
                        stock_dist_08,
                        stock_dist_09,
                        stock_dist_10,
                        stock_num,
                        ware_num);

        rc = GenericWrite(&hnd, Buffer, numBytes);
        if (rc != 0) { goto stock_done; }

      } /* end for... */
    } /* end for... */

    rc = GenericClose(&hnd);

stock_done:

    timestamp2 = current_time();
    elapse = timestamp2 - timestamp1;
    if (rc == 0) {
      if (!quiet_mode) {
        fprintf(stdout,"\nSTOCK table generated in %8.2f seconds.\n\n",elapse);
        fflush(stdout);
      }
    } else {
      fprintf(stderr,"\nSTOCK table FAILED at (S %d W %d) after %8.2f seconds.\n\n",stock_num,
ware_num, elapse);
      fflush(stderr);
    }
}

/*----------------------------------------------------------------------*/
/* generate warehouse table                                             */
/*----------------------------------------------------------------------*/
void gen_ware_tbl( void )
{
    sqlint32 ware_num = 0 ;
    char ware_name[11] ;
    char ware_street_1[21] ;
    char ware_street_2[21] ;
    char ware_city[21] ;
    char ware_state[3]  ;
    char ware_zip[10]  ;
    double ware_tax ;
    double ware_YTD ;

    IOH_NUM numBytes;
    ioHandle hnd;
    char Buffer[1024];

    timestamp1 = current_time();

    rc = GenericOpen(&hnd, outtype1, outname1);
    if (rc != 0) { goto ware_done; }

    for (ware_num = ware_start; ware_num <= ware_end; ware_num++)
    {
      if (!quiet_mode && ((ware_num % 500) == 0)) {
        fprintf(stdout, "Warehouse #%d\n", ware_num);
        fflush(stdout);
      }

      create_random_a_string( ware_name,      6,10) ; /* create name */
      create_random_a_string( ware_street_1, 10,20) ; /* create street 1 */
      create_random_a_string( ware_street_2, 10,20) ; /* create street 2 */
      create_random_a_string( ware_city,     10,20) ; /* create city */
      create_random_a_string( ware_state,     2,2) ;  /* create state */
      create_random_n_string( ware_zip,       4,4) ;  /* create zip */
      strcat(ware_zip, "11111");

      ware_tax = rand_decimal(0, 2000,4);
      ware_YTD = 300000.00;

      numBytes = sprintf(Buffer, fmtWare,
                      ware_name,
                      ware_street_1,
                      ware_street_2,
                      ware_city,
                      ware_state,
                      ware_zip,
                              ware_tax,
                              ware_YTD,
                      ware_num);

      rc = GenericWrite(&hnd, Buffer, numBytes);
      if (rc != 0) { goto ware_done; }

    } /* end for */

    rc = GenericClose(&hnd);

ware_done:

    timestamp2 = current_time();
    elapse = timestamp2 - timestamp1;
    if (rc == 0) {
      if (!quiet_mode) {
        fprintf(stdout,"\nWAREHOUSE table generated in %8.2f seconds.\n\n",elapse);
        fflush(stdout);
      }
    } else {
      fprintf(stderr,"\nWAREHOUSE table FAILED at (W %d) after %8.2f seconds.\n\n",ware_num, elapse);
      fflush(stderr);
    }
}

/*----------------------------------------------------------------------*/
/* generate dist table                                                  */
/*----------------------------------------------------------------------*/
void gen_dist_tbl( void )
{
    sqlint32 ware_num = 0 ;
    sqlint32 dist_num = 0 ;
    char dist_name[11];
    char dist_street_1[21];
    char dist_street_2[21];
    char dist_city[21];
    char dist_state[3];
    char dist_zip[10];
    double dist_tax;
    sqlint32 next_o_id;
    double dist_YTD;

    IOH_NUM numBytes;
    ioHandle hnd;
    char Buffer[1024];

    next_o_id = CUSTOMERS_PER_DISTRICT + 1;
    timestamp1 = current_time();

    rc = GenericOpen(&hnd, outtype1, outname1);
    if (rc != 0) { goto dist_done; }

    for (ware_num = ware_start; ware_num <= ware_end; ware_num++)
    {
      if (!quiet_mode) {
        fprintf(stdout, "DISTRICT for Warehouse #%d\n", ware_num);
        fflush(stdout);
      }
      for (dist_num = 1; dist_num <= DISTRICTS_PER_WAREHOUSE; dist_num++)
      {
        create_random_a_string( dist_name,      6,10) ; /* create name */
        create_random_a_string( dist_street_1, 10,20) ; /* create street 1 */
        create_random_a_string( dist_street_2, 10,20) ; /* create street 2 */
        create_random_a_string( dist_city,     10,20) ; /* create city */
        create_random_a_string( dist_state,     2,2) ;  /* create state */
        create_random_n_string( dist_zip,       4,4) ;  /* create zip */
        strcat(dist_zip, "11111");
        dist_tax = rand_decimal(0, 2000,4);
        dist_YTD = 30000.00;

        numBytes = sprintf(Buffer, fmtDist,
                        next_o_id,
                        dist_tax,
                        dist_YTD,
                        dist_name,
                        dist_street_1,
                        dist_street_2,
                        dist_city,
                        dist_state,
                        dist_zip,
                        dist_num,
                        ware_num);

        rc = GenericWrite(&hnd, Buffer, numBytes);
        if (rc != 0) { goto dist_done; }

      } /* end for... */
    } /* end for... */

    rc = GenericClose(&hnd);

dist_done:

    timestamp2 = current_time();
    elapse = timestamp2 - timestamp1;
    if (rc == 0) {
      if (!quiet_mode) {
        fprintf(stdout,"\nDISTRICT table generated in %8.2f seconds.\n\n",elapse);
        fflush(stdout);
      }
    } else {
      fprintf(stderr,"\nDISTRICT table FAILED at (W %d D %d) after %8.2f
seconds.\n\n",ware_num,dist_num,elapse);
      fflush(stderr);
    }
}

/*----------------------------------------------------------------------*/
/* generate customer table                                              */
/*----------------------------------------------------------------------*/
void gen_cust_tbl( void )
{
    sqlint32 ware_num = 0 ;
    sqlint32 dist_num = 0 ;
    sqlint32 cust_num = 0 ;
    char cust_last[17];
```

```
char cust_middle[3];
char cust_first[17];
char cust_street_1[21];
char cust_street_2[21];
char cust_city[21];
char cust_state[3];
char cust_zip[10];
char cust_phone[17];
char cust_credit[3];
char cust_data[501];
char cust_since[27];
double cust_discount;
double cust_balance;
double cust_YTD_payment;
double cust_credit_lim;

IOH_NUM numBytes;
ioHandle hnd;
char Buffer[1024];
int len, pos;

timestamp1 = current_time();

rc = GenericOpen(&hnd, outtype1, outname1);
if (rc != 0) { goto cust_done; }

strcpy(cust_middle, "OE");

createTimestampString(cust_since);

for (cust_num = 1; cust_num <= CUSTOMERS_PER_DISTRICT; cust_num++)
{
  if (!quiet_mode) {
    fprintf(stdout, "CUSTOMER #%d:\n", cust_num);
    fflush(stdout);
  }

  for (ware_num = ware_start; ware_num <= ware_end; ware_num++)
  {
    for (dist_num = 1; dist_num <= DISTRICTS_PER_WAREHOUSE; dist_num++)
    {
      if (cust_num <= 1000)                    /* create last name */
        create_random_last_name( cust_last, cust_num);
      else                                     /* create last name */
        create_random_last_name( cust_last, 0);
      create_random_a_string( cust_first,     8,16) ; /* create first name */
      create_random_a_string( cust_street_1, 10,20) ; /* create street 1 */
      create_random_a_string( cust_street_2, 10,20) ; /* create street 2 */
      create_random_a_string( cust_city,     10,20) ; /* create city */
      create_random_a_string( cust_state,     2,2) ;  /* create state */
      create_random_n_string( cust_zip,       4,4) ;  /* create zip */
      strcat(cust_zip, "11111");

      /* create phone number */
      create_random_n_string( cust_phone, 16,16) ;
      if ( rand_integer( 1, 100 ) <= 10 )
        strcpy( cust_credit, "BC" ) ;
      else
        strcpy( cust_credit, "GC" ) ;

      /* create discount rate */
      cust_discount = rand_decimal(0,5000,4);

      /* create customer data */
      create_random_a_string(cust_data, 300, 500);

      /* pad customer data (only for non-rctload) */
      if (using_rctload == 0) {
        for (pos=strlen(cust_data); pos<500; pos++)
          cust_data[pos] = ' ';
        cust_data[500] = '\0';
      }
      cust_credit_lim = 50000.00;
      cust_balance = -10.00;
      cust_YTD_payment = 10.00;

      if (cust_num == 1 && dist_num == 1 && ware_num == 1)
      {
        sprintf(cust_first,"C_LAST_LOAD=%d",C_C_LAST_LOAD);
      }

      numBytes = sprintf(Buffer, fmtCust,
                         cust_num,
                         cust_state,
                         cust_zip,
                         cust_phone,
                         cust_since,
                         cust_credit_lim,
                         cust_middle,
                         cust_credit,
                         cust_discount,
                         cust_data,
                         cust_last,
                         cust_first,
                         cust_street_1,
                         cust_street_2,
                         cust_city,
                         dist_num,
                         ware_num,
                         0,
                         cust_balance,
                         cust_YTD_payment,
                         1);

      rc = GenericWrite(&hnd, Buffer, numBytes);
      if (rc != 0) { goto cust_done; }

    } /* end for district... */
  } /* end for warehouse... */
} /* end for customer... */

rc = GenericClose(&hnd);

cust_done:

timestamp2 = current_time();
elapse = timestamp2 - timestamp1;
if (rc == 0) {
  if (!quiet_mode) {
    fprintf(stdout,"\nCUSTOMER table generated in %8.2f seconds.\n\n",elapse);
    fflush(stdout);
  }
} else {
  fprintf(stderr,"\nCUSTOMER table FAILED at (W %d D %d C %d) after %8.2f seconds.\n\n",ware_num,
dist_num, cust_num, elapse);
  fflush(stderr);
}
}

/*----------------------------------------------------------------------*/
/* generate hist table                                                  */
/*----------------------------------------------------------------------*/
void gen_hist_tbl( void )
{
  sqlint32 ware_num = 0 ;
  sqlint32 dist_num = 0 ;
  sqlint32 cust_num = 0 ;
  char hist_data[25] ;
  char h_date[27] ;

  IOH_NUM numBytes;
  ioHandle hnd;
  char Buffer[1024];

  timestamp1 = current_time();

  rc = GenericOpen(&hnd, outtype1, outname1);
  if (rc != 0) { goto hist_done; }

  createTimestampString(h_date);

  for (ware_num = ware_start; ware_num <= ware_end; ware_num++)
  {
    if (!quiet_mode) {
      fprintf(stdout, "HISTORY for Warehouse #%d:\n", ware_num);
      fflush(stdout);
    }
    for (dist_num = 1; dist_num <= DISTRICTS_PER_WAREHOUSE; dist_num++)
    {
      for (cust_num = 1; cust_num <= CUSTOMERS_PER_DISTRICT; cust_num++)
      {
        /* create history data */
        create_random_a_string( hist_data, 12,24) ;

        numBytes = sprintf(Buffer, fmtHist,
                           cust_num,
                           dist_num,
                           ware_num,
                           dist_num,
                           ware_num,
                           h_date,
                                   10.00,
                           hist_data);

        rc = GenericWrite(&hnd, Buffer, numBytes);
        if (rc != 0) { goto hist_done; }

      } /* end for customer... */
    } /* end for district... */
  } /* end for warehouse... */

  rc = GenericClose(&hnd);

hist_done:

  timestamp2 = current_time();
  elapse = timestamp2 - timestamp1;
  if (rc == 0) {
    if (!quiet_mode) {
      fprintf(stdout,"\nHISTORY table generated in %8.2f seconds.\n\n",elapse);
      fflush(stdout);
    }
  } else {
    fprintf(stderr,"\nHISTORY table FAILED at (W %d D %d C %d) after %8.2f seconds.\n\n",ware_num,
dist_num, cust_num, elapse);
    fflush(stderr);
  }
}

/*----------------------------------------------------------------------*/
/* generate nu_ord table                                                */
/*----------------------------------------------------------------------*/
void gen_nu_ord_tbl( void )
{
  sqlint32 ware_num = 0 ;
  sqlint32 dist_num = 0 ;
  sqlint32 nu_ord_id = 0 ;
  int nu_ord_hi ;

  IOH_NUM numBytes;
  ioHandle hnd;
  char Buffer[1024];

  /* compute maximum and minimum
    order numbers for this
    district */
  nu_ord_hi = CUSTOMERS_PER_DISTRICT - NU_ORDERS_PER_DISTRICT + 1;
  if (nu_ord_hi < 0) {
```

```
    nu_ord_hi = CUSTOMERS_PER_DISTRICT - (CUSTOMERS_PER_DISTRICT / 3) + 1;
    fprintf(stderr,"\n**** WARNING **** NU_ORDERS_PER_DISTRICT is >
CUSTOMERS_PER_DISTRICT\n");
    fprintf(stderr,"                Check the values in file lval.h\n");
    fprintf(stderr,"                Loading New-Order with 1/3 of CUSTOMERS_PER_DISTRICT\n");
  }

  timestamp1 = current_time();

  rc = GenericOpen(&hnd, outtype1, outname1);
  if (rc != 0) { goto neword_done; }

  /* We generate in O/W/D order for non-RCT tables.  With the
   * data clustered on O_ID, this gives us good bufferpool
   * characteristics.  We also create a btree index in W/D/O
   * order, to satisfy MIN(O_ID) queries.
   *
   * For RCT tables *with* RCT Jump Cache, we *should* generate
   * the data in W/D/O order (to match the table definition.)
   * We don't since it would push schema decisions into flat file
   * generation (and I don't want to do that.)  It's just as easy
   * to sort the flat files afterwards.
   */

  for (nu_ord_id = nu_ord_hi;
       nu_ord_id <= CUSTOMERS_PER_DISTRICT;
       nu_ord_id++)
  {
    if (!quiet_mode) {
      fprintf(stdout, "NEW_ORDER for Customer #%d:\n", nu_ord_id);
      fflush(stdout);
    }
    for (ware_num = ware_start; ware_num <= ware_end; ware_num++)
    {
      for (dist_num = 1; dist_num <= DISTRICTS_PER_WAREHOUSE; dist_num++)
      {
        numBytes = sprintf(Buffer, fmtNewOrd,
                           nu_ord_id,
                           dist_num,
                           ware_num);

        rc = GenericWrite(&hnd, Buffer, numBytes);
        if (rc != 0) { goto neword_done; }

      } /* end for... */
    } /* end for... */
  } /* end for... */

  rc = GenericClose(&hnd);

neword_done:

  timestamp2 = current_time();
  elapse = timestamp2 - timestamp1;
  if (rc == 0) {
    if (!quiet_mode) {
      fprintf(stdout,"\nNEW_ORDER table generated in %8.2f seconds.\n\n",elapse);
      fflush(stdout);
    }
  } else {
    fprintf(stderr,"\nNEW_ORDER table FAILED at (W %d D %d O %d) after %8.2f
seconds.\n\n",ware_num, dist_num, nu_ord_id, elapse);
    fflush(stderr);
  }
}

/*----------------------------------------------------------------------*/
/* generate order and order_line tables                                 */
/*----------------------------------------------------------------------*/
void gen_ordr_tbl( void )
{
  sqlint32 ware_num = 0 ;
  sqlint32 dist_num = 0 ;
  sqlint32 cust_num = 0 ;
  sqlint32 ord_num = 0 ;
  sqlint32 ordr_carrier_id;
  sqlint32 ordr_ol_cnt;
  sqlint32 oline_ol_num;
  sqlint32 oline_item_num;

  double oline_amount;
  char oline_dist_info[25];

  IOH_NUM numBytes;
  ioHandle hnd1, hnd2;
  char Buffer[1024];

  char currtmstmp[27];
  char nulltmstmp[27] = "0001-01-01 00:00:00";

  oline_dist_info[24] = '\0';

  timestamp1 = current_time();

  rc1 = GenericOpen(&hnd1, outtype1, outname1);
  if (rc1 != 0) { goto ool_done; }
  rc2 = GenericOpen(&hnd2, outtype2, outname2);
  if (rc2 != 0) { goto ool_done; }

  createTimestampString(currtmstmp);

  for (ware_num = ware_start; ware_num <= ware_end; ware_num++)
  {
    if (!quiet_mode) {
      fprintf(stdout, "ORDERS & ORDER_LINE for Warehouse #%d\n", ware_num);
      fflush(stdout);
    }
    for (dist_num = 1; dist_num <= DISTRICTS_PER_WAREHOUSE; dist_num++)
    {
      if (!quiet_mode) {
        fprintf(stdout, "District #%d\t", dist_num);
        fflush(stdout);
      }

      seed_1_3000();

      for (ord_num = 1; ord_num <= CUSTOMERS_PER_DISTRICT; ord_num++)
      {
        if (ord_num < 2101)
          ordr_carrier_id = rand_integer( 1, 10 ) ;
        else
          ordr_carrier_id = 0;

        cust_num = random_1_3000();
        ordr_ol_cnt = rand_integer(MIN_OL_PER_ORDER,MAX_OL_PER_ORDER);

        numBytes = sprintf(Buffer, fmtOrdr,
                           cust_num,
                           currtmstmp,
                           ordr_carrier_id,
                           ordr_ol_cnt,
                           1,
                           ord_num,
                           ware_num,
                           dist_num);

        rc1 = GenericWrite(&hnd1, Buffer, numBytes);
        if (rc1 != 0) { goto ool_done; }

        for ( oline_ol_num = 1; oline_ol_num <= ordr_ol_cnt; oline_ol_num++ )
        {
          oline_item_num = rand_integer(1, ITEMS) ;
          create_random_a_string( oline_dist_info, 24, 24) ;

          numBytes = sprintf(Buffer, fmtOLine,
                             ((ord_num < 2101) ? currtmstmp : nulltmstmp),
                             ((ord_num < 2101) ? 0.00 : rand_decimal(1,999999,2)),
                             oline_item_num,
                             ware_num,
                             5,
                             oline_dist_info,
                             ord_num,
                             dist_num,
                             ware_num,
                             oline_ol_num);

          rc2 = GenericWrite(&hnd2, Buffer, numBytes);
          if (rc2 != 0) { goto ool_done; }

        } /* for order_line */
      } /* for order */
    } /* for dist */
  } /* for ware */

  rc1 = GenericClose(&hnd2);
  rc2 = GenericClose(&hnd1);

ool_done:

  timestamp2 = current_time();
  elapse = timestamp2 - timestamp1;
  if (rc1 == 0 && rc2 == 0) {
    if (!quiet_mode) {
      fprintf(stdout,"\nORDERS & ORDER_LINE tables generated in %8.2f seconds.\n\n",elapse);
      fflush(stdout);
    }
  } else {
    fprintf(stderr,"\nORDERS & ORDER_LINE tables FAILED at (W %d D %d O %d OL %d) after %8.2f
seconds.\n\n",ware_num, dist_num, ord_num, oline_ol_num, elapse);
    fflush(stderr);
  }
}

// This routine will initalize the printf format strings and replace the
// delimiter with the one provided.  The pipe symbol is the default.
void InitFormatStrings(char delim)
{
  char *p;

  // Check if Using Default Delimiter
  if (delim == '|') return;

  // Replace Delimiters
  while (p = strchr(fmtWare,'|'))   { *p = delim; }
  while (p = strchr(fmtDist,'|'))   { *p = delim; }
  while (p = strchr(fmtItem,'|'))   { *p = delim; }
  while (p = strchr(fmtStock,'|'))  { *p = delim; }
  while (p = strchr(fmtCust,'|'))   { *p = delim; }
  while (p = strchr(fmtHist,'|'))   { *p = delim; }
  while (p = strchr(fmtOrdr,'|'))   { *p = delim; }
  while (p = strchr(fmtOLine,'|'))  { *p = delim; }
  while (p = strchr(fmtNewOrd,'|')) { *p = delim; }
}

void ScalingReport(void)
{
  /* Print Scaling Values */
  fprintf(stdout,"Scaling Values in Use\n");
  fprintf(stdout,"-----------------------------\n");
  fprintf(stdout,"Warehouses:           %d\n", WAREHOUSES);
  fprintf(stdout,"Districts/Warehouse:   %d\n", DISTRICTS_PER_WAREHOUSE);
  fprintf(stdout,"Customers/District:    %d\n", CUSTOMERS_PER_DISTRICT);
  fprintf(stdout,"Items:                 %d\n", ITEMS);
  fprintf(stdout,"Stock/Warehouse:       %d\n", STOCK_PER_WAREHOUSE);
  fprintf(stdout,"Min Order Lines/Order: %d\n", MIN_OL_PER_ORDER);
  fprintf(stdout,"Max Order Lines/Order: %d\n", MAX_OL_PER_ORDER);
  fprintf(stdout,"New Orders/District:   %d\n", NU_ORDERS_PER_DISTRICT);
  fprintf(stdout,"-----------------------------\n");
}
```

dbgen/tpccrnd.c

```
/*****************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****************************************************************************/

/*
 * tpccrnd.c - Random generation functions for TPC-C
 *
 */

#include <stdio.h>
#include <string.h>
#include <math.h>
#include "db2tpcc.h"
#include "tpccmisc.h"
#include "lval.h"

static char tbl_cust[CUSTOMERS_PER_DISTRICT];

static char alnum[] =
   "0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ";

static char *last_name_parts[] =
{
   "BAR",
   "OUGHT",
   "ABLE",
   "PRI",
   "PRES",
   "ESE",
   "ANTI",
   "CALLY",
   "ATION",
   "EING"
};

/*
*****************************************************************************
*   rand_integer
*
*   create a uniform random numeric value of type integer, of random
*   value between lo and hi. Number is NOT placed in BUFFER, and IS
*   simply RETURNED.
*
*   Routine RETURNS the VALUE.
*
*   parameters
*   ----------
*   lo end of acceptable value range
*   hi end of acceptable value range
*
*   output
*   ------
*   random integer value  RETURNED
*
*****************************************************************************
*/

int rand_integer ( int val_lo, int val_hi )
{
  return((random()%(val_hi-val_lo+1))+val_lo);
}

/*
*****************************************************************************
*   rand_decimal
*
*   create a uniform random numeric value of type double, of random
*   value between lo and hi with val_dec fractional digits.
*   Number is NOT placed in BUFFER, and IS simply RETURNED.
*
*   Routine RETURNS the VALUE.
*
*   parameters
*   ----------
*   lo end of acceptable value range
*   hi end of acceptable value range
*   number of fractional digits
*
*   output
*   ------
*   random double value  RETURNED
*
*****************************************************************************
*/

double rand_decimal ( int val_lo, int val_hi, int val_dec )
{
  return(rand_integer(val_lo,val_hi)/pow(10.0,(double)val_dec));
}

/*
*****************************************************************************
*   seed_1_3000
*
*
*****************************************************************************
*/

void seed_1_3000( void )
{
  int i;

  for (i = 0; i < CUSTOMERS_PER_DISTRICT; i++) {
    tbl_cust[i] = 0;
  }
}

/*
*****************************************************************************
*   random_1_3000
*
*
*****************************************************************************
*/

int random_1_3000( void )
{
  static int i;
  static int x;

  x = rand_integer(0, CUSTOMERS_PER_DISTRICT - 1);

  for (i = 0; i < CUSTOMERS_PER_DISTRICT; i++)
  {
    if (tbl_cust[x] == 0)
    {
      tbl_cust[x] = 1;
      return(x+1);
    } else {
      x++;
    }
    if (x == CUSTOMERS_PER_DISTRICT)
      x=0;
  }

  printf("\nfatal error in random_1_3000 \n");
  abort();
}


/*
*****************************************************************************
*   initialize_random
*****************************************************************************
*/

void initialize_random(void)
{
  int t = current_time();

  srand(t);
  srandom(t);
}


/*
*****************************************************************************
*   create_random_a_string
*
*   create a random alphanumeric string, of random length between lo and
*   hi and place them in designated buffer. Routine returns the actual
*   length.
*
*   parameters
*   ----------
*   lo end of acceptable length range
*   hi end of acceptable length range
*
*   output
*   ------
*   actual length
*   random alphanumeric string
*
*****************************************************************************
*/

int create_random_a_string( char *out_buffer, int length_lo, int length_hi )
{
  int i, actual_length ;

  actual_length = rand_integer( length_lo, length_hi ) ;

  for (i = 0; i < actual_length; i++ )
  {
    out_buffer[i] = alnum[rand_integer( 0, 61 )] ;
  }
  out_buffer[actual_length] = '\0' ;

  return (actual_length);
}


/*
*****************************************************************************
*   create_random_n_string
*
*   create a random numeric string, of random length between lo and
*   hi and place them in designated buffer. Routine returns the actual
*   length.
*
*   parameters
*   ----------
*   lo end of acceptable length range
```

```
*   hi end of acceptable length range
*
*   output
*   ------
*   actual length
*   random numeric string
*
*****************************************************************************
*/

int create_random_n_string( char *out_buffer, int length_lo, int length_hi )
{
  int i, actual_length ;

  actual_length = rand_integer( length_lo, length_hi ) ;

  for (i = 0; i < actual_length; i++ )
  {
    out_buffer[i] = (char)rand_integer( 48,57 ) ;
  }
  out_buffer[actual_length] = '\0' ;

  return (actual_length);
}

/*
*****************************************************************************
*   NURand_val
*
*   create a non-uniform random numeric value of type integer, of random
*   value between lo and hi. Number is NOT placed in BUFFER, and IS
*   simply RETURNED.
*
*   Routine RETURNS the VALUE.
*
*   parameters
*   ----------
*   lo end of acceptable value range
*   hi end of acceptable value range
*
*   output
*   ------
*   random integer value  RETURNED
*
*****************************************************************************
*/

int NURand_val ( int A, int x, int y, int C )
{
  return((((rand_integer(0,A)|rand_integer(x,y))+C)%(y-x+1))+x);
}

/*
*****************************************************************************
*   create_a_string_with_original
*
*   create a random alphanumeric string, of random length between lo and
*   hi and place them in designated buffer. Routine returns the actual
*   length.
*
*   the word "ORIGINAL" is placed at a random location in the buffer at
*   random, for a given percent of the records.
*
*   percent_to_set must be an integer value from 0 to 100.
*   if 0, no records will be set. If 100, all records will be set.
*
*   CANNOT USE ON STRINGS OF LENGTH LESS THAN 8 ! LOWER LIMIT MUST BE > 8 !
*
*   parameters
*   ----------
*   lo end of acceptable length range
*   hi end of acceptable length range
*   percentage of records to set to ORIGINAL
*
*   output
*   ------
*   actual length
*   random alphanumeric string with the word "ORIGINAL" is placed at a
*       random location
*
*****************************************************************************
*/

int create_a_string_with_original( char *out_buffer, int length_lo,
                                   int length_hi, int percent_to_set )
{
  int actual_length, start_pos ;

  actual_length = create_random_a_string( out_buffer, length_lo, length_hi ) ;

  if ( rand_integer( 1, 100 ) <= percent_to_set )
  {
    start_pos = rand_integer( 0, actual_length-8 ) ;
    strncpy(out_buffer+start_pos,"ORIGINAL",8) ;
  }

  return (actual_length);
}


/*****************************************************************************
 *
 * create_random_last_name
 *
 * parameters:
 *    out_buffer - target buffer for the generated last name
 *
 * description:
 *    create_random_last_name generates a random number from 0 to 999
 *    inclusive. a random name is generated by associating a random string
 *    with each digit of the generated number. the three strings are
 *    concatenated to generate the name
 *
 *****************************************************************************/

int create_random_last_name(char *out_buffer, int cust_num)
{
  int random_num;

  if (cust_num == 0)
    random_num = NURand_val( A_C_LAST, 0, 999, C_C_LAST_LOAD );
  else
    random_num = cust_num - 1;

  strcpy(out_buffer, last_name_parts[random_num / 100]);
  random_num %= 100;
  strcat(out_buffer, last_name_parts[random_num / 10]);
  random_num %= 10;
  strcat(out_buffer, last_name_parts[random_num]);

  return(strlen(out_buffer));
}
```

include/db2tpcc.h

```
/*****************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****************************************************************************/

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

#ifndef __DB2TPCC_H
#define __DB2TPCC_H

#include <sys/types.h>

#include "lval.h"

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

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

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

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

#define MIN_WAREHOUSE 1
#define MAX_WAREHOUSE WAREHOUSES

/*****************************************************************************/
/* NURand Constants                                                         */
/* C_C_LAST_RUN and C_C_LAST_LOAD must adhere to clause 2.1.6.              */
/* Analysis indicates that a C_LAST delta of 85 is optimal.                 */
/*****************************************************************************/
#define C_C_LAST_RUN          88
#define C_C_LAST_LOAD         173
#define C_C_ID            319
#define C_OL_I_ID         3849
#define A_C_LAST          255
#define A_C_ID            1023
#define A_OL_I_ID         8191

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

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

#define SPGENERAL_PAD 3
#define SPGENERAL_ADJUST sizeof(int16_t)

struct in_neword_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  struct   in_items_struct {
```

```
    int32_t  s_OL_I_ID;
    int32_t  s_OL_SUPPLY_W_ID;
    int16_t  s_OL_QUANTITY;
    int16_t  pad1[3];
  } in_item[15];
  int32_t  s_C_ID;
  int32_t  s_W_ID;
  int16_t  s_D_ID;
  int16_t  s_O_OL_CNT;          /* init by SUT */
  int16_t  s_all_local;
  int16_t  duplicate_items;
};

struct out_neword_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  struct   items_struct {
    float  s_I_PRICE;
    float  s_OL_AMOUNT;
    int16_t  s_S_QUANTITY;
    int16_t  pad2;
    char     s_I_NAME[25];
    char     s_brand_generic;
  } item[15];
  float  s_W_TAX;
  float  s_D_TAX;
  float  s_C_DISCOUNT;
  float  s_total_amount;
  int32_t  s_O_ID;
  int16_t  s_O_OL_CNT;
  int16_t  s_transtatus;
  int16_t  deadlocks;
  char     s_C_LAST[17];
  char     s_C_CREDIT[3];
  char     s_O_ENTRY_D_time[27];

};

struct in_payment_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  float  s_H_AMOUNT;
  int32_t  s_W_ID;
  int32_t  s_C_W_ID;
  int32_t  s_C_ID;
  int16_t  s_C_D_ID;
  int16_t  s_D_ID;
  char     s_C_LAST[17];
};

struct out_payment_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  double  s_C_CREDIT_LIM;
  double  s_C_BALANCE;
  float  s_C_DISCOUNT;
  int32_t  s_C_ID;
  int16_t  s_transtatus;
  int16_t  deadlocks;
  char     s_W_STREET_1[21];
  char     s_W_STREET_2[21];
  char     s_W_CITY[21];
  char     s_W_STATE[3];
  char     s_W_ZIP[10];
  char     s_D_STREET_1[21];
  char     s_D_STREET_2[21];
  char     s_D_CITY[21];
  char     s_D_STATE[3];
  char     s_D_ZIP[10];
  char     s_C_FIRST[17];
  char     s_C_MIDDLE[3];
  char     s_C_LAST[17];
  char     s_C_STREET_1[21];
  char     s_C_STREET_2[21];
  char     s_C_CITY[21];
  char     s_C_STATE[3];
  char     s_C_ZIP[10];
  char     s_C_PHONE[17];
  char     s_C_CREDIT[3];
  char     s_C_DATA[201];
  char     s_H_DATE_time[27];
  char     s_C_SINCE_time[27];
};

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

struct out_ordstat_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  double  s_C_BALANCE;
  int32_t  s_C_ID;
  int32_t  s_O_ID;
  int16_t  s_O_CARRIER_ID;
  int16_t  s_ol_cnt;
  int16_t  pad1[2];
  struct   oitems_struct {
    double  s_OL_AMOUNT;
    int32_t  s_OL_I_ID;
    int32_t  s_OL_SUPPLY_W_ID;
    int16_t  s_OL_QUANTITY;
    int16_t  pad2;
    char     s_OL_DELIVERY_D_time[27];
  } item[15];
  int16_t  s_transtatus;
  int16_t  deadlocks;
  char     s_C_FIRST[17];
  char     s_C_MIDDLE[3];
  char     s_C_LAST[17];
  char     s_O_ENTRY_D_time[27];
  int16_t  pad3[2];
};

struct in_delivery_struct {
  int16_t  len;
  int16_t  pad[SPGENERAL_PAD];
  int32_t  s_W_ID;
  int16_t  s_O_CARRIER_ID;
};

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

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

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

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

#ifdef __cplusplus
extern "C" {
#endif

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

#ifdef __cplusplus
}
#endif

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

#ifdef __cplusplus
extern "C" {
#endif

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


#ifdef __cplusplus
}
#endif

#endif // __DB2TPCC_H
```

_include/lval.h

```
/* lval.h - generated automatically at 20080316.1422 */

#ifndef __LVAL_H
#define __LVAL_H
#define WAREHOUSES 51968
#define DISTRICTS_PER_WAREHOUSE 10
#define CUSTOMERS_PER_DISTRICT 3000
#define ITEMS 100000
#define STOCK_PER_WAREHOUSE 100000
#define MIN_OL_PER_ORDER 5
#define MAX_OL_PER_ORDER 15
#define NU_ORDERS_PER_DISTRICT 900
#endif // __LVAL_H
```

_include/platform.h

```
/*****************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
```

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

/*
 * platform.h - Platform Isolation Layer
 *
 */

#ifndef __PLATFORM_H
#define __PLATFORM_H

/* *******************************************************************/
/* Generic Macros                                                    */
/* *******************************************************************/
#define GEN_ERRCODE          errno

/* *******************************************************************/
/* Windows I/O Macros                                                */
/* *******************************************************************/

/* *******************************************************************/
/* UNIX I/O Macros                                                   */
/* *******************************************************************/
#include <fcntl.h>

#define IOH_INIT(hnd, type, name)                                  \
  hnd->fd = -1;                                                    \
  hnd->type = type;                                                \
  hnd->name = name;

#define IOH_CREATE(hnd)                                            \
  if (hnd->type == IOH_PIPE) {                                     \
    rc = mkfifo(hnd->name, 0666);                                  \
  } else {                                                         \
    rc = 0;                                                        \
  }

#define IOH_OPEN(hnd)                                              \
  if (hnd->type == IOH_FILE_APPEND) {                              \
    hnd->fd = open(hnd->name, O_WRONLY | O_CREAT | O_APPEND, 0666);      \
  } else {                                                         \
    hnd->fd = open(hnd->name, O_WRONLY | O_CREAT | O_TRUNC, 0666);       \
  }                                                                \
  if (hnd->fd == -1) {                                             \
    rc = -1;                                                       \
  } else {                                                         \
    rc = 0;                                                        \
  }

#define IOH_WRITE(hnd, buff, num, num2)                            \
  rc = write(hnd->fd, buff, num);                                  \
  if (rc >= 0) {                                                   \
    num2 = rc;                                                     \
    rc = 0;                                                        \
  }

#define IOH_FLUSH(hnd)          rc = 0;
#define IOH_CLOSE(hnd)          rc = close(hnd->fd);
#define IOH_DELETE(hnd)         if (hnd->type == IOH_PIPE) { rc = unlink(hnd->name); }

typedef unsigned int IOH_NUM;
typedef int IOH_HND;

/*******************************************************************/
/* UNIX Semaphore Macros                                           */
/*******************************************************************/
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/sem.h>

union semun {
  int val;
  struct semid_ds *buf;
  unsigned short int *array;
} semUnion;

struct sembuf semBuf;

#define SEM_HANDLE int

#define SEM_INIT(hnd, x, name)                                     \
  if ( (hnd = semget(IPC_PRIVATE, 1 , IPC_CREAT | IPC_EXCL | S_IRUSR | S_IWUSR | S_IRGRP |
S_IWGRP | S_IROTH | S_IWOTH)) == -1)                               \
      API_ERROR(__LINE__, "semget", (rc=GEN_ERRCODE));             \
  semUnion.val = x;                                                \
  if ( semctl(hnd, 0, SETVAL, semUnion) < 0 )                      \
      API_ERROR(__LINE__, "semctl SETVAL", (rc=GEN_ERRCODE));

#define SEM_WAIT(hnd)                                              \
  semBuf.sem_num = 0;                                              \
  semBuf.sem_op = -1;                                              \
  semBuf.sem_flg = SEM_UNDO;                                       \
  if ( semop(hnd, &semBuf, 1) < 0 )                                \
      API_ERROR(__LINE__, "semop wait", (rc=GEN_ERRCODE));

#define SEM_FREE(hnd)                                              \
  semBuf.sem_num = 0;                                              \
  semBuf.sem_op = 1;                                               \
  semBuf.sem_flg = SEM_UNDO;                                       \
  if ( semop(hnd, &semBuf, 1) < 0 )                                \
      API_ERROR(__LINE__, "semop free", (rc=GEN_ERRCODE));

#define SEM_DESTROY(hnd)                                           \
  if ( semctl(hnd, 0, IPC_RMID, 0) )                               \
      API_ERROR(__LINE__, "semctl IPC_RMID", (rc=GEN_ERRCODE));


/* *******************************************************************/
/* Common I/O Macros and Definitions                                 */
/* *******************************************************************/
#define IOH_FILE 1
#define IOH_PIPE 2
#define IOH_FILE_APPEND 3

#define IOH_ERRMSG(hnd, msg)                                       \
  if (rc != 0) {                                                   \
    fprintf(stderr, "Error %d %s fd %d (%d, %s)\n", GEN_ERRCODE, msg,     \
                hnd->fd, hnd->type, hnd->name);                    \
    return rc;                                                     \
  }

struct _ioh {
  IOH_HND fd;
  int type;
  char *name;
};

typedef struct _ioh ioHandle;

/* *******************************************************************/
/* Generic I/O Routine Prototypes                                    */
/* *******************************************************************/
int GenericOpen(ioHandle *hnd, int type, char *name);
int GenericWrite(ioHandle *hnd, char *Buffer, unsigned int numBytes);
int GenericClose(ioHandle *hnd);

/* *******************************************************************/
/* Generic I/O Routines                                              */
/* *******************************************************************/
int GenericOpen(ioHandle *hnd, int type, char *name)
{
  int rc = 0;

  IOH_INIT(hnd, type, name)

  IOH_CREATE(hnd)
  IOH_ERRMSG(hnd, "creating")

  IOH_OPEN(hnd)
  IOH_ERRMSG(hnd, "opening")

  return rc;
}

int GenericWrite(ioHandle *hnd, char *Buffer, unsigned int numBytes)
{
  int rc = 0;
  int numBytesWritten = -1;

  IOH_WRITE(hnd, Buffer, numBytes, numBytesWritten)
  IOH_ERRMSG(hnd, "writing")
  if (numBytes != numBytesWritten) {
    fprintf(stderr, "Truncated data writing to fd %d (%d, %s)\n", hnd->fd, hnd->type, hnd->name);
    rc = -1;
  }
  return rc;
}

int GenericClose(ioHandle *hnd)
{
  int rc = 0;

  IOH_FLUSH(hnd)
  IOH_ERRMSG(hnd, "flushing")

  IOH_CLOSE(hnd)
  IOH_ERRMSG(hnd, "closing")

  IOH_DELETE(hnd)
  IOH_ERRMSG(hnd, "deleting")

  return rc;
}

#endif // __PLATFORM_H
```

include/tpccmisc.h

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

/*
 * tpccmisc.h - Miscellanouse Routines
 *
 */

#ifndef __TPCCMISC_H
#define __TPCCMISC_H


extern double current_time_ms(void);
```

```
extern double current_time(void);

#include <time.h>
#define createTimestampString(buf)                    \
{                                                     \
  time_t now;                                         \
  struct tm *tm;                                      \
  time(&now);                                         \
  tm = localtime(&now);                               \
  sprintf(buf,                                        \
          "%4.4d-%2.2d-%2.2d %2.2d:%2.2d:%2.2d",      \
          tm->tm_year + 1900, tm->tm_mon + 1, tm->tm_mday, \
          tm->tm_hour, tm->tm_min, tm->tm_sec);       \
}

#endif // __TPCCMISC_H
```

_include/tpccrnd.h

```
/*****************************************************************************
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****************************************************************************/

/*
 * tpccrnd.h - Random generation functions for TPC-C
 *
 */

#ifndef __TPCCRND_H
#define __TPCCRND_H

void initialize_random(void);
int rand_integer( int val_lo, int val_hi ) ;
double rand_decimal( int val_lo, int val_hi, int val_dec ) ;
int NUrand_val( int A, int val_lo, int val_hi, int C ) ;

void seed_1_3000( void );
int random_1_3000( void );

int create_random_a_string( char *out_buffer,
                            int length_lo,
                            int length_hi );
int create_random_n_string( char *out_buffer,
                            int length_lo,
                            int length_hi );
int create_a_string_with_original( char *out_buffer,
                                   int length_lo,
                                   int length_hi,
                                   int percent_to_set ) ;
int create_random_last_name(char *out_buffer, int cust_num);

#endif // __TPCCRND_H
```

_tpccenv.sh

```
##############################################################################
## Licensed Materials - Property of IBM
##
## Governed under the terms of the International
## License Agreement for Non-Warranted Sample Code.
##
## (C) COPYRIGHT International Business Machines Corp. 1996 - 2006
## All Rights Reserved.
##
## US Government Users Restricted Rights - Use, duplication or
## disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
##############################################################################

#
# tpccenv.sh - UNIX Environment Setup
#

# The Kit Version
export TPCC_VERSION=CK080131

# The DB2 Instance Name (for DB2)
export DB2INSTANCE=${USER}

# The OS being used (i.e. "UNIX", "LINUX", "WINDOWS")
export PLATFORM=UNIX
export SERVER_PLATFORM=UNIX

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

# Specifies whether or not to use dari stored proc's for the TPC-C driver.  Set to either DARIVERSION or
NONDARI;
#export TPCC_SPTYPE=NOSP
#export TPCC_SPTYPE=SPGENERAL2
export TPCC_SPTYPE=SPGENERAL
#export TPCC_SPTYPE=DARI2SQLDA

export DB2VERSION=v9

# The schema name is typically the SQL autorization ID (or username).
# This is required for runstats and EEE.
export TPCC_SCHEMA=${USER}
export SERVER_TPCC_SCHEMA=${USER}

# DB2 EE/EEE Configuration
export DB2EDITION=EE
#export DB2EDITION=EEE
export DB2NODE=0
export DB2NODES=1;              # set to the number of nodes you have.  Set to 1 for EE.

# TPCC General Configuration
export TPCC_DBNAME=TPCC
export TPCC_ROOT=${HOME}/tpc-c.ibm
export TPCC_SQLLIB=${HOME}/sqllib
export TPCC_RUNDATA=${HOME}/tpccdata

# TPCC Debug Configuration
# This is the path where all error and debug logs are placed.
# To get debugging from within the stored procedures, you must
# set DB2ENVLIST="TPCC_DEBUGDIR" in tpcc.config.
export TPCC_DEBUGDIR=/tmp

# Specifies where stored procedures should be placed and if they should
# be fenced.
export TPCC_SPDIR=${TPCC_SQLLIB}/function
export TPCC_FENCED=NO
```

Appendix - D: Pricing Information

Microsoft Corporation
One Microsoft Way
Redmond, WA 98052-6399

Tel 425 882 8080
Fax 425 936 7329
http://www.microsoft.com/

Microsoft

May 28, 2008

IBM Corporation
Lotus Douglas
11501 Burnet Road
Austin, TX 78758

Here is the information you requested regarding pricing for several Microsoft products to be used in conjunction with your TPC-C benchmark testing.

All pricing shown is in US Dollars ($).

Part Number	Description	Unit Price	Quantity	Price
P70-00275	**Windows Server 2003 Web Edition** *Full Server License* *Discount Schedule: No Discounts Applied*	$399	1	$399
127-00012	**Visual Studio Standard 2005** *Full License* *No Discount Applied*	$250	1	$250
N/A	**Microsoft Problem Resolution Services** *Professional Support* *(1 Incident)*	$245	1	$245

All products are currently orderable through Microsoft's normal distribution channels. A list of Microsoft's resellers can be found at **http://www.microsoft.com/products/info/render.aspx?view=22&type=mnp&content=22/licensing**

Defect support is included in the purchase price. Additional support is available from Microsoft PSS on an incident by incident basis at $245 per call.

This quote is valid for the next 90 days.

If we can be of any further assistance, please contact Jamie Reding at
(425) 703-0510 or jamiere@microsoft.com.

Reference ID: PClodo0805280000037580.
Please include this Reference ID in any correspondence regarding this price quote.

Compagnie IBM France
Tour Descartes – La Défense 5
2, avenue Gambetta
92066 Paris La Défense Cedex

Bull
Rue Jean Jaures
78340 Les Clayes-sous-Bois

Date: May 27, 2008

Dear Mr Claude Hernandez,

Thank you for your interest in IBM Hardware. Please find the requested quote for the IBM storage and X system packages corresponding to the referenced TPC benchmark:

Description	Part Number	Unit Price	Quantity	Ext Price
IBM Storage				
IBM System Storage DS3400 Express	1726-42E	8749	34	297 466
IBM 1M SAS cable	39R6529	119	192	22 848
IBM System Storage EXP3000	1727-01X	3199	96	307 104
IBM Hot-Swap 3.5 inch 73.4GB 15K SAS HDD	40K1043	309	1550	478 950
IBM TotalStorage SAN32B-3	2005-B5K	15810	3	47 430
8-port Activation	22R5505	8100	6	48 600
4 Gbps SW SFP Transceivers - 4 pack	2414	550	24	13 200
IBM S2 42U Standard Rack	93074RX	1489	7	10 423
ServicePac for 3-Year 24x7x4 Support (DS3400)	44J8073	1300	34	44 200
ServicePac for 3-Year 24x7x4 Support (EXP3000)	41L2768	760	96	72 960
ServicePac for 3-Year 24x7x4 Support (SAN32B-2)	41E9167	3300	3	9 900
ServicePac for 3-Year 24x7x4 Support (Rack)	41L2760	300	7	2 100
			Subtotal	**$ 1 355 181**
IBM Xservers				
IBM System x3550 (Dual-core Xeon 2.0GHz 4MB L2 Cache)	7878AC1	1927	16	30 832
512MB PC2-5300 CL5 ECC DDR2 Chipkill FBDIMM 667MHz	546	115	32	3 680
73GB 15K Hot-Swap SAS HDD	5161	309	16	4 944
ServicePac for 3-Year 24x7x4 Support	96P2250	586	16	9 376
NetBAY S24 2U Standard Rack Cabinet	93074RX	1489	1	1 489
Optical 3-Button Mouse-USB	40K9201	19	1	19
Preferred Pro FullSize PS/2 Keyboard	40K9584	29	1	29
IBM C117 17" CRT Monitor	49387NU	149	1	149
			Subtotal	**$ 50 518**

SAS au capital de 542 323 943 ,70 Euros
Siège social : Tour Descartes – La Défense 5
2, avenue Gambetta – 92400 Courbevoie
552 118 465 RCS Nanterre
Informations sur le site : www.ibm.com
Siret 552 118 465 02430 – Tel. 01 49 05 70 00

Total	**$**	**1 405 699**
Discount	$	421 710
Discount rate		30%

Effective Date: May 27, 2008
Termination Date: September 30, 2008

Philippe Choplin
Bull Account Executive
IBM

Compagnie IBM France
Tour Descartes – La Défense 5
2, avenue Gambetta
92066 Paris La Défense Cedex

Bull
Rue Jean Jaures
78340 Les Clayes-sous-Bois

Date: June 6, 2008

Dear Mr Claude Hernandez,

Thank you for your interest in IBM Software. Please find the requested quote for the IBM storage packages corresponding to the referenced TPC benchmark:

Description	Part Number	Unit Price	Quantity	Ext Price
IBM DB2 DataBase				
DB2 9.5 ESE Lic&Mtce (278.52 perVU,p6 VU rating-120, 8 procs)		279	960	267 379
DB2 9.5 ESE MtceRenewal (13.27 perVU,8 procs, 2 years)		13	1920	25 478
			Total	**$ 292 857**
			Discount	$ 108 357
			Discount rate	37%

Effective Date: June 6, 2008
Termination Date: September 30, 2008

Philippe Choplin
Bull Account Executive
IBM

SAS au capital de 542 323 943 ,70 Euros
Siège social : Tour Descartes – La Défense 5
2, avenue Gambetta – 92400 Courbevoie
552 118 465 RCS Nanterre
Informations sur le site : www.ibm.com
Siret 552 118 465 02430 – Tél. 01 49 05 70 00