IBM System p 570
Model 9117-MMA
Using
AIX 5L Version 5.3
and
DB2 Enterprise 9

TPC Benchmark™ C
Full Disclosure Report

IBM
First Edition
May 21, 2007
Special Notices

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 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: May 21, 2007

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 IBM 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 IBM licensed program are not intended to state or imply that only IBM'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, IBM 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 IBM 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:

TPC Benchmark Administrator
IBM Commercial Performance
Mail Stop 9571
11501 Burnet Road
Austin, TX 78758
FAX Number (512) 838-1852

© Copyright International Business Machines Corporation, 2007. 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.

NOTE: US. Government Users - Documentation related to restricted rights: Use, duplication, or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.
**IBM System p 570 Model 9117-MMA**

**TPC-C Rev. 5.8**

**Report Date:** May 21, 2007

<table>
<thead>
<tr>
<th>Total System Cost</th>
<th>TPC-C Throughput</th>
<th>Price/Performance</th>
<th>Availability Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>5,713,181 USD</td>
<td>1,616,162</td>
<td>$3.54 USD</td>
<td>November 21, 2007</td>
</tr>
</tbody>
</table>

**Database Server**
- Processor Chip/Core/Thread: 8/16/32 POWER6 4.7GHz
- Database Manager: DB2 9.1
- Operating System: AIX 5L V5.3
- Other Software: Microsoft Visual C++, Microsoft COM+
- No. Users: 1,279,200

---

**System Components**

<table>
<thead>
<tr>
<th>System Components</th>
<th>Server</th>
<th>Each of the 64 Clients</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Quantity</td>
<td>Description</td>
</tr>
<tr>
<td><strong>Processors Chips /Cores/Threads</strong></td>
<td>8/16/32</td>
<td>4.7GHz POWER6™</td>
</tr>
<tr>
<td><strong>Memory</strong></td>
<td>24</td>
<td>32 GB</td>
</tr>
<tr>
<td><strong>Disk Controllers</strong></td>
<td>4</td>
<td>Integrated SAS</td>
</tr>
<tr>
<td></td>
<td>21</td>
<td>4Gb Dual-port FC Adapters</td>
</tr>
<tr>
<td></td>
<td>21</td>
<td>IBM System Storage DS4800</td>
</tr>
<tr>
<td><strong>Disk Drives</strong></td>
<td>3312</td>
<td>36.4GB 15K RPM 4Gb FC</td>
</tr>
<tr>
<td></td>
<td>168</td>
<td>73.4GB 15K RPM 4Gb FC</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>73.4GB 15K RPM SAS</td>
</tr>
<tr>
<td><strong>Total Storage</strong></td>
<td>117.05 TB</td>
<td></td>
</tr>
<tr>
<td><strong>Terminals</strong></td>
<td>1</td>
<td>System Console</td>
</tr>
<tr>
<td>Description</td>
<td>Part No.</td>
<td>Source</td>
</tr>
<tr>
<td>-----------------------------------------------------------------------------</td>
<td>------------</td>
<td>--------</td>
</tr>
<tr>
<td><strong>Server Hardware</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Server 1.9117 Model MMA</td>
<td>9117-MMA</td>
<td>1</td>
</tr>
<tr>
<td>Op Panel (MMA)</td>
<td>1845</td>
<td>1</td>
</tr>
<tr>
<td>P6 Processor Power Regulator</td>
<td>5625</td>
<td>1</td>
</tr>
<tr>
<td>System CEC Enclosure with Bezel</td>
<td>5626</td>
<td>1</td>
</tr>
<tr>
<td>AC Power Supply, 200-240V, 1500 Watt</td>
<td>5628</td>
<td>1</td>
</tr>
<tr>
<td>Media Enclosure and Backplane</td>
<td>5629</td>
<td>1</td>
</tr>
<tr>
<td>Service Processor Interface</td>
<td>5648</td>
<td>1</td>
</tr>
<tr>
<td>Processor Enclosure and Backplane</td>
<td>5663</td>
<td>1</td>
</tr>
<tr>
<td>I/O Backplane</td>
<td>5666</td>
<td>1</td>
</tr>
<tr>
<td>System Midplane</td>
<td>5667</td>
<td>1</td>
</tr>
<tr>
<td>SAS DASD Backplane, 6-pk</td>
<td>5668</td>
<td>1</td>
</tr>
<tr>
<td>Line Cord, DRWR TO IBM PDU, 14', 200-240V/10A,</td>
<td>6458</td>
<td>1</td>
</tr>
<tr>
<td>Rack Mount kit for IBM 19' rack</td>
<td>7164</td>
<td>1</td>
</tr>
<tr>
<td>Power Distribution Backplane</td>
<td>7870</td>
<td>1</td>
</tr>
<tr>
<td>P6 P6 SMP Fabric Cable, DRWR/DRWR</td>
<td>3660</td>
<td>1</td>
</tr>
<tr>
<td>P6 P6 SMP Fabric Cable, DRWR/NC/DRWR</td>
<td>3664</td>
<td>1</td>
</tr>
<tr>
<td>P6 SMP Fabric Cable DRWR/NC/NC/DRWR</td>
<td>3665</td>
<td>1</td>
</tr>
<tr>
<td>Enhanced FSP cable, 4 enclosures</td>
<td>5660</td>
<td>1</td>
</tr>
<tr>
<td>IDE Slimline DVD-ROM Drive</td>
<td>5756</td>
<td>1</td>
</tr>
<tr>
<td>4.7GHz POWER6 -2 Core Processor Card, 0-core active</td>
<td>7380</td>
<td>1</td>
</tr>
<tr>
<td>One Processor Activation for Processor Feature #7380</td>
<td>5403</td>
<td>1</td>
</tr>
<tr>
<td>256GB Memory (8x32GB) DDR2 POWER6 memory</td>
<td>8129</td>
<td>1</td>
</tr>
<tr>
<td>Activation of 256 GB DDR2 POWER6 Memory</td>
<td>5681</td>
<td>1</td>
</tr>
<tr>
<td>I/O Riser, 2x Serial, 2x p5IO2C E'net (Evans)</td>
<td>5636</td>
<td>1</td>
</tr>
<tr>
<td>73.4GB SAS DASD, 15K RPM</td>
<td>3646</td>
<td>1</td>
</tr>
<tr>
<td>4 Gigabit Fibre Channel PCI-X Adapter</td>
<td>5759</td>
<td>1</td>
</tr>
<tr>
<td>IBM 10/100/1000 Base-TX Ethernet PCI-X Adapter</td>
<td>5701</td>
<td>1</td>
</tr>
<tr>
<td>GX Dual Port- 12X Channel Attach</td>
<td>1802</td>
<td>1</td>
</tr>
<tr>
<td>IO Drawer 7314-G30</td>
<td>7314-G30</td>
<td>1</td>
</tr>
<tr>
<td>Planer and Tray Assembly</td>
<td>6590</td>
<td>1</td>
</tr>
<tr>
<td>I/O Drawer Mounting Enclosure</td>
<td>7314</td>
<td>1</td>
</tr>
<tr>
<td>AC Power Supply 300 Watt</td>
<td>6270</td>
<td>1</td>
</tr>
<tr>
<td>1.5M 12X ENHANCED IB CABLE</td>
<td>1830</td>
<td>1</td>
</tr>
<tr>
<td>Power Controll SPCN</td>
<td>6631</td>
<td>1</td>
</tr>
<tr>
<td>Line cord</td>
<td>6458</td>
<td>1</td>
</tr>
<tr>
<td>Dual Port 12X Channel Adapter</td>
<td>6446</td>
<td>1</td>
</tr>
<tr>
<td>POWER CONTROL CABLE, 3M, (SPCN)</td>
<td>6006</td>
<td>1</td>
</tr>
<tr>
<td>Rack Model T00</td>
<td>7014-T00</td>
<td>1</td>
</tr>
<tr>
<td>Front Trim Kit For 1.8 Meter Rack (Black)</td>
<td>6246</td>
<td>1</td>
</tr>
<tr>
<td>Side Panel (Black)</td>
<td>6098</td>
<td>1</td>
</tr>
<tr>
<td>PDU to 14', 200-240V/24A, UTG0247, PT#12</td>
<td>6654</td>
<td>1</td>
</tr>
<tr>
<td>HMC 1.7310-C05 Desktop Hardw.Mgmt.Console</td>
<td>7310-C05</td>
<td>1</td>
</tr>
<tr>
<td>IBM ThinkVision C170 17-inch Color Monitor</td>
<td>3631</td>
<td>1</td>
</tr>
<tr>
<td>Power Cord (6-foot), To Wall Plug Type #4</td>
<td>6470</td>
<td>1</td>
</tr>
<tr>
<td>Ethernet Cable, 6M, HMC to System Unit</td>
<td>7801</td>
<td>1</td>
</tr>
<tr>
<td>Keyboard - English, #103P</td>
<td>8800</td>
<td>1</td>
</tr>
<tr>
<td>Mouse - Attachment Cable</td>
<td>8841</td>
<td>1</td>
</tr>
</tbody>
</table>

**Storage**

- DS4800 Disk System Model 82  
  - 1815-82A: 1 53,995 21 1,133,895
- DS4800 8-Storage Partitions  
  - 8870: 1 10,000 21 210,000
- (22R4255) DS4800 AIX Host Kit  
  - 7711: 1 7,000 21 147,000
- DS4000 EXP810 Enclosure  
  - 1812-B1A: 1 6,000 248 1,488,000
- 72GB/15K Drive 4Gb FC disks  
  - 5413: 1 1,679 168 282,072
- 36GB/15K Drive 4Gb FC disks  
  - 5412: 1 892 3,312 2,954,304
- Short Wave SFP  
  - 2410: 1 998 185 184,630

**Subtotal** 2,142,159 127,310
## IBM System p 570 Model 9117-MMA

**Report Date:** May 21, 2007

---

### Server Software

<table>
<thead>
<tr>
<th>Item Description</th>
<th>Quantity</th>
<th>Price</th>
<th>Unit</th>
<th>Subtotal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fiber Cable 25m</td>
<td>1</td>
<td>189</td>
<td>1</td>
<td>189</td>
</tr>
<tr>
<td>Fiber Cable 1m</td>
<td>1</td>
<td>79</td>
<td>1</td>
<td>79</td>
</tr>
<tr>
<td>3 Year Warranty Service Upgrade 1812-81A 24x7x4</td>
<td>1</td>
<td>960</td>
<td>1</td>
<td>960</td>
</tr>
<tr>
<td>3 Year Warranty Service Upgrade 1815-82A 24x7x4</td>
<td>1</td>
<td>3,200</td>
<td>1</td>
<td>3,200</td>
</tr>
</tbody>
</table>

**Subtotal:** $6,447,023

### 3 Year Warranty Service Upgrade

<table>
<thead>
<tr>
<th>Item Description</th>
<th>Quantity</th>
<th>Price</th>
<th>Unit</th>
<th>Subtotal</th>
</tr>
</thead>
<tbody>
<tr>
<td>AIX 5.3 (media only)</td>
<td>1</td>
<td>50</td>
<td>1</td>
<td>50</td>
</tr>
<tr>
<td>AIX Software per Processor</td>
<td>1</td>
<td>1,225</td>
<td>16</td>
<td>19,600</td>
</tr>
<tr>
<td>Software Maintenance for AIX, 3 Year</td>
<td>1</td>
<td>1,958</td>
<td>16</td>
<td>31,328</td>
</tr>
<tr>
<td>F5 3 Yr SWMA for AIX per Processor</td>
<td>1</td>
<td>468</td>
<td>16</td>
<td>7,936</td>
</tr>
<tr>
<td>Partition Load Manager SW Maint: 3 year</td>
<td>1</td>
<td>627</td>
<td>15</td>
<td>880</td>
</tr>
<tr>
<td>F5 3 Yr SWMA for AIX per Processor</td>
<td>1</td>
<td>55</td>
<td>1</td>
<td>55</td>
</tr>
<tr>
<td>F5 3 yr Services 7x24 Support per Processor</td>
<td>1</td>
<td>496</td>
<td>16</td>
<td>7,936</td>
</tr>
</tbody>
</table>

**Subtotal:** $305,280

### V1O Software Maintenance (3Y)

<table>
<thead>
<tr>
<th>Item Description</th>
<th>Quantity</th>
<th>Price</th>
<th>Unit</th>
<th>Subtotal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Per Processor F5 VIO 3 Yr Maintenance</td>
<td>1</td>
<td>245</td>
<td>16</td>
<td>3,920</td>
</tr>
<tr>
<td>Per Processor F5 VIO 3 Yr Maint 24x7 Support</td>
<td>1</td>
<td>64</td>
<td>16</td>
<td>1,024</td>
</tr>
</tbody>
</table>

**Subtotal:** $1,024

### Initial Software Support 3 Year

<table>
<thead>
<tr>
<th>Item Description</th>
<th>Quantity</th>
<th>Price</th>
<th>Unit</th>
<th>Subtotal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Per Processor Software Support 3 Year</td>
<td>1</td>
<td>675</td>
<td>1</td>
<td>675</td>
</tr>
<tr>
<td>Per Processor 24x7 Software Support 3 Year</td>
<td>1</td>
<td>236</td>
<td>1</td>
<td>236</td>
</tr>
</tbody>
</table>

**Subtotal:** $539,698

### Client Hardware and Software

<table>
<thead>
<tr>
<th>Item Description</th>
<th>Quantity</th>
<th>Price</th>
<th>Unit</th>
<th>Subtotal</th>
</tr>
</thead>
<tbody>
<tr>
<td>xSeries 346 Express Model</td>
<td>1</td>
<td>2,175</td>
<td>1</td>
<td>2,175</td>
</tr>
<tr>
<td>3.2GHz 800MHz 2MB L2 Cache Xeon Processor</td>
<td>1</td>
<td>499</td>
<td>64</td>
<td>31,936</td>
</tr>
<tr>
<td>1GB (2x512MB Kit) PC2-3200</td>
<td>1</td>
<td>238</td>
<td>64</td>
<td>15,232</td>
</tr>
<tr>
<td>36GB 15K Hot Swap SCSI</td>
<td>1</td>
<td>269</td>
<td>64</td>
<td>17,216</td>
</tr>
<tr>
<td>NetBAY S2 42U Standard Rack Cabinet</td>
<td>1</td>
<td>1,489</td>
<td>26</td>
<td>38,714</td>
</tr>
<tr>
<td>Optical 3-Button Mouse - USB</td>
<td>1</td>
<td>19</td>
<td>1</td>
<td>19</td>
</tr>
<tr>
<td>Preferred Pro Full Size PS/2 Keyboard</td>
<td>1</td>
<td>29</td>
<td>1</td>
<td>29</td>
</tr>
<tr>
<td>IBM C117 17&quot; CRT Monitor</td>
<td>1</td>
<td>149</td>
<td>1</td>
<td>149</td>
</tr>
</tbody>
</table>

**Subtotal:** $242,495

### Third Party Hardware/Software

<table>
<thead>
<tr>
<th>Item Description</th>
<th>Quantity</th>
<th>Price</th>
<th>Unit</th>
<th>Subtotal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Visual C++ Standard Edition</td>
<td>2</td>
<td>109</td>
<td>1</td>
<td>109</td>
</tr>
<tr>
<td>Microsoft Windows 2003 Server</td>
<td>2</td>
<td>719</td>
<td>64</td>
<td>46,016</td>
</tr>
<tr>
<td>Microsoft Problem Resolution Services</td>
<td>2</td>
<td>245</td>
<td>1</td>
<td>245</td>
</tr>
<tr>
<td>3Com Baseline Switch 2824 24-port unmanaged Gigabit</td>
<td>3</td>
<td>290</td>
<td>6</td>
<td>1,740</td>
</tr>
</tbody>
</table>

**Subtotal:** $48,110

**Total:** $9,419,485

**Total IBM Discounts:** $4,273,465

**Three-Year Cost of Ownership:** $5,146,020

---

### Notes:

For pricing details and contact information please see appendix E.

**Pricing Sources:**
1. IBM
2. Microsoft
3. CDW

*Discounts are based on US list prices for similar quantities & configurations including pre-payment for maintenance. The discount of 43% applies to the totality of all items with price source of "1".*

---

**Audited by:** Francois Raab, Info Sizing (www.infosizing.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 reflect standard pricing policies for the listed components. For complete details, see the pricing sections of the TPC benchmark specifications. If you find that the stated prices are not available according to these terms, please inform the TPC at pricing@tpc.org. Thank you.
MQTH, computed Maximum Qualified Throughput: 1,616,162 tpmC

**Response Times (in seconds)**

<table>
<thead>
<tr>
<th></th>
<th>90th%</th>
<th>Average</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>New Order</td>
<td>0.48</td>
<td>0.25</td>
<td>20.91</td>
</tr>
<tr>
<td>Payment</td>
<td>0.46</td>
<td>0.24</td>
<td>20.43</td>
</tr>
<tr>
<td>Order-Status</td>
<td>0.47</td>
<td>0.25</td>
<td>16.9</td>
</tr>
<tr>
<td>Delivery (interactive)</td>
<td>0.10</td>
<td>0.10</td>
<td>19.62</td>
</tr>
<tr>
<td>Delivery (deferred)</td>
<td>0.14</td>
<td>0.24</td>
<td>2.55</td>
</tr>
<tr>
<td>Stock-Level</td>
<td>0.55</td>
<td>0.30</td>
<td>13.44</td>
</tr>
<tr>
<td>Menu</td>
<td>0.10</td>
<td>0.10</td>
<td>20.40</td>
</tr>
</tbody>
</table>

Response time delay added for emulated components was 0.1 seconds

**Transaction Mix, in percent of total transactions**

<table>
<thead>
<tr>
<th></th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>New Order</td>
<td>44.96%</td>
</tr>
<tr>
<td>Payment</td>
<td>43.01%</td>
</tr>
<tr>
<td>Order-Status</td>
<td>4.01%</td>
</tr>
<tr>
<td>Delivery</td>
<td>4.01%</td>
</tr>
<tr>
<td>Stock-Level</td>
<td>4.01%</td>
</tr>
</tbody>
</table>

**Keying/Think Times (in seconds)**

<table>
<thead>
<tr>
<th></th>
<th>Min.</th>
<th>Average</th>
<th>Max.</th>
</tr>
</thead>
<tbody>
<tr>
<td>New Order</td>
<td>18.00/0.01</td>
<td>18.01/12.02</td>
<td>18.04/120.20</td>
</tr>
<tr>
<td>Payment</td>
<td>3.00/0.01</td>
<td>3.01/12.02</td>
<td>3.04/120.20</td>
</tr>
<tr>
<td>Order-Status</td>
<td>2.00/0.01</td>
<td>2.01/10.01</td>
<td>2.03/100.10</td>
</tr>
<tr>
<td>Delivery</td>
<td>2.00/0.01</td>
<td>2.01/5.02</td>
<td>2.03/50.20</td>
</tr>
<tr>
<td>Stock-Level</td>
<td>2.00/0.01</td>
<td>2.01/5.02</td>
<td>2.03/50.20</td>
</tr>
</tbody>
</table>

**Test Duration**

- Ramp-up Time: 19 minutes
- Measurement interval: 2 hours 20 minutes
- Transactions during measurement interval (all types): 503,267,530
- Ramp-down time: 15 minutes

**Checkpoints**

- Number of checkpoints: N/A
- Checkpoint interval: N/A
### Table of Content

<table>
<thead>
<tr>
<th>Clause</th>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>General Items</td>
<td>11</td>
</tr>
<tr>
<td>1</td>
<td>Clause 1: Logical Data Base Design Related Items</td>
<td>13</td>
</tr>
<tr>
<td>1.1</td>
<td>Table Definitions</td>
<td>13</td>
</tr>
<tr>
<td>1.2</td>
<td>Database Organization</td>
<td>13</td>
</tr>
<tr>
<td>1.3</td>
<td>Insert and/or Delete Operations</td>
<td>13</td>
</tr>
<tr>
<td>1.4</td>
<td>Horizontal or Vertical Partitioning</td>
<td>13</td>
</tr>
<tr>
<td>2</td>
<td>Clause 2: Transaction &amp; Terminal Profiles Related Items</td>
<td>14</td>
</tr>
<tr>
<td>2.1</td>
<td>Verification for the Random Number Generator</td>
<td>14</td>
</tr>
<tr>
<td>2.2</td>
<td>Input/Output Screens</td>
<td>14</td>
</tr>
<tr>
<td>2.3</td>
<td>Priced Terminal Features</td>
<td>14</td>
</tr>
<tr>
<td>2.4</td>
<td>Presentation Managers</td>
<td>14</td>
</tr>
<tr>
<td>2.5</td>
<td>Home and Remote Order-lines</td>
<td>14</td>
</tr>
<tr>
<td>2.6</td>
<td>New-Order Rollback Transactions</td>
<td>14</td>
</tr>
<tr>
<td>2.7</td>
<td>Number of Items per Order</td>
<td>14</td>
</tr>
<tr>
<td>2.8</td>
<td>Home and Remote Payment Transactions</td>
<td>15</td>
</tr>
<tr>
<td>2.9</td>
<td>Non-Primary Key Transactions</td>
<td>15</td>
</tr>
<tr>
<td>2.10</td>
<td>Skipped Delivery Transactions</td>
<td>15</td>
</tr>
<tr>
<td>2.11</td>
<td>Mix of Transaction Types</td>
<td>16</td>
</tr>
<tr>
<td>2.12</td>
<td>Queuing Mechanism of Delivery</td>
<td>16</td>
</tr>
<tr>
<td>3</td>
<td>Clause 3: Transaction and System Properties</td>
<td>17</td>
</tr>
<tr>
<td>3.1</td>
<td>Atomicity Requirements</td>
<td>17</td>
</tr>
<tr>
<td>3.2</td>
<td>Consistency Requirements</td>
<td>17</td>
</tr>
<tr>
<td>3.3</td>
<td>Isolation Requirements</td>
<td>18</td>
</tr>
<tr>
<td>3.4</td>
<td>Durability Requirements</td>
<td>18</td>
</tr>
<tr>
<td>4</td>
<td>Clause 4: Scaling and Data Base Population Related Items</td>
<td>20</td>
</tr>
<tr>
<td>4.1</td>
<td>Cardinality of Tables</td>
<td>20</td>
</tr>
<tr>
<td>4.2</td>
<td>Distribution of Tables and Logs</td>
<td>20</td>
</tr>
<tr>
<td>4.3</td>
<td>Data Base Model Implemented</td>
<td>20</td>
</tr>
<tr>
<td>4.4</td>
<td>Partitions/Replications Mapping</td>
<td>21</td>
</tr>
<tr>
<td>4.5</td>
<td>60-Day Space Calculations</td>
<td>27</td>
</tr>
<tr>
<td>5</td>
<td>Clause 5: Performance Metrics and Response Time Related Items</td>
<td>28</td>
</tr>
<tr>
<td>5.1</td>
<td>Response Times</td>
<td>28</td>
</tr>
<tr>
<td>5.2</td>
<td>Keying and Think Times</td>
<td>28</td>
</tr>
<tr>
<td>5.3</td>
<td>Response Time Frequency Distribution</td>
<td>29</td>
</tr>
<tr>
<td>5.4</td>
<td>Performance Curve for Response Time versus Throughput</td>
<td>31</td>
</tr>
<tr>
<td>5.5</td>
<td>Think Time Frequency Distribution</td>
<td>32</td>
</tr>
<tr>
<td>5.6</td>
<td>Throughput versus Elapsed Time</td>
<td>32</td>
</tr>
<tr>
<td>5.7</td>
<td>Steady State Determination</td>
<td>33</td>
</tr>
<tr>
<td>5.8</td>
<td>Work Performed During Steady State</td>
<td>33</td>
</tr>
<tr>
<td>5.9</td>
<td>Measurement Interval</td>
<td>34</td>
</tr>
<tr>
<td>6</td>
<td>Clause 6: SUT, Driver, and Communication Definition Related Items</td>
<td>35</td>
</tr>
<tr>
<td>6.1</td>
<td>RTE Availability</td>
<td>35</td>
</tr>
<tr>
<td>6.2</td>
<td>Functionality and Performance of Emulated Components</td>
<td>35</td>
</tr>
<tr>
<td>6.3</td>
<td>Network Bandwidth</td>
<td>35</td>
</tr>
<tr>
<td>6.4</td>
<td>Operator Intervention</td>
<td>35</td>
</tr>
<tr>
<td>7</td>
<td>Clause 7: Pricing Related Items</td>
<td>36</td>
</tr>
<tr>
<td>7.1</td>
<td>Hardware and Programs Used</td>
<td>36</td>
</tr>
<tr>
<td>7.2</td>
<td>Three Year Cost of System Configuration</td>
<td>36</td>
</tr>
<tr>
<td>7.3</td>
<td>Availability Dates</td>
<td>36</td>
</tr>
<tr>
<td>7.4</td>
<td>Statement of tpmC and Price/Performance</td>
<td>36</td>
</tr>
</tbody>
</table>
Abstract

This report documents the full disclosure information required by the TPC Benchmark™ C Standard Specification Revision 5.8 dated April, 2006, for measurements on the IBM System p 570 Model 9117-MMA. The software used on the IBM System p 570 Model 9117-MMA includes AIX 5L Version 5.3 operating system, DB2 9.1 database manager. Microsoft COM+ is used as transaction manager.

<table>
<thead>
<tr>
<th>Company Name</th>
<th>System Name</th>
<th>Data Base Software</th>
<th>Operating System Software</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM Corporation</td>
<td>IBM System p 570 Model 9117-MMA</td>
<td>DB2 9.1</td>
<td>AIX 5L Version 5.3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Total System Cost</th>
<th>TPC-C Throughput</th>
<th>Price/Performance</th>
</tr>
</thead>
<tbody>
<tr>
<td>$5,713,181 USD</td>
<td>1,616,162</td>
<td>$3.54 USD</td>
</tr>
</tbody>
</table>
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 5.8 in April 2006.

This is the full disclosure report for benchmark testing of the IBM System p 570 Model 9117-MMA and DB2 9.1 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 IBM 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 International Business Machines Corporation.

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)
IBM System p 570 Model 9117-MMA Benchmark Configuration

64Clients
IBM System x346
3.2 GHz Intel® Xeon™
2x 2MB L2 Cache
2GB Memory
1 36GB Internal Drive
2 Integrated 10/100/1000 Ethernet

IBM® System p 570
8x 4.7GHz POWER6™ Dual-core Processor Chips
8x 32MB L3 Cache
768GB Memory
4 Internal SAS DASD Controllers
2 73.4GB Internal SAS Drives
21 IBM 4Gb dual-port FC Adapters
4 Dual port 10/100/1000 Ethernet

Storage
21 IBM TotalStorage DS4800
248 EXP810 Storage Expansion
3312 36.4GB 15K RPM Drives
168 73.4GB 15K RPM Drives

IBM System p 570 Model 9117-MMA Priced Configuration

64Clients
IBM System x346
3.2 GHz Intel® Xeon™
2x 2MB L2 Cache
2GB Memory
1 36GB Internal Drive
2 Integrated 10/100/1000 Ethernet

IBM® System p 570
8x 4.7GHz POWER6™ Dual-core Processor Chips
8x 32MB L3 Cache
768GB Memory
4 Internal SAS DASD Controllers
2 73.4GB Internal SAS Drives
21 IBM 4Gb dual-port FC Adapters
4 Dual port 10/100/1000 Ethernet

Storage
21 IBM TotalStorage DS4800
248 EXP810 Storage Expansion
3312 36.4GB 15K RPM Drives
168 73.4GB 15K RPM 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 database.*

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

1.2. Database Organization

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

Physical space was allocated to DB2 9.1 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 database implementation that precludes inserts beyond the limits defined in Clause 1.4.11 must be disclosed. This includes the maximum number of rows that can be inserted and the maximum key value for these new rows.*

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 9.1 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.

Each table partition for STOCK, CUSTOMER, ORDERS and ORDERLINE contains data associated with a range of 1,066 warehouses.

Each table partition for WAREHOUSE, DISTRICT, NEWORDER and HISTORY contains data associated with a range of 3,198 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.

\[
\text{double neg_exp_4(double average) = } -\text{average} \times \left(\frac{1}{\text{RANDOM}_4\_Z} \times \log(1 - \text{RANDOM}_4\_K \times \text{drand48}())\right);
\]

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 shows 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.

<table>
<thead>
<tr>
<th>New Order</th>
<th>IBM System p 570 Model 9117-MMA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Percentage of Home order lines</td>
<td>99.01%</td>
</tr>
<tr>
<td>Percentage of Remote order lines</td>
<td>0.99%</td>
</tr>
<tr>
<td>Percentage of Rolled Back Transactions</td>
<td>1.00%</td>
</tr>
<tr>
<td>Average Number of Items per order</td>
<td>10.00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Payment</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Percentage of Home transactions</td>
<td>85.01%</td>
</tr>
<tr>
<td>Percentage of Remote transactions</td>
<td>14.99%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Non-Primary Key Access</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Percentage of Payment using C_LAST</td>
<td>59.99%</td>
</tr>
<tr>
<td>Percentage of Order-Status using C_LAST</td>
<td>60.00%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Delivery</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Delivery transactions skipped</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Transaction Mix</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>New-Order</td>
<td>44.96%</td>
</tr>
<tr>
<td>Payment</td>
<td>43.01%</td>
</tr>
<tr>
<td>Order-Status</td>
<td>4.01%</td>
</tr>
<tr>
<td>Delivery</td>
<td>4.01%</td>
</tr>
<tr>
<td>Stock-Level</td>
<td>4.01%</td>
</tr>
</tbody>
</table>

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 database transactions are atomic; the system will either perform all individual operations on the data, or will assure that no partially-completed operations leave any effects on the data.

3.1.1. 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 database transaction to take the database from one consistent state to another, assuming that the database 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 preceding District and Warehouse.
Additionally, that same relationship exists for the most recent Order ID \([\text{max}(o\_id)]\) for the New Order table associated with the same District and Warehouse. Those relationships can be illustrated as follows:

\[ d\_\text{next}\_o\_id - 1 = \text{max}(o\_id) = \text{max}(no\_o\_id) \]

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 \([\text{max}(no\_o\_id)]\) minus the first Order ID \([\text{min}(no\_o\_id)]\) 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:

\[ \text{max}(no\_o\_id) - \text{min}(no\_o\_id) + 1 = \text{number of rows in New Order for the Warehouse/District} \]

4. For each District within a Warehouse, the sum of Order Line counts \([\text{sum}(o\_ol\_cnt)]\) 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:

\[ \text{sum}(o\_ol\_cnt) = \text{number of rows in the Order Line table for the Warehouse/District} \]

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 database 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 database 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 database 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 127,920 warehouses, the number of active warehouses during the benchmark.

<table>
<thead>
<tr>
<th>Table Name</th>
<th>Number of Rows</th>
</tr>
</thead>
<tbody>
<tr>
<td>Warehouse</td>
<td>127,920</td>
</tr>
<tr>
<td>District</td>
<td>1,279,200</td>
</tr>
<tr>
<td>Customer</td>
<td>3,837,600,000</td>
</tr>
<tr>
<td>History</td>
<td>3,837,600,000</td>
</tr>
<tr>
<td>Order</td>
<td>3,837,600,000</td>
</tr>
<tr>
<td>New Order</td>
<td>1,151,280,000</td>
</tr>
<tr>
<td>Order Line</td>
<td>38,375,754,139</td>
</tr>
<tr>
<td>Stock</td>
<td>12,792,000,000</td>
</tr>
<tr>
<td>Item</td>
<td>100,000</td>
</tr>
</tbody>
</table>

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.

One dual-port FC adapter connected to a DS4800 storage controller was used for the log. The storage controller contained eight RAID5 arrays each with 15 disk drives. The log logical volume was striped across the eight arrays (hdisks). Each of the disks used for the log had 36GB of storage capacity and the RAID5 LUN was 968.64GB in size.

There are 20 dual-port FC adapters connected to 20 storage controllers. Each of the storage controllers contained 168 disks for a total of 3360 disks. All storage controllers were used evenly.

All database data was evenly distributed on 120 storage arrays. Each array is created using 28 disks. Each array contains 12 LUNs and each LUN corresponds to one DB2 container.

RAID0 was used to create the disk arrays. Six of the arrays used 73.4GB disks, while the remainder were created with 36.4GB 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 9.1. DB2 9.1 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:

<table>
<thead>
<tr>
<th>ARRAY GROUP NAME</th>
<th>DATABASE PARTITION</th>
<th>LUNS</th>
</tr>
</thead>
<tbody>
<tr>
<td>F01V1</td>
<td>1</td>
<td>D1F01V1ITEM, D1F01V1WARE, D1F01V1DIST, D1F01V1CSTI, D1F01V1NORA, D1F01V1ORDR, D1F01V1VISTK, D1F01V1ORDI, D1F01V1HIST, D1F01V1NORB</td>
</tr>
<tr>
<td>F01V2</td>
<td>2</td>
<td>D1F01V2ITEM, D1F01V2WARE, D1F01V2DIST, D1F01V2CSTI, D1F01V2NORA, D1F01V2ORDR, D1F01V2VISTK, D1F01V2ORDI, D1F01V2HIST, D1F01V2NORB</td>
</tr>
<tr>
<td>F01V3</td>
<td>3</td>
<td>D1F01V3ITEM, D1F01V3WARE, D1F01V3DIST, D1F01V3CSTI, D1F01V3NORA, D1F01V3ORDR, D1F01V3VISTK, D1F01V3ORDI, D1F01V3HIST, D1F01V3NORB</td>
</tr>
<tr>
<td>F01V4</td>
<td>4</td>
<td>D1F01V4ITEM, D1F01V4WARE, D1F01V4DIST, D1F01V4CSTI, D1F01V4NORA, D1F01V4ORDR, D1F01V4VISTK, D1F01V4ORDI, D1F01V4HIST, D1F01V4NORB</td>
</tr>
<tr>
<td>F01V5</td>
<td>5</td>
<td>D1F01V5ITEM, D1F01V5WARE, D1F01V5DIST, D1F01V5CSTI, D1F01V5NORA, D1F01V5ORDR, D1F01V5VISTK, D1F01V5ORDI, D1F01V5HIST, D1F01V5NORB</td>
</tr>
<tr>
<td>F01V6</td>
<td>6</td>
<td>D1F01V6ITEM, D1F01V6WARE, D1F01V6DIST, D1F01V6CSTI, D1F01V6NORA, D1F01V6ORDR, D1F01V6VISTK, D1F01V6ORDI, D1F01V6HIST, D1F01V6NORB</td>
</tr>
<tr>
<td>F02V1</td>
<td>7</td>
<td>D1F02V1ITEM, D1F02V1WARE, D1F02V1DIST, D1F02V1CSTI, D1F02V1NORA, D1F02V1ORDR, D1F02V1VISTK, D1F02V1ORDI, D1F02V1HIST, D1F02V1NORB</td>
</tr>
<tr>
<td>F02V2</td>
<td>8</td>
<td>D1F02V2ITEM, D1F02V2WARE, D1F02V2DIST, D1F02V2CSTI, D1F02V2NORA, D1F02V2ORDR, D1F02V2VISTK, D1F02V2ORDI, D1F02V2HIST, D1F02V2NORB</td>
</tr>
<tr>
<td>F02V3</td>
<td>9</td>
<td>D1F02V3ITEM, D1F02V3WARE, D1F02V3DIST, D1F02V3CSTI, D1F02V3NORA, D1F02V3ORDR, D1F02V3VISTK, D1F02V3ORDI, D1F02V3HIST, D1F02V3NORB</td>
</tr>
<tr>
<td>F02V4</td>
<td>10</td>
<td>D1F02V4ITEM, D1F02V4WARE, D1F02V4DIST, D1F02V4CSTI, D1F02V4NORA, D1F02V4ORDR, D1F02V4VISTK, D1F02V4ORDI, D1F02V4HIST, D1F02V4NORB</td>
</tr>
<tr>
<td>F02V5</td>
<td>11</td>
<td>D1F02V5ITEM, D1F02V5WARE, D1F02V5DIST, D1F02V5CSTI, D1F02V5NORA, D1F02V5ORDR, D1F02V5VISTK, D1F02V5ORDI, D1F02V5HIST, D1F02V5NORB</td>
</tr>
<tr>
<td>F02V6</td>
<td>12</td>
<td>D1F02V6ITEM, D1F02V6WARE, D1F02V6DIST, D1F02V6CSTI, D1F02V6NORA, D1F02V6ORDR, D1F02V6VISTK, D1F02V6ORDI, D1F02V6HIST, D1F02V6NORB</td>
</tr>
<tr>
<td>F03V1</td>
<td>13</td>
<td>D1F03V1ITEM, D1F03V1WARE, D1F03V1DIST, D1F03V1CSTI, D1F03V1NORA, D1F03V1ORDR, D1F03V1VISTK, D1F03V1ORDI, D1F03V1HIST, D1F03V1NORB</td>
</tr>
<tr>
<td>F03V2</td>
<td>14</td>
<td>D1F03V2ITEM, D1F03V2WARE, D1F03V2DIST, D1F03V2CSTI, D1F03V2NORA, D1F03V2ORDR, D1F03V2VISTK, D1F03V2ORDI, D1F03V2HIST, D1F03V2NORB</td>
</tr>
<tr>
<td>F03V3</td>
<td>15</td>
<td>D1F03V3ITEM, D1F03V3WARE, D1F03V3DIST, D1F03V3CSTI, D1F03V3NORA, D1F03V3ORDR, D1F03V3VISTK, D1F03V3ORDI, D1F03V3HIST, D1F03V3NORB</td>
</tr>
<tr>
<td>F03V4</td>
<td>16</td>
<td>D1F03V4ITEM, D1F03V4WARE, D1F03V4DIST, D1F03V4CSTI, D1F03V4NORA, D1F03V4ORDR, D1F03V4VISTK, D1F03V4ORDI, D1F03V4HIST, D1F03V4NORB</td>
</tr>
<tr>
<td>F03V5</td>
<td>17</td>
<td>D1F03V5ITEM, D1F03V5WARE, D1F03V5DIST, D1F03V5CSTI, D1F03V5NORA, D1F03V5ORDR, D1F03V5VISTK, D1F03V5ORDI, D1F03V5HIST, D1F03V5NORB</td>
</tr>
<tr>
<td>F03V6</td>
<td>18</td>
<td>D1F03V6ITEM, D1F03V6WARE, D1F03V6DIST, D1F03V6CSTI, D1F03V6NORA, D1F03V6ORDR, D1F03V6VISTK, D1F03V6ORDI, D1F03V6HIST, D1F03V6NORB</td>
</tr>
<tr>
<td>F04V1</td>
<td>19</td>
<td>D1F04V1ITEM, D1F04V1WARE, D1F04V1DIST, D1F04V1CSTI, D1F04V1NORA, D1F04V1ORL, D1F04V1STK, D1F04V1CST, D1F04V1ORD, D1F04V1HIST, D1F04V1NORB</td>
</tr>
<tr>
<td>-------</td>
<td>----</td>
<td>---------------------------------------------------------------</td>
</tr>
<tr>
<td>F04V2</td>
<td>20</td>
<td>D1F04V2ITEM, D1F04V2WARE, D1F04V2DIST, D1F04V2CSTI, D1F04V2NORA, D1F04V2ORL, D1F04V2STK, D1F04V2CST, D1F04V2ORD, D1F04V2HIST, D1F04V2NORB</td>
</tr>
<tr>
<td>F04V3</td>
<td>21</td>
<td>D1F04V3ITEM, D1F04V3WARE, D1F04V3DIST, D1F04V3CSTI, D1F04V3NORA, D1F04V3ORL, D1F04V3STK, D1F04V3CST, D1F04V3ORD, D1F04V3HIST, D1F04V3NORB</td>
</tr>
<tr>
<td>F04V4</td>
<td>22</td>
<td>D1F04V4ITEM, D1F04V4WARE, D1F04V4DIST, D1F04V4CSTI, D1F04V4NORA, D1F04V4ORL, D1F04V4STK, D1F04V4CST, D1F04V4ORD, D1F04V4HIST, D1F04V4NORB</td>
</tr>
<tr>
<td>F04V5</td>
<td>23</td>
<td>D1F04V5ITEM, D1F04V5WARE, D1F04V5DIST, D1F04V5CSTI, D1F04V5NORA, D1F04V5ORL, D1F04V5STK, D1F04V5CST, D1F04V5ORD, D1F04V5HIST, D1F04V5NORB</td>
</tr>
<tr>
<td>F04V6</td>
<td>24</td>
<td>D1F04V6ITEM, D1F04V6WARE, D1F04V6DIST, D1F04V6CSTI, D1F04V6NORA, D1F04V6ORL, D1F04V6STK, D1F04V6CST, D1F04V6ORD, D1F04V6HIST, D1F04V6NORB</td>
</tr>
<tr>
<td>F05V1</td>
<td>25</td>
<td>D1F05V1ITEM, D1F05V1WARE, D1F05V1DIST, D1F05V1CSTI, D1F05V1NORA, D1F05V1ORL, D1F05V1STK, D1F05V1CST, D1F05V1ORD, D1F05V1HIST, D1F05V1NORB</td>
</tr>
<tr>
<td>F05V2</td>
<td>26</td>
<td>D1F05V2ITEM, D1F05V2WARE, D1F05V2DIST, D1F05V2CSTI, D1F05V2NORA, D1F05V2ORL, D1F05V2STK, D1F05V2CST, D1F05V2ORD, D1F05V2HIST, D1F05V2NORB</td>
</tr>
<tr>
<td>F05V3</td>
<td>27</td>
<td>D1F05V3ITEM, D1F05V3WARE, D1F05V3DIST, D1F05V3CSTI, D1F05V3NORA, D1F05V3ORL, D1F05V3STK, D1F05V3CST, D1F05V3ORD, D1F05V3HIST, D1F05V3NORB</td>
</tr>
<tr>
<td>F05V4</td>
<td>28</td>
<td>D1F05V4ITEM, D1F05V4WARE, D1F05V4DIST, D1F05V4CSTI, D1F05V4NORA, D1F05V4ORL, D1F05V4STK, D1F05V4CST, D1F05V4ORD, D1F05V4HIST, D1F05V4NORB</td>
</tr>
<tr>
<td>F05V5</td>
<td>29</td>
<td>D1F05V5ITEM, D1F05V5WARE, D1F05V5DIST, D1F05V5CSTI, D1F05V5NORA, D1F05V5ORL, D1F05V5STK, D1F05V5CST, D1F05V5ORD, D1F05V5HIST, D1F05V5NORB</td>
</tr>
<tr>
<td>F05V6</td>
<td>30</td>
<td>D1F05V6ITEM, D1F05V6WARE, D1F05V6DIST, D1F05V6CSTI, D1F05V6NORA, D1F05V6ORL, D1F05V6STK, D1F05V6CST, D1F05V6ORD, D1F05V6HIST, D1F05V6NORB</td>
</tr>
<tr>
<td>F06V1</td>
<td>31</td>
<td>D1F06V1ITEM, D1F06V1WARE, D1F06V1DIST, D1F06V1CSTI, D1F06V1NORA, D1F06V1ORL, D1F06V1STK, D1F06V1CST, D1F06V1ORD, D1F06V1HIST, D1F06V1NORB</td>
</tr>
<tr>
<td>F06V2</td>
<td>32</td>
<td>D1F06V2ITEM, D1F06V2WARE, D1F06V2DIST, D1F06V2CSTI, D1F06V2NORA, D1F06V2ORL, D1F06V2STK, D1F06V2CST, D1F06V2ORD, D1F06V2HIST, D1F06V2NORB</td>
</tr>
<tr>
<td>F06V3</td>
<td>33</td>
<td>D1F06V3ITEM, D1F06V3WARE, D1F06V3DIST, D1F06V3CSTI, D1F06V3NORA, D1F06V3ORL, D1F06V3STK, D1F06V3CST, D1F06V3ORD, D1F06V3HIST, D1F06V3NORB</td>
</tr>
<tr>
<td>F06V4</td>
<td>34</td>
<td>D1F06V4ITEM, D1F06V4WARE, D1F06V4DIST, D1F06V4CSTI, D1F06V4NORA, D1F06V4ORL, D1F06V4STK, D1F06V4CST, D1F06V4ORD, D1F06V4HIST, D1F06V4NORB</td>
</tr>
<tr>
<td>F06V5</td>
<td>35</td>
<td>D1F06V5ITEM, D1F06V5WARE, D1F06V5DIST, D1F06V5CSTI, D1F06V5NORA, D1F06V5ORL, D1F06V5STK, D1F06V5CST, D1F06V5ORD, D1F06V5HIST, D1F06V5NORB</td>
</tr>
<tr>
<td>F06V6</td>
<td>36</td>
<td>D1F06V6ITEM, D1F06V6WARE, D1F06V6DIST, D1F06V6CSTI, D1F06V6NORA, D1F06V6ORL, D1F06V6STK, D1F06V6CST, D1F06V6ORD, D1F06V6HIST, D1F06V6NORB</td>
</tr>
<tr>
<td>F07V1</td>
<td>37</td>
<td>D1F07V1ITEM, D1F07V1WARE, D1F07V1DIST, D1F07V1CSTI, D1F07V1NORA, D1F07V1ORL, D1F07V1STK, D1F07V1CST, D1F07V1ORD, D1F07V1HIST, D1F07V1NORB</td>
</tr>
<tr>
<td>F07V2</td>
<td>38</td>
<td>D1F07V2ITEM, D1F07V2WARE, D1F07V2DIST, D1F07V2CSTI, D1F07V2NORA, D1F07V2ORL, D1F07V2STK, D1F07V2CST, D1F07V2ORD, D1F07V2HIST, D1F07V2NORB</td>
</tr>
<tr>
<td>F07V3</td>
<td>39</td>
<td>D1F07V3ITEM, D1F07V3WARE, D1F07V3DIST, D1F07V3CSTI, D1F07V3NORA, D1F07V3ORL, D1F07V3STK, D1F07V3CST, D1F07V3ORD, D1F07V3HIST, D1F07V3NORB</td>
</tr>
<tr>
<td>F07V4</td>
<td>40</td>
<td>D1F07V4ITEM, D1F07V4WARE, D1F07V4DIST, D1F07V4CSTI, D1F07V4NORA, D1F07V4ORL, D1F07V4STK, D1F07V4CST, D1F07V4ORD, D1F07V4HIST, D1F07V4NORB</td>
</tr>
<tr>
<td>F07V5</td>
<td>41</td>
<td>D1F07V5ITEM, D1F07V5WARE, D1F07V5DIST, D1F07V5CSTI, D1F07V5NORA, D1F07V5ORL, D1F07V5STK, D1F07V5CST, D1F07V5ORD, D1F07V5HIST, D1F07V5NORB</td>
</tr>
<tr>
<td>F07V6</td>
<td>42</td>
<td>D1F07V6ITEM, D1F07V6WARE, D1F07V6DIST, D1F07V6CSTI, D1F07V6NORA, D1F07V6ORL, D1F07V6STK, D1F07V6ORDI, D1F07V6ORD, D1F07V6HIST, D1F07V6NORB</td>
</tr>
<tr>
<td>F08V1</td>
<td>43</td>
<td>D1F08V1ITEM, D1F08V1WARE, D1F08V1DIST, D1F08V1CSTI, D1F08V1NORA, D1F08V1ORL, D1F08V1STK, D1F08V1CST, D1F08V1ORDI, D1F08V1ORD, D1F08V1HIST, D1F08V1NORB</td>
</tr>
<tr>
<td>F08V2</td>
<td>44</td>
<td>D1F08V2ITEM, D1F08V2WARE, D1F08V2DIST, D1F08V2CSTI, D1F08V2NORA, D1F08V2ORL, D1F08V2STK, D1F08V2CST, D1F08V2ORDI, D1F08V2ORD, D1F08V2HIST, D1F08V2NORB</td>
</tr>
<tr>
<td>F08V3</td>
<td>45</td>
<td>D1F08V3ITEM, D1F08V3WARE, D1F08V3DIST, D1F08V3CSTI, D1F08V3NORA, D1F08V3ORL, D1F08V3STK, D1F08V3CST, D1F08V3ORDI, D1F08V3ORD, D1F08V3HIST, D1F08V3NORB</td>
</tr>
<tr>
<td>F08V4</td>
<td>46</td>
<td>D1F08V4ITEM, D1F08V4WARE, D1F08V4DIST, D1F08V4CSTI, D1F08V4NORA, D1F08V4ORL, D1F08V4STK, D1F08V4CST, D1F08V4ORDI, D1F08V4ORD, D1F08V4HIST, D1F08V4NORB</td>
</tr>
<tr>
<td>F08V5</td>
<td>47</td>
<td>D1F08V5ITEM, D1F08V5WARE, D1F08V5DIST, D1F08V5CSTI, D1F08V5NORA, D1F08V5ORL, D1F08V5STK, D1F08V5CST, D1F08V5ORDI, D1F08V5ORD, D1F08V5HIST, D1F08V5NORB</td>
</tr>
<tr>
<td>F08V6</td>
<td>48</td>
<td>D1F08V6ITEM, D1F08V6WARE, D1F08V6DIST, D1F08V6CSTI, D1F08V6NORA, D1F08V6ORL, D1F08V6STK, D1F08V6CST, D1F08V6ORDI, D1F08V6ORD, D1F08V6HIST, D1F08V6NORB</td>
</tr>
<tr>
<td>F09V1</td>
<td>49</td>
<td>D1F09V1ITEM, D1F09V1WARE, D1F09V1DIST, D1F09V1CSTI, D1F09V1NORA, D1F09V1ORL, D1F09V1STK, D1F09V1CST, D1F09V1ORDI, D1F09V1ORD, D1F09V1HIST, D1F09V1NORB</td>
</tr>
<tr>
<td>F09V2</td>
<td>50</td>
<td>D1F09V2ITEM, D1F09V2WARE, D1F09V2DIST, D1F09V2CSTI, D1F09V2NORA, D1F09V2ORL, D1F09V2STK, D1F09V2CST, D1F09V2ORDI, D1F09V2ORD, D1F09V2HIST, D1F09V2NORB</td>
</tr>
<tr>
<td>F09V3</td>
<td>51</td>
<td>D1F09V3ITEM, D1F09V3WARE, D1F09V3DIST, D1F09V3CSTI, D1F09V3NORA, D1F09V3ORL, D1F09V3STK, D1F09V3CST, D1F09V3ORDI, D1F09V3ORD, D1F09V3HIST, D1F09V3NORB</td>
</tr>
<tr>
<td>F09V4</td>
<td>52</td>
<td>D1F09V4ITEM, D1F09V4WARE, D1F09V4DIST, D1F09V4CSTI, D1F09V4NORA, D1F09V4ORL, D1F09V4STK, D1F09V4CST, D1F09V4ORDI, D1F09V4ORD, D1F09V4HIST, D1F09V4NORB</td>
</tr>
<tr>
<td>F09V5</td>
<td>53</td>
<td>D1F09V5ITEM, D1F09V5WARE, D1F09V5DIST, D1F09V5CSTI, D1F09V5NORA, D1F09V5ORL, D1F09V5STK, D1F09V5CST, D1F09V5ORDI, D1F09V5ORD, D1F09V5HIST, D1F09V5NORB</td>
</tr>
<tr>
<td>F09V6</td>
<td>54</td>
<td>D1F09V6ITEM, D1F09V6WARE, D1F09V6DIST, D1F09V6CSTI, D1F09V6NORA, D1F09V6ORL, D1F09V6STK, D1F09V6CST, D1F09V6ORDI, D1F09V6ORD, D1F09V6HIST, D1F09V6NORB</td>
</tr>
<tr>
<td>F10V1</td>
<td>55</td>
<td>D1F10V1ITEM, D1F10V1WARE, D1F10V1DIST, D1F10V1CSTI, D1F10V1NORA, D1F10V1ORL, D1F10V1STK, D1F10V1CST, D1F10V1ORDI, D1F10V1ORD, D1F10V1HIST, D1F10V1NORB</td>
</tr>
<tr>
<td>F10V2</td>
<td>56</td>
<td>D1F10V2ITEM, D1F10V2WARE, D1F10V2DIST, D1F10V2CSTI, D1F10V2NORA, D1F10V2ORL, D1F10V2STK, D1F10V2CST, D1F10V2ORDI, D1F10V2ORD, D1F10V2HIST, D1F10V2NORB</td>
</tr>
<tr>
<td>F10V3</td>
<td>57</td>
<td>D1F10V3ITEM, D1F10V3WARE, D1F10V3DIST, D1F10V3CSTI, D1F10V3NORA, D1F10V3ORL, D1F10V3STK, D1F10V3CST, D1F10V3ORDI, D1F10V3ORD, D1F10V3HIST, D1F10V3NORB</td>
</tr>
<tr>
<td>F10V4</td>
<td>58</td>
<td>D1F10V4ITEM, D1F10V4WARE, D1F10V4DIST, D1F10V4CSTI, D1F10V4NORA, D1F10V4ORL, D1F10V4STK, D1F10V4CST, D1F10V4ORDI, D1F10V4ORD, D1F10V4HIST, D1F10V4NORB</td>
</tr>
<tr>
<td>F10V5</td>
<td>59</td>
<td>D1F10V5ITEM, D1F10V5WARE, D1F10V5DIST, D1F10V5CSTI, D1F10V5NORA, D1F10V5ORL, D1F10V5STK, D1F10V5CST, D1F10V5ORDI, D1F10V5ORD, D1F10V5HIST, D1F10V5NORB</td>
</tr>
<tr>
<td>F10V6</td>
<td>60</td>
<td>D1F10V6ITEM, D1F10V6WARE, D1F10V6DIST, D1F10V6CSTI, D1F10V6NORA, D1F10V6ORL, D1F10V6STK, D1F10V6CST, D1F10V6ORDI, D1F10V6ORD, D1F10V6HIST, D1F10V6NORB</td>
</tr>
<tr>
<td>F11V1</td>
<td>61</td>
<td>D1F11V1ITEM, D1F11V1WARE, D1F11V1DIST, D1F11V1CSTI, D1F11V1NORA, D1F11V1ORL, D1F11V1STK, D1F11V1CST, D1F11V1ORDI, D1F11V1ORD, D1F11V1HIST, D1F11V1NORB</td>
</tr>
<tr>
<td>F11V2</td>
<td>62</td>
<td>D1F11V2ITEM, D1F11V2WARE, D1F11V2DIST, D1F11V2CSTI, D1F11V2NORA, D1F11V2ORL, D1F11V2STK, D1F11V2CST, D1F11V2ORDI, D1F11V2ORD, D1F11V2HIST, D1F11V2NORB</td>
</tr>
<tr>
<td>Code</td>
<td>Value</td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>-------</td>
<td></td>
</tr>
<tr>
<td>F11V5</td>
<td>65</td>
<td></td>
</tr>
<tr>
<td></td>
<td>D1F11V5ITEM, D1F11V5WARE, D1F11V5DIST, D1F11V5CSTI, D1F11V5NORA, D1F11V5ORL, D1F11V5STK, D1F11V5ORD, D1F11V5ORDI, D1F11V5HIST, D1F11V5NORB</td>
<td></td>
</tr>
<tr>
<td>F11V6</td>
<td>66</td>
<td></td>
</tr>
<tr>
<td>F12V1</td>
<td>67</td>
<td></td>
</tr>
<tr>
<td></td>
<td>D1F12V1ITEM, D1F12V1WARE, D1F12V1DIST, D1F12V1CSTI, D1F12V1NORA, D1F12V1ORL, D1F12V1STK, D1F12V1ORD, D1F12V1ORDI, D1F12V1HIST, D1F12V1NORB</td>
<td></td>
</tr>
<tr>
<td>F12V2</td>
<td>68</td>
<td></td>
</tr>
<tr>
<td></td>
<td>D1F12V2ITEM, D1F12V2WARE, D1F12V2DIST, D1F12V2CSTI, D1F12V2NORA, D1F12V2ORL, D1F12V2STK, D1F12V2ORD, D1F12V2ORDI, D1F12V2HIST, D1F12V2NORB</td>
<td></td>
</tr>
<tr>
<td>F12V3</td>
<td>69</td>
<td></td>
</tr>
<tr>
<td></td>
<td>D1F12V3ITEM, D1F12V3WARE, D1F12V3DIST, D1F12V3CSTI, D1F12V3NORA, D1F12V3ORL, D1F12V3STK, D1F12V3ORD, D1F12V3ORDI, D1F12V3HIST, D1F12V3NORB</td>
<td></td>
</tr>
<tr>
<td>F12V4</td>
<td>70</td>
<td></td>
</tr>
<tr>
<td></td>
<td>D1F12V4ITEM, D1F12V4WARE, D1F12V4DIST, D1F12V4CSTI, D1F12V4NORA, D1F12V4ORL, D1F12V4STK, D1F12V4ORD, D1F12V4ORDI, D1F12V4HIST, D1F12V4NORB</td>
<td></td>
</tr>
<tr>
<td>F12V5</td>
<td>71</td>
<td></td>
</tr>
<tr>
<td></td>
<td>D1F12V5ITEM, D1F12V5WARE, D1F12V5DIST, D1F12V5CSTI, D1F12V5NORA, D1F12V5ORL, D1F12V5STK, D1F12V5ORD, D1F12V5ORDI, D1F12V5HIST, D1F12V5NORB</td>
<td></td>
</tr>
<tr>
<td>F12V6</td>
<td>72</td>
<td></td>
</tr>
<tr>
<td></td>
<td>D1F12V6ITEM, D1F12V6WARE, D1F12V6DIST, D1F12V6CSTI, D1F12V6NORA, D1F12V6ORL, D1F12V6STK, D1F12V6ORD, D1F12V6ORDI, D1F12V6HIST, D1F12V6NORB</td>
<td></td>
</tr>
<tr>
<td>F13V1</td>
<td>73</td>
<td></td>
</tr>
<tr>
<td></td>
<td>D1F13V1ITEM, D1F13V1WARE, D1F13V1DIST, D1F13V1CSTI, D1F13V1NORA, D1F13V1ORL, D1F13V1STK, D1F13V1ORD, D1F13V1ORDI, D1F13V1HIST, D1F13V1NORB</td>
<td></td>
</tr>
<tr>
<td>F13V2</td>
<td>74</td>
<td></td>
</tr>
<tr>
<td></td>
<td>D1F13V2ITEM, D1F13V2WARE, D1F13V2DIST, D1F13V2CSTI, D1F13V2NORA, D1F13V2ORL, D1F13V2STK, D1F13V2ORD, D1F13V2ORDI, D1F13V2HIST, D1F13V2NORB</td>
<td></td>
</tr>
<tr>
<td>F13V3</td>
<td>75</td>
<td></td>
</tr>
<tr>
<td></td>
<td>D1F13V3ITEM, D1F13V3WARE, D1F13V3DIST, D1F13V3CSTI, D1F13V3NORA, D1F13V3ORL, D1F13V3STK, D1F13V3ORD, D1F13V3ORDI, D1F13V3HIST, D1F13V3NORB</td>
<td></td>
</tr>
<tr>
<td>F13V4</td>
<td>76</td>
<td></td>
</tr>
<tr>
<td></td>
<td>D1F13V4ITEM, D1F13V4WARE, D1F13V4DIST, D1F13V4CSTI, D1F13V4NORA, D1F13V4ORL, D1F13V4STK, D1F13V4ORD, D1F13V4ORDI, D1F13V4HIST, D1F13V4NORB</td>
<td></td>
</tr>
<tr>
<td>F13V5</td>
<td>77</td>
<td></td>
</tr>
<tr>
<td></td>
<td>D1F13V5ITEM, D1F13V5WARE, D1F13V5DIST, D1F13V5CSTI, D1F13V5NORA, D1F13V5ORL, D1F13V5STK, D1F13V5ORD, D1F13V5ORDI, D1F13V5HIST, D1F13V5NORB</td>
<td></td>
</tr>
<tr>
<td>F13V6</td>
<td>78</td>
<td></td>
</tr>
<tr>
<td></td>
<td>D1F13V6ITEM, D1F13V6WARE, D1F13V6DIST, D1F13V6CSTI, D1F13V6NORA, D1F13V6ORL, D1F13V6STK, D1F13V6ORD, D1F13V6ORDI, D1F13V6HIST, D1F13V6NORB</td>
<td></td>
</tr>
<tr>
<td>F14V1</td>
<td>79</td>
<td></td>
</tr>
<tr>
<td></td>
<td>D1F14V1ITEM, D1F14V1WARE, D1F14V1DIST, D1F14V1CSTI, D1F14V1NORA, D1F14V1ORL, D1F14V1STK, D1F14V1ORD, D1F14V1ORDI, D1F14V1HIST, D1F14V1NORB</td>
<td></td>
</tr>
<tr>
<td>F14V2</td>
<td>80</td>
<td></td>
</tr>
<tr>
<td></td>
<td>D1F14V2ITEM, D1F14V2WARE, D1F14V2DIST, D1F14V2CSTI, D1F14V2NORA, D1F14V2ORL, D1F14V2STK, D1F14V2ORD, D1F14V2ORDI, D1F14V2HIST, D1F14V2NORB</td>
<td></td>
</tr>
<tr>
<td>F14V3</td>
<td>81</td>
<td></td>
</tr>
<tr>
<td></td>
<td>D1F14V3ITEM, D1F14V3WARE, D1F14V3DIST, D1F14V3CSTI, D1F14V3NORA, D1F14V3ORL, D1F14V3STK, D1F14V3ORD, D1F14V3ORDI, D1F14V3HIST, D1F14V3NORB</td>
<td></td>
</tr>
<tr>
<td>F14V4</td>
<td>82</td>
<td></td>
</tr>
<tr>
<td></td>
<td>D1F14V4ITEM, D1F14V4WARE, D1F14V4DIST, D1F14V4CSTI, D1F14V4NORA, D1F14V4ORL, D1F14V4STK, D1F14V4ORD, D1F14V4ORDI, D1F14V4HIST, D1F14V4NORB</td>
<td></td>
</tr>
<tr>
<td>F14V5</td>
<td>83</td>
<td></td>
</tr>
<tr>
<td></td>
<td>D1F14V5ITEM, D1F14V5WARE, D1F14V5DIST, D1F14V5CSTI, D1F14V5NORA, D1F14V5ORL, D1F14V5STK, D1F14V5ORD, D1F14V5ORDI, D1F14V5HIST, D1F14V5NORB</td>
<td></td>
</tr>
<tr>
<td>F14V6</td>
<td>84</td>
<td></td>
</tr>
<tr>
<td></td>
<td>D1F14V6ITEM, D1F14V6WARE, D1F14V6DIST, D1F14V6CSTI, D1F14V6NORA, D1F14V6ORL, D1F14V6STK, D1F14V6ORD, D1F14V6ORDI, D1F14V6HIST, D1F14V6NORB</td>
<td></td>
</tr>
<tr>
<td>F15V1</td>
<td>85</td>
<td></td>
</tr>
<tr>
<td></td>
<td>D1F15V1ITEM, D1F15V1WARE, D1F15V1DIST, D1F15V1CSTI, D1F15V1NORA, D1F15V1ORL, D1F15V1STK, D1F15V1ORD, D1F15V1ORDI, D1F15V1HIST, D1F15V1NORB</td>
<td></td>
</tr>
<tr>
<td>F15V2</td>
<td>86</td>
<td></td>
</tr>
<tr>
<td></td>
<td>D1F15V2ITEM, D1F15V2WARE, D1F15V2DIST, D1F15V2CSTI, D1F15V2NORA, D1F15V2ORL, D1F15V2STK, D1F15V2ORD, D1F15V2ORDI, D1F15V2HIST, D1F15V2NORB</td>
<td></td>
</tr>
<tr>
<td>F15V3</td>
<td>87</td>
<td></td>
</tr>
<tr>
<td></td>
<td>D1F15V3ITEM, D1F15V3WARE, D1F15V3DIST, D1F15V3CSTI, D1F15V3NORA, D1F15V3ORL, D1F15V3STK, D1F15V3ORD, D1F15V3ORDI, D1F15V3HIST, D1F15V3NORB</td>
<td></td>
</tr>
<tr>
<td>F15V4</td>
<td>88</td>
<td>D1F15V4ITEM, D1F15V4WARE, D1F15V4DIST, D1F15V4CSTI, D1F15V4NORA, D1F15V4ORL, D1F15V4STK, D1F15V4ORDI, D1F15V4ORD, D1F15V4HIST, D1F15V4NORB</td>
</tr>
<tr>
<td>-------</td>
<td>----</td>
<td>-----------------------------------------------------------------</td>
</tr>
<tr>
<td>F15V5</td>
<td>89</td>
<td>D1F15V5ITEM, D1F15V5WARE, D1F15V5DIST, D1F15V5CSTI, D1F15V5NORA, D1F15V5ORL, D1F15V5STK, D1F15V5ORDI, D1F15V5ORD, D1F15V5HIST, D1F15V5NORB</td>
</tr>
<tr>
<td>F15V6</td>
<td>90</td>
<td>D1F15V6ITEM, D1F15V6WARE, D1F15V6DIST, D1F15V6CSTI, D1F15V6NORA, D1F15V6ORL, D1F15V6STK, D1F15V6ORDI, D1F15V6ORD, D1F15V6HIST, D1F15V6NORB</td>
</tr>
<tr>
<td>F16V1</td>
<td>91</td>
<td>D1F16V1ITEM, D1F16V1WARE, D1F16V1DIST, D1F16V1CSTI, D1F16V1NORA, D1F16V1ORL, D1F16V1STK, D1F16V1ORDI, D1F16V1ORD, D1F16V1HIST, D1F16V1NORB</td>
</tr>
<tr>
<td>F16V2</td>
<td>92</td>
<td>D1F16V2ITEM, D1F16V2WARE, D1F16V2DIST, D1F16V2CSTI, D1F16V2NORA, D1F16V2ORL, D1F16V2STK, D1F16V2ORDI, D1F16V2ORD, D1F16V2HIST, D1F16V2NORB</td>
</tr>
<tr>
<td>F16V3</td>
<td>93</td>
<td>D1F16V3ITEM, D1F16V3WARE, D1F16V3DIST, D1F16V3CSTI, D1F16V3NORA, D1F16V3ORL, D1F16V3STK, D1F16V3ORDI, D1F16V3ORD, D1F16V3HIST, D1F16V3NORB</td>
</tr>
<tr>
<td>F16V4</td>
<td>94</td>
<td>D1F16V4ITEM, D1F16V4WARE, D1F16V4DIST, D1F16V4CSTI, D1F16V4NORA, D1F16V4ORL, D1F16V4STK, D1F16V4ORDI, D1F16V4ORD, D1F16V4HIST, D1F16V4NORB</td>
</tr>
<tr>
<td>F16V5</td>
<td>95</td>
<td>D1F16V5ITEM, D1F16V5WARE, D1F16V5DIST, D1F16V5CSTI, D1F16V5NORA, D1F16V5ORL, D1F16V5STK, D1F16V5ORDI, D1F16V5ORD, D1F16V5HIST, D1F16V5NORB</td>
</tr>
<tr>
<td>F16V6</td>
<td>96</td>
<td>D1F16V6ITEM, D1F16V6WARE, D1F16V6DIST, D1F16V6CSTI, D1F16V6NORA, D1F16V6ORL, D1F16V6STK, D1F16V6ORDI, D1F16V6ORD, D1F16V6HIST, D1F16V6NORB</td>
</tr>
<tr>
<td>F17V1</td>
<td>97</td>
<td>D1F17V1ITEM, D1F17V1WARE, D1F17V1DIST, D1F17V1CSTI, D1F17V1NORA, D1F17V1ORL, D1F17V1STK, D1F17V1ORDI, D1F17V1ORD, D1F17V1HIST, D1F17V1NORB</td>
</tr>
<tr>
<td>F17V2</td>
<td>98</td>
<td>D1F17V2ITEM, D1F17V2WARE, D1F17V2DIST, D1F17V2CSTI, D1F17V2NORA, D1F17V2ORL, D1F17V2STK, D1F17V2ORDI, D1F17V2ORD, D1F17V2HIST, D1F17V2NORB</td>
</tr>
<tr>
<td>F17V3</td>
<td>99</td>
<td>D1F17V3ITEM, D1F17V3WARE, D1F17V3DIST, D1F17V3CSTI, D1F17V3NORA, D1F17V3ORL, D1F17V3STK, D1F17V3ORDI, D1F17V3ORD, D1F17V3HIST, D1F17V3NORB</td>
</tr>
<tr>
<td>F17V4</td>
<td>100</td>
<td>D1F17V4ITEM, D1F17V4WARE, D1F17V4DIST, D1F17V4CSTI, D1F17V4NORA, D1F17V4ORL, D1F17V4STK, D1F17V4ORDI, D1F17V4ORD, D1F17V4HIST, D1F17V4NORB</td>
</tr>
<tr>
<td>F17V5</td>
<td>101</td>
<td>D1F17V5ITEM, D1F17V5WARE, D1F17V5DIST, D1F17V5CSTI, D1F17V5NORA, D1F17V5ORL, D1F17V5STK, D1F17V5ORDI, D1F17V5ORD, D1F17V5HIST, D1F17V5NORB</td>
</tr>
<tr>
<td>F17V6</td>
<td>102</td>
<td>D1F17V6ITEM, D1F17V6WARE, D1F17V6DIST, D1F17V6CSTI, D1F17V6NORA, D1F17V6ORL, D1F17V6STK, D1F17V6ORDI, D1F17V6ORD, D1F17V6HIST, D1F17V6NORB</td>
</tr>
<tr>
<td>F18V1</td>
<td>103</td>
<td>D1F18V1ITEM, D1F18V1WARE, D1F18V1DIST, D1F18V1CSTI, D1F18V1NORA, D1F18V1ORL, D1F18V1STK, D1F18V1ORDI, D1F18V1ORD, D1F18V1HIST, D1F18V1NORB</td>
</tr>
<tr>
<td>F18V2</td>
<td>104</td>
<td>D1F18V2ITEM, D1F18V2WARE, D1F18V2DIST, D1F18V2CSTI, D1F18V2NORA, D1F18V2ORL, D1F18V2STK, D1F18V2ORDI, D1F18V2ORD, D1F18V2HIST, D1F18V2NORB</td>
</tr>
<tr>
<td>F18V3</td>
<td>105</td>
<td>D1F18V3ITEM, D1F18V3WARE, D1F18V3DIST, D1F18V3CSTI, D1F18V3NORA, D1F18V3ORL, D1F18V3STK, D1F18V3ORDI, D1F18V3ORD, D1F18V3HIST, D1F18V3NORB</td>
</tr>
<tr>
<td>F18V4</td>
<td>106</td>
<td>D1F18V4ITEM, D1F18V4WARE, D1F18V4DIST, D1F18V4CSTI, D1F18V4NORA, D1F18V4ORL, D1F18V4STK, D1F18V4ORDI, D1F18V4ORD, D1F18V4HIST, D1F18V4NORB</td>
</tr>
<tr>
<td>F18V5</td>
<td>107</td>
<td>D1F18V5ITEM, D1F18V5WARE, D1F18V5DIST, D1F18V5CSTI, D1F18V5NORA, D1F18V5ORL, D1F18V5STK, D1F18V5ORDI, D1F18V5ORD, D1F18V5HIST, D1F18V5NORB</td>
</tr>
<tr>
<td>F18V6</td>
<td>108</td>
<td>D1F18V6ITEM, D1F18V6WARE, D1F18V6DIST, D1F18V6CSTI, D1F18V6NORA, D1F18V6ORL, D1F18V6STK, D1F18V6ORDI, D1F18V6ORD, D1F18V6HIST, D1F18V6NORB</td>
</tr>
<tr>
<td>F19V1</td>
<td>109</td>
<td>D1F19V1ITEM, D1F19V1WARE, D1F19V1DIST, D1F19V1CSTI, D1F19V1NORA, D1F19V1ORL, D1F19V1STK, D1F19V1ORDI, D1F19V1ORD, D1F19V1HIST, D1F19V1NORB</td>
</tr>
<tr>
<td>F19V2</td>
<td>110</td>
<td>D1F19V2ITEM, D1F19V2WARE, D1F19V2DIST, D1F19V2CSTI, D1F19V2NORA, D1F19V2ORL, D1F19V2STK, D1F19V2ORDI, D1F19V2ORD, D1F19V2HIST, D1F19V2NORB</td>
</tr>
<tr>
<td>F19V3</td>
<td>111</td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>-----</td>
<td></td>
</tr>
<tr>
<td>D1F19V3ITEM, D1F19V3WARE, D1F19V3DIST, D1F19V3CSTI, D1F19V3NORA, D1F19V3ORL, D1F19V3STK, D1F19V3CST, D1F19V3ORD, D1F19V3ORDI, D1F19V3HIST, D1F19V3NORB</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>F19V4</th>
<th>112</th>
</tr>
</thead>
<tbody>
<tr>
<td>D1F19V4ITEM, D1F19V4WARE, D1F19V4DIST, D1F19V4CSTI, D1F19V4NORA, D1F19V4ORL, D1F19V4STK, D1F19V4CST, D1F19V4ORD, D1F19V4ORDI, D1F19V4HIST, D1F19V4NORB</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>F19V5</th>
<th>113</th>
</tr>
</thead>
<tbody>
<tr>
<td>D1F19V5ITEM, D1F19V5WARE, D1F19V5DIST, D1F19V5CSTI, D1F19V5NORA, D1F19V5ORL, D1F19V5STK, D1F19V5CST, D1F19V5ORD, D1F19V5ORDI, D1F19V5HIST, D1F19V5NORB</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>F19V6</th>
<th>114</th>
</tr>
</thead>
<tbody>
<tr>
<td>D1F19V6ITEM, D1F19V6WARE, D1F19V6DIST, D1F19V6CSTI, D1F19V6NORA, D1F19V6ORL, D1F19V6STK, D1F19V6CST, D1F19V6ORD, D1F19V6ORDI, D1F19V6HIST, D1F19V6NORB</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>F20V1</th>
<th>115</th>
</tr>
</thead>
<tbody>
<tr>
<td>D1F20V1ITEM, D1F20V1WARE, D1F20V1DIST, D1F20V1CSTI, D1F20V1NORA, D1F20V1ORL, D1F20V1STK, D1F20V1CST, D1F20V1ORD, D1F20V1ORDI, D1F20V1HIST, D1F20V1NORB</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>F20V2</th>
<th>116</th>
</tr>
</thead>
<tbody>
<tr>
<td>D1F20V2ITEM, D1F20V2WARE, D1F20V2DIST, D1F20V2CSTI, D1F20V2NORA, D1F20V2ORL, D1F20V2STK, D1F20V2CST, D1F20V2ORD, D1F20V2ORDI, D1F20V2HIST, D1F20V2NORB</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>F20V3</th>
<th>117</th>
</tr>
</thead>
<tbody>
<tr>
<td>D1F20V3ITEM, D1F20V3WARE, D1F20V3DIST, D1F20V3CSTI, D1F20V3NORA, D1F20V3ORL, D1F20V3STK, D1F20V3CST, D1F20V3ORD, D1F20V3ORDI, D1F20V3HIST, D1F20V3NORB</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>F20V4</th>
<th>118</th>
</tr>
</thead>
<tbody>
<tr>
<td>D1F20V4ITEM, D1F20V4WARE, D1F20V4DIST, D1F20V4CSTI, D1F20V4NORA, D1F20V4ORL, D1F20V4STK, D1F20V4CST, D1F20V4ORD, D1F20V4ORDI, D1F20V4HIST, D1F20V4NORB</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>F20V5</th>
<th>119</th>
</tr>
</thead>
<tbody>
<tr>
<td>D1F20V5ITEM, D1F20V5WARE, D1F20V5DIST, D1F20V5CSTI, D1F20V5NORA, D1F20V5ORL, D1F20V5STK, D1F20V5CST, D1F20V5ORD, D1F20V5ORDI, D1F20V5HIST, D1F20V5NORB</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>F20V6</th>
<th>120</th>
</tr>
</thead>
<tbody>
<tr>
<td>D1F20V6ITEM, D1F20V6WARE, D1F20V6DIST, D1F20V6CSTI, D1F20V6NORA, D1F20V6ORL, D1F20V6STK, D1F20V6CST, D1F20V6ORD, D1F20V6ORDI, D1F20V6HIST, D1F20V6NORB</td>
<td></td>
</tr>
</tbody>
</table>

**Table 4-2:** IBM System p 570 Model 9117-MMA 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.

<table>
<thead>
<tr>
<th>Table</th>
<th>Rows</th>
<th>Table</th>
<th>Index</th>
<th>5% Space</th>
<th>Total Space</th>
</tr>
</thead>
<tbody>
<tr>
<td>Warehouse</td>
<td>127,920</td>
<td>35</td>
<td>0</td>
<td>2</td>
<td>37</td>
</tr>
<tr>
<td>District</td>
<td>1,279,200</td>
<td>190</td>
<td>0</td>
<td>10</td>
<td>200</td>
</tr>
<tr>
<td>Item</td>
<td>100,000</td>
<td>10</td>
<td>0</td>
<td>1</td>
<td>11</td>
</tr>
<tr>
<td>Stock</td>
<td>12,792,000,000</td>
<td>4,164,600</td>
<td>0</td>
<td>208,230</td>
<td>4,372,830</td>
</tr>
<tr>
<td>Customer</td>
<td>3,837,600,000</td>
<td>2,998,680</td>
<td>185,520</td>
<td>159,210</td>
<td>3,343,410</td>
</tr>
<tr>
<td>New-Order</td>
<td>1,151,280,000</td>
<td>88,800</td>
<td>0</td>
<td>4,440</td>
<td>93,240</td>
</tr>
<tr>
<td>Orders</td>
<td>3,837,600,000</td>
<td>147,917</td>
<td>108,240</td>
<td>0</td>
<td>256,157</td>
</tr>
<tr>
<td>Order-Line</td>
<td>38,376,000,000</td>
<td>2,574,292</td>
<td>0</td>
<td>0</td>
<td>2,574,292</td>
</tr>
<tr>
<td>History</td>
<td>3,837,600,000</td>
<td>236,960</td>
<td>0</td>
<td>0</td>
<td>236,960</td>
</tr>
<tr>
<td>Additional Overhead</td>
<td>3,837,600,000</td>
<td>2,215,311</td>
<td>0</td>
<td>0</td>
<td>2,215,311</td>
</tr>
</tbody>
</table>

| Free Space     | 514,953  | 30 Minute log Computations |
| Dynamic Space  | 2,959,169| Log Written (KB) 117,282,225 |
| Static Space   | 10,133,278| New-Order Txns 48,484,860 |
| Daily Spread   | 598,186  | Log Written per New-Order (KB) 2.42 |

**Data Storage Requirement**

- 60 Days (MB) 46,024,430
- 60 Days (GB) 44,946

**Log Storage Requirement**

- 8 Hours (GB) 1,789.58

**Disk Sizing**

<table>
<thead>
<tr>
<th>Disk Type</th>
<th>Formatted Capacity (GB)</th>
<th># of Disks</th>
<th>SUT Capacity (GB)</th>
<th># of Disks</th>
<th>Priced Capacity (GB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB FastT 36 GB</td>
<td>32.76</td>
<td>3,192</td>
<td>104,570</td>
<td>3,192</td>
<td>104,570</td>
</tr>
<tr>
<td>DB FastT 73 GB</td>
<td>67.86</td>
<td>168</td>
<td>11,400</td>
<td>168</td>
<td>11,400</td>
</tr>
<tr>
<td>LOG FastT RAID5</td>
<td>31.17</td>
<td>120</td>
<td>3,740</td>
<td>120</td>
<td>3,740</td>
</tr>
<tr>
<td>OS SAS 73GB</td>
<td>73.00</td>
<td>2</td>
<td>146</td>
<td>2</td>
<td>146</td>
</tr>
</tbody>
</table>

**Total Capacity (GB)**

119,857
5 \hspace{1em} \textbf{Clause 5: Performance Metrics and Response Time Related Items}

5.1. \hspace{1em} \textbf{Response Times}

\textit{Ninetyeth 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. \hspace{1em} \textbf{Keying and Think Times}

\textit{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.

<table>
<thead>
<tr>
<th>Response Times</th>
<th>New Order</th>
<th>Payment</th>
<th>Order Status</th>
<th>Delivery (int./def.)</th>
<th>Stock Level</th>
<th>Menus</th>
</tr>
</thead>
<tbody>
<tr>
<td>90 %</td>
<td>0.48</td>
<td>0.46</td>
<td>0.47</td>
<td>0.10/0.16</td>
<td>0.55</td>
<td>0.10</td>
</tr>
<tr>
<td>Average</td>
<td>0.25</td>
<td>0.24</td>
<td>0.25</td>
<td>0.10/0.12</td>
<td>0.30</td>
<td>0.10</td>
</tr>
<tr>
<td>Maximum</td>
<td>20.91</td>
<td>20.43</td>
<td>16.9</td>
<td>19.62/0.53</td>
<td>13.44</td>
<td>20.43</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Think Times</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum</td>
<td>0.01</td>
<td>0.01</td>
<td>0.01</td>
<td>0.01</td>
<td>0.01</td>
<td>N/A</td>
</tr>
<tr>
<td>Average</td>
<td>12.02</td>
<td>12.02</td>
<td>10.01</td>
<td>5.02</td>
<td>5.02</td>
<td>N/A</td>
</tr>
<tr>
<td>Maximum</td>
<td>120.20</td>
<td>120.20</td>
<td>100.10</td>
<td>50.20</td>
<td>50.20</td>
<td>N/A</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Keying Times</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum</td>
<td>18.00</td>
<td>3.00</td>
<td>2.00</td>
<td>2.00</td>
<td>2.00</td>
<td>N/A</td>
</tr>
<tr>
<td>Average</td>
<td>18.01</td>
<td>3.01</td>
<td>2.01</td>
<td>2.01</td>
<td>2.01</td>
<td>N/A</td>
</tr>
<tr>
<td>Maximum</td>
<td>18.04</td>
<td>3.04</td>
<td>2.03</td>
<td>2.03</td>
<td>2.03</td>
<td>N/A</td>
</tr>
</tbody>
</table>

\begin{table}[h]
\centering
\begin{tabular}{|l|c|c|c|c|c|
\hline
\textbf{Response Times} & \textbf{New Order} & \textbf{Payment} & \textbf{Order Status} & \textbf{Delivery (int./def.)} & \textbf{Stock Level} & \textbf{Menus} \\
\hline
90 % & 0.48 & 0.46 & 0.47 & 0.10/0.16 & 0.55 & 0.10 \\
Average & 0.25 & 0.24 & 0.25 & 0.10/0.12 & 0.30 & 0.10 \\
Maximum & 20.91 & 20.43 & 16.9 & 19.62/0.53 & 13.44 & 20.43 \\
\hline
\textbf{Think Times} & & & & & \\
\hline
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 \\
\hline
\textbf{Keying Times} & & & & & \\
\hline
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 & 18.04 & 3.04 & 2.03 & 2.03 & 2.03 & N/A \\
\hline
\end{tabular}
\caption{Think and Keying Times}
\end{table}
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
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](image)

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](image)
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 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.
    (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 9.1 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 9.1 proceeds to update the database as follows:

When DB2 9.1 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 9.1 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 9.1 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 9.1. 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.

IBM used an internally developed RTE 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 4 Gigabit Ethernet switches each with a rate of 1000Mbits full-duplex. Each group of 16 clients were connected to one of the Gigabit Ethernet switches 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 in Appendix - 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 IBM US list prices.

A 43% discount 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.

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: November 21, 2007.

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.

<table>
<thead>
<tr>
<th>System</th>
<th>tpmC</th>
<th>3-year System Cost</th>
<th>$/tpmC</th>
<th>Availability Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM System p 570 Model 9117-MMA</td>
<td>1,616,162</td>
<td>5,713,181 USD</td>
<td>$3.54  USD</td>
<td>November 21, 2007</td>
</tr>
</tbody>
</table>

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 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:
Benchmark Sponsor: John J. Makis
IBM System p Performance
11501 Burnet Road
Austin, TX 78758

May 17, 2007

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

Platform: IBM System p 570 Model 9117-MMA c/s
Operating system: AIX 5L Version 5.3
Database Manager: DB2 9.1
Transaction Manager: Microsoft COM+

The results were:

<table>
<thead>
<tr>
<th>CPU's (Speed)</th>
<th>Memory</th>
<th>Disks</th>
<th>NewOrder 90% Response Time</th>
<th>tpmC</th>
</tr>
</thead>
<tbody>
<tr>
<td>8 x Dual Core POWRR6 (4.7GHz)</td>
<td>768 GB (8 x 32 MB L3)</td>
<td>3312 x 36.4 GB FC 168 x 73.4 GB FC 2 x 73.4 GB SAS</td>
<td>0.48 Seconds</td>
<td>1,616,162.8</td>
</tr>
</tbody>
</table>

Server: IBM System p 570 Model 9117-MMA

Sixty-four Clients: IBM System x 346 (each with)

| 2 x Xeon (3.2 GHz) (2 MB L2) | 2.0 GB | 1 x 36 GB SCSI | n/a | n/a |

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

- The transactions were correctly implemented
- The database records were the proper size
- The database was properly scaled and populated
- The ACID properties were met
- Input data was generated according to the specified percentages
- The transaction cycle times included the required keying and think times
- The reported response times were correctly measured.
- At least 90% of all delivery transactions met the 80 Second completion time limit
- All 90% response times were under the specified maximums
- The measurement interval was representative of steady state conditions
- The reported measurement interval was 2 hours and 20 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
**Appendix - A: Client Server Code**

### A.1 Client/Terminal Handler Code

#### Makefile

```
# Licensed Materials - Property of IBM
# Governed under the terms of the International
# License Agreement for Non-Warranted Sample Code.
# 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)
# # Preprocessor, Compiler and Linker Flags
# ########################################################################
# OS File Extensions & Path Separator
OBJEXT=.obj LIBEXT=.lib SHLIBEXT=.dll BINEXT=.exe
SLASH=/ CMDSEP=\n
# Library Configuration
AR=ar.exe ARFLAGS= ARFLAGS_LIB= ARFLAGS_OUT= OUT:
# OS Commands
ERASE=del /F ERASEDIR=rm /d /S MOVE=move COPY=copy
# OS File Extensions & Path Separator
OBJEXT=.obj LIBEXT=lib SHLIBEXT=dl BINEXT=exe SLASH=/ CMD EXT=F

# User Targets
# all: $(OBJ) $(LIB) $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) $(LIBEXT)
# $(AR) $(ARFLAGS) $(ARFLAGS_LIB) $(OBJECTS) $(LIBRARY) $(LIBEXT)
# $(LIB) $(LIBRARY) (Continued)
```
struct in_item_struct * two = (struct in_item_struct *) b;
if (one->OL_I_ID == two->OL_I_ID) {
    return (one->OL_I_ID - two->OL_I_ID);
} else {
    return (one->OL_SUPPLY_W_ID - two->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 < in_neword->s_O_OL_CNT; itemIndex++) {
        switch (in_neword->in_item[itemIndex].s_OL_I_ID) {
            case UNUSED_ITEM_ID:
                sqlerror(NEWORD_SQL, "NEW", __FILE__, __LINE__, &sqlca);
                clientRc = FATAL_SQLERROR;
                return (clientRc);
                break;
        }
        SWAP_BYTE(in_neword->in_item[itemIndex].s_OL_SUPPLY_W_ID);
        SWAP_BYTE(in_neword->in_item[itemIndex].s_OL_SUPPLY_W_ID);
    }
    clientRc = FATAL_SQLERROR;
    return (clientRc);
}

payment_sql (struct in_payment_struct * in_payment, struct out_payment_struct * payment) {
    struct sqlca sqlca;
    int clientRc = TRAN_OK;
    for (itemIndex = 0; itemIndex < in_payment->s_O_OL_CNT; itemIndex++) {
        SWAP_BYTE(in_payment->in_item[itemIndex].s_OL_I_ID);
        SWAP_BYTE(in_payment->in_item[itemIndex].s_OL_SUPPLY_W_ID);
    }
    if (sqlca.sqlcode == 0) {
        float wtax = in_neword->s_W_TAX;
        float dtax = in_neword->s_D_TAX;
        float cdisc = in_neword->s_C_DISCOUNT;
        float factor = (1.0 - cdisc) * (1.0 + wtax + dtax);
        neword->s_total_amount *= factor;
    }
}

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);
}
### SQL END DECLARE SECTION;

```c
// 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_w_id in_payment->s_C_W_ID
#define d_w_id in_payment->s_D_W_ID
#define d_id_in_d_in_d_id_in_c_in_c_id in_c_id
#define in_c_id in_payment->s_C_ID
#define c_data_prefix_c_id_type struct c_data_prefix_c_id_type { short len ; char data[ 34 ] ; } c_data_prefix_c_id ;

EXEC SQL END DECLARE SECTION;
```

### EXEC SQL BEGIN COMPOUND NOT ATOMIC STATISTIC

```c
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
          , :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_discount , :c_balance, :c_data :c_data_indicator, :h_date
FROM TABLE ( PAY_C_LAST(   :w_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, C_DATE
                      )             ;
COMMIT ;            END COMPOUND ;
```

### EXEC SQL ROLLBACK WORK;

```c
EXEC SQL ROLLBACK WORK ;        if ( sqlca.sqlcode != 0 )
```

### EXEC SQL BEGIN COMPOUND NOT ATOMIC STATISTIC

```c
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
          , :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_ID(   :w_id
          , :d_id
          , :c_w_id
          , :c_d_id
          , :in_c_id
          )
          , c_data_prefix_c_id = sprintf( c_data_prefix_c_id.data, " %2.2d %6.6d %2.2d %6.6d %06.2f",  c_d_id , c_w_id , d_id , w_id , h_amount ) ;

EXEC SQL BEGIN COMPOUND NOT ATOMIC STATISTIC

```c
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
          , :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_discount , :c_balance, :c_data :c_data_indicator, :h_date
FROM TABLE ( PAY_C_LAST(   :w_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, C_DATE
                      )             ;
COMMIT ;            END COMPOUND ;
```

### EXEC SQL BEGIN DECLARE SECTION;

```c
struct c_data_prefix_c_id_type { short len ; char data[ 34 ] ; } c_data_prefix_c_id ;
```
EXEC SQL END DECLARE SECTION;

int clientRc = TRAN_OK;
int itemIndex = 0;

in_ord = (struct vc_del_in *) in_delivery;
in_ord-len = sizeof(struct vc_del_in_struct) - SPGENERAL_ADJUST;
out_ord = (struct vc_del_out *) in_delivery;
out_ord-len = sizeof(struct vc_del_out_struct) - SPGENERAL_ADJUST;

ifdef DEBUGIT
del_debug(delivery, in_delivery, "DEL failed");
#endif /* DEBUGIT */

ifdef SWAP_ENDIAN
SWAP_BYTE(in_ord->s_C_ID);
SWAP_BYTE(in_ord->s_D_ID);
SWAP_BYTE(in_ord->s_D_CARRIER_ID);
SWAP_BYTE(in_ord->s_D_SUPPLY_W_ID);
SWAP_BYTE(in_ord->s_D_QUANTITY);
#endif /* SWAP_ENDIAN */

EXEC SQL BEGIN DECLARE SECTION;
struct sqlca sqlca;

EXEC SQL END DECLARE SECTION;

if (sqlca.sqlcode == 0)
{
#ifdef DEBUGIT
stocklev->deadlocks = -1;
stocklev->s_transtatus = TRAN_OK;
stocklev->s_transtatus = FATAL_SQLERROR;
#endif /* DEBUGIT */

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

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 BEGIN DECLARE SECTION;
struct vc_ord_in
{
short len;
char data[14];
} * in_ord;

struct vc_ord_out
{
short len;
char data[50];
} * out_ord;

EXEC SQL END DECLARE SECTION;

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

EXEC SQL BEGIN DECLARE SECTION;
struct in_stocklev_struct * in_stocklev
{
struct vc_del_in
{
short len;
char data[14];
} * in_del;

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

EXEC SQL BEGIN DECLARE SECTION;
struct in_stocklev_struct * in_stocklev
{
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;

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

EXEC SQL BEGIN DECLARE SECTION;
struct vc_ord_in
{
short len;
char data[14];
} * in_ord;

struct vc_ord_out
{
short len;
char data[50];
} * out_ord;

EXEC SQL END DECLARE SECTION;

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
ordstat->s_C_ID = in_ordstat->s_C_ID ;
#endif /* DEBUGIT */

EXEC SQL BEGIN DECLARE SECTION;
struct in_stocklev_struct * in_stocklev
{
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;

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

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
;

EXEC SQL END DECLARE SECTION;
util_obj_db2 = tpccctx$(OBJEXT)

all: dbgen connect $(util_obj_db2) disconnect

dbgen: $(util_obj)

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)

# ######################################################################## # Dependencies
# ########################################################################
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 connect_to_TM(char *in_dbname);
- return connect_to_TM(authn, char *in_username, char *in_password)
- SQL_STRUCTURE sqlca; sqlca;
- int ConnectSQLCODE = 0;
- EXEC SQL BEGIN DECLARE SECTION;
- if (sqlca.sqlcode != 0)    {
- EXEC SQL ROLLBACK WORK ;
- return (clientRc) ;
- }
- EXEC SQL CONNECT TO :dbname IN SHARE MODE;
- if (ConnectSQLCODE != 0)    {
- EXEC SQL CONNECT TO :dbname IN SHARE MODE USER :username USING :password;
- return (clientRc) ;
- }
- EXEC SQL ROLLBACK WORK ;
- 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 
# 
# EXEC SQL BEGIN DECLARE SECTION;
# if (sqlca.sqlcode != 0)    {
# EXEC SQL ROLLBACK WORK ;
# return (clientRc) ;
# }
# EXEC SQL CONNECT TO :dbname IN SHARE MODE;
# if (ConnectSQLCODE != 0)    {
# EXEC SQL CONNECT TO :dbname IN SHARE MODE USER :username USING :password;
# return (clientRc) ;
# }
# EXEC SQL ROLLBACK WORK ;
# return (clientRc) ;
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. 
** 
# # Makefile - Makefile for Src.Common 
# 
# EXEC SQL BEGIN DECLARE SECTION;
# if (sqlca.sqlcode != 0)    {
# EXEC SQL ROLLBACK WORK ;
# return (clientRc) ;
# }
# EXEC SQL CONNECT TO :dbname IN SHARE MODE;
# if (ConnectSQLCODE != 0)    {
# EXEC SQL CONNECT TO :dbname IN SHARE MODE USER :username USING :password;
# return (clientRc) ;
# }
# EXEC SQL ROLLBACK WORK ;
# return (clientRc) ;
```c
if (SQLCODE != 0) { return SQLCODE; }
if (DisconnectSQLCODE) {
    return DisconnectSQLCODE;
} else {
    return 0;
}

int create_context(void) {
    SQL_STRUCTURE 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;
    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;
    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;
    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 0;
}

int get_context(void **ctx) {
    SQL_STRUCTURE sqlca;
    SQLCODE = sqleGetCurrentCtx(ctx, NULL, &sqlca);
    if (SQLCODE) { return SQLCODE; }
    return 0;
}
```

---

**Src.Common/tpcdbg.c**

```c
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];
    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:
            strcat(err_fn, "new.err.out");
            strcpy(tranName, "NEW_ORDER");
            break;
        case DELIVERY_SQL:
            strcat(err_fn, "del.err.out");
            strcpy(tranName, "DELIVERY");
            break;
        case PAYMENT_SQL:
            strcat(err_fn, "pay.err.out");
            strcpy(tranName, "PAYMENT");
            break;
        case ORDSTAT_SQL:
            strcat(err_fn, "ord.err.out");
            strcpy(tranName, "ORDER_STAT");
            break;
        case STOCKLEV_SQL:
            strcat(err_fn, "stk.err.out");
            strcpy(tranName, "STOCK_LVL");
            break;
        default:
            strcat(err_fn, "cli.err.out");
            strcpy(tranName, "CLIENT");
            break;
    }
    
    /* 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 \n", psqlca->sqlcode);
    ```
#define CLIENT_SQL   0
#define NEWORD_SQL  1
#define PAYMENT_SQL  2
#define ORDSTAT_SQL  3
#define STOCKLEV_SQL  5
#define SPGENERAL_PAD sizeof(int16_t)

struct in_neword_struct {   int16_t  len;
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;};

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];};
A.2 Client Transaction Code

Makefile.config

```bash
# Compiler Configuration
CLFLAGS_DEBUG may be set to "-g", "-DDEBUGIT" or "-g -DDEBUGIT" or left blank
CC=xlc
CLFLAGS_DSO=-gflags=-c -qarch=ansi -qplusxsmc -DSQLUNIX -DSQLAIX -q64 -D_LARGE_FILES
CLFLAGS_OUT=-o
CLFLAGS_DEBUG=

# Linker Configuration
LD_EXECUTE=-o
LD_STORP=-n
CLFLAGS_EXEC=-lm -q64
CLFLAGS_SHLIB=-qmkshrobj
CLFLAGS_STORP=$(CLFLAGS_SHLIB) -bE:$@.exp -lc -b64
CLFLAGS_LIB=-L$(TPCC_SQLLIB)/lib -ld2
CLFLAGS_OUT=-o

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

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

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

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

#define NACOMPCHK(last)
   if (sqlca.sqlcode == SQL_RC_E911) { 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; } 
   }

#define NACOMPCHK(last)
   if (sqlca.sqlcode == SQL_RC_E911) { 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; } 
   }

#define NACOMPCHK(last)
   if (sqlca.sqlcode == SQL_RC_E911) { 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; } 
   }

#define NACOMPCHK(last)
   if (sqlca.sqlcode == SQL_RC_E911) { 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; } 
   }
```
** US Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

```c
#include <string.h>

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

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

EXEC SQL BEGIN DECLARE SECTION;
char dbname[9];    char username[128];    char password[15];    EXEC SQL END DECLARE SECTION;
```

** (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.

```c
#include <time.h> #include "sqlca.h" #include "sql.h"

EXEC SQL CONNECT TO :dbname IN SHARE MODE;
```
char err_fn[DEBUG_PATH_SIZE + DEBUG_FILENAME_SZ];
int tranName[16];
char timeStamp[27];
int k;

int initPredictor();

void del_print(struct out_delivery_struct *delivery_ptr, struct in_delivery_struct *in_delivery, char *filename, char *msg);

void new_print(struct out_neword_struct *neword_ptr, struct in_neword_struct *in_neword, char *filename, char *msg);

char tranName[16];
char timeStamp[27];
int k;
fprintf(debug_fp, "\t\t\s_C_LAST = %s
",                ordstat_ptr->s_C_BALANCE);    fprintf(debug_fp, "\t\t\s_O_ID = %d (%X)
",            ordstat_ptr->s_O_ID, ordstat_ptr->s_O_ID);

fprintf(debug_fp, "\t\t\s_O_ENTRY_D = %s
",                ordstat_ptr->s_O_ENTRY_D_time);    fprintf(debug_fp, "\t\t\s_O_CARRIER_ID = %d (%X)
",            ordstat_ptr->s_O_CARRIER_ID, ordstat_ptr->s_O_CARRIER_ID);

fprintf(debug_fp, "\t\t\ol_cnt = %d (%X)
",                ordstat_ptr->deadlocks, ordstat_ptr->deadlocks);     fprintf(debug_fp, "\t\titems {
items = ordstat_ptr->s_ol_cnt;
for (j = 0; j < items; j++) {
    if (j != 0)          fprintf(debug_fp, "\n void ord_debug (struct out_ordstat_struct *ordstat_ptr,                 struct in_ordstat_struct  *in_ordstat,                 char *msg)
{"fprintf(debug_fp, "\nitems = in_neword->s_O_OL_CNT;
for (j = 0; j < items; j++) {
    fprintf(debug_fp, "\t\\ts_OL_AMOUNT[%d] = %.2f
", j, neword_ptr->item[j].s_OL_AMOUNT);
    fprintf(debug_fp, "\t\\ts_S_QUANTITY[%d] = %d (%X)
", j, neword_ptr->item[j].s_S_QUANTITY, neword_ptr->item[j].s_S_QUANTITY);
    fprintf(debug_fp, "\t\\ts_brand_generic = %d (%X)
", j, neword_ptr->item[j].s_brand_generic);
}    fprintf(debug_fp, "\t\}
}
\n\n/**/ 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);
}
```c
fprintf(debug_fp, "Stock level debug information follows %s (%s)\n", timeStamp, msg);    fprintf(debug_fp, " PID %d \n", getpid());    fprintf(debug_fp, "=================================================\n");    fprintf(debug_fp, " in_stocklev_struct \n");    fprintf(debug_fp, " s_D_ID = %d (%X)\n", in_stocklev->s_D_ID, in_stocklev->s_D_ID);    fprintf(debug_fp, " s_threshold = %d (%X)\n", in_stocklev->s_threshold, in_stocklev->s_threshold);    fprintf(debug_fp, " s_low_stock = %d (%X)\n", in_stocklev->s_low_stock, in_stocklev->s_low_stock);    fclose(debug_fp);
```
**rebind: connect catalog**

```
  db2 bind tpcc_all_sql.bnd $(BND_OPTS) QUERYOPT 7
```

**Install Targets**

```
  $(COPY) ords $(TPCC_SPDIR)
  $(COPY) news $(TPCC_SPDIR)
  $(COPY) dels $(TPCC_SPDIR)
```

**Build Rules**

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

**tpcc_all_sql.c:**

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

**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: $(TPCC_ROOT)/include/db2tpcc.h
```

**Src.Srv/cat-func.ddl**

```
```
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 CUSTOMER
WHERE C_W_ID = ORD.C_LAST.W_ID
AND C_D_ID = ORD.C_LAST.D_ID
AND C_ID = ORD.C_LAST.C_ID
) AS V1
WHERE NUM = (COUNT + BIGNUT(1)) / BIGNUT(2) ;
); SET (O_ID, O_CARRIER_ID, O_ENTRY_D)
= (SELECT O_ID
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

-- -- 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_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
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 C_ID INTEGER ;
DECLARE O_ID INTEGER;
DECLARE O_CARRIER_ID SMALLINT;   DECLARE O_ENTRY_D    TIMESTAMP;
DECLARE O_CARRIER_ID SMALLINT;
DECLARE O_ENTRY_D Enough
/* 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_W_ID = PAY.C_LAST.W_ID
AND C_D_ID = PAY.C_LAST.D_ID
AND C_ID = PAY.C_LAST.C_ID
) AS V1
WHERE NUM = (COUNT + BIGNUT(1)) / BIGNUT(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, C_DATA                 , 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
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
   , 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
   AND D_STREET_1 = PAY_C_LAST.D_STREET_1
   AND D_STREET_2 = PAY_C_LAST.D_STREET_2
   AND D_CITY = PAY_C_LAST.D_CITY
   AND D_STATE = PAY_C_LAST.D_STATE
   AND D_ZIP = PAY_C_LAST.D_ZIP
WHERE D_W_ID = PAY_C_LAST.W_ID                                AND D_ID   = PAY_C_LAST.D_ID                           ) AS U
WITH D_NAME AS ( 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.D_AMOUNT
   ) AS U
)

/* Determine the C_ID */
SET ( C_ID ) = ( SELECT C_ID
FROM ( SELECT C_ID
   , COUNT(*) OVER() AS COUNT
   , ROWNUMBER() OVER (ORDER BY C_FIRST) AS NUM
   FROM CUSTOMER
WHERE C_LAST = PAY_C_LAST.C_LAST
   AND C_W_ID = PAY_C_LAST.C_W_ID
   AND C_D_ID = PAY_C_LAST.C_D_ID
   ) AS T
WHERE NUM = (COUNT + BIGINT( 1 ) ) / BIGINT( 2 )
)

/* Done - return the collected data */
RETURN VALUES (   W_STREET_1, W_STREET_2, W_CITY, W_STATE, W_ZIP
   , D_NAME, D_STREET_1, D_STREET_2, D_CITY, D_STATE, D_ZIP
   , H_DATE )
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 C.H Date TIMESTAMP ;

/* Finally insert into the warehouse */
VALUES ( PAY.C_ID, PAY.C_D_ID, PAY.C_W_ID, PAY.D_ID, PAY.W_ID, VAR.W_Name || CHAR( 3, 4 ) || VAR.D_Name, VAR.H_Date, PAY.C.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, C.H_Date ) ;

DECLARE C_State CHAR(2) ;
DECLARE C_Zip CHAR(9) ;
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 INT ;

SET OL_DIST_INFO = CASE WHEN S.Quantity = NEW OL.ALL.LQty THEN S.Quantity ELSE S.Quantity + 91 END ;

SELECT OL_DIST_INFO || S_Data AS S_DATA ;

FROM NEW TABLE ( UPDATE STOCK
INCLUDE ( OL_DIST_INFO CHAR(24) )
SET S.Quantity = CASE WHEN S.Quantity = NEW OL.ALL.LQty...
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.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.W_ID
AND C_W_ID = W_ID
AND C_D_ID = D_ID ) ;

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

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"
EXEC SQL DECLARATE 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
, OL_NUMBER
, I_ID                                                  , I_SUPPLY_W_ID                                                  , I_QTY
FROM  Table( VALUES
(TIMESTAMP('0001-01-01 00:00:00')) AS OL_DELIVERY_D
, I_PRICE, I_NAME, I_DATA, OL_DIST_INFO, S_DATA, S_QUANTITY
, OL_W_ID
, OL_NUMBER
, OL_I_ID, OL_SUPPLY_W_ID

FROM Table( VALUES ( SMALLINT( 1 )         , :id0  , :ol_quantity0  , :supply_w_id0  )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT( 2 )         , :id1  , :ol_quantity1  , :supply_w_id1  )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT( 3 )         , :id2  , :ol_quantity2  , :supply_w_id2  )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT( 4 )         , :id3  , :ol_quantity3  , :supply_w_id3  )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT( 5 )         , :id4  , :ol_quantity4  , :supply_w_id4  )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT( 6 )         , :id5  , :ol_quantity5  , :supply_w_id5  )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT( 7 )         , :id6  , :ol_quantity6  , :supply_w_id6  )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT( 8 )         , :id7  , :ol_quantity7  , :supply_w_id7  )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT( 9 )         , :id8  , :ol_quantity8  , :supply_w_id8  )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT(10)         , :id9  , :ol_quantity9  , :supply_w_id9  )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT(11)         , :id10, :ol_quantity10, :supply_w_id10 )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT(12)         , :id11, :ol_quantity11, :supply_w_id11 )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT(13)         , :id12, :ol_quantity12, :supply_w_id12 )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT(14)         , :id13, :ol_quantity13, :supply_w_id13 )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT(15)         , :id14, :ol_quantity14, :supply_w_id14 )
	) AS ITEMLIST
)
AS X ( OL_NUMBER , I_ID  , I_QTY          , I_SUPPLY_W_ID  )
)
WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
)
SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA , S_QUANTITY
, OL_W_ID
, OL_NUMBER
, OL_I_ID
, OL_SUPPLY_W_ID
, OL_DELIVERY_D
, OL_QTY
, OL_SUPPLY_W_ID
, OL_QUANTITY
FROM Table( VALUES ( SMALLINT( 1 )         , :id0  , :ol_quantity0  , :supply_w_id0  )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT( 2 )         , :id1  , :ol_quantity1  , :supply_w_id1  )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT( 3 )         , :id2  , :ol_quantity2  , :supply_w_id2  )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT( 4 )         , :id3  , :ol_quantity3  , :supply_w_id3  )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT( 5 )         , :id4  , :ol_quantity4  , :supply_w_id4  )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT( 6 )         , :id5  , :ol_quantity5  , :supply_w_id5  )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT( 7 )         , :id6  , :ol_quantity6  , :supply_w_id6  )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT( 8 )         , :id7  , :ol_quantity7  , :supply_w_id7  )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT( 9 )         , :id8  , :ol_quantity8  , :supply_w_id8  )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT(10)         , :id9  , :ol_quantity9  , :supply_w_id9  )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT(11)         , :id10, :ol_quantity10, :supply_w_id10 )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT(12)         , :id11, :ol_quantity11, :supply_w_id11 )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT(13)         , :id12, :ol_quantity12, :supply_w_id12 )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT(14)         , :id13, :ol_quantity13, :supply_w_id13 )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT(15)         , :id14, :ol_quantity14, :supply_w_id14 )
	) AS ITEMLIST
)
AS X ( OL_NUMBER , I_ID  , I_QTY          , I_SUPPLY_W_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 Table( VALUES ( SMALLINT( 1 )         , :id0  , :ol_quantity0  , :supply_w_id0  )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT( 2 )         , :id1  , :ol_quantity1  , :supply_w_id1  )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT( 3 )         , :id2  , :ol_quantity2  , :supply_w_id2  )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT( 4 )         , :id3  , :ol_quantity3  , :supply_w_id3  )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT( 5 )         , :id4  , :ol_quantity4  , :supply_w_id4  )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT( 6 )         , :id5  , :ol_quantity5  , :supply_w_id5  )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT( 7 )         , :id6  , :ol_quantity6  , :supply_w_id6  )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT( 8 )         , :id7  , :ol_quantity7  , :supply_w_id7  )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT( 9 )         , :id8  , :ol_quantity8  , :supply_w_id8  )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT(10)         , :id9  , :ol_quantity9  , :supply_w_id9  )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT(11)         , :id10, :ol_quantity10, :supply_w_id10 )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT(12)         , :id11, :ol_quantity11, :supply_w_id11 )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT(13)         , :id12, :ol_quantity12, :supply_w_id12 )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT(14)         , :id13, :ol_quantity13, :supply_w_id13 )
	) AS ITEMLIST
, I_QTY
FROM Table( VALUES ( SMALLINT(15)         , :id14, :ol_quantity14, :supply_w_id14 )
	) AS ITEMLIST
)
AS NEW_OL_ALL
WHERE NEW_OL_ALL.I_PRICE IS NOT NULL

EXEC SQL DECLARATE SECTION;
int storedProcRc;
int inputItemCount;
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)
                WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
                ) AS NEW_OL_ALL
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
              ,OL_DELIVERY_D
              ,I_QTY
              ,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
          ,:id0 AS I_ID
          ,:supply_w_id0 AS I_SUPPLY_W_ID
          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
                                      )
                      ) 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 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
              ,OL_DELIVERY_D
              ,I_QTY
              ,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
          ,:id0 AS I_ID
          ,:supply_w_id0 AS I_SUPPLY_W_ID
          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
                                      )
                      ) 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 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
              ,OL_DELIVERY_D
              ,I_QTY
              ,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
          ,:id0 AS I_ID
          ,:supply_w_id0 AS I_SUPPLY_W_ID
          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
                                      )
                      ) 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 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
              ,OL_DELIVERY_D
              ,I_QTY
              ,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
          ,:id0 AS I_ID
          ,:supply_w_id0 AS I_SUPPLY_W_ID
          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
                                      )
                      ) 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 DATA
)
AS INS
;
EXEC SQL DECLARE ISOL_Remote_7 CURSOR FOR
WITH DATA AS ( SELECT O_ID, W_ID, OL_NUMBER, I_ID, SUPPLY_W_ID, I_QTY
FROM Table( VALUES
( SMALLINT(1) , :id0 , :ol_quantity0 , :supply_w_id0 )
,...
( SMALLINT(5) , :id4 , :ol_quantity4 , :supply_w_id4 )
,...
( SMALLINT(6) , :id6 , :ol_quantity6 , :supply_w_id6 )
,...
( SMALLINT(8) , :id8 , :ol_quantity8 , :supply_w_id8 )
) AS X ( OL_NUMBER, I_ID, I_QTY, SUPPLY_W_ID )
)
AS ITEM
,
TABLE( NEW_OL_ALL(I_ID, I_QTY, W_ID, SUPPLY_W_ID) ) AS NEW_OL_ALL
WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
)
)
AS INS
;
EXEC SQL DECLARE ISOL_Remote_8 CURSOR FOR
WITH DATA AS ( SELECT O_ID, W_ID, OL_NUMBER, I_ID, 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) , :id6 , :ol_quantity6 , :supply_w_id6 )
,...
( SMALLINT(7) , :id7 , :ol_quantity7 , :supply_w_id7 )
) AS X ( OL_NUMBER, I_ID, I_QTY, SUPPLY_W_ID )
)
AS ITEM
,
TABLE( NEW_OL_ALL(I_ID, I_QTY, W_ID, SUPPLY_W_ID, OL_NUMBER, SUPPLY_W_ID) ) AS NEW_OL_ALL
WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
)
)
AS INS
;
EXEC SQL DECLARE ISOL_Remote_9 CURSOR FOR
WITH DATA AS ( SELECT O_ID, W_ID, OL_NUMBER, I_ID, 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, SUPPLY_W_ID )
)
AS ITEM
,
TABLE( NEW_OL_ALL(I_ID, I_QTY, W_ID, SUPPLY_W_ID, OL_NUMBER, SUPPLY_W_ID, OL_NUMBER, SUPPLY_W_ID) ) AS NEW_OL_ALL
WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
)
) FROM NEW TABLE ( INSERT INTO ORDER_LINE
( O_ID, O_D_ID, O_W_ID, OL_NUMBER, OL_I_ID, OL_SUPPLY_W_ID, OL_DELIVERY_D, OL_QUANTITY, OL_AMOUNT, OL_DIST_INFO ) 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_DIST_INFO, I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY )
                      INCLUDE ( I_PRICE DECIMAL(5,2), I_NAME CHAR(24), I_DATA VARCHAR(50), S_DATA VARCHAR(50), S_QUANTITY SMALLINT )
               ) AS NEW_OL_LOCAL
WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
)

EXEC SQL DECLARE ISOL_Local_3 CURSOR FOR
WITH DATA AS (  SELECT  O_ID, D_ID, W_ID, OL_NUMBER, OL_I_ID, OL_SUPPLY_W_ID, OL_DELIVERY_D, OL_QUANTITY, 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, OL_I_ID, OL_SUPPLY_W_ID, OL_DELIVERY_D, OL_QUANTITY, 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, 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 )
INCLUDE (   I_PRICE DECIMAL(5,2), S_QUANTITY SMALLINT )
SELECT   O_ID, D_ID, W_ID, OL_NUMBER, I_ID, I_SUPPLY_W_ID, OL_DELIVERY_D, OL_QUANTITY, OL_DIST_INFO, TOTAL_PRICE, I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY
FROM DATA
)

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, 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 )
WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
)

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_DIST_INFO, I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY )
                      INCLUDE ( I_PRICE DECIMAL(5,2), I_NAME CHAR(24), I_DATA VARCHAR(50), S_DATA VARCHAR(50), S_QUANTITY SMALLINT )
               ) AS NEW_OL_LOCAL
WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
)
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
)
AS OLITEM)

, 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
)
AS OLITEM)

, 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
)

EXEC SQL DECLARE ISOL_Local_10 CURSOR FOR

WITH DATA AS ( SELECT O_ID
, D_ID
, W_ID
, OL_NUMBER
, OL_I_ID
, OL_SUPPLY_W_ID
, OL_DELIVERY_D
, TOTAL_PRICE
, I_ID
, I_QTY
, I_NAME
, I_DATA
, S_DATA
, S_QUANTITY
FROM DATA
)
AS INS
;

FROM SELECT :next_o_id as O_ID
, :w_id AS W_ID
, :id AS D_ID
, :id AS O_ID
, :id AS OL_NUMBER
, :id AS I_ID
, :id AS 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

EXEC SQL DECLARE ISOL_Local_11 CURSOR FOR

WITH DATA AS ( SELECT O_ID
, D_ID
, W_ID
, OL_NUMBER
, OL_I_ID
, OL_SUPPLY_W_ID
, OL_DELIVERY_D
, TOTAL_PRICE
, I_ID
, I_QTY
, I_NAME
, I_DATA
, S_DATA
, S_QUANTITY
FROM DATA
)
AS INS
;

FROM SELECT :next_o_id as O_ID
, :w_id AS W_ID
, :id AS D_ID
, :id AS O_ID
, :id AS OL_NUMBER
, :id AS I_ID
, :id AS 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

EXEC SQL DECLARE ISOL_Local_12 CURSOR FOR

WITH DATA AS ( SELECT O_ID
, D_ID
, W_ID
, OL_NUMBER
, OL_I_ID
, OL_SUPPLY_W_ID
, OL_DELIVERY_D
, TOTAL_PRICE
, I_ID
, I_QTY
, I_NAME
, I_DATA
, S_DATA
, S_QUANTITY
FROM DATA
)
AS INS
;
EXEC SQL
SELECT D_TAX, D_NEXT_O_ID INTO :dist_tax, :next_o_id
FROM OLD TABLE (UPDATE DISTRICT
WHERE D_W_ID = :w_id
AND D_ID = :d_id
)
FROM NEW_TABLE (VALUES
   ( 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
FROM NEW_TABLE ( NEW_OL_LOCAL ( I_ID, I_QTY )
   , OL_NUMBER
   , I_ID
   , OL_DELIVERY_D
   , OL_SUPPLY_W_ID
   , OL_DIST_INFO
) AS 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_SUPPLY_W_ID
   , OL_DELIVERY_D
   , OL_QTY
   , OL_TOTALPRICE
   , OL_DIST_INFO
   , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY
) AS AS INS
FROM DATA
AS INS
EXECL DECLARE ISOL_Local_15 CURSOR FOR
WITH DATA AS ( SELECT O_ID
   , D_ID
   , W_ID
   , OL_NUMBER
   , OL_I_ID
   , SUPPLY_W_ID
   , DELIVERY_D
   , QTY
   , TOTALPRICE
   , DIST_INFO
   , OL_I_ID
   , OL_SUPPLY_W_ID
   , OL_DELIVERY_D
   , OL_QUANTITY
   , OL_AMOUNT
   , OL_DIST_INFO
   , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY
) AS 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_QTY
   , OL_TOTALPRICE
   , OL_DIST_INFO
   , I_PRICE
   , I_NAME
   , I_DATA
   , S_DATA
   , S_QUANTITY
) AS 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_QTY
   , OL_TOTALPRICE
   , OL_DIST_INFO
   , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY
) AS AS INS
FROM DATA
AS INS
EXEC SQL DECLARE ISOL_Local_15 CURSOR FOR
WITH DATA AS ( SELECT O_ID
   , D_ID
   , W_ID
   , OL_NUMBER
   , OL_I_ID
   , SUPPLY_W_ID
   , DELIVERY_D
   , QTY
   , TOTALPRICE
   , DIST_INFO
   , OL_I_ID
   , OL_SUPPLY_W_ID
   , OL_DELIVERY_D
   , OL_QUANTITY
   , OL_AMOUNT
   , OL_DIST_INFO
   , I_PRICE
   , I_NAME
   , I_DATA
   , S_DATA
   , S_QUANTITY
) AS 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_QTY
   , OL_TOTALPRICE
   , OL_DIST_INFO
   , I_PRICE
   , I_NAME
   , I_DATA
   , S_DATA
   , S_QUANTITY
) AS 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_QTY
   , OL_TOTALPRICE
   , OL_DIST_INFO
   , I_PRICE
   , I_NAME
   , I_DATA
   , S_DATA
   , S_QUANTITY
) AS AS INS
FROM DATA
AS INS
EXEC SQL DECLARE ISOL_Local_15 CURSOR FOR
WITH DATA AS ( SELECT O_ID
   , D_ID
   , W_ID
   , OL_NUMBER
   , OL_I_ID
   , SUPPLY_W_ID
   , DELIVERY_D
   , QTY
   , TOTALPRICE
   , DIST_INFO
   , OL_I_ID
   , OL_SUPPLY_W_ID
   , OL_DELIVERY_D
   , OL_QUANTITY
   , OL_AMOUNT
   , OL_DIST_INFO
   , I_PRICE
   , I_NAME
   , I_DATA
   , S_DATA
   , S_QUANTITY
) AS 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_QTY
   , OL_TOTALPRICE
   , OL_DIST_INFO
   , I_PRICE
   , I_NAME
   , I_DATA
   , S_DATA
   , S_QUANTITY
) AS 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_QTY
   , OL_TOTALPRICE
   , OL_DIST_INFO
   , I_PRICE
   , I_NAME
   , I_DATA
   , S_DATA
   , S_QUANTITY
) AS AS INS
FROM DATA
AS INS
EXEC SQL DECLARE ISOL_Local_15 CURSOR FOR
WITH DATA AS ( SELECT O_ID
   , D_ID
   , W_ID
   , OL_NUMBER
   , OL_I_ID
   , SUPPLY_W_ID
   , DELIVERY_D
   , QTY
   , TOTALPRICE
   , DIST_INFO
   , OL_I_ID
   , OL_SUPPLY_W_ID
   , OL_DELIVERY_D
   , OL_QUANTITY
   , OL_AMOUNT
   , OL_DIST_INFO
   , I_PRICE
   , I_NAME
   , I_DATA
   , S_DATA
   , S_QUANTITY
) AS 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_QTY
   , OL_TOTALPRICE
   , OL_DIST_INFO
   , I_PRICE
   , I_NAME
   , I_DATA
   , S_DATA
   , S_QUANTITY
) AS 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_QTY
   , OL_TOTALPRICE
   , OL_DIST_INFO
   , I_PRICE
   , I_NAME
   , I_DATA
   , S_DATA
   , S_QUANTITY
) AS AS INS
FROM DATA
AS INS
EXEC SQL DECLARE ISOL_Local_15 CURSOR FOR
WITH DATA AS ( SELECT O_ID
   , D_ID
   , W_ID
   , OL_NUMBER
   , OL_I_ID
   , SUPPLY_W_ID
   , DELIVERY_D
   , QTY
   , TOTALPRICE
   , DIST_INFO
   , OL_I_ID
   , OL_SUPPLY_W_ID
   , OL_DELIVERY_D
   , OL_QUANTITY
   , OL_AMOUNT
   , OL_DIST_INFO
   , I_PRICE
   , I_NAME
   , I_DATA
   , S_DATA
   , S_QUANTITY
) AS 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_QTY
   , OL_TOTALPRICE
   , OL_DIST_INFO
   , I_PRICE
   , I_NAME
   , I_DATA
   , S_DATA
   , S_QUANTITY
) AS 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_QTY
   , OL_TOTALPRICE
   , OL_DIST_INFO
   , I_PRICE
   , I_NAME
   , I_DATA
   , S_DATA
   , S_QUANTITY
) AS AS INS
FROM DATA
AS INS
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
}

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;
EXEC SQL OPEN ISOL_Remote_7;
NEW_CURSOR_OPEN_ERROR
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_9;
NEW_CURSOR_OPEN_ERROR
EXEC SQL FETCH ISOL_Remote_9 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
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
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
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
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
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
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
EXEC SQL FETCH ISOL_Remote_15 INTO item_price, item_name, i_data, stockDistrictInformation, s_data, s_quantity;
NEW_CURSOR_ERROR
break;
default:
&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 )    )       {          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
if( sqlca.sqlcode != 0 )
{          neword->s_transtatus = INVALID_ITEM ;          EXEC SQL ROLLBACK WORK ;
| }     else
{          neword->s_transtatus = FATAL_SQLERROR;
| }     goto ferror;       }
/*---------------------------------------------*/
for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )    {
EXEC SQL FETCH ISOL_Remote_9 INTO item_price, item_name, i_data, stockDistrictInformation, s_data, s_quantity;
EXEC SQL COMMIT;
return ( storedProcRc ) ;
sql_error:
break ;
if ( sqlca.sqlcode >= 0 )    {       storedProcRc = SQLZ_HOLD_PROC ;
| }     else    {       storedProcRc = SQLZ_DISCONNECT_PROC ;
| }
/*---------------------------------------------*/  mexit:
for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )    {
EXEC SQL FETCH ISOL_Remote_9 INTO item_price, item_name, i_data, stockDistrictInformation, s_data, s_quantity;
EXEC SQL ROLLBACK WORK;     if ( sqlca.sqlcode != 0 )
{          neword->s_transtatus = INVALID_ITEM ;          EXEC SQL ROLLBACK WORK ;
| }     else
{          neword->s_transtatus = FATAL_SQLERROR;
| }     goto ferror;       }
/*---------------------------------------------*/
if ( neword->s_O_OL_CNT == in_neword->s_O_OL_CNT  )        {
neword->s_transtatus = TRAN_OK ;
EXEC SQL COMMIT;
}
for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ; inputItemArrayIndex++ )    {
EXEC SQL FETCH ISOL_Remote_9 INTO item_price, item_name, i_data, stockDistrictInformation, s_data, s_quantity;
EXEC SQL COMMIT;
if ( neword->s_O_OL_CNT != in_neword->s_O_OL_CNT )
{          neword->s_transtatus = TRAN_OK ;
| }
| }     else
{          neword->s_transtatus = FATAL_SQLERROR;
| }     EXEC SQL ROLLBACK WORK ;
if ( sqlca.sqlcode != 0 )
{          neword->s_transtatus = FATAL_SQLERROR;
| }
static unsigned char skip[256] = {8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8};

static int is_ORIGINAL(char *string, short length) {
    char            *cur_string;
    char            *end_string;
    cur_string = string + 7;
    end_string = string + length;
    while (cur_string < end_string)     {
        int skip_dist = skips[*cur_string];
        while (skip_dist > 0) {
            cur_string += skip_dist;
            skip_dist = skips[*cur_string];
        }
    }
    return (cur_string[4] == 'G');
}

EXEC SQL BEGIN DECLARE SECTION;
    struct s_data_type { short len ; char data[ 16 ] ; } c_last_input;
    sqlint32   o_id ;      //##sqlint32   c_id ;      short      o_carrier_id ;      //##sqlint64   o_entry_d ;
    struct s_data_type { short len ; char data[ 16 ] ; } c_last_input ;
    double c_balance ;
    //##struct s_data_type { short len ; char data[ 16 ] ; } c_last_input ;
EXEC SQL END DECLARE SECTION;

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 ) ;
    }
    else    {
        goto mexit ;
    }

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 )
{
    EXEC SQL
        SELECT O_ID, O_CARRIER_ID, O_ENTRY_D, C_BALANCE, C_FIRST, C_MIDDLE, C_ID
        // From input values
        INTO :o_id, :o_carrier_id, :o_entry_d, :c_balance, :c_first, :c_middle, :c_last
        FROM TABLE ( ORD_C_LAST( :w_id , :d_id )
                ) AS ORD_C_LAST
exit:
    return ( result ) ;
}

TPC Benchmark™ C Full Disclosure Report - IBM System p 570 Model 9117-MMA
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) ;
go to 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 );
sqerror(ORDSTAT_SQL, "COMMIT", __FILE__, __LINE__ , &sqlca);
go to 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 ;
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 {
  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];
  char s_C_LAST[17];
};

struct in_neword_struct {
  int16_t len;
  int16_t pad[SPGENERAL_PAD];
  int16_t duplicate_items;
  struct items_struct { float s_I_PRICE; int16_t s_S_QUANTITY; char s_I_NAME[25]; } item[15];
  int32_t s_C_ID;
};

struct out_neword_struct {
  int16_t len;
  int16_t pad[SPGENERAL_PAD];
  struct items_struct { double s_OL_SUPPLY_W_ID; double s_OL_AMOUNT; int32_t s_OL_I_ID; int16_t s_S_QUANTITY; float s_I_PRICE; char s_I_NAME[25]; } oitems_struct[15];
  int32_t s_D_ID;
  int32_t s_O_OL_CNT;
  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_LAST[17];
  char s_SINCE_time[27];
};

struct oitems_struct {
  double s_OL_SUPPLY_W_ID;
  double s_OL_AMOUNT;
  int32_t s_OL_I_ID;
};

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

#include "db2tpcc.h"
#include <sys/types.h>
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);

#endif // __TPCCDBG_H

# endif

# tpccenv.sh

# Licensed Materials - Property of IBM
# ...

# The Kit Version
# export TPCC_VERSION=CK070418
# 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 daii stored proc's for the TPC-C driver. Set to either DARIVERSION or NONDARI;
# export TPCC_SPTYPE=HOSP
# export TPCC_SPTYPE=ISPGENERAL2
# export TPCC_SPTYPE=SPGENERAL
# export TPCC_SPTYPE=DARI2SQLDA
export DB2VERSION=v8

# The schema name is typically the SQL authorization 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 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_SDIR=$TPCC_SQLIB/tdproc
# export TPCC_FENCED=NO
**Appendix - B: Tunable Parameters**

**B.1 Database Parameters.**

**db.cfg.out**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database configuration for Database TPCC</td>
<td>= 0x0b00</td>
</tr>
<tr>
<td>Database release level</td>
<td>= 0x0b00</td>
</tr>
<tr>
<td>Database territory</td>
<td>= US</td>
</tr>
<tr>
<td>Database code page</td>
<td>= 819</td>
</tr>
<tr>
<td>Database code set</td>
<td>= ISO8859-1</td>
</tr>
<tr>
<td>Database country/region code</td>
<td>= 1</td>
</tr>
<tr>
<td>Database collating sequence</td>
<td>= T奋力M</td>
</tr>
<tr>
<td>Alternate collating sequence (ALT_COLLATE)</td>
<td>= 4096</td>
</tr>
<tr>
<td>Dynamic SQL Query management (DYN_QUERY_MGMT)</td>
<td>= DISABLE</td>
</tr>
<tr>
<td>Discovery support for this database (DISCOVER_DB)</td>
<td>= ENABLE</td>
</tr>
<tr>
<td>Restrict access</td>
<td>= NO</td>
</tr>
<tr>
<td>Default query optimization class (DFT_QUERYOPT)</td>
<td>= 5</td>
</tr>
<tr>
<td>Degree of parallelism (DFT DEGREE)</td>
<td>= 1</td>
</tr>
<tr>
<td>Continue upon arithmetic exceptions (DFT_SOMATHFAIL)</td>
<td>= NO</td>
</tr>
<tr>
<td>Default refresh age (DFT_REFRESH AGE)</td>
<td>= 0</td>
</tr>
<tr>
<td>Default maintained table types for ugl (DFT MTB TYPES)</td>
<td>= SYSTEM</td>
</tr>
<tr>
<td>Number of frequent values retained (NUM FREQUENCES)</td>
<td>= 10</td>
</tr>
<tr>
<td>Number of quantities retained (NUM QUANTITIES)</td>
<td>= 20</td>
</tr>
<tr>
<td>Backup pending</td>
<td>= NO</td>
</tr>
<tr>
<td>Rollforward pending</td>
<td>= YES</td>
</tr>
<tr>
<td>Restore pending</td>
<td>= NO</td>
</tr>
<tr>
<td>Multi-page file allocation enabled</td>
<td>= YES</td>
</tr>
<tr>
<td>Log retain for recovery status</td>
<td>= RECOVERY</td>
</tr>
<tr>
<td>User exit for logging status</td>
<td>= NO</td>
</tr>
<tr>
<td>Self tuning memory (SELF_TUNING_MEM)</td>
<td>= OFF</td>
</tr>
<tr>
<td>Size of database shared memory (4KB) (DATABASE_MEMORY)</td>
<td>= 199909080</td>
</tr>
<tr>
<td>Database memory threshold (DB_MEM_THRESHOLD)</td>
<td>= 10</td>
</tr>
<tr>
<td>Max storage for lock list (4KB) (LOCKLIST)</td>
<td>= 32000</td>
</tr>
<tr>
<td>Percent of lock list applications (MAXLOCKS)</td>
<td>= 20</td>
</tr>
<tr>
<td>Package cache size (4KB) (PCKCACHESIZE)</td>
<td>= 20000</td>
</tr>
<tr>
<td>Sort heap size for sorted sorts (4KB) (SHEAPPHERE_SHR)</td>
<td>= 12000</td>
</tr>
<tr>
<td>Sort list heap (4KB) (SORTHERE)</td>
<td>= 16</td>
</tr>
<tr>
<td>Database heap (4KB) (DBHEAP)</td>
<td>= 524288</td>
</tr>
<tr>
<td>Catalog cache size (4KB) (CATALOGCACHESZ)</td>
<td>= (MAXAPPLS*4)</td>
</tr>
<tr>
<td>Log buffer size (4KB) (LOGBUFFER)</td>
<td>= 66000</td>
</tr>
<tr>
<td>Unlocked heap size (4KB) (UTL_HEAP_SZ)</td>
<td>= 5000</td>
</tr>
<tr>
<td>Buffer pool pages (BUFFPAGE)</td>
<td>= 11000</td>
</tr>
<tr>
<td>Max size of appl. group mem set (4KB) (APPGROUP_MEM_SZ)</td>
<td>= 20000</td>
</tr>
<tr>
<td>Max appl. control heap size (4KB) (APPL_CTL_HEAP_SZ)</td>
<td>= 12</td>
</tr>
<tr>
<td>SQL statement heap (4KB) (STMTHEAP)</td>
<td>= 65000</td>
</tr>
<tr>
<td>Default application heap (4KB) (APPLHEAPSZ)</td>
<td>= 2500</td>
</tr>
<tr>
<td>Statistics heap size (4KB) (STAT_HEAP_SZ)</td>
<td>= 4384</td>
</tr>
<tr>
<td>Internal for checking deadlock (ms) (CLCHKTIMES)</td>
<td>= 3000</td>
</tr>
<tr>
<td>Lock timeouts (sec) (LOCKTIMEOUT)</td>
<td>= 1</td>
</tr>
<tr>
<td>Changed pages threshold (CHNGPPS_THRESH)</td>
<td>= 99</td>
</tr>
<tr>
<td>Number of asynchronous page cleaners (NUM OCLEANERS)</td>
<td>= 1</td>
</tr>
<tr>
<td>Number of ID servers (NUM IOSERVERS)</td>
<td>= 1</td>
</tr>
<tr>
<td>Index sort flag (INDEXSORT)</td>
<td>= YES</td>
</tr>
<tr>
<td>Default prefix size (pages) (DFT_PREFETCH_SZ)</td>
<td>= AUTOMATIC</td>
</tr>
<tr>
<td>Track modified pages (TRACKMOD)</td>
<td>= OFF</td>
</tr>
<tr>
<td>Default number of containers</td>
<td>= 1</td>
</tr>
<tr>
<td>Default tablespace extension (pages) (DFT EXTEND_SZ)</td>
<td>= 12</td>
</tr>
<tr>
<td>Max number of active applications (MAXAPPLS)</td>
<td>= 5050</td>
</tr>
<tr>
<td>Average number of active applications (AVG APPLS)</td>
<td>= 100</td>
</tr>
<tr>
<td>Max DB files open per application (MAXFILEP)</td>
<td>= 800</td>
</tr>
<tr>
<td>Log file size (4KB) (LOGFILESZ)</td>
<td>= 262144</td>
</tr>
<tr>
<td>Number of primary log files (LOG PRIMARY)</td>
<td>= 250</td>
</tr>
<tr>
<td>Number of secondary log files (LOGSECOND)</td>
<td>= 0</td>
</tr>
<tr>
<td>Changed path to log files (NEWLOGPATH)</td>
<td>= -1</td>
</tr>
<tr>
<td>Path to log files</td>
<td>= -1</td>
</tr>
<tr>
<td>Overflow log path (OVERFLOWLOGPATH)</td>
<td>= 1</td>
</tr>
<tr>
<td>Mirror log path (MIRRORLOGPATH)</td>
<td>= 2</td>
</tr>
<tr>
<td>First active log file size (50000000 LOG)</td>
<td>= 1000000000</td>
</tr>
<tr>
<td>Block log on disk full (BLK LOG_DISK_FULL)</td>
<td>= NO</td>
</tr>
<tr>
<td>Percent max primary log space by transaction (MAX LOG)</td>
<td>= 0</td>
</tr>
<tr>
<td>Num. of active log files for 1 transaction (NUM LOG SPAN)</td>
<td>= 0</td>
</tr>
<tr>
<td>Group commit count (MINCNOCOMIT)</td>
<td>= 3</td>
</tr>
<tr>
<td>Percent log file reclaimed before soft chop (SOFTMAX)</td>
<td>= 10025</td>
</tr>
<tr>
<td>Log retain for recovery enabled (LOGRETAIL)</td>
<td>= RECOVERY</td>
</tr>
<tr>
<td>User exit for logging enabled (USEREXIT)</td>
<td>= OFF</td>
</tr>
<tr>
<td>HADR database role</td>
<td>= STANDARD</td>
</tr>
<tr>
<td>HADR local host name (HADR_LOCAL_HOST)</td>
<td>=</td>
</tr>
<tr>
<td>HADR local service name (HADR_LOCAL_SVC)</td>
<td>=</td>
</tr>
<tr>
<td>HADR remote host name (HADR_REMOTE_HOST)</td>
<td>=</td>
</tr>
<tr>
<td>HADR remote service name (HADR_REMOTE_SVC)</td>
<td>=</td>
</tr>
<tr>
<td>HADR instance name of remote server (HADR_REMOTE INST)</td>
<td>=</td>
</tr>
<tr>
<td>HADR timeout value (HADR_TIMEOUT)</td>
<td>= 120</td>
</tr>
<tr>
<td>HADR log write synchronization mode (HADR SYMCKMODE)</td>
<td>= NEARSYNC</td>
</tr>
<tr>
<td>First log arch method (LOGARCHMETH1)</td>
<td>= LOGRETAIL</td>
</tr>
<tr>
<td>Options for logarchmeth1 (LOGARCHOPT1)</td>
<td>=</td>
</tr>
<tr>
<td>Second log arch method (LOGARCHMETH2)</td>
<td>= OFF</td>
</tr>
<tr>
<td>Options for logarchmeth2 (LOGARCHOPT2)</td>
<td>=</td>
</tr>
<tr>
<td>Failover log arch path (FAILARCHPATH)</td>
<td>=</td>
</tr>
<tr>
<td>Number of log archive retries on error (NUMARCHRETRY)</td>
<td>= 5</td>
</tr>
<tr>
<td>Arch log retry Delay (secs) (ARCHRETIRETRYDELAY)</td>
<td>= 20</td>
</tr>
<tr>
<td>Vendor options (VENDOROPT)</td>
<td>=</td>
</tr>
<tr>
<td>Auto restart enabled (AUTORESTART)</td>
<td>= ON</td>
</tr>
<tr>
<td>Index re-creation time and redo index build (INDEXREC)</td>
<td>= SYSTEM (RESTART)</td>
</tr>
<tr>
<td>Log pages during index build (LOGINDEXBUILD)</td>
<td>= OFF</td>
</tr>
<tr>
<td>Default number of loadrec sessions (DFT LOADREC SES)</td>
<td>= 1</td>
</tr>
<tr>
<td>Number of database backups to retain (NUM DB BACKUPS)</td>
<td>= 12</td>
</tr>
<tr>
<td>Recovery history retention (days) (RECS_HS RETENTION)</td>
<td>= 366</td>
</tr>
<tr>
<td>TSM management class (TSM MGMTCLASS)</td>
<td>=</td>
</tr>
<tr>
<td>TSM node name (TSM NODENAME)</td>
<td>=</td>
</tr>
<tr>
<td>TSM owner (TSM OWNER)</td>
<td>=</td>
</tr>
<tr>
<td>TSM password (TSM PASSWORD)</td>
<td>=</td>
</tr>
<tr>
<td>Automatic maintenance (AUTO MAINT)</td>
<td>= OFF</td>
</tr>
<tr>
<td>Automatic database backup (AUTO_DB_BACKUP)</td>
<td>= OFF</td>
</tr>
<tr>
<td>Automatic table maintenance (AUTO_TBL_MAINT)</td>
<td>= OFF</td>
</tr>
<tr>
<td>Automatic runstats (AUTO RUNSTATS)</td>
<td>= OFF</td>
</tr>
<tr>
<td>Automatic statistics profiling (AUTO STATS PROF)</td>
<td>= OFF</td>
</tr>
<tr>
<td>Automatic profile updates (AUTO_PROF_UPD)</td>
<td>= OFF</td>
</tr>
<tr>
<td>Automatic reorganization (AUTO_REORG)</td>
<td>= OFF</td>
</tr>
</tbody>
</table>

**dbm.cfg.out**

Database Manager Configuration

- **Node type**: Database Server with local clients
- **Database manager release level**: 0x0b00
- **CPU speed (millisecond/instruction)**: (CPUSPEED) = 2.7567121e-07
- **Max number of concurrently active databases (NUMDB)**: 1
- **Federated Database System Support (FEDERATED)**: OFF
- **Transaction processor monitor name (TP_MON_NAME)**: TVMON
- **Default charge-back account (DFT_ACCOUNT_STR)**: CHARGE
- **Java Development Kit installation path (JDK_PATH)**: /home/tpcc/java/jdk64
- **Diagnostic error capture level (DIAGLEVEL)**: 1
- **Diagnostic data directory path (DIAGPATH)**: /mnt/tppc

- **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
- **Monitor**: (DFT_MON_TIMEOUT) = OFF
- **Unit of work**: (DFT_MON_UOW) = OFF
- **SYSADM group name**: SYSADM (GROUP) = STANDARD
- **SYSCtrl group name**: SYSCtrl (GROUP) = SYSCtrl
- **SYSAudit group name**: SYSAudit (GROUP) = SYSAudit
- **SYSEM group name**: SYSEM (GROUP) = SYSEM
- **Client UserId-Password Plugin**: (CLNT_PW_PLUGIN) = Tivoli
- **Client Kerberos Plugin**: (CLNT_KB_PLUGIN) = Tivoli
- **Group Plugin**: (GROUP_PLUGIN) = Tivoli
- **GSS Plugin for Local Authentication**: (LOCAL_GSSPLUGIN) = Tivoli
- **GSS Plugin Mode**: (GSS_PLUGIN MODE) = UNENCED
- **Server List of GSS Plugins**: (GSSPLUGIN_GSSPLUGIN_LIST) = Tivoli
- **Server User-Id Password Plugin**: (SERVER PW PLUGIN) = Tivoli
- **Server Connection Authentication**: (SERVER_AUTH) = NOT SPECIFIED
- **Database manager authentication**: (AUTHENTICATION) = CLIENT
- **Cataloging allowed without authority**: (CATALOG_NOAUTH) = YES
- **Trusted all clients**: (TRUST_ALLCLIENTS) = YES
- **Trusted client authentication**: (TRUST_CLNTAUTH) = CLIENT
- **Bypass federated authentication**: (Bypass AUTHENTICATION) = NO
- **Default database path (DBFILEPATH)**: /home/tpcc

- **Database manager release level**: 0x0b00
- **Database monitor heap size (4KB) (MON_HEAP_SZ) = 4906
- **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) = AUTO
- **Backup buffer default size (4KB) (BACKBUFSIZE) = 1024
- **Restore buffer default size (4KB) (RESTBUFSIZE) = 1024
- **Sort heap threshold (4KB) (SHEAPPHERE_THRES) = 0
- **Directory cache support (DIR_CACHE) = YES

TPC Benchmark™ C Full Disclosure Report - IBM System p 570 Model 9117-MMA Page 81 of 318
Application support layer heap size (KB) [ASLHEAPSZ] = 15
Max requester I/O block size (bytes) [RQI0BLK] = 4096
Query heap size (KB) [QUERY_HEAP_SZ] = 1000

Workload impact by throttled utilities [UTIL_IMPACT_LIM] = 10

Priority of agents [AGENTPRI] = 60
Max number of existing agents [MAXAGENTS] = 5050
Agent pool size [NUM_POOLAGENTS] = 0
Initial number of agents in pool [NUM_INTAGENTS] = 0
Max number of coordinating agents [MAX_COORDAGENTS] = MAXAGENTS
Max no. of coordinating agents [MAXCOORDAGENTS] = MAXCOORDAGENTS
Max number of client connections [MAX_CONNECTIONS] = MAXCOORDAGENTS

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 response interval (sec) [RESP_INTERVAL] = 180

SPM name [SPM_NAME] =
SPM log size [SPM_LOG_FILE_SZ] = 256
SPM log path [SPM_LOG_PATH] =
TCP/IP Service name [TCP/IPNAME] =
Discover server instance [DISCOVER_INST] = ENABLE

Maximum query degree of parallelism [MAX_QUERYDEGREES] = ANY
Enable intra-partition parallelism [INTRA_PARALLELS] = NO

No. of int. communication buffers (4KB) [FCM_NUM_BUFFERS] = AUTOMATIC
No. of int. communication channels [FCM_NUM_CHANNELS] = AUTOMATIC

db2set.cfg.out

```
[i] DB2_LARGE_PAGE_MEM=DB:16GB
[i] DB2_Resource_policy=home/tpcc/tpc-c.ibm/cfg/affinity.cfg
[i] DB2_SSLindy_COMM_BUFFER=Y
[i] DB2_USE_ALTERNATE_PAGE_CLEANING=YES
[i] DB2_MAX_NON_TABLE_LOCKS=500
[i] DB2_KEEPTABLELOCKCONNECTION
[i] DB2_NUM_OWP_DAEMONS=0
[i] DB2_EVENT_LOG_CONFIG=OFF
[i] DB2_NO_FORK_CHECK=ON
[i] DB2_ALLOCATION_SIZE=1577216
[i] DB2_APM_PERFORMANCE=ALL
[i] DB2ASSUMEUPDATE=ON
[i] DB2CHECKCLIENT_TIMEOUT=0
[i] DB2_HASH_IGNOFF
[i] DB2CHKSQD=off
[i] DB2COLLECT_TS_REC_INFO=off
[i] DB2COLLECT_MEMORY_AFFINITY
[i] DB2_CONN=tpc
[i] DB2CPUThrottle
[i] DB2_TRUSTED_BIND=ON
```

TPC Benchmark™ C Full Disclosure Report - IBM System p 570 Model 9117-MMA Page 82 of 318
B.2 Transaction Monitor Parameters

tppCom.tpcc com_settings.txt

Transactions not supported
Enable object pooling
Maximum pool size 23
Maximum pool size 24
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,cc,1b,13,04,00,00,00,00
"TcpWindowSize"=dword:00040000
"TcpBufferPool"=dword:00000004
"TcpAutoLevelWindow"=dword:00000001
"TcpAutoLevelInterval"=dword:00000001
"TcpInitAlloc"=dword:00000001
"TcpSendWindow"=dword:00000001
"TcpSlidingWindow"=dword:00000001
"TcpWindowSize"=dword:00000000

[HKEY_LOCAL_MACHINE\SOFTWARE\TPCC]
defps                     1      1      1      0      1      boolean           D --------------------------------------------------------------------------------
"ScheduleManagerLoad"=dword:00000001
"ScheduleManagerPort"=dword:00000001
"ScheduleManagerThreads"=dword:00000001
"ScheduleManagerConnections"=dword:00000001
"ScheduleManagerTimeout"=dword:00000001
"ScheduleManagerWindowSize"=dword:00000001
"ScheduleManagerWindowSize"=dword:00000000
"ScheduleManagerWindowSize"=dword:00000000

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\Interfaces\{408E911E-8040-49F7-AF9A-4A9953C9BF2B}\]
dlvyLogPath"="c:\InetPub\wwwroot\tpcc\dlvy"
dlvyQueueLen"=dword:00004e20
dlvyThreads"=dword:00000004

TCPip parameters registry.reg

Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\Interfaces\{408E911E-8040-49F7-AF9A-4A9953C9BF2B}\]
dlvyLogPath"="c:\InetPub\wwwroot\tpcc\dlvy"
dlvyQueueLen"=dword:00004e20
dlvyThreads"=dword:00000004

Tppc software registry.reg

Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SOFTWARE\TPCC]
defps                     1      1      1      0      1      boolean           D --------------------------------------------------------------------------------
TppcBenchmarkSettings.exe
TppcCom.tpcc com_settings.txt
TppcCom.tpcc com_settings.txt

B.3 AIX Parameters

IBM System p 570

TPC Benchmark™ C Full Disclosure Report - IBM System p 570 Model 9117-MMA Page 83 of 318
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value 1</th>
<th>Value 2</th>
<th>Value 3</th>
<th>Value 4</th>
<th>Value 5</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>j2_atimeUpdateSymlink</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>D</td>
</tr>
<tr>
<td>j2_inodeCacheSize</td>
<td>40</td>
<td>400</td>
<td>40</td>
<td>1</td>
<td>1000</td>
<td>D</td>
</tr>
<tr>
<td>j2_maxPageReadAhead</td>
<td>128</td>
<td>128</td>
<td>128</td>
<td>0</td>
<td>64K</td>
<td>4KB</td>
</tr>
<tr>
<td>j2_maxRandomWrite</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>64K</td>
<td>4KB</td>
</tr>
<tr>
<td>j2_maxUsableMaxTransfer</td>
<td>512</td>
<td>512</td>
<td>512</td>
<td>1</td>
<td>4K</td>
<td>pages</td>
</tr>
<tr>
<td>j2_metadataCacheSize</td>
<td>40</td>
<td>400</td>
<td>40</td>
<td>1</td>
<td>1000</td>
<td>D</td>
</tr>
<tr>
<td>j2_nRandomCluster</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>64K</td>
<td>16KB</td>
</tr>
<tr>
<td>mempools</td>
<td>8</td>
<td>8</td>
<td>d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>minfree</td>
<td>960</td>
<td>960</td>
<td>960</td>
<td>200K</td>
<td>4K</td>
<td>pages</td>
</tr>
<tr>
<td>npsaxm</td>
<td>516K</td>
<td>516K</td>
<td>516K</td>
<td>0</td>
<td>16M</td>
<td>4KB</td>
</tr>
<tr>
<td>num_spec_dataset</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>numpffsiks</td>
<td>16512K</td>
<td>16512K</td>
<td>4KB</td>
<td>blocks</td>
<td></td>
<td></td>
</tr>
<tr>
<td>page_stale_method</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>boolean</td>
<td></td>
</tr>
<tr>
<td>pagecoloring</td>
<td>n/a</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>boolean</td>
<td></td>
</tr>
<tr>
<td>pinnable_frames</td>
<td>3209K</td>
<td>3209K</td>
<td>4KB</td>
<td>pages</td>
<td></td>
<td></td>
</tr>
<tr>
<td>pta_balance_threshold</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>99</td>
<td>%</td>
</tr>
<tr>
<td>relais_percentage</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>32K-1</td>
<td>D</td>
</tr>
<tr>
<td>rgclean</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>boolean</td>
<td></td>
</tr>
<tr>
<td>rgcontrol</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>3</td>
<td>D</td>
</tr>
<tr>
<td>scrub</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>boolean</td>
<td></td>
</tr>
<tr>
<td>scrubclean</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>boolean</td>
<td></td>
</tr>
<tr>
<td>soft_min_lgpgs_vmpool</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>90</td>
<td>%</td>
</tr>
<tr>
<td>spec_datasetsize</td>
<td>512</td>
<td>512</td>
<td>512</td>
<td>0</td>
<td></td>
<td>B</td>
</tr>
<tr>
<td>strict_masklen</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>boolean</td>
</tr>
<tr>
<td>strict_maskperm</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>boolean</td>
</tr>
<tr>
<td>v_gshmm</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>boolean</td>
</tr>
<tr>
<td>vm_modlist_threshold</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-2</td>
<td>2G-1</td>
<td>D</td>
</tr>
<tr>
<td>vmm_fork_policy</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>boolean</td>
</tr>
<tr>
<td>vmm_mpsize_support</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>boolean</td>
</tr>
</tbody>
</table>

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 = Boot: can only be changed using bootload 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 be only be incremented
- d = deprecated: deprecated and cannot be changed

Value conventions:
- K = Kib: 2^10
- G = Giga: 2^30
- P = Pets: 2^90
- M = Meg: 2^20
- T = Tera: 2^40
- E = Exa: 2^60

NAME      CUR   DEF   BOOT   MIN   MAX   UNIT  TYPE  DEPENDENCIES
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value 1</th>
<th>Value 2</th>
<th>Value 3</th>
<th>Value 4</th>
<th>Value 5</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>pv_min_pbuf</td>
<td>512</td>
<td>512</td>
<td>512</td>
<td>512</td>
<td>2G-1</td>
<td>D</td>
</tr>
<tr>
<td>sync_release_ilock</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>boolean D</td>
</tr>
</tbody>
</table>

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

C.1 Database Creation Scripts

DDL/ALTTBS_P F 0.ddl

connect to TPCC;
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_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 ts_customer_001 prefetchsize 0;
arTABLESPACE ts_customer_002 prefetchsize 0;
arTABLESPACE ts_customer_003 prefetchsize 0;
arTABLESPACE ts_customer_004 prefetchsize 0;
arTABLESPACE ts_customer_005 prefetchsize 0;
arTABLESPACE ts_customer_006 prefetchsize 0;
arTABLESPACE ts_customer_007 prefetchsize 0;
arTABLESPACE ts_customer_008 prefetchsize 0;
arTABLESPACE ts_customer_009 prefetchsize 0;
arTABLESPACE ts_customer_010 prefetchsize 0;
arTABLESPACE ts_customer_011 prefetchsize 0;
arTABLESPACE ts_customer_012 prefetchsize 0;
arTABLESPACE ts_customer_013 prefetchsize 0;
arTABLESPACE ts_customer_014 prefetchsize 0;
arTABLESPACE ts_customer_015 prefetchsize 0;
arTABLESPACE ts_customer_016 prefetchsize 0;
arTABLESPACE ts_customer_017 prefetchsize 0;
arTABLESPACE ts_customer_018 prefetchsize 0;
arTABLESPACE ts_customer_019 prefetchsize 0;
arTABLESPACE ts_customer_020 prefetchsize 0;
arTABLESPACE ts_customer_021 prefetchsize 0;
arTABLESPACE ts_customer_022 prefetchsize 0;
arTABLESPACE ts_customer_023 prefetchsize 0;
arTABLESPACE ts_customer_024 prefetchsize 0;
arTABLESPACE ts_customer_025 prefetchsize 0;
arTABLESPACE ts_customer_026 prefetchsize 0;
arTABLESPACE ts_customer_027 prefetchsize 0;
arTABLESPACE ts_customer_028 prefetchsize 0;
arTABLESPACE ts_customer_029 prefetchsize 0;
arTABLESPACE ts_customer_030 prefetchsize 0;
arTABLESPACE ts_customer_031 prefetchsize 0;
arTABLESPACE ts_customer_032 prefetchsize 0;
arTABLESPACE ts_customer_033 prefetchsize 0;
arTABLESPACE ts_customer_034 prefetchsize 0;
arTABLESPACE ts_customer_035 prefetchsize 0;
arTABLESPACE ts_customer_036 prefetchsize 0;
arTABLESPACE ts_customer_037 prefetchsize 0;
arTABLESPACE ts_customer_038 prefetchsize 0;
arTABLESPACE ts_customer_039 prefetchsize 0;
arTABLESPACE ts_customer_040 prefetchsize 0;
arTABLESPACE ts_customer_041 prefetchsize 0;
arTABLESPACE ts_customer_042 prefetchsize 0;
arTABLESPACE ts_customer_043 prefetchsize 0;
arTABLESPACE ts_customer_044 prefetchsize 0;
arTABLESPACE ts_customer_045 prefetchsize 0;
arTABLESPACE ts_customer_046 prefetchsize 0;
arTABLESPACE ts_customer_047 prefetchsize 0;
arTABLESPACE ts_customer_048 prefetchsize 0;
arTABLESPACE ts_customer_049 prefetchsize 0;
arTABLESPACE ts_customer_050 prefetchsize 0;
arTABLESPACE ts_customer_051 prefetchsize 0;
arTABLESPACE ts_customer_052 prefetchsize 0;
arTABLESPACE ts_customer_053 prefetchsize 0;
arTABLESPACE ts_customer_054 prefetchsize 0;
arTABLESPACE ts_customer_055 prefetchsize 0;
arTABLESPACE ts_customer_056 prefetchsize 0;
arTABLESPACE ts_customer_057 prefetchsize 0;
arTABLESPACE ts_customer_058 prefetchsize 0;
arTABLESPACE ts_customer_059 prefetchsize 0;
arTABLESPACE ts_customer_060 prefetchsize 0;
arTABLESPACE ts_customer_061 prefetchsize 0;
arTABLESPACE ts_customer_062 prefetchsize 0;
arTABLESPACE ts_customer_063 prefetchsize 0;
arTABLESPACE ts_customer_064 prefetchsize 0;
arTABLESPACE ts_customer_065 prefetchsize 0;
arTABLESPACE ts_customer_066 prefetchsize 0;
arTABLESPACE ts_customer_067 prefetchsize 0;
arTABLESPACE ts_customer_068 prefetchsize 0;
arTABLESPACE ts_customer_069 prefetchsize 0;
arTABLESPACE ts_customer_070 prefetchsize 0;
arTABLESPACE ts_customer_071 prefetchsize 0;
arTABLESPACE ts_customer_072 prefetchsize 0;
arTABLESPACE ts_customer_073 prefetchsize 0;
arTABLESPACE ts_customer_074 prefetchsize 0;
arTABLESPACE ts_customer_075 prefetchsize 0;
arTABLESPACE ts_customer_076 prefetchsize 0;
arTABLESPACE ts_customer_077 prefetchsize 0;
arTABLESPACE ts_customer_078 prefetchsize 0;
arTABLESPACE ts_customer_079 prefetchsize 0;
arTABLESPACE ts_customer_080 prefetchsize 0;
arTABLESPACE ts_customer_081 prefetchsize 0;
arTABLESPACE ts_customer_082 prefetchsize 0;
arTABLESPACE ts_customer_083 prefetchsize 0;
arTABLESPACE ts_customer_084 prefetchsize 0;
arTABLESPACE ts_customer_085 prefetchsize 0;
arTABLESPACE ts_customer_086 prefetchsize 0;
arTABLESPACE ts_customer_087 prefetchsize 0;
arTABLESPACE ts_customer_088 prefetchsize 0;
arTABLESPACE ts_customer_089 prefetchsize 0;
arTABLESPACE ts_customer_090 prefetchsize 0;
arTABLESPACE ts_customer_091 prefetchsize 0;
arTABLESPACE ts_customer_092 prefetchsize 0;
arTABLESPACE ts_customer_093 prefetchsize 0;
arTABLESPACE ts_customer_094 prefetchsize 0;
arTABLESPACE ts_customer_095 prefetchsize 0;
arTABLESPACE ts_customer_096 prefetchsize 0;
arTABLESPACE ts_customer_097 prefetchsize 0;
arTABLESPACE ts_customer_098 prefetchsize 0;
arTABLESPACE ts_customer_099 prefetchsize 0;
arTABLESPACE ts_customer_100 prefetchsize 0;
arTABLESPACE ts_customer_101 prefetchsize 0;
arTABLESPACE ts_customer_102 prefetchsize 0;
arTABLESPACE ts_customer_103 prefetchsize 0;
arTABLESPACE ts_customer_104 prefetchsize 0;
arTABLESPACE ts_customer_105 prefetchsize 0;
arTABLESPACE ts_customer_106 prefetchsize 0;
arTABLESPACE ts_customer_107 prefetchsize 0;
arTABLESPACE ts_customer_108 prefetchsize 0;
arTABLESPACE ts_customer_109 prefetchsize 0;
arTABLESPACE ts_customer_110 prefetchsize 0;
arTABLESPACE ts_customer_111 prefetchsize 0;
arTABLESPACE ts_customer_112 prefetchsize 0;
arTABLESPACE ts_customer_113 prefetchsize 0;
arTABLESPACE ts_customer_114 prefetchsize 0;
arTABLESPACE ts_customer_115 prefetchsize 0;
arTABLESPACE ts_customer_116 prefetchsize 0;
arTABLESPACE ts_customer_117 prefetchsize 0;
arTABLESPACE ts_customer_118 prefetchsize 0;
arTABLESPACE ts_customer_119 prefetchsize 0;
arTABLESPACE ts_customer_120 prefetchsize 0;
arTABLESPACE ts_customer_121 prefetchsize 0;
arTABLESPACE ts_customer_122 prefetchsize 0;
arTABLESPACE ts_customer_123 prefetchsize 0;
arTABLESPACE ts_customer_124 prefetchsize 0;
arTABLESPACE ts_customer_125 prefetchsize 0;
arTABLESPACE ts_customer_126 prefetchsize 0;
arTABLESPACE ts_customer_127 prefetchsize 0;
arTABLESPACE ts_customer_128 prefetchsize 0;
arTABLESPACE ts_customer_129 prefetchsize 0;
arTABLESPACE ts_customer_130 prefetchsize 0;
arTABLESPACE ts_customer_131 prefetchsize 0;
arTABLESPACE ts_customer_132 prefetchsize 0;
arTABLESPACE ts_customer_133 prefetchsize 0;
arTABLESPACE ts_customer_134 prefetchsize 0;
arTABLESPACE ts_customer_135 prefetchsize 0;
arTABLESPACE ts_customer_136 prefetchsize 0;
arTABLESPACE ts_customer_137 prefetchsize 0;
arTABLESPACE ts_customer_138 prefetchsize 0;
arTABLESPACE ts_customer_139 prefetchsize 0;
arTABLESPACE ts_customer_140 prefetchsize 0;
arTABLESPACE ts_customer_141 prefetchsize 0;
arTABLESPACE ts_customer_142 prefetchsize 0;
arTABLESPACE ts_customer_143 prefetchsize 0;
arTABLESPACE ts_customer_144 prefetchsize 0;
arTABLESPACE ts_customer_145 prefetchsize 0;
arTABLESPACE ts_customer_146 prefetchsize 0;
arTABLESPACE ts_customer_147 prefetchsize 0;
arTABLESPACE ts_customer_148 prefetchsize 0;
arTABLESPACE ts_customer_149 prefetchsize 0;
arTABLESPACE ts_customer_150 prefetchsize 0;
arTABLESPACE ts_customer_151 prefetchsize 0;
arTABLESPACE ts_customer_152 prefetchsize 0;
arTABLESPACE ts_customer_153 prefetchsize 0;
arTABLESPACE ts_customer_154 prefetchsize 0;
arTABLESPACE ts_customer_155 prefetchsize 0;
arTABLESPACE ts_customer_156 prefetchsize 0;
arTABLESPACE ts_customer_157 prefetchsize 0;
arTABLESPACE ts_customer_158 prefetchsize 0;
arTABLESPACE ts_customer_159 prefetchsize 0;
arTABLESPACE ts_customer_160 prefetchsize 0;
arTABLESPACE ts_customer_161 prefetchsize 0;
arTABLESPACE ts_customer_162 prefetchsize 0;
ALTER TABLE CUSTOMER1 ADD CONSTRAINT CUSTOMER1CKC CHECK (C_W_ID BETWEEN 1 AND 1066);
SET INTEGRITY FOR CUSTOMER1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;
SET INTEGRITY FOR CUSTOMER2 OFF;
ALTER TABLE CUSTOMER1 DROP CONSTRAINT CUSTOMER1CKC;
ALTER TABLE CUSTOMER1 ADD CONSTRAINT CUSTOMER1CKC CHECK (C_W_ID BETWEEN 1067 AND 2132);
SET INTEGRITY FOR CUSTOMER1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;
SET INTEGRITY FOR CUSTOMER2 OFF;
ALTER TABLE CUSTOMER1 DROP CONSTRAINT CUSTOMER1CKC;
ALTER TABLE CUSTOMER1 ADD CONSTRAINT CUSTOMER1CKC CHECK (C_W_ID BETWEEN 3199 AND 4264);
SET INTEGRITY FOR CUSTOMER1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;
SET INTEGRITY FOR CUSTOMER2 OFF;
ALTER TABLE CUSTOMER1 DROP CONSTRAINT CUSTOMER1CKC;
ALTER TABLE CUSTOMER1 ADD CONSTRAINT CUSTOMER1CKC CHECK (C_W_ID BETWEEN 4265 AND 5330);
SET INTEGRITY FOR CUSTOMER1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;
SET INTEGRITY FOR CUSTOMER2 OFF;
ALTER TABLE CUSTOMER1 DROP CONSTRAINT CUSTOMER1CKC;
ALTER TABLE CUSTOMER1 ADD CONSTRAINT CUSTOMER1CKC CHECK (C_W_ID BETWEEN 5331 AND 6396);
SET INTEGRITY FOR CUSTOMER1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;
SET INTEGRITY FOR CUSTOMER2 OFF;
ALTER TABLE CUSTOMER1 DROP CONSTRAINT CUSTOMER1CKC;
ALTER TABLE CUSTOMER1 ADD CONSTRAINT CUSTOMER1CKC CHECK (C_W_ID BETWEEN 6397 AND 7462);
SET INTEGRITY FOR CUSTOMER1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;
SET INTEGRITY FOR CUSTOMER2 OFF;
ALTER TABLE CUSTOMER1 DROP CONSTRAINT CUSTOMER1CKC;
ALTER TABLE CUSTOMER1 ADD CONSTRAINT CUSTOMER1CKC CHECK (C_W_ID BETWEEN 7463 AND 8528);
SET INTEGRITY FOR CUSTOMER1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;
SET INTEGRITY FOR CUSTOMER2 OFF;
ALTER TABLE CUSTOMER1 DROP CONSTRAINT CUSTOMER1CKC;
ALTER TABLE CUSTOMER1 ADD CONSTRAINT CUSTOMER1CKC CHECK (C_W_ID BETWEEN 8529 AND 9594);
SET INTEGRITY FOR CUSTOMER1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;
SET INTEGRITY FOR CUSTOMER2 OFF;
ALTER TABLE CUSTOMER1 DROP CONSTRAINT CUSTOMER1CKC;
ALTER TABLE CUSTOMER1 ADD CONSTRAINT CUSTOMER1CKC CHECK (C_W_ID BETWEEN 9595 AND 10660);
SET INTEGRITY FOR CUSTOMER1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;
SET INTEGRITY FOR CUSTOMER2 OFF;
ALTER TABLE CUSTOMER1 DROP CONSTRAINT CUSTOMER1CKC;
ALTER TABLE CUSTOMER1 ADD CONSTRAINT CUSTOMER1CKC CHECK (C_W_ID BETWEEN 10661 AND 11726);
SET INTEGRITY FOR CUSTOMER1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;
SET INTEGRITY FOR CUSTOMER2 OFF;
ALTER TABLE CUSTOMER1 DROP CONSTRAINT CUSTOMER1CKC;
ALTER TABLE CUSTOMER1 ADD CONSTRAINT CUSTOMER1CKC CHECK (C_W_ID BETWEEN 11727 AND 12792);
SET INTEGRITY FOR CUSTOMER1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;
SET INTEGRITY FOR CUSTOMER2 OFF;
ALTER TABLE CUSTOMER1 DROP CONSTRAINT CUSTOMER1CKC;
ALTER TABLE CUSTOMER1 ADD CONSTRAINT CUSTOMER1CKC CHECK (C_W_ID BETWEEN 12793 AND 13858);
SET INTEGRITY FOR CUSTOMER1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;
SET INTEGRITY FOR CUSTOMER2 OFF;
ALTER TABLE CUSTOMER1 DROP CONSTRAINT CUSTOMER1CKC;
ALTER TABLE CUSTOMER1 ADD CONSTRAINT CUSTOMER1CKC CHECK (C_W_ID BETWEEN 13859 AND 14924);
SET INTEGRITY FOR CUSTOMER1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;
SET INTEGRITY FOR CUSTOMER2 OFF;
ALTER TABLE CUSTOMER1 DROP CONSTRAINT CUSTOMER1CKC;
ALTER TABLE CUSTOMER1 ADD CONSTRAINT CUSTOMER1CKC CHECK (C_W_ID BETWEEN 14925 AND 15990);
SET INTEGRITY FOR CUSTOMER1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;
SET INTEGRITY FOR CUSTOMER2 OFF;
ALTER TABLE CUSTOMER1 DROP CONSTRAINT CUSTOMER1CKC;
ALTER TABLE CUSTOMER1 ADD CONSTRAINT CUSTOMER1CKC CHECK (C_W_ID BETWEEN 15991 AND 17056);
SET INTEGRITY FOR CUSTOMER1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;
SET INTEGRITY FOR CUSTOMER2 OFF;
ALTER TABLE CUSTOMER1 DROP CONSTRAINT CUSTOMER1CKC;
ALTER TABLE CUSTOMER1 ADD CONSTRAINT CUSTOMER1CKC CHECK (C_W_ID BETWEEN 17057 AND 18122);
SET INTEGRITY FOR CUSTOMER1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;
SET INTEGRITY FOR CUSTOMER2 OFF;
ALTER TABLE CUSTOMER1 DROP CONSTRAINT CUSTOMER1CKC;
ALTER TABLE CUSTOMER1 ADD CONSTRAINT CUSTOMER1CKC CHECK (C_W_ID BETWEEN 18123 AND 19188);
SET INTEGRITY FOR CUSTOMER1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;
SET INTEGRITY FOR CUSTOMER2 OFF;
ALTER TABLE CUSTOMER1 DROP CONSTRAINT CUSTOMER1CKC;
ALTER TABLE CUSTOMER1 ADD CONSTRAINT CUSTOMER1CKC CHECK (C_W_ID BETWEEN 19189 AND 20254);
SET INTEGRITY FOR CUSTOMER1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;
SET INTEGRITY FOR CUSTOMER2 OFF;
ALTER TABLE CUSTOMER1 DROP CONSTRAINT CUSTOMER1CKC;
ALTER TABLE CUSTOMER1 ADD CONSTRAINT CUSTOMER1CKC CHECK (C_W_ID BETWEEN 20255 AND 21320);
SET INTEGRITY FOR CUSTOMER1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;
SET INTEGRITY FOR CUSTOMER2 OFF;
ALTER TABLE CUSTOMER1 DROP CONSTRAINT CUSTOMER1CKC;
ALTER TABLE CUSTOMER1 ADD CONSTRAINT CUSTOMER1CKC CHECK (C_W_ID BETWEEN 21321 AND 22386);
SET INTEGRITY FOR CUSTOMER1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;
SET INTEGRITY FOR CUSTOMER2 OFF;
connect reset; connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER116 OFF;
ALTER TABLE CUSTOMER116 DROP CONSTRAINT CUSTOMER116CKC;
ALTER TABLE CUSTOMER116 ADD CONSTRAINT CUSTOMER116CKC CHECK (C_W_ID BETWEEN 122591 AND 123565);
SET INTEGRITY FOR CUSTOMER116 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;

SET INTEGRITY FOR CUSTOMER117 OFF;
ALTER TABLE CUSTOMER117 DROP CONSTRAINT CUSTOMER117CKC;
ALTER TABLE CUSTOMER117 ADD CONSTRAINT CUSTOMER117CKC CHECK (C_W_ID BETWEEN 123857 AND 12472);
SET INTEGRITY FOR CUSTOMER117 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC 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 125769 AND 126965);
SET INTEGRITY FOR DISTRICT5 ALL IMMEDIATE UNCHECKED;
connect reset;

connect to TPC 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 12793 AND 15990);
SET INTEGRITY FOR DISTRICT6 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;

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 12793 AND 15990);
SET INTEGRITY FOR DISTRICT11 ALL IMMEDIATE UNCHECKED;
connect reset;

connect to TPC 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 125769 AND 126965);
SET INTEGRITY FOR DISTRICT12 ALL IMMEDIATE UNCHECKED;
connect reset;

connect to TPC 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 125769 AND 126965);
SET INTEGRITY FOR DISTRICT13 ALL IMMEDIATE UNCHECKED;
connect reset;

connect to TPC 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 12793 AND 15990);
SET INTEGRITY FOR DISTRICT14 ALL IMMEDIATE UNCHECKED;
connect reset;

connect to TPC 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 12793 AND 15990);
SET INTEGRITY FOR DISTRICT15 ALL IMMEDIATE UNCHECKED;
connect reset;

connect to TPC 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 15991 AND 19188);
SET INTEGRITY FOR DISTRICT16 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;

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 51169 AND 54366);
SET INTEGRITY FOR DISTRICT17 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;

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 54367 AND 57564);
SET INTEGRITY FOR DISTRICT18 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;

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 57565 AND 60762);
SET INTEGRITY FOR DISTRICT19 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;

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 60763 AND 63960);
SET INTEGRITY FOR DISTRICT20 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;

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 63961 AND 67158);
SET INTEGRITY FOR DISTRICT21 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;

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 67159 AND 70356);
SET INTEGRITY FOR DISTRICT22 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;

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 70357 AND 73554);
SET INTEGRITY FOR DISTRICT23 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;

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 73555 AND 76752);
SET INTEGRITY FOR DISTRICT24 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;

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 76753 AND 79950);
SET INTEGRITY FOR DISTRICT25 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;

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 79951 AND 83148);
SET INTEGRITY FOR DISTRICT26 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;

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 83149 AND 86346);
SET INTEGRITY FOR DISTRICT27 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;

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 86347 AND 89544);
SET INTEGRITY FOR DISTRICT28 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;

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 89545 AND 92742);
SET INTEGRITY FOR DISTRICT29 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;

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 92743 AND 95940);
SET INTEGRITY FOR DISTRICT30 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPC in share mode;
connect reset; connect to TPCC in share mode;
ALTER TABLE DISTRICT27 DROP CONSTRAINT DISTRICT27CKC;
ALTER TABLE DISTRICT27 ADD CONSTRAINT DISTRICT27CKC CHECK (D_W_ID BETWEEN 83149
AND 86346);
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 89545
AND 92742);
SET INTEGRITY FOR DISTRICT29 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE DISTRICT29 DROP CONSTRAINT DISTRICT29CKC;
ALTER TABLE DISTRICT29 ADD CONSTRAINT DISTRICT29CKC CHECK (D_W_ID BETWEEN 89545
AND 92742);
SET INTEGRITY FOR DISTRICT29 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE DISTRICT30 DROP CONSTRAINT DISTRICT30CKC;
ALTER TABLE DISTRICT30 ADD CONSTRAINT DISTRICT30CKC CHECK (D_W_ID BETWEEN 92743
AND 95940);
SET INTEGRITY FOR DISTRICT30 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE DISTRICT31 DROP CONSTRAINT DISTRICT31CKC;
ALTER TABLE DISTRICT31 ADD CONSTRAINT DISTRICT31CKC CHECK (D_W_ID BETWEEN 99139
AND 102335);
SET INTEGRITY FOR DISTRICT31 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE DISTRICT32 DROP CONSTRAINT DISTRICT32CKC;
ALTER TABLE DISTRICT32 ADD CONSTRAINT DISTRICT32CKC CHECK (D_W_ID BETWEEN 99139
AND 102335);
SET INTEGRITY FOR DISTRICT32 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE DISTRICT33 DROP CONSTRAINT DISTRICT33CKC;
ALTER TABLE DISTRICT33 ADD CONSTRAINT DISTRICT33CKC CHECK (D_W_ID BETWEEN 105535
AND 108732);
SET INTEGRITY FOR DISTRICT33 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE DISTRICT34 DROP CONSTRAINT DISTRICT34CKC;
ALTER TABLE DISTRICT34 ADD CONSTRAINT DISTRICT34CKC CHECK (D_W_ID BETWEEN 105535
AND 108732);
SET INTEGRITY FOR DISTRICT34 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE DISTRICT35 DROP CONSTRAINT DISTRICT35CKC;
ALTER TABLE DISTRICT35 ADD CONSTRAINT DISTRICT35CKC CHECK (D_W_ID BETWEEN 105535
AND 108732);
SET INTEGRITY FOR DISTRICT35 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE DISTRICT36 DROP CONSTRAINT DISTRICT36CKC;
ALTER TABLE DISTRICT36 ADD CONSTRAINT DISTRICT36CKC CHECK (D_W_ID BETWEEN 111911
AND 115259);
SET INTEGRITY FOR DISTRICT36 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE DISTRICT37 DROP CONSTRAINT DISTRICT37CKC CHECK (D_W_ID BETWEEN 15129
AND 155320);
SET INTEGRITY FOR DISTRICT37 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE DISTRICT38 DROP CONSTRAINT DISTRICT38CKC;
ALTER TABLE DISTRICT38 ADD CONSTRAINT DISTRICT38CKC CHECK (D_W_ID BETWEEN 118327
AND 121524);
SET INTEGRITY FOR DISTRICT38 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE HISTORY7 ADD CONSTRAINT HISTORY7CKC CHECK (H_W_ID BETWEEN 19189
AND 22386);
SET INTEGRITY FOR HISTORY7 OFF;
ALTER TABLE HISTORY8 DROP CONSTRAINT HISTORY8CKC;
ALTER TABLE HISTORY8 ADD CONSTRAINT HISTORY8CKC CHECK (H_W_ID BETWEEN 22387
AND 25584);
SET INTEGRITY FOR HISTORY8 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE HISTORY9 DROP CONSTRAINT HISTORY9CKC;
ALTER TABLE HISTORY9 ADD CONSTRAINT HISTORY9CKC CHECK (H_W_ID BETWEEN 25585
AND 28782);
SET INTEGRITY FOR HISTORY9 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE HISTORY10 DROP CONSTRAINT HISTORY10CKC;
ALTER TABLE HISTORY10 ADD CONSTRAINT HISTORY10CKC CHECK (H_W_ID BETWEEN 28783
AND 31980);
SET INTEGRITY FOR HISTORY10 OFF;
ALTER TABLE HISTORY11 DROP CONSTRAINT HISTORY11CKC;
ALTER TABLE HISTORY11 ADD CONSTRAINT HISTORY11CKC CHECK (H_W_ID BETWEEN 31981
AND 35178);
SET INTEGRITY FOR HISTORY11 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE HISTORY12 DROP CONSTRAINT HISTORY12CKC;
ALTER TABLE HISTORY12 ADD CONSTRAINT HISTORY12CKC CHECK (H_W_ID BETWEEN 35179
AND 38376);
SET INTEGRITY FOR HISTORY12 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE HISTORY13 DROP CONSTRAINT HISTORY13CKC;
ALTER TABLE HISTORY13 ADD CONSTRAINT HISTORY13CKC CHECK (H_W_ID BETWEEN 38377
AND 41574);
SET INTEGRITY FOR HISTORY13 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE HISTORY14 DROP CONSTRAINT HISTORY14CKC;
ALTER TABLE HISTORY14 ADD CONSTRAINT HISTORY14CKC CHECK (H_W_ID BETWEEN 41575
AND 44772);
SET INTEGRITY FOR HISTORY14 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE HISTORY15 DROP CONSTRAINT HISTORY15CKC;
ALTER TABLE HISTORY15 ADD CONSTRAINT HISTORY15CKC CHECK (H_W_ID BETWEEN 44773
AND 47970);
SET INTEGRITY FOR HISTORY15 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE HISTORY16 DROP CONSTRAINT HISTORY16CKC;
ALTER TABLE HISTORY16 ADD CONSTRAINT HISTORY16CKC CHECK (H_W_ID BETWEEN 47971
AND 51168);
SET INTEGRITY FOR HISTORY16 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode.

`DDL/CRCONST_HISTORY.ddl`

connect to TPCC in share mode;
ALTER TABLE HISTORY1 DROP CONTRAINT HISTORY1CKC;
ALTER TABLE HISTORY1 ADD CONSTRAINT HISTORY1CKC CHECK (H_W_ID BETWEEN 1 AND
3198);
SET INTEGRITY FOR HISTORY1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE HISTORY2 DROP CONSTRAINT HISTORY2CKC;
ALTER TABLE HISTORY2 ADD CONSTRAINT HISTORY2CKC CHECK (H_W_ID BETWEEN 3199
AND 6396);
SET INTEGRITY FOR HISTORY2 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE HISTORY3 DROP CONSTRAINT HISTORY3CKC;
ALTER TABLE HISTORY3 ADD CONSTRAINT HISTORY3CKC CHECK (H_W_ID BETWEEN 6397
AND 9644);
SET INTEGRITY FOR HISTORY3 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE HISTORY4 DROP CONSTRAINT HISTORY4CKC;
ALTER TABLE HISTORY4 ADD CONSTRAINT HISTORY4CKC CHECK (H_W_ID BETWEEN 9595
AND 12792);
SET INTEGRITY FOR HISTORY4 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE HISTORY5 DROP CONSTRAINT HISTORY5CKC;
ALTER TABLE HISTORY5 ADD CONSTRAINT HISTORY5CKC CHECK (H_W_ID BETWEEN 12793
AND 15990);
SET INTEGRITY FOR HISTORY5 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE HISTORY6 DROP CONSTRAINT HISTORY6CKC;
ALTER TABLE HISTORY6 ADD CONSTRAINT HISTORY6CKC CHECK (H_W_ID BETWEEN 15991
AND 19188);
SET INTEGRITY FOR HISTORY6 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE HISTORY7 DROP CONSTRAINT HISTORY7CKC;
ALTER TABLE HISTORY7 ADD CONSTRAINT HISTORY7CKC CHECK (H_W_ID BETWEEN 19189
AND 23386);
SET INTEGRITY FOR HISTORY7 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE HISTORY8 DROP CONSTRAINT HISTORY8CKC;
ALTER TABLE HISTORY8 ADD CONSTRAINT HISTORY8CKC CHECK (H_W_ID BETWEEN 22387
AND 25584);
SET INTEGRITY FOR HISTORY8 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE HISTORY9 DROP CONSTRAINT HISTORY9CKC;
ALTER TABLE HISTORY9 ADD CONSTRAINT HISTORY9CKC CHECK (H_W_ID BETWEEN 25585
AND 28782);
SET INTEGRITY FOR HISTORY9 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE HISTORY10 DROP CONSTRAINT HISTORY10CKC;
ALTER TABLE HISTORY10 ADD CONSTRAINT HISTORY10CKC CHECK (H_W_ID BETWEEN 28783
AND 31980);
SET INTEGRITY FOR HISTORY10 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE HISTORY11 DROP CONSTRAINT HISTORY11CKC;
ALTER TABLE HISTORY11 ADD CONSTRAINT HISTORY11CKC CHECK (H_W_ID BETWEEN 31981
AND 35178);
SET INTEGRITY FOR HISTORY11 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE HISTORY12 DROP CONSTRAINT HISTORY12CKC;
ALTER TABLE HISTORY12 ADD CONSTRAINT HISTORY12CKC CHECK (H_W_ID BETWEEN 35179
AND 38376);
SET INTEGRITY FOR HISTORY12 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE HISTORY13 DROP CONSTRAINT HISTORY13CKC;
ALTER TABLE HISTORY13 ADD CONSTRAINT HISTORY13CKC CHECK (H_W_ID BETWEEN 38377
AND 41574);
SET INTEGRITY FOR HISTORY13 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE HISTORY14 DROP CONSTRAINT HISTORY14CKC;
ALTER TABLE HISTORY14 ADD CONSTRAINT HISTORY14CKC CHECK (H_W_ID BETWEEN 41575
AND 44772);
SET INTEGRITY FOR HISTORY14 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE HISTORY15 DROP CONSTRAINT HISTORY15CKC;
ALTER TABLE HISTORY15 ADD CONSTRAINT HISTORY15CKC CHECK (H_W_ID BETWEEN 44773
AND 47970);
SET INTEGRITY FOR HISTORY15 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE HISTORY16 DROP CONSTRAINT HISTORY16CKC;
ALTER TABLE HISTORY16 ADD CONSTRAINT HISTORY16CKC CHECK (H_W_ID BETWEEN 47971
AND 51168);
SET INTEGRITY FOR HISTORY16 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode.
ALTER TABLE HISTORY27 ADD CONSTRAINT HISTORY27CKC CHECK (H_W_ID BETWEEN 83149 AND 86346);
SET INTEGRITY FOR HISTORY27 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 51567 AND 57564);
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 60763 AND 63960);
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 60763 AND 63960);
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 63961 AND 67158);
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 67159 AND 70356);
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 70357 AND 73554);
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 73555 AND 76752);
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 76753 AND 79950);
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 79951 AND 83148);
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 83149 AND 86346);
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 83149 AND 86346);
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 83149 AND 86346);
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 92743 AND 95940);
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 95941 AND 99138);
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 BETWEEN 102336 AND 105534);
SET INTEGRITY FOR HISTORY32 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY33 OFF;
ALTER TABLE HISTORY33 DROP CONSTRAINT HISTORY33CKC;
ALTER TABLE HISTORY33 ADD CONSTRAINT HISTORY33CKC CHECK (H_W_ID BETWEEN 108733 AND 115129);
SET INTEGRITY FOR HISTORY33 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY34 OFF;
ALTER TABLE HISTORY34 DROP CONSTRAINT HISTORY34CKC;
ALTER TABLE HISTORY34 ADD CONSTRAINT HISTORY34CKC CHECK (H_W_ID BETWEEN 105535 AND 108732);
SET INTEGRITY FOR HISTORY34 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY35 OFF;
ALTER TABLE HISTORY35 DROP CONSTRAINT HISTORY35CKC;
ALTER TABLE HISTORY35 ADD CONSTRAINT HISTORY35CKC CHECK (H_W_ID BETWEEN 111528 AND 115129);
SET INTEGRITY FOR HISTORY35 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY36 OFF;
ALTER TABLE HISTORY36 DROP CONSTRAINT HISTORY36CKC;
ALTER TABLE HISTORY36 ADD CONSTRAINT HISTORY36CKC CHECK (H_W_ID BETWEEN 115129 AND 118326);
SET INTEGRITY FOR HISTORY36 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY37 OFF;
ALTER TABLE HISTORY37 DROP CONSTRAINT HISTORY37CKC;
ALTER TABLE HISTORY37 ADD CONSTRAINT HISTORY37CKC CHECK (H_W_ID BETWEEN 118326 AND 121524);
SET INTEGRITY FOR HISTORY37 ALL IMMEDIATE UNCHECKED;
connect reset;
SET INTEGRITY FOR NEW_ORDERA17 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA8 OFF;
ALTER TABLE NEW_ORDERA28 ADD CONSTRAINT NEW_ORDERA28CKC CHECK ((NO_W_ID BETWEEN 86347 AND 89544) AND (NO_O_ID <= 3678));
SET INTEGRITY FOR NEW_ORDERA18 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;
SET INTEGRITY FOR NEW_ORDERA8 ALL IMMEDIATE UNCHECKED;
connect reset;
ALTER TABLE NEW_ORDERA29 ADD CONSTRAINT NEW_ORDERA29CKC CHECK ((NO_W_ID BETWEEN 89545 AND 92742) AND (NO_O_ID <= 3678));
SET INTEGRITY FOR NEW_ORDERA29 ALL IMMEDIATE UNCHECKED;
connect reset;
ALTER TABLE NEW_ORDERA21 ADD CONSTRAINT NEW_ORDERA21CKC CHECK ((NO_W_ID BETWEEN 63951 AND 67158) AND (NO_O_ID <= 3678));
SET INTEGRITY FOR NEW_ORDERA21 ALL IMMEDIATE UNCHECKED;
connect reset;
ALTER TABLE NEW_ORDERA31 DROP CONSTRAINT NEW_ORDERA31CKC;
ALTER TABLE NEW_ORDERA31 ADD CONSTRAINT NEW_ORDERA31CKC CHECK ((NO_W_ID BETWEEN 102337 AND 105534) AND (NO_O_ID <= 3678));
SET INTEGRITY FOR NEW_ORDERA31 ALL IMMEDIATE UNCHECKED;
connect reset;
ALTER TABLE NEW_ORDERA17 ADD CONSTRAINT NEW_ORDERA17CKC CHECK ((NO_W_ID BETWEEN 54367 AND 57565) AND (NO_O_ID BETWEEN 28792 AND 31987) AND (NO_O_ID <= 3678));
SET INTEGRITY FOR NEW_ORDERA17 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA10 OFF;
ALTER TABLE NEW_ORDERA30 DROP CONSTRAINT NEW_ORDERA30CKC;
ALTER TABLE NEW_ORDERA30 ADD CONSTRAINT NEW_ORDERA30CKC CHECK ((NO_W_ID BETWEEN 35179 AND 38375) AND (NO_O_ID <= 3678));
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_ORDERA22 DROP CONSTRAINT NEW_ORDERA22CKC;
ALTER TABLE NEW_ORDERA22 ADD CONSTRAINT NEW_ORDERA22CKC CHECK ((NO_W_ID BETWEEN 67159 AND 70356) AND (NO_O_ID <= 3678));
SET INTEGRITY FOR NEW_ORDERA11 ALL IMMEDIATE UNCHECKED;
connect reset;
ALTER TABLE NEW_ORDERA22 ADD CONSTRAINT NEW_ORDERA22CKC CHECK ((NO_W_ID BETWEEN 31981 AND 35178) AND (NO_O_ID <= 3678));
SET INTEGRITY FOR NEW_ORDERA11 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA13 OFF;
ALTER TABLE NEW_ORDERA34 DROP CONSTRAINT NEW_ORDERA34CKC;
ALTER TABLE NEW_ORDERA34 ADD CONSTRAINT NEW_ORDERA34CKC CHECK ((NO_W_ID BETWEEN 105535 AND 108732) AND (NO_O_ID <= 3678));
SET INTEGRITY FOR NEW_ORDERA13 ALL IMMEDIATE UNCHECKED;
connect reset;
ALTER TABLE NEW_ORDERA13 ADD CONSTRAINT NEW_ORDERA13CKC CHECK ((NO_W_ID BETWEEN 28783 AND 31980) AND (NO_O_ID <= 3678));
SET INTEGRITY FOR NEW_ORDERA13 ALL IMMEDIATE UNCHECKED;
connect reset;
ALTER TABLE NEW_ORDERA24 ADD CONSTRAINT NEW_ORDERA24CKC CHECK ((NO_W_ID BETWEEN 73555 AND 76752) AND (NO_O_ID <= 3678));
SET INTEGRITY FOR NEW_ORDERA24 ALL IMMEDIATE UNCHECKED;
connect reset;
ALTER TABLE NEW_ORDERA24 DROP CONSTRAINT NEW_ORDERA24CKC;
ALTER TABLE NEW_ORDERA24 ADD CONSTRAINT NEW_ORDERA24CKC CHECK ((NO_W_ID BETWEEN 70357 AND 73554) AND (NO_O_ID <= 3678));
SET INTEGRITY FOR NEW_ORDERA24 ALL IMMEDIATE UNCHECKED;
connect reset;
ALTER TABLE NEW_ORDERA24 ADD CONSTRAINT NEW_ORDERA24CKC CHECK ((NO_W_ID BETWEEN 63951 AND 67158) AND (NO_O_ID <= 3678));
SET INTEGRITY FOR NEW_ORDERA24 ALL IMMEDIATE UNCHECKED;
connect reset;
ALTER TABLE NEW_ORDERA32 DROP CONSTRAINT NEW_ORDERA32CKC;
ALTER TABLE NEW_ORDERA32 ADD CONSTRAINT NEW_ORDERA32CKC CHECK ((NO_W_ID BETWEEN 105535 AND 108732) AND (NO_O_ID <= 3678));
SET INTEGRITY FOR NEW_ORDERA32 ALL IMMEDIATE UNCHECKED;
connect reset;
ALTER TABLE NEW_ORDERA32 ADD CONSTRAINT NEW_ORDERA32CKC CHECK ((NO_W_ID BETWEEN 67159 AND 70356) AND (NO_O_ID <= 3678));
SET INTEGRITY FOR NEW_ORDERA32 ALL IMMEDIATE UNCHECKED;
connect reset;
ALTER TABLE NEW_ORDERA34 DROP CONSTRAINT NEW_ORDERA34CKC;
ALTER TABLE NEW_ORDERA34 ADD CONSTRAINT NEW_ORDERA34CKC CHECK ((NO_W_ID BETWEEN 102337 AND 105534) AND (NO_O_ID <= 3678));
SET INTEGRITY FOR NEW_ORDERA34 ALL IMMEDIATE UNCHECKED;
connect reset;
ALTER TABLE NEW_ORDERA34 DROP CONSTRAINT NEW_ORDERA34CKC;
ALTER TABLE NEW_ORDERA34 ADD CONSTRAINT NEW_ORDERA34CKC CHECK ((NO_W_ID BETWEEN 105535 AND 108732) AND (NO_O_ID <= 3678));
SET INTEGRITY FOR NEW_ORDERA34 ALL IMMEDIATE UNCHECKED;
connect reset;
ALTER TABLE NEW_ORDERA35 ADD CONSTRAINT NEW_ORDERA35CKC CHECK ((NO_W_ID BETWEEN 108733 AND 111930) AND (NO_O_ID <= 3678));
SET INTEGRITY FOR NEW_ORDERA35 ALL IMMEDIATE UNCHECKED;
connect reset;
ALTER TABLE NEW_ORDERA35 DROP CONSTRAINT NEW_ORDERA35CKC;
ALTER TABLE NEW_ORDERA35 ADD CONSTRAINT NEW_ORDERA35CKC CHECK ((NO_W_ID BETWEEN 105535 AND 108732) AND (NO_O_ID <= 3678));
SET INTEGRITY FOR NEW_ORDERA35 ALL IMMEDIATE UNCHECKED;
connect reset;
ALTER TABLE NEW_ORDERA35 ADD CONSTRAINT NEW_ORDERA35CKC CHECK ((NO_W_ID BETWEEN 118327 AND 121524) AND (NO_O_ID <= 3678));
SET INTEGRITY FOR NEW_ORDERA35 ALL IMMEDIATE UNCHECKED;
connect reset;
ALTER TABLE NEW_ORDERA36 DROP CONSTRAINT NEW_ORDERA36CKC;
ALTER TABLE NEW_ORDERA36 ADD CONSTRAINT NEW_ORDERA36CKC CHECK ((NO_W_ID BETWEEN 111931 AND 115128) AND (NO_O_ID <= 3678));
SET INTEGRITY FOR NEW_ORDERA36 ALL IMMEDIATE UNCHECKED;
connect reset;
ALTER TABLE NEW_ORDERA36 ADD CONSTRAINT NEW_ORDERA36CKC CHECK ((NO_W_ID BETWEEN 105535 AND 108732) AND (NO_O_ID <= 3678));
SET INTEGRITY FOR NEW_ORDERA36 ALL IMMEDIATE UNCHECKED;
connect reset;
ALTER TABLE NEW_ORDERA37 DROP CONSTRAINT NEW_ORDERA37CKC;
ALTER TABLE NEW_ORDERA37 ADD CONSTRAINT NEW_ORDERA37CKC CHECK ((NO_W_ID BETWEEN 115129 AND 118326) AND (NO_O_ID <= 3678));
SET INTEGRITY FOR NEW_ORDERA37 ALL IMMEDIATE UNCHECKED;
connect reset;
ALTER TABLE NEW_ORDERA37 ADD CONSTRAINT NEW_ORDERA37CKC CHECK ((NO_W_ID BETWEEN 111931 AND 115128) AND (NO_O_ID <= 3678));
SET INTEGRITY FOR NEW_ORDERA37 ALL IMMEDIATE UNCHECKED;
connect reset;
ALTER TABLE NEW_ORDERA38 ADD CONSTRAINT NEW_ORDERA38CKC CHECK ((NO_W_ID BETWEEN 118327 AND 121524) AND (NO_O_ID <= 3678));
SET INTEGRITY FOR NEW_ORDERA38 ALL IMMEDIATE UNCHECKED;
connect reset;
ALTER TABLE NEW_ORDERA38 DROP CONSTRAINT NEW_ORDERA38CKC;
ALTER TABLE NEW_ORDERA38 ADD CONSTRAINT NEW_ORDERA38CKC CHECK ((NO_W_ID BETWEEN 54367 AND 57565) AND (NO_O_ID BETWEEN 28792 AND 31987) AND (NO_O_ID <= 3678));
SET INTEGRITY FOR NEW_ORDERA38 ALL IMMEDIATE UNCHECKED;
connect reset;
ALTER TABLE NEW_ORDERA38 DROP CONSTRAINT NEW_ORDERA38CKC;
ALTER TABLE NEW_ORDERA38 ADD CONSTRAINT NEW_ORDERA38CKC CHECK ((NO_W_ID BETWEEN 54367 AND 57565) AND (NO_O_ID BETWEEN 28792 AND 31987) AND (NO_O_ID <= 3678));
SET INTEGRITY FOR NEW_ORDERA38 ALL IMMEDIATE UNCHECKED;
connect reset;
ALTER TABLE NEW_ORDERA39 ADD CONSTRAINT NEW_ORDERA39CKC CHECK ((NO_W_ID BETWEEN 95454 AND 98651) AND (NO_O_ID <= 3678));
SET INTEGRITY FOR NEW_ORDERA39 ALL IMMEDIATE UNCHECKED;
connect reset;
ALTER TABLE NEW_ORDERA39 DROP CONSTRAINT NEW_ORDERA39CKC;
ALTER TABLE NEW_ORDERA39 ADD CONSTRAINT NEW_ORDERA39CKC CHECK ((NO_W_ID BETWEEN 95454 AND 98651) AND (NO_O_ID <= 3678));
SET INTEGRITY FOR NEW_ORDERA39 ALL IMMEDIATE UNCHECKED;
connect reset;
ALTER TABLE ORDERS9 ADD CONSTRAINT ORDERS9CKC CHECK (O_W_ID BETWEEN 8529 AND 92742) AND (NO_O_ID >= 3679);

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 9595 AND 11728) AND (NO_O_ID >= 3679);
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 12472 AND 14924) AND (NO_O_ID >= 3679);
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 12473 AND 15123) AND (NO_O_ID >= 3679);
SET INTEGRITY FOR ORDERS12 ALL IMMEDIATE UNCHECKED;
connect reset; connect to TPCC in share mode;
ALTER TABLE NEW_ORDERB29 DROP CONSTRAINT NEW_ORDERB29CKC;
ALTER TABLE NEW_ORDERB29 ADD CONSTRAINT NEW_ORDERB29CKC CHECK ((NO_W_ID BETWEEN 92743 AND 102337) AND (NO_O_ID >= 3679));
SET INTEGRITY FOR NEW_ORDERB29 ALL IMMEDIATE UNCHECKED;
connect reset; connect to TPCC in share mode;
ALTER TABLE NEW_ORDERB30 DROP CONSTRAINT NEW_ORDERB30CKC;
ALTER TABLE NEW_ORDERB30 ADD CONSTRAINT NEW_ORDERB30CKC CHECK ((NO_W_ID BETWEEN 102338 AND 111932) AND (NO_O_ID >= 3679));
SET INTEGRITY FOR NEW_ORDERB30 ALL IMMEDIATE UNCHECKED;
connect reset; connect to TPCC in share mode;
ALTER TABLE NEW_ORDERB31 DROP CONSTRAINT NEW_ORDERB31CKC;
ALTER TABLE NEW_ORDERB31 ADD CONSTRAINT NEW_ORDERB31CKC CHECK ((NO_W_ID BETWEEN 111933 AND 124730) AND (NO_O_ID >= 3679));
SET INTEGRITY FOR NEW_ORDERB31 ALL IMMEDIATE UNCHECKED;
connect reset; connect to TPCC in share mode;
ALTER TABLE NEW_ORDERB32 DROP CONSTRAINT NEW_ORDERB32CKC;
ALTER TABLE NEW_ORDERB32 ADD CONSTRAINT NEW_ORDERB32CKC CHECK ((NO_W_ID BETWEEN 124731 AND 13859) AND (NO_O_ID >= 3679));
SET INTEGRITY FOR NEW_ORDERB32 ALL IMMEDIATE UNCHECKED;
connect reset; connect to TPCC in share mode;
ALTER TABLE NEW_ORDERB33 DROP CONSTRAINT NEW_ORDERB33CKC;
ALTER TABLE NEW_ORDERB33 ADD CONSTRAINT NEW_ORDERB33CKC CHECK ((NO_W_ID BETWEEN 13859 AND 18122) AND (NO_O_ID >= 3679));
SET INTEGRITY FOR NEW_ORDERB33 ALL IMMEDIATE UNCHECKED;
connect reset; connect to TPCC in share mode;
ALTER TABLE NEW_ORDERB34 DROP CONSTRAINT NEW_ORDERB34CKC;
ALTER TABLE NEW_ORDERB34 ADD CONSTRAINT NEW_ORDERB34CKC CHECK ((NO_W_ID BETWEEN 18122 AND 2132) AND (NO_O_ID >= 3679));
SET INTEGRITY FOR NEW_ORDERB34 ALL IMMEDIATE UNCHECKED;
connect reset; connect to TPCC in share mode;
ALTER TABLE NEW_ORDERB35 DROP CONSTRAINT NEW_ORDERB35CKC;
ALTER TABLE NEW_ORDERB35 ADD CONSTRAINT NEW_ORDERB35CKC CHECK ((NO_W_ID BETWEEN 2132 AND 5330) AND (NO_O_ID >= 3679));
SET INTEGRITY FOR NEW_ORDERB35 ALL IMMEDIATE UNCHECKED;
connect reset; connect to TPCC in share mode;
ALTER TABLE NEW_ORDERB36 DROP CONSTRAINT NEW_ORDERB36CKC;
ALTER TABLE NEW_ORDERB36 ADD CONSTRAINT NEW_ORDERB36CKC CHECK ((NO_W_ID BETWEEN 5331 AND 102337) AND (NO_O_ID >= 3679));
SET INTEGRITY FOR NEW_ORDERB36 ALL IMMEDIATE UNCHECKED;
connect reset; connect to TPCC in share mode;
ALTER TABLE NEW_ORDERB37 DROP CONSTRAINT NEW_ORDERB37CKC;
ALTER TABLE NEW_ORDERB37 ADD CONSTRAINT NEW_ORDERB37CKC CHECK ((NO_W_ID BETWEEN 102338 AND 111932) AND (NO_O_ID >= 3679));
SET INTEGRITY FOR NEW_ORDERB37 ALL IMMEDIATE UNCHECKED;
connect reset; connect to TPCC in share mode;
ALTER TABLE NEW_ORDERB38 DROP CONSTRAINT NEW_ORDERB38CKC;
ALTER TABLE NEW_ORDERB38 ADD CONSTRAINT NEW_ORDERB38CKC CHECK ((NO_W_ID BETWEEN 111933 AND 124730) AND (NO_O_ID >= 3679));
SET INTEGRITY FOR NEW_ORDERB38 ALL IMMEDIATE UNCHECKED;
connect reset; connect to TPCC in share mode;
ALTER TABLE NEW_ORDERB39 DROP CONSTRAINT NEW_ORDERB39CKC;
ALTER TABLE NEW_ORDERB39 ADD CONSTRAINT NEW_ORDERB39CKC CHECK ((NO_W_ID BETWEEN 124731 AND 15123) AND (NO_O_ID >= 3679));
SET INTEGRITY FOR NEW_ORDERB39 ALL IMMEDIATE UNCHECKED;
connect reset; connect to TPCC in share mode;
ALTER TABLE ORDERS61 ADD CONSTRAINT ORDERS61CKC CHECK (O_W_ID BETWEEN 63961 AND 66092);
SET INTEGRITY FOR ORDERS61 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE ORDERS62 DROP CONSTRAINT ORDERS62CKC;
ALTER TABLE ORDERS62 ADD CONSTRAINT ORDERS62CKC CHECK (O_W_ID BETWEEN 54367 AND 54432);
SET INTEGRITY FOR ORDERS62 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE ORDERS63 DROP CONSTRAINT ORDERS63CKC;
ALTER TABLE ORDERS63 ADD CONSTRAINT ORDERS63CKC CHECK (O_W_ID BETWEEN 55433 AND 56496);
SET INTEGRITY FOR ORDERS63 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE ORDERS64 DROP CONSTRAINT ORDERS64CKC;
ALTER TABLE ORDERS64 ADD CONSTRAINT ORDERS64CKC CHECK (O_W_ID BETWEEN 56499 AND 57564);
SET INTEGRITY FOR ORDERS64 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE ORDERS51 DROP CONSTRAINT ORDERS51CKC;
ALTER TABLE ORDERS51 ADD CONSTRAINT ORDERS51CKC CHECK (O_W_ID BETWEEN 63961 AND 65058);
SET INTEGRITY FOR ORDERS51 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE ORDERS52 DROP CONSTRAINT ORDERS52CKC;
ALTER TABLE ORDERS52 DROP CONSTRAINT ORDERS52CKC CHECK (O_W_ID BETWEEN 65057 AND 66052);
SET INTEGRITY FOR ORDERS52 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE ORDERS53 DROP CONSTRAINT ORDERS53CKC;
ALTER TABLE ORDERS53 ADD CONSTRAINT ORDERS53CKC CHECK (O_W_ID BETWEEN 66053 AND 67158);
SET INTEGRITY FOR ORDERS53 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE ORDERS54 DROP CONSTRAINT ORDERS54CKC;
ALTER TABLE ORDERS54 DROP CONSTRAINT ORDERS54CKC CHECK (O_W_ID BETWEEN 68225 AND 69296);
SET INTEGRITY FOR ORDERS54 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE ORDERS55 DROP CONSTRAINT ORDERS55CKC;
ALTER TABLE ORDERS55 DROP CONSTRAINT ORDERS55CKC CHECK (O_W_ID BETWEEN 57565 AND 58683);
SET INTEGRITY FOR ORDERS55 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE ORDERS56 DROP CONSTRAINT ORDERS56CKC;
ALTER TABLE ORDERS56 DROP CONSTRAINT ORDERS56CKC CHECK (O_W_ID BETWEEN 58931 AND 59896);
SET INTEGRITY FOR ORDERS56 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE ORDERS57 DROP CONSTRAINT ORDERS57CKC;
ALTER TABLE ORDERS57 DROP CONSTRAINT ORDERS57CKC CHECK (O_W_ID BETWEEN 59907 AND 60762);
SET INTEGRITY FOR ORDERS57 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE ORDERS58 DROP CONSTRAINT ORDERS58CKC;
ALTER TABLE ORDERS58 DROP CONSTRAINT ORDERS58CKC CHECK (O_W_ID BETWEEN 60763 AND 63620);
SET INTEGRITY FOR ORDERS58 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE ORDERS59 DROP CONSTRAINT ORDERS59CKC;
ALTER TABLE ORDERS59 DROP CONSTRAINT ORDERS59CKC CHECK (O_W_ID BETWEEN 63629 AND 63824);
SET INTEGRITY FOR ORDERS59 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE ORDERS60 DROP CONSTRAINT ORDERS60CKC;
ALTER TABLE ORDERS60 DROP CONSTRAINT ORDERS60CKC CHECK (O_W_ID BETWEEN 63829 AND 63960);
SET INTEGRITY FOR ORDERS60 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE ORDERS61 DROP CONSTRAINT ORDERS61CKC;
ALTER TABLE ORDERS61 ADD CONSTRAINT ORDERS61CKC CHECK (O_W_ID BETWEEN 63961 AND 66058);
SET INTEGRITY FOR ORDERS61 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE ORDERS62 DROP CONSTRAINT ORDERS62CKC;
ALTER TABLE ORDERS62 DROP CONSTRAINT ORDERS62CKC CHECK (O_W_ID BETWEEN 65027 AND 66052);
SET INTEGRITY FOR ORDERS62 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE ORDERS63 DROP CONSTRAINT ORDERS63CKC;
ALTER TABLE ORDERS63 DROP CONSTRAINT ORDERS63CKC CHECK (O_W_ID BETWEEN 66053 AND 67158);
SET INTEGRITY FOR ORDERS63 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE ORDERS64 DROP CONSTRAINT ORDERS64CKC;
ALTER TABLE ORDERS64 DROP CONSTRAINT ORDERS64CKC CHECK (O_W_ID BETWEEN 67159 AND 68224);
SET INTEGRITY FOR ORDERS64 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE ORDERS65 DROP CONSTRAINT ORDERS65CKC;
ALTER TABLE ORDERS65 DROP CONSTRAINT ORDERS65CKC CHECK (O_W_ID BETWEEN 68225 AND 69296);
SET INTEGRITY FOR ORDERS65 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE ORDERS66 DROP CONSTRAINT ORDERS66CKC;
ALTER TABLE ORDERS66 DROP CONSTRAINT ORDERS66CKC CHECK (O_W_ID BETWEEN 69291 AND 70256);
SET INTEGRITY FOR ORDERS66 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE ORDERS67 DROP CONSTRAINT ORDERS67CKC;
ALTER TABLE ORDERS67 DROP CONSTRAINT ORDERS67CKC CHECK (O_W_ID BETWEEN 70357 AND 71422);
SET INTEGRITY FOR ORDERS67 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE ORDERS68 DROP CONSTRAINT ORDERS68CKC;
ALTER TABLE ORDERS68 DROP CONSTRAINT ORDERS68CKC CHECK (O_W_ID BETWEEN 71423 AND 72488);
SET INTEGRITY FOR ORDERS68 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE ORDERS69 DROP CONSTRAINT ORDERS69CKC;
ALTER TABLE ORDERS69 DROP CONSTRAINT ORDERS69CKC CHECK (O_W_ID BETWEEN 72489 AND 73554);
SET INTEGRITY FOR ORDERS69 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE ORDERS70 DROP CONSTRAINT ORDERS70CKC;
ALTER TABLE ORDERS70 DROP CONSTRAINT ORDERS70CKC CHECK (O_W_ID BETWEEN 73555 AND 74620);
SET INTEGRITY FOR ORDERS70 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE ORDERS71 DROP CONSTRAINT ORDERS71CKC;
ALTER TABLE ORDERS71 DROP CONSTRAINT ORDERS71CKC CHECK (O_W_ID BETWEEN 74621 AND 75686);
SET INTEGRITY FOR ORDERS71 ALL IMMEDIATE UNCHECKED;
ALTER TABLE ORDERS82 DROP CONSTRAINT ORDERS82CKC;
ALTER TABLE ORDERS83 ADD CONSTRAINT ORDERS83CKC CHECK (O_W_ID BETWEEN 86347 AND 87412);
SET INTEGRITY FOR ORDERS82 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS83 OFF;
ALTER TABLE ORDERS83 DROP CONSTRAINT ORDERS83CKC;
ALTER TABLE ORDERS83 ADD CONSTRAINT ORDERS83CKC CHECK (O_W_ID BETWEEN 87413 AND 88478);
SET INTEGRITY FOR ORDERS83 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS84 OFF;
ALTER TABLE ORDERS84 DROP CONSTRAINT ORDERS84CKC;
ALTER TABLE ORDERS84 ADD CONSTRAINT ORDERS84CKC CHECK (O_W_ID BETWEEN 88479 AND 89544);
SET INTEGRITY FOR ORDERS84 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS85 OFF;
ALTER TABLE ORDERS85 DROP CONSTRAINT ORDERS85CKC;
ALTER TABLE ORDERS85 ADD CONSTRAINT ORDERS85CKC CHECK (O_W_ID BETWEEN 90510 AND 91575);
SET INTEGRITY FOR ORDERS85 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS86 OFF;
ALTER TABLE ORDERS86 DROP CONSTRAINT ORDERS86CKC;
ALTER TABLE ORDERS86 ADD CONSTRAINT ORDERS86CKC CHECK (O_W_ID BETWEEN 91676 AND 92742);
SET INTEGRITY FOR ORDERS86 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS87 OFF;
ALTER TABLE ORDERS87 DROP CONSTRAINT ORDERS87CKC;
ALTER TABLE ORDERS87 ADD CONSTRAINT ORDERS87CKC CHECK (O_W_ID BETWEEN 92743 AND 93808);
SET INTEGRITY FOR ORDERS87 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS89 OFF;
ALTER TABLE ORDERS89 DROP CONSTRAINT ORDERS89CKC;
ALTER TABLE ORDERS89 ADD CONSTRAINT ORDERS89CKC CHECK (O_W_ID BETWEEN 93809 AND 94874);
SET INTEGRITY FOR ORDERS89 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS90 OFF;
ALTER TABLE ORDERS90 DROP CONSTRAINT ORDERS90CKC;
ALTER TABLE ORDERS90 ADD CONSTRAINT ORDERS90CKC CHECK (O_W_ID BETWEEN 94875 AND 95940);
SET INTEGRITY FOR ORDERS90 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS91 OFF;
ALTER TABLE ORDERS91 DROP CONSTRAINT ORDERS91CKC;
ALTER TABLE ORDERS91 ADD CONSTRAINT ORDERS91CKC CHECK (O_W_ID BETWEEN 95941 AND 97007);
SET INTEGRITY FOR ORDERS91 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS92 OFF;
ALTER TABLE ORDERS92 DROP CONSTRAINT ORDERS92CKC;
ALTER TABLE ORDERS92 ADD CONSTRAINT ORDERS92CKC CHECK (O_W_ID BETWEEN 97007 AND 98072);
SET INTEGRITY FOR ORDERS92 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS93 OFF;
ALTER TABLE ORDERS93 DROP CONSTRAINT ORDERS93CKC;
ALTER TABLE ORDERS93 ADD CONSTRAINT ORDERS93CKC CHECK (O_W_ID BETWEEN 98073 AND 99138);
SET INTEGRITY FOR ORDERS93 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS94 OFF;
ALTER TABLE ORDERS94 DROP CONSTRAINT ORDERS94CKC;
ALTER TABLE ORDERS94 ADD CONSTRAINT ORDERS94CKC CHECK (O_W_ID BETWEEN 99139 AND 100204);
SET INTEGRITY FOR ORDERS94 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS95 OFF;
ALTER TABLE ORDERS95 DROP CONSTRAINT ORDERS95CKC;
ALTER TABLE ORDERS95 ADD CONSTRAINT ORDERS95CKC CHECK (O_W_ID BETWEEN 100205 AND 101270);
SET INTEGRITY FOR ORDERS95 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS96 OFF;
ALTER TABLE ORDERS96 DROP CONSTRAINT ORDERS96CKC;
ALTER TABLE ORDERS96 ADD CONSTRAINT ORDERS96CKC CHECK (O_W_ID BETWEEN 101271 AND 102336);
SET INTEGRITY FOR ORDERS96 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS97 OFF;
ALTER TABLE ORDERS97 DROP CONSTRAINT ORDERS97CKC;
ALTER TABLE ORDERS97 ADD CONSTRAINT ORDERS97CKC CHECK (O_W_ID BETWEEN 102337 AND 103402);
SET INTEGRITY FOR ORDERS97 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS98 OFF;
ALTER TABLE ORDERS98 DROP CONSTRAINT ORDERS98CKC;
ALTER TABLE ORDERS98 ADD CONSTRAINT ORDERS98CKC CHECK (O_W_ID BETWEEN 103403 AND 104468);
SET INTEGRITY FOR ORDERS98 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS99 OFF;
ALTER TABLE ORDERS99 DROP CONSTRAINT ORDERS99CKC;
ALTER TABLE ORDERS99 ADD CONSTRAINT ORDERS99CKC CHECK (O_W_ID BETWEEN 104469 AND 105534);
SET INTEGRITY FOR ORDERS99 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS100 OFF;
ALTER TABLE ORDERS100 DROP CONSTRAINT ORDERS100CKC;
ALTER TABLE ORDERS100 ADD CONSTRAINT ORDERS100CKC CHECK (O_W_ID BETWEEN 105535 AND 106600);
SET INTEGRITY FOR ORDERS100 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS101 OFF;
ALTER TABLE ORDERS101 DROP CONSTRAINT ORDERS101CKC;
ALTER TABLE ORDERS101 ADD CONSTRAINT ORDERS101CKC CHECK (O_W_ID BETWEEN 106601 AND 107666);
SET INTEGRITY FOR ORDERS101 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS102 OFF;
ALTER TABLE ORDERS102 DROP CONSTRAINT ORDERS102CKC;
ALTER TABLE ORDERS102 ADD CONSTRAINT ORDERS102CKC CHECK (O_W_ID BETWEEN 107667 AND 108732);
SET INTEGRITY FOR ORDERS102 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE ORDER_LINE3 DROP CONSTRAINT ORDER_LINE3CKC;
ALTER TABLE ORDER_LINE3 ADD CONSTRAINT ORDER_LINE3CKC CHECK (O_W_ID BETWEEN 2133 AND 3189);
SET INTEGRITY FOR ORDER_LINE3 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS113 ALL IMMEDIATE UNCHECKED;
connect reset;
ALTER TABLE ORDERS113 ADD CONSTRAINT ORDERS113CKC CHECK (O_W_ID BETWEEN 119393 AND 120458); 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 3198 AND 4264); SET INTEGRITY FOR ORDER_LINE4 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS114 ALL IMMEDIATE UNCHECKED;
connect reset;
ALTER TABLE ORDERS114 DROP CONSTRAINT ORDERS114CKC;
ALTER TABLE ORDER_LINE14 DROP CONSTRAINT ORDER_LINE14CKC;
ALTER TABLE ORDER_LINE14 ADD CONSTRAINT ORDER_LINE14CKC CHECK (OL_W_ID BETWEEN 120459 AND 121524); SET INTEGRITY FOR ORDER_LINE14 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode; 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 4265 AND 5330); SET INTEGRITY FOR ORDER_LINE5 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS117 OFF; ALTER TABLE ORDER_LINE17 DROP CONSTRAINT ORDER_LINE17CKC;
ALTER TABLE ORDER_LINE17 ADD CONSTRAINT ORDER_LINE17CKC CHECK (OL_W_ID BETWEEN 123657 AND 124722); SET INTEGRITY FOR ORDER_LINE17 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode; 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 7463 AND 8528); SET INTEGRITY FOR ORDER_LINE6 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode; 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 8529 AND 9594); SET INTEGRITY FOR ORDER_LINE9 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode; 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 9595 AND 10660); SET INTEGRITY FOR ORDER_LINE10 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode; 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 1069 AND 2132); SET INTEGRITY FOR ORDER_LINE11 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode; 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 1067 AND 2132); SET INTEGRITY FOR ORDER_LINE12 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode; 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 12793 AND 13858); SET INTEGRITY FOR ORDER_LINE13 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode; 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 15991 AND 17056); SET INTEGRITY FOR ORDER_LINE14 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode; 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 18123 AND 19188); SET INTEGRITY FOR ORDER_LINE15 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode; 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 20255 AND 21320); SET INTEGRITY FOR ORDER_LINE16 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode; 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 22387 AND 23452); SET INTEGRITY FOR ORDER_LINE17 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode; 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 23453 AND 24518); SET INTEGRITY FOR ORDER_LINE18 ALL IMMEDIATE UNCHECKED;
connect reset;
ALTER TABLE ORDER_LINE86 ADD CONSTRAINT ORDER_LINE86CKC CHECK (OL_W_ID BETWEEN 90611 AND 91676); SET INTEGRITY FOR ORDER_LINE86 ALL IMMEDIATE UNCHECKED;

connect reset; connect to TPCC in share mode; ALTER TABLE ORDER_LINE87 DROP CONSTRAINT ORDER_LINE87CKC; ALTER TABLE ORDER_LINE87 ADD CONSTRAINT ORDER_LINE87CKC CHECK (OL_W_ID BETWEEN 92743 AND 93808); SET INTEGRITY FOR ORDER_LINE87 ALL IMMEDIATE UNCHECKED;

connect reset; connect to TPCC in share mode; ALTER TABLE ORDER_LINE91 ADD CONSTRAINT ORDER_LINE91CKC CHECK (OL_W_ID BETWEEN 95941 AND 97006); SET INTEGRITY FOR ORDER_LINE91 ALL IMMEDIATE UNCHECKED;

connect reset; connect to TPCC in share mode; ALTER TABLE ORDER_LINE92 ADD CONSTRAINT ORDER_LINE92CKC CHECK (OL_W_ID BETWEEN 98073 AND 99138); SET INTEGRITY FOR ORDER_LINE92 ALL IMMEDIATE UNCHECKED;

connect reset; connect to TPCC in share mode; ALTER TABLE ORDER_LINE93 DROP CONSTRAINT ORDER_LINE93CKC; ALTER TABLE ORDER_LINE93 ADD CONSTRAINT ORDER_LINE93CKC CHECK (OL_W_ID BETWEEN 100205 AND 101270); SET INTEGRITY FOR ORDER_LINE93 ALL IMMEDIATE UNCHECKED;

connect reset; connect to TPCC in share mode; ALTER TABLE ORDER_LINE94 DROP CONSTRAINT ORDER_LINE94CKC; ALTER TABLE ORDER_LINE94 ADD CONSTRAINT ORDER_LINE94CKC CHECK (OL_W_ID BETWEEN 101271 AND 102336); SET INTEGRITY FOR ORDER_LINE94 ALL IMMEDIATE UNCHECKED;

connect reset; connect to TPCC in share mode; ALTER TABLE ORDER_LINE95 DROP CONSTRAINT ORDER_LINE95CKC; ALTER TABLE ORDER_LINE95 ADD CONSTRAINT ORDER_LINE95CKC CHECK (OL_W_ID BETWEEN 102337 AND 103402); SET INTEGRITY FOR ORDER_LINE95 ALL IMMEDIATE UNCHECKED;

connect reset; connect to TPCC in share mode; ALTER TABLE ORDER_LINE96 DROP CONSTRAINT ORDER_LINE96CKC; ALTER TABLE ORDER_LINE96 ADD CONSTRAINT ORDER_LINE96CKC CHECK (OL_W_ID BETWEEN 103403 AND 104468); SET INTEGRITY FOR ORDER_LINE96 ALL IMMEDIATE UNCHECKED;

connect reset; connect to TPCC in share mode; ALTER TABLE ORDER_LINE97 DROP CONSTRAINT ORDER_LINE97CKC; ALTER TABLE ORDER_LINE97 ADD CONSTRAINT ORDER_LINE97CKC CHECK (OL_W_ID BETWEEN 104469 AND 105534); SET INTEGRITY FOR ORDER_LINE97 ALL IMMEDIATE UNCHECKED;

connect reset; connect to TPCC in share mode; ALTER TABLE ORDER_LINE98 DROP CONSTRAINT ORDER_LINE98CKC; ALTER TABLE ORDER_LINE98 ADD CONSTRAINT ORDER_LINE98CKC CHECK (OL_W_ID BETWEEN 105535 AND 106600); SET INTEGRITY FOR ORDER_LINE98 ALL IMMEDIATE UNCHECKED;

connect reset; connect to TPCC in share mode; ALTER TABLE ORDER_LINE99 DROP CONSTRAINT ORDER_LINE99CKC; ALTER TABLE ORDER_LINE99 ADD CONSTRAINT ORDER_LINE99CKC CHECK (OL_W_ID BETWEEN 106601 AND 107666); SET INTEGRITY FOR ORDER_LINE99 ALL IMMEDIATE UNCHECKED;

connect reset; connect to TPCC in share mode; ALTER TABLE ORDER_LINE100 DROP CONSTRAINT ORDER_LINE100CKC; ALTER TABLE ORDER_LINE100 ADD CONSTRAINT ORDER_LINE100CKC CHECK (OL_W_ID BETWEEN 107667 AND 108732); SET INTEGRITY FOR ORDER_LINE100 ALL IMMEDIATE UNCHECKED;

connect reset; connect to TPCC in share mode; ALTER TABLE ORDER_LINE101 DROP CONSTRAINT ORDER_LINE101CKC; ALTER TABLE ORDER_LINE101 ADD CONSTRAINT ORDER_LINE101CKC CHECK (OL_W_ID BETWEEN 108733 AND 109798); SET INTEGRITY FOR ORDER_LINE101 ALL IMMEDIATE UNCHECKED;

connect reset; connect to TPCC in share mode; ALTER TABLE ORDER_LINE102 DROP CONSTRAINT ORDER_LINE102CKC; ALTER TABLE ORDER_LINE102 ADD CONSTRAINT ORDER_LINE102CKC CHECK (OL_W_ID BETWEEN 109799 AND 110864); SET INTEGRITY FOR ORDER_LINE102 ALL IMMEDIATE UNCHECKED;

connect reset; connect to TPCC in share mode; ALTER TABLE ORDER_LINE103 DROP CONSTRAINT ORDER_LINE103CKC; ALTER TABLE ORDER_LINE103 ADD CONSTRAINT ORDER_LINE103CKC CHECK (OL_W_ID BETWEEN 110865 AND 111930); SET INTEGRITY FOR ORDER_LINE103 ALL IMMEDIATE UNCHECKED;

connect reset; connect to TPCC in share mode; ALTER TABLE ORDER_LINE104 DROP CONSTRAINT ORDER_LINE104CKC; ALTER TABLE ORDER_LINE104 ADD CONSTRAINT ORDER_LINE104CKC CHECK (OL_W_ID BETWEEN 111931 AND 112996); SET INTEGRITY FOR ORDER_LINE104 ALL IMMEDIATE UNCHECKED;

connect reset; connect to TPCC in share mode; ALTER TABLE ORDER_LINE105 DROP CONSTRAINT ORDER_LINE105CKC; ALTER TABLE ORDER_LINE105 ADD CONSTRAINT ORDER_LINE105CKC CHECK (OL_W_ID BETWEEN 112997 AND 114062); SET INTEGRITY FOR ORDER_LINE105 ALL IMMEDIATE UNCHECKED;

connect reset; connect to TPCC in share mode; ALTER TABLE ORDER_LINE106 DROP CONSTRAINT ORDER_LINE106CKC; ALTER TABLE ORDER_LINE106 ADD CONSTRAINT ORDER_LINE106CKC CHECK (OL_W_ID BETWEEN 114063 AND 115128); SET INTEGRITY FOR ORDER_LINE106 ALL IMMEDIATE UNCHECKED;

connect reset; connect to TPCC in share mode; ALTER TABLE ORDER_LINE107 DROP CONSTRAINT ORDER_LINE107CKC; ALTER TABLE ORDER_LINE107 ADD CONSTRAINT ORDER_LINE107CKC CHECK (OL_W_ID BETWEEN 115129 AND 116194); SET INTEGRITY FOR ORDER_LINE107 ALL IMMEDIATE UNCHECKED;

connect reset; connect to TPCC in share mode; ALTER TABLE ORDER_LINE108 DROP CONSTRAINT ORDER_LINE108CKC; ALTER TABLE ORDER_LINE108 ADD CONSTRAINT ORDER_LINE108CKC CHECK (OL_W_ID BETWEEN 116195 AND 117260); SET INTEGRITY FOR ORDER_LINE108 ALL IMMEDIATE UNCHECKED;

connect reset; connect to TPCC in share mode; ALTER TABLE ORDER_LINE109 DROP CONSTRAINT ORDER_LINE109CKC; ALTER TABLE ORDER_LINE109 ADD CONSTRAINT ORDER_LINE109CKC CHECK (OL_W_ID BETWEEN 117261 AND 118326); SET INTEGRITY FOR ORDER_LINE109 ALL IMMEDIATE UNCHECKED;

connect reset; connect to TPCC in share mode; ALTER TABLE ORDER_LINE110 DROP CONSTRAINT ORDER_LINE110CKC; ALTER TABLE ORDER_LINE110 ADD CONSTRAINT ORDER_LINE110CKC CHECK (OL_W_ID BETWEEN 118327 AND 119392); SET INTEGRITY FOR ORDER_LINE110 ALL IMMEDIATE UNCHECKED;

connect reset; connect to TPCC in share mode; ALTER TABLE ORDER_LINE111 DROP CONSTRAINT ORDER_LINE111CKC; ALTER TABLE ORDER_LINE111 ADD CONSTRAINT ORDER_LINE111CKC CHECK (OL_W_ID BETWEEN 119393 AND 120458); SET INTEGRITY FOR ORDER_LINE111 ALL IMMEDIATE UNCHECKED;

connect reset; connect to TPCC in share mode; ALTER TABLE ORDER_LINE112 DROP CONSTRAINT ORDER_LINE112CKC; ALTER TABLE ORDER_LINE112 ADD CONSTRAINT ORDER_LINE112CKC CHECK (OL_W_ID BETWEEN 120459 AND 121524); SET INTEGRITY FOR ORDER_LINE112 ALL IMMEDIATE UNCHECKED;

connect reset; connect to TPCC in share mode; ALTER TABLE ORDER_LINE113 DROP CONSTRAINT ORDER_LINE113CKC; ALTER TABLE ORDER_LINE113 ADD CONSTRAINT ORDER_LINE113CKC CHECK (OL_W_ID BETWEEN 121525 AND 122590); SET INTEGRITY FOR ORDER_LINE113 ALL IMMEDIATE UNCHECKED;

connect reset; connect to TPCC in share mode; ALTER TABLE ORDER_LINE114 DROP CONSTRAINT ORDER_LINE114CKC; ALTER TABLE ORDER_LINE114 ADD CONSTRAINT ORDER_LINE114CKC CHECK (OL_W_ID BETWEEN 122591 AND 123656); SET INTEGRITY FOR ORDER_LINE114 ALL IMMEDIATE UNCHECKED;

connect reset; connect to TPCC in share mode; ALTER TABLE ORDER_LINE115 DROP CONSTRAINT ORDER_LINE115CKC; ALTER TABLE ORDER_LINE115 ADD CONSTRAINT ORDER_LINE115CKC CHECK (OL_W_ID BETWEEN 123657 AND 124722); SET INTEGRITY FOR ORDER_LINE115 ALL IMMEDIATE UNCHECKED;
null
ALTER TABLE STOCK28 DROP CONSTRAINT STOCK28CKC;
ALTER TABLE STOCK28 ADD CONSTRAINT STOCK28CKC CHECK (S_W_ID BETWEEN 28783 AND 29948);
SET INTEGRITY FOR STOCK28 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK29 DROP CONSTRAINT STOCK29CKC;
ALTER TABLE STOCK29 ADD CONSTRAINT STOCK29CKC CHECK (S_W_ID BETWEEN 29849 AND 30564);
SET INTEGRITY FOR STOCK29 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK30 ADD CONSTRAINT STOCK30CKC CHECK (S_W_ID BETWEEN 30015 AND 31880);
SET INTEGRITY FOR STOCK30 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK31 DROP CONSTRAINT STOCK31CKC;
ALTER TABLE STOCK31 ADD CONSTRAINT STOCK31CKC CHECK (S_W_ID BETWEEN 31881 AND 33546);
SET INTEGRITY FOR STOCK31 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK32 DROP CONSTRAINT STOCK32CKC;
ALTER TABLE STOCK32 ADD CONSTRAINT STOCK32CKC CHECK (S_W_ID BETWEEN 33047 AND 34312);
SET INTEGRITY FOR STOCK32 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK33 DROP CONSTRAINT STOCK33CKC;
ALTER TABLE STOCK33 ADD CONSTRAINT STOCK33CKC CHECK (S_W_ID BETWEEN 34113 AND 35178);
SET INTEGRITY FOR STOCK33 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK34 DROP CONSTRAINT STOCK34CKC;
ALTER TABLE STOCK34 ADD CONSTRAINT STOCK34CKC CHECK (S_W_ID BETWEEN 35179 AND 36244);
SET INTEGRITY FOR STOCK34 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK35 DROP CONSTRAINT STOCK35CKC;
ALTER TABLE STOCK35 ADD CONSTRAINT STOCK35CKC CHECK (S_W_ID BETWEEN 36245 AND 37320);
SET INTEGRITY FOR STOCK35 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK36 DROP CONSTRAINT STOCK36CKC;
ALTER TABLE STOCK36 ADD CONSTRAINT STOCK36CKC CHECK (S_W_ID BETWEEN 37311 AND 38376);
SET INTEGRITY FOR STOCK36 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK37 DROP CONSTRAINT STOCK37CKC;
ALTER TABLE STOCK37 ADD CONSTRAINT STOCK37CKC CHECK (S_W_ID BETWEEN 38377 AND 39442);
SET INTEGRITY FOR STOCK37 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK38 DROP CONSTRAINT STOCK38CKC;
ALTER TABLE STOCK38 ADD CONSTRAINT STOCK38CKC CHECK (S_W_ID BETWEEN 39443 AND 40508);
SET INTEGRITY FOR STOCK38 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK39 DROP CONSTRAINT STOCK39CKC;
ALTER TABLE STOCK39 ADD CONSTRAINT STOCK39CKC CHECK (S_W_ID BETWEEN 40509 AND 41574);
SET INTEGRITY FOR STOCK39 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK40 DROP CONSTRAINT STOCK40CKC;
ALTER TABLE STOCK40 ADD CONSTRAINT STOCK40CKC CHECK (S_W_ID BETWEEN 41575 AND 43254);
SET INTEGRITY FOR STOCK40 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK41 DROP CONSTRAINT STOCK41CKC;
ALTER TABLE STOCK41 ADD CONSTRAINT STOCK41CKC CHECK (S_W_ID BETWEEN 43255 AND 43796);
SET INTEGRITY FOR STOCK41 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK42 DROP CONSTRAINT STOCK42CKC;
ALTER TABLE STOCK42 ADD CONSTRAINT STOCK42CKC CHECK (S_W_ID BETWEEN 43707 AND 44772);
SET INTEGRITY FOR STOCK42 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK43 DROP CONSTRAINT STOCK43CKC;
ALTER TABLE STOCK43 ADD CONSTRAINT STOCK43CKC CHECK (S_W_ID BETWEEN 44773 AND 45830);
SET INTEGRITY FOR STOCK43 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK44 DROP CONSTRAINT STOCK44CKC;
ALTER TABLE STOCK44 ADD CONSTRAINT STOCK44CKC CHECK (S_W_ID BETWEEN 45839 AND 46964);
SET INTEGRITY FOR STOCK44 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK45 DROP CONSTRAINT STOCK45CKC;
ALTER TABLE STOCK45 ADD CONSTRAINT STOCK45CKC CHECK (S_W_ID BETWEEN 46905 AND 47970);
SET INTEGRITY FOR STOCK45 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK46 DROP CONSTRAINT STOCK46CKC;
ALTER TABLE STOCK46 ADD CONSTRAINT STOCK46CKC CHECK (S_W_ID BETWEEN 47971 AND 49037);
SET INTEGRITY FOR STOCK46 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK47 DROP CONSTRAINT STOCK47CKC;
ALTER TABLE STOCK47 ADD CONSTRAINT STOCK47CKC CHECK (S_W_ID BETWEEN 49038 AND 50103);
SET INTEGRITY FOR STOCK47 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK48 DROP CONSTRAINT STOCK48CKC;
ALTER TABLE STOCK48 ADD CONSTRAINT STOCK48CKC CHECK (S_W_ID BETWEEN 50104 AND 51168);
SET INTEGRITY FOR STOCK48 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK49 DROP CONSTRAINT STOCK49CKC;
ALTER TABLE STOCK49 ADD CONSTRAINT STOCK49CKC CHECK (S_W_ID BETWEEN 51169 AND 52230);
SET INTEGRITY FOR STOCK49 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK50 DROP CONSTRAINT STOCK50CKC;
ALTER TABLE STOCK50 ADD CONSTRAINT STOCK50CKC CHECK (S_W_ID BETWEEN 52231 AND 53300);
SET INTEGRITY FOR STOCK50 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK51 DROP CONSTRAINT STOCK51CKC;
ALTER TABLE STOCK51 ADD CONSTRAINT STOCK51CKC CHECK (S_W_ID BETWEEN 53301 AND 54360);
SET INTEGRITY FOR STOCK51 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK52 DROP CONSTRAINT STOCK52CKC;
ALTER TABLE STOCK52 ADD CONSTRAINT STOCK52CKC CHECK (S_W_ID BETWEEN 54361 AND 55432);
SET INTEGRITY FOR STOCK52 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK53 DROP CONSTRAINT STOCK53CKC;
ALTER TABLE STOCK53 ADD CONSTRAINT STOCK53CKC CHECK (S_W_ID BETWEEN 55433 AND 56498);
SET INTEGRITY FOR STOCK53 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK54 DROP CONSTRAINT STOCK54CKC;
ALTER TABLE STOCK54 ADD CONSTRAINT STOCK54CKC CHECK (S_W_ID BETWEEN 56499 AND 57564);
SET INTEGRITY FOR STOCK54 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK55 DROP CONSTRAINT STOCK55CKC;
ALTER TABLE STOCK55 ADD CONSTRAINT STOCK55CKC CHECK (S_W_ID BETWEEN 57565 AND 58630);
SET INTEGRITY FOR STOCK55 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK56 DROP CONSTRAINT STOCK56CKC;
ALTER TABLE STOCK56 ADD CONSTRAINT STOCK56CKC CHECK (S_W_ID BETWEEN 58631 AND 59696);
SET INTEGRITY FOR STOCK56 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK57 DROP CONSTRAINT STOCK57CKC;
ALTER TABLE STOCK57 ADD CONSTRAINT STOCK57CKC CHECK (S_W_ID BETWEEN 59697 AND 60762);
SET INTEGRITY FOR STOCK57 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK58 DROP CONSTRAINT STOCK58CKC;
ALTER TABLE STOCK58 ADD CONSTRAINT STOCK58CKC CHECK (S_W_ID BETWEEN 60763 AND 61828);
SET INTEGRITY FOR STOCK58 ALL IMMEDIATE UNCHECKED;
ALTER TABLE STOCK59 ADD CONSTRAINT STOCK59CKC CHECK (S_W_ID BETWEEN 61829 AND 62864);
SET INTEGRITY FOR STOCK59 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 62895 AND 63960);
SET INTEGRITY FOR STOCK59 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK59 OFF;
ALTER TABLE STOCK60 DROP CONSTRAINT STOCK60CKC;
ALTER TABLE STOCK60 ADD CONSTRAINT STOCK60CKC CHECK (S_W_ID BETWEEN 63961 AND 65026);
SET INTEGRITY FOR STOCK60 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 66093 AND 67358);
SET INTEGRITY FOR STOCK60 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK60 OFF;
ALTER TABLE STOCK61 DROP CONSTRAINT STOCK61CKC;
ALTER TABLE STOCK61 ADD CONSTRAINT STOCK61CKC CHECK (S_W_ID BETWEEN 68225 AND 69290);
SET INTEGRITY FOR STOCK61 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK61 OFF;
ALTER TABLE STOCK61 ADD CONSTRAINT STOCK61CKC CHECK (S_W_ID BETWEEN 69291 AND 70356);
SET INTEGRITY FOR STOCK61 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK61 OFF;
ALTER TABLE STOCK62 DROP CONSTRAINT STOCK62CKC;
ALTER TABLE STOCK62 ADD CONSTRAINT STOCK62CKC CHECK (S_W_ID BETWEEN 70357 AND 71422);
SET INTEGRITY FOR STOCK62 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 72489 AND 73554);
SET INTEGRITY FOR STOCK62 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK62 OFF;
ALTER TABLE STOCK63 DROP CONSTRAINT STOCK63CKC;
ALTER TABLE STOCK63 ADD CONSTRAINT STOCK63CKC CHECK (S_W_ID BETWEEN 73555 AND 74620);
SET INTEGRITY FOR STOCK63 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 74621 AND 75686);
SET INTEGRITY FOR STOCK63 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 75687 AND 76752);
SET INTEGRITY FOR STOCK63 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 76753 AND 77818);
SET INTEGRITY FOR STOCK63 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 77819 AND 78885);
SET INTEGRITY FOR STOCK63 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 78886 AND 79950);
SET INTEGRITY FOR STOCK63 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 79951 AND 80026);
SET INTEGRITY FOR STOCK63 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK63 OFF;
ALTER TABLE STOCK64 DROP CONSTRAINT STOCK64CKC;
ALTER TABLE STOCK64 ADD CONSTRAINT STOCK64CKC CHECK (S_W_ID BETWEEN 80027 AND 81017);
SET INTEGRITY FOR STOCK64 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 BETWEEN 81018 AND 82082);
SET INTEGRITY FOR STOCK64 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 BETWEEN 82083 AND 83148);
SET INTEGRITY FOR STOCK64 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK64 OFF;
ALTER TABLE STOCK65 DROP CONSTRAINT STOCK65CKC;
ALTER TABLE STOCK65 ADD CONSTRAINT STOCK65CKC CHECK (S_W_ID BETWEEN 83149 AND 84214);
SET INTEGRITY FOR STOCK65 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK65 OFF;
ALTER TABLE STOCK65 DROP CONSTRAINT STOCK65CKC;
ALTER TABLE STOCK65 ADD CONSTRAINT STOCK65CKC CHECK (S_W_ID BETWEEN 84215 AND 85280);
SET INTEGRITY FOR STOCK65 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK65 OFF;
ALTER TABLE STOCK65 DROP CONSTRAINT STOCK65CKC;
ALTER TABLE STOCK65 ADD CONSTRAINT STOCK65CKC CHECK (S_W_ID BETWEEN 85281 AND 86346);
SET INTEGRITY FOR STOCK65 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK65 OFF;
ALTER TABLE STOCK65 DROP CONSTRAINT STOCK65CKC;
ALTER TABLE STOCK65 ADD CONSTRAINT STOCK65CKC CHECK (S_W_ID BETWEEN 86347 AND 87412);
SET INTEGRITY FOR STOCK65 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK65 OFF;
ALTER TABLE STOCK65 DROP CONSTRAINT STOCK65CKC;
ALTER TABLE STOCK65 ADD CONSTRAINT STOCK65CKC CHECK (S_W_ID BETWEEN 87413 AND 88478);
SET INTEGRITY FOR STOCK65 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK65 OFF;
ALTER TABLE STOCK65 DROP CONSTRAINT STOCK65CKC;
ALTER TABLE STOCK65 ADD CONSTRAINT STOCK65CKC CHECK (S_W_ID BETWEEN 88479 AND 89544);
SET INTEGRITY FOR STOCK65 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK65 OFF;
ALTER TABLE STOCK65 DROP CONSTRAINT STOCK65CKC;
ALTER TABLE STOCK65 ADD CONSTRAINT STOCK65CKC CHECK (S_W_ID BETWEEN 89545 AND 90610);
SET INTEGRITY FOR STOCK65 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK65 OFF;
ALTER TABLE STOCK65 DROP CONSTRAINT STOCK65CKC;
ALTER TABLE STOCK65 ADD CONSTRAINT STOCK65CKC CHECK (S_W_ID BETWEEN 90611 AND 91676);
SET INTEGRITY FOR STOCK65 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK65 OFF;
ALTER TABLE STOCK65 DROP CONSTRAINT STOCK65CKC;
ALTER TABLE STOCK65 ADD CONSTRAINT STOCK65CKC CHECK (S_W_ID BETWEEN 91677 AND 92742);
SET INTEGRITY FOR STOCK65 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK65 OFF;
ALTER TABLE STOCK65 DROP CONSTRAINT STOCK65CKC;
ALTER TABLE STOCK65 ADD CONSTRAINT STOCK65CKC CHECK (S_W_ID BETWEEN 92743 AND 93808);
SET INTEGRITY FOR STOCK65 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK65 OFF;
ALTER TABLE STOCK65 DROP CONSTRAINT STOCK65CKC;
ALTER TABLE STOCK65 ADD CONSTRAINT STOCK65CKC CHECK (S_W_ID BETWEEN 93809 AND 94874);
SET INTEGRITY FOR STOCK65 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK65 OFF;
ALTER TABLE STOCK65 DROP CONSTRAINT STOCK65CKC;
ALTER TABLE STOCK65 ADD CONSTRAINT STOCK65CKC CHECK (S_W_ID BETWEEN 94875 AND 95940);
SET INTEGRITY FOR STOCK65 ALL IMMEDIATE UNCHECKED;
SET INTEGRITY FOR STOCK101 ALL IMMEDIATE UNCHECKED; connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK102 OFF;
ALTER TABLE STOCK102 DROP CONSTRAINT STOCK102CKC;
ALTER TABLE STOCK102 ADD CONSTRAINT STOCK102CKC CHECK (S_W_ID BETWEEN 107667 AND 108732);
SET INTEGRITY FOR STOCK102 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK103 OFF;
ALTER TABLE STOCK103 DROP CONSTRAINT STOCK103CKC;
ALTER TABLE STOCK103 ADD CONSTRAINT STOCK103CKC CHECK (S_W_ID BETWEEN 108733 AND 109798);
SET INTEGRITY FOR STOCK103 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK104 DROP CONSTRAINT STOCK104CKC;
ALTER TABLE STOCK104 ADD CONSTRAINT STOCK104CKC CHECK (S_W_ID BETWEEN 109799 AND 110864);
SET INTEGRITY FOR STOCK104 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK105 OFF;
ALTER TABLE STOCK105 DROP CONSTRAINT STOCK105CKC;
ALTER TABLE STOCK105 ADD CONSTRAINT STOCK105CKC CHECK (S_W_ID BETWEEN 110865 AND 111930);
SET INTEGRITY FOR STOCK105 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK106 DROP CONSTRAINT STOCK106CKC;
ALTER TABLE STOCK106 ADD CONSTRAINT STOCK106CKC CHECK (S_W_ID BETWEEN 111931 AND 112996);
SET INTEGRITY FOR STOCK106 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK107 DROP Constraint STOCK107CKC;
ALTER TABLE STOCK107 ADD Constraint STOCK107CKC CHECK (S_W_ID BETWEEN 112997 AND 114062);
SET INTEGRITY FOR STOCK107 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK108 DROP CONSTRAINT STOCK108CKC;
ALTER TABLE STOCK108 ADD CONSTRAINT STOCK108CKC CHECK (S_W_ID BETWEEN 114063 AND 115128);
SET INTEGRITY FOR STOCK108 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK109 DROP CONSTRAINT STOCK109CKC;
ALTER TABLE STOCK109 ADD CONSTRAINT STOCK109CKC CHECK (S_W_ID BETWEEN 115129 AND 116194);
SET INTEGRITY FOR STOCK109 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK110 DROP CONSTRAINT STOCK110CKC;
ALTER TABLE STOCK110 ADD CONSTRAINT STOCK110CKC CHECK (S_W_ID BETWEEN 116195 AND 117260);
SET INTEGRITY FOR STOCK110 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK111 DROP CONSTRAINT STOCK111CKC;
ALTER TABLE STOCK111 ADD CONSTRAINT STOCK111CKC CHECK (S_W_ID BETWEEN 117261 AND 118326);
SET INTEGRITY FOR STOCK111 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK112 DROP CONSTRAINT STOCK112CKC;
ALTER TABLE STOCK112 ADD CONSTRAINT STOCK112CKC CHECK (S_W_ID BETWEEN 118327 AND 119393);
SET INTEGRITY FOR STOCK112 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK113 DROP CONSTRAINT STOCK113CKC;
ALTER TABLE STOCK113 ADD CONSTRAINT STOCK113CKC CHECK (S_W_ID BETWEEN 119394 AND 120459);
SET INTEGRITY FOR STOCK113 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK114 DROP CONSTRAINT STOCK114CKC;
ALTER TABLE STOCK114 ADD CONSTRAINT STOCK114CKC CHECK (S_W_ID BETWEEN 120460 AND 121525);
SET INTEGRITY FOR STOCK114 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK115 DROP CONSTRAINT STOCK115CKC;
ALTER TABLE STOCK115 ADD CONSTRAINT STOCK115CKC CHECK (S_W_ID BETWEEN 121526 AND 122591);
SET INTEGRITY FOR STOCK115 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK116 DROP CONSTRAINT STOCK116CKC;
ALTER TABLE STOCK116 ADD CONSTRAINT STOCK116CKC CHECK (S_W_ID BETWEEN 122592 AND 123657);
SET INTEGRITY FOR STOCK116 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK117 DROP CONSTRAINT STOCK117CKC;
ALTER TABLE STOCK117 ADD CONSTRAINT STOCK117CKC CHECK (S_W_ID BETWEEN 123658 AND 124722);
SET INTEGRITY FOR STOCK117 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK118 DROP CONSTRAINT STOCK118CKC;
ALTER TABLE STOCK118 ADD CONSTRAINT STOCK118CKC CHECK (S_W_ID BETWEEN 124723 AND 125788);
SET INTEGRITY FOR STOCK118 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK119 DROP CONSTRAINT STOCK119CKC;
ALTER TABLE STOCK119 ADD CONSTRAINT STOCK119CKC CHECK (S_W_ID BETWEEN 125789 AND 126864);
SET INTEGRITY FOR STOCK119 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
ALTER TABLE STOCK120 DROP CONSTRAINT STOCK120CKC;
ALTER TABLE STOCK120 ADD CONSTRAINT STOCK120CKC CHECK (S_W_ID >= 126865);
SET INTEGRITY FOR STOCK120 ALL IMMEDIATE UNCHECKED;
CREATE INDEX CUST_IDXB90
ON CUSTOMER90(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_IDXB91;
CREATE INDEX CUST_IDXB91
ON CUSTOMER91(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_IDXB92;
CREATE INDEX CUST_IDXB92
ON CUSTOMER92(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_IDXB93;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB94;
CREATE INDEX CUST_IDXB94
ON CUSTOMER94(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_IDXB95;
CREATE INDEX CUST_IDXB95
ON CUSTOMER95(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_IDXB96;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB97;
CREATE INDEX CUST_IDXB97
ON CUSTOMER97(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_IDXB98;
CREATE INDEX CUST_IDXB98
ON CUSTOMER98(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_IDXB99;
CREATE INDEX CUST_IDXB99
ON CUSTOMER99(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_IDXB100;
CREATE INDEX CUST_IDXB100
ON CUSTOMER100(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_IDXB101;
CREATE INDEX CUST_IDXB101
ON CUSTOMER101(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_IDXB102;
CREATE INDEX CUST_IDXB102
ON CUSTOMER102(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_IDXB103;
CREATE INDEX CUST_IDXB103
ON CUSTOMER103(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_IDXB104;
CREATE INDEX CUST_IDXB104
ON CUSTOMER104(C_LAST, C_W_ID, C_D_ID, C_FIRST, C_ID) PCTFREE 0;
connect reset;
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
connect reset; connect to TPCC in share mode;
DROP INDEX ORDR_IDXB28;
CREATE INDEX ORDR_IDXB28
connect reset;
connect to TPCC in share mode;
RUN TPCC
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB16;
CREATE INDEX ORDR_IDXB16
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB29;
CREATE INDEX ORDR_IDXB29
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB18;
CREATE INDEX ORDR_IDXB18
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB30;
CREATE INDEX ORDR_IDXB30
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB31;
CREATE INDEX ORDR_IDXB31
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB37;
CREATE INDEX ORDR_IDXB37
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB2;
CREATE INDEX ORDR_IDXB2
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB34;
CREATE INDEX ORDR_IDXB34
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB35;
CREATE INDEX ORDR_IDXB35
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB39;
CREATE INDEX ORDR_IDXB39
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB4;
CREATE INDEX ORDR_IDXB4
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB50;
CREATE INDEX ORDR_IDXB50
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB6;
CREATE INDEX ORDR_IDXB6
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB9;
CREATE INDEX ORDR_IDXB9
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB10;
CREATE INDEX ORDR_IDXB10
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB11;
CREATE INDEX ORDR_IDXB11
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB12;
CREATE INDEX ORDR_IDXB12
connect reset;
connect to TPCC in share mode;
<table>
<thead>
<tr>
<th>C_ID</th>
<th>INTEGER       NOT NULL</th>
</tr>
</thead>
<tbody>
<tr>
<td>C_STATE</td>
<td>CHAR(2)       NOT NULL</td>
</tr>
<tr>
<td>C_ZIP</td>
<td>CHAR(9)       NOT NULL</td>
</tr>
<tr>
<td>C_PHONE</td>
<td>CHAR(16)      NOT NULL</td>
</tr>
<tr>
<td>C_CREDIT</td>
<td>CHAR(2)       NOT NULL</td>
</tr>
<tr>
<td>C_DISCOUNT</td>
<td>REAL          NOT NULL</td>
</tr>
<tr>
<td>C_SINCE</td>
<td>TIMESTAMP     NOT NULL</td>
</tr>
<tr>
<td>C_CREDIT_LIM</td>
<td>DECIMAL(12,2) NOT NULL</td>
</tr>
<tr>
<td>C_DATA</td>
<td>VARCHAR(500)  NOT NULL</td>
</tr>
<tr>
<td>C_LAST</td>
<td>VARCHAR(16)   NOT NULL</td>
</tr>
<tr>
<td>C_FIRST</td>
<td>VARCHAR(16)   NOT NULL</td>
</tr>
<tr>
<td>C_STREET_1</td>
<td>VARCHAR(20)   NOT NULL</td>
</tr>
<tr>
<td>C_STREET_2</td>
<td>VARCHAR(20)   NOT NULL</td>
</tr>
<tr>
<td>C_W_ID</td>
<td>INTEGER       NOT NULL</td>
</tr>
<tr>
<td>C_DELIVERY_CNT</td>
<td>INTEGER NOT NULL</td>
</tr>
<tr>
<td>C_BALANCE</td>
<td>DECIMAL(12,2) NOT NULL</td>
</tr>
<tr>
<td>C_YTD_PAYMENT</td>
<td>DECIMAL(12,2) NOT NULL</td>
</tr>
<tr>
<td>C_PAYMENT_CNT</td>
<td>INTEGER NOT NULL</td>
</tr>
</tbody>
</table>

INDEX IN ts_customer_008 ORGANIZE BY KEY SEQUENCE (C_ID STARTING FROM 1 ENDING AT 3000, C_W_ID STARTING FROM 8529 ENDING AT 9594, C_D_ID STARTING FROM 1 ENDING AT 10) ALLOW OVERFLOW;

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_CREDIT CHAR(2) NOT NULL, C_DISCOUNT REAL NOT NULL, C_SINCE TIMESTAMP NOT NULL, C_CREDIT_LIM DECIMAL(12,2) NOT NULL, C_DATA VARCHAR(500) NOT NULL, C_LAST VARCHAR(16) NOT NULL, C_FIRST VARCHAR(16) NOT NULL, C_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 )

INDEX IN ts_customer_013 ORGANIZE BY KEY SEQUENCE (C_ID STARTING FROM 1 ENDING AT 3000, C_W_ID STARTING FROM 11729 ENDING AT 12793, C_D_ID STARTING FROM 1 ENDING AT 10) ALLOW OVERFLOW;

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_CREDIT CHAR(2) NOT NULL, C_DISCOUNT REAL NOT NULL, C_SINCE TIMESTAMP NOT NULL, C_CREDIT_LIM DECIMAL(12,2) NOT NULL, C_DATA VARCHAR(500) NOT NULL, C_LAST VARCHAR(16) NOT NULL, C_FIRST VARCHAR(16) NOT NULL, C_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 )

INDEX IN ts_customer_014 ORGANIZE BY KEY SEQUENCE (C_ID STARTING FROM 1 ENDING AT 3000, C_W_ID STARTING FROM 12793 ENDING AT 13858, C_D_ID STARTING FROM 1 ENDING AT 10) ALLOW OVERFLOW;

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_CREDIT CHAR(2) NOT NULL, C_DISCOUNT REAL NOT NULL, C_SINCE TIMESTAMP NOT NULL, C_CREDIT_LIM DECIMAL(12,2) NOT NULL, C_DATA VARCHAR(500) NOT NULL, C_LAST VARCHAR(16) NOT NULL, C_FIRST VARCHAR(16) NOT NULL, C_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 )

INDEX IN ts_customer_015 ORGANIZE BY KEY SEQUENCE (C_ID STARTING FROM 1 ENDING AT 3000, C_W_ID STARTING FROM 12793 ENDING AT 13858, C_D_ID STARTING FROM 1 ENDING AT 10) ALLOW OVERFLOW;
connect to TPCC in share mode;
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_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,  
  C_STREET_1      VARCHAR(20)   NOT NULL,  
  C_STREET_2      VARCHAR(20)   NOT NULL,  
  C_CITY          VARCHAR(20)   NOT NULL,  
  C_DELIVERY_CNT  INTEGER       NOT NULL,  
  C_ID STARTING FROM 1 ENDING AT 3000,  
  C_W_ID STARTING FROM 13859 ENDING AT 14924,  
  C_D_ID STARTING FROM 1 ENDING AT 10  
)
ALLOW OVERFLOW;

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_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,  
  C_STREET_1      VARCHAR(20)   NOT NULL,  
  C_STREET_2      VARCHAR(20)   NOT NULL,  
  C_CITY          VARCHAR(20)   NOT NULL,  
  C_DELIVERY_CNT  INTEGER       NOT NULL,  
  C_ID STARTING FROM 1 ENDING AT 3000,  
  C_W_ID STARTING FROM 14925 ENDING AT 15990,  
  C_D_ID STARTING FROM 1 ENDING AT 10  
)
ALLOW OVERFLOW;

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_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,  
  C_STREET_1      VARCHAR(20)   NOT NULL,  
  C_STREET_2      VARCHAR(20)   NOT NULL,  
  C_CITY          VARCHAR(20)   NOT NULL,  
  C_DELIVERY_CNT  INTEGER       NOT NULL,  
  C_ID STARTING FROM 1 ENDING AT 3000,  
  C_W_ID STARTING FROM 19189 ENDING AT 20254,  
  C_D_ID STARTING FROM 1 ENDING AT 10  
)
ALLOW OVERFLOW;
DROP TABLE CUSTOMER23;
CREATE TABLE CUSTOMER23
(
C_ID INTEGER NOT NULL,
C_PHONE CHAR(16) NOT NULL,
C_STATE CHAR(2) NOT NULL,
C_ZIP CHAR(9) NOT NULL,
C_SINCE TIMESTAMP NOT NULL,
C_CREDIT CHAR(2) NOT NULL,
C_CREDIT_LIM DECIMAL(12,2) NOT NULL,
C_MIDDLE CHAR(2) NOT NULL,
C_DISCOUNT REAL NOT NULL,
C_DATA VARCHAR(500) NOT NULL,
C_DELIVERY_CNT INTEGER NOT NULL,
C_CONFLICT INTEGER NOT NULL
);
INT 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 24519 ENDING AT 25584,
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_PHONE CHAR(16) NOT NULL,
C_STATE CHAR(2) NOT NULL,
C_ZIP CHAR(9) NOT NULL,
C_SINCE TIMESTAMP NOT NULL,
C_CREDIT CHAR(2) NOT NULL,
C_DISCOUNT REAL NOT NULL,
C_DATA VARCHAR(500) 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
);
INT 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 21321 ENDING AT 22386,
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_PHONE CHAR(16) NOT NULL,
C_STATE CHAR(2) NOT NULL,
C_ZIP CHAR(9) NOT NULL,
C_SINCE TIMESTAMP NOT NULL,
C_CREDIT CHAR(2) NOT NULL,
C_DISCOUNT REAL NOT NULL,
C_DATA VARCHAR(500) 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
);
INT 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 26651 ENDING AT 27716,
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_PHONE CHAR(16) NOT NULL,
C_STATE CHAR(2) NOT NULL,
C_ZIP CHAR(9) NOT NULL,
C_SINCE TIMESTAMP NOT NULL,
C_CREDIT CHAR(2) NOT NULL,
C_DISCOUNT REAL NOT NULL,
C_DATA VARCHAR(500) 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
);
INT 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 27716 ENDING AT 28781,
C_D_ID STARTING FROM 1 ENDING AT 10
) ALLOW OVERFLOW;
connect reset;
C_DELIVERY_CNT INTEGER NOT NULL,
C_BALANCE DECIMAL(12,2) NOT NULL,
C_CITY VARCHAR(20) NOT NULL,
C_D_ID SMALLINT 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,
)
INDEX IN ts_customer_035
ORGANIZE BY KEY SEQUENCE (C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 36245 ENDING AT 37310,
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_EXPIRE CHAR(2) NOT NULL,
C_DISCOUNT REAL NOT NULL,
C_DATA VARCHAR(500) NOT NULL,
C_CITY VARCHAR(20) 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,
)
INDEX IN ts_customer_036
ORGANIZE BY KEY SEQUENCE (C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 37311 ENDING AT 38376,
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_EXPIRE CHAR(2) NOT NULL,
C_DISCOUNT REAL NOT NULL,
C_DATA VARCHAR(500) NOT NULL,
C_CITY VARCHAR(20) NOT NULL,
C_DELIVERY_CNT INTEGER NOT NULL,
C_BALANCE DECIMAL(12,2) NOT NULL,
C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
C_DEAL_BUYER CHAR(2) NOT NULL,
C_SINCE_BUYER CHAR(2) NOT NULL,
C_ZIP_BUYER CHAR(9) NOT NULL,
C_DATA_BUYER VARCHAR(500) NOT NULL,
C_CITY_BUYER VARCHAR(20) 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,
)
INDEX IN ts_customer_037
ORGANIZE BY KEY SEQUENCE (C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 38377 ENDING AT 39442,
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_EXPIRE CHAR(2) NOT NULL,
C_DISCOUNT REAL NOT NULL,
C_DATA VARCHAR(500) NOT NULL,
C_CITY VARCHAR(20) 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,
)
INDEX IN ts_customer_038
ORGANIZE BY KEY SEQUENCE (C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 39443 ENDING AT 40508,
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,
CREATE TABLE CUSTOMER44
  (  C_ID            INTEGER       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,
     INDEX IN is_customer_041
     ORGANIZE BY KEY SEQUENCE
       (  C_ID STARTING FROM 1 ENDING AT 3000,
           C_W_ID STARTING FROM 42641 ENDING AT 43706,
           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_MIDDLE        CHAR(2)       NOT NULL,
     C_CREDIT        CHAR(2)       NOT NULL,
     C_DISCOUNT      REAL          NOT NULL,
     C_DATA          VARCHAR(500)  NOT NULL,
     C_FIRST         VARCHAR(16)   NOT NULL,
     C_STREET_1      VARCHAR(20)   NOT NULL,
     C_STREET_2      VARCHAR(20)   NOT NULL,
     C_BALANCE       DECIMAL(12,2) NOT NULL,
     C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
     C_PAYMENT_CNT   INTEGER       NOT NULL
     INDEX IN is_customer_042
     ORGANIZE BY KEY SEQUENCE
       (  C_ID STARTING FROM 1 ENDING AT 3000,
           C_W_ID STARTING FROM 43707 ENDING AT 44772,
           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
     INDEX IN is_customer_043
     ORGANIZE BY KEY SEQUENCE
       (  C_ID STARTING FROM 1 ENDING AT 3000,
           C_W_ID STARTING FROM 41575 ENDING AT 42640,
           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_DELIVERY_CNT  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_D_ID          SMALLINT      NOT NULL,
     C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
     INDEX IN is_customer_044
     ORGANIZE BY KEY SEQUENCE
       (  C_ID STARTING FROM 1 ENDING AT 3000,
           C_W_ID STARTING FROM 45839 ENDING AT 46904,
           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_DATA          VARCHAR(500)  NOT NULL,
     C_LAST          VARCHAR(16)   NOT NULL,
     C_D_ID          SMALLINT      NOT NULL,
     C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,
     INDEX IN is_customer_045
     ORGANIZE BY KEY SEQUENCE
       (  C_ID STARTING FROM 1 ENDING AT 3000,
           C_W_ID STARTING FROM 44773 ENDING AT 45838,
           C_D_ID STARTING FROM 1 ENDING AT 10
       )
  )
ALLOW OVERFLOW;
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_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_BALANCE DECIMAL(12,2) NOT NULL,
  C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
  C_PAYMENT_CNT INTEGER NOT NULL
) IN ts_customer_048
ORGANIZE BY KEY SEQUENCE (C_ID STARTING FROM 1 ENDING AT 3000, C_W_ID STARTING FROM 50103 ENDING AT 51168, C_D_ID SMALLINT NOT NULL, C_CREDIT_LIM DECIMAL(12,2) NOT NULL, C_DATA VARCHAR(500) NOT NULL, C_BALANCE DECIMAL(12,2) NOT NULL, C_YTD_PAYMENT DECIMAL(12,2) NOT NULL, C_PAYMENT_CNT INTEGER NOT NULL)
INDEX IN is_customer_048
INDEX IN s_customer_048
ORGANIZE BY KEY SEQUENCE (C_ID STARTING FROM 1 ENDING AT 3000, C_W_ID STARTING FROM 50103 ENDING AT 51168, C_D_ID SMALLINT NOT NULL, C_CREDIT_LIM DECIMAL(12,2) NOT NULL, C_DATA VARCHAR(500) NOT NULL, C_BALANCE DECIMAL(12,2) NOT NULL, C_YTD_PAYMENT DECIMAL(12,2) NOT NULL, C_PAYMENT_CNT INTEGER NOT NULL)
INDEX IN is_customer_049
INDEX IN s_customer_049
ORGANIZE BY KEY SEQUENCE (C_ID STARTING FROM 1 ENDING AT 3000, C_W_ID SMALLINT NOT NULL, C_D_ID SMALLINT NOT NULL, C_BONUS FLOAT NOT NULL, C_REWARD FLOAT NOT NULL, C_CALL_center VARCHAR(16) NOT NULL, C_STREET_1 VARCHAR(20) NOT NULL, C_DELIVERY_CNT INTEGER NOT NULL)
INDEX IN is_customer_050
INDEX IN s_customer_050
ORGANIZE BY KEY SEQUENCE (C_ID STARTING FROM 1 ENDING AT 3000, C_W_ID SMALLINT NOT NULL, C_D_ID SMALLINT NOT NULL, C_BONUS FLOAT NOT NULL, C_REWARD FLOAT NOT NULL, C_CALL_center VARCHAR(16) NOT NULL, C_STREET_1 VARCHAR(20) NOT NULL, C_DELIVERY_CNT INTEGER NOT NULL)
INDEX IN is_customer_051
INDEX IN s_customer_051
ORGANIZE BY KEY SEQUENCE (C_ID STARTING FROM 1 ENDING AT 3000, C_W_ID SMALLINT NOT NULL, C_D_ID SMALLINT NOT NULL, C_BONUS FLOAT NOT NULL, C_REWARD FLOAT NOT NULL, C_CALL_center VARCHAR(16) NOT NULL, C_STREET_1 VARCHAR(20) NOT NULL, C_DELIVERY_CNT INTEGER NOT NULL)
ORGANIZE BY KEY SEQUENCE:
C_ID STARTING FROM 1 ENDING AT 10
C_W_ID STARTING FROM 69291 ENDING AT 70356,
C_D_ID STARTING FROM 1 ENDING AT 10

DROP TABLE CUSTOMER69;

ORGANIZE BY KEY SEQUENCE
C_W_ID STARTING FROM 69291 ENDING AT 70356,
C_D_ID STARTING FROM 1 ENDING AT 10
ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode
DROP TABLE CUSTOMER67;
CREATE TABLE CUSTOMER67
(C_ID INTEGER NOT NULL, C_STATE CHAR(2) NOT NULL, C_ZIP CHAR(9) NOT NULL,
C_PHONE CHAR(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_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
ORGANIZE BY KEY SEQUENCE
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 70357 ENDING AT 71422,
C_D_ID STARTING FROM 1 ENDING AT 10
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode
DROP TABLE CUSTOMER66;
CREATE TABLE CUSTOMER66
(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_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_DELIVERY_CNT INTEGER NOT NULL,
C_STREET_2 VARCHAR(20) NOT NULL, C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
C_DATA VARCHAR(500) NOT NULL, C_SINCE TIMESTAMP NOT NULL,
C_CREDIT_LIM DECIMAL(12,2) NOT NULL, C_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
C_PAYMENT_CNT INTEGER NOT NULL)

INDEX IN is_customer_065
INDEX IN is_customer_069
ORGANIZE BY KEY SEQUENCE
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 72489 ENDING AT 73554,
C_D_ID STARTING FROM 1 ENDING AT 10
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode
DROP TABLE CUSTOMER70;
CREATE TABLE CUSTOMER70
(C_ID INTEGER NOT NULL, C_STATE CHAR(2) NOT NULL, C_ZIP CHAR(9) NOT NULL,
C_PHONE CHAR(16) NOT NULL, C_CREDIT_LIM DECIMAL(12,2) NOT NULL,
C_MIDDLE CHAR(2) NOT NULL, C_DISCOUNT REAL NOT NULL,
C_DATA VARCHAR(500) NOT NULL, C_SINCE TIMESTAMP NOT NULL,
C_DISCOUNT REAL NOT NULL, C_DATA VARCHAR(500) NOT NULL,
C_LAST VARCHAR(16) NOT NULL, C пенсионный математический аналитик
C_FIRST VARCHAR(16) NOT NULL, C_STREET_1 VARCHAR(20) NOT NULL,
C_DELIVERY_CNT INTEGER NOT NULL, C_LAST VARCHAR(16) NOT NULL,
C_ID SMALLINT NOT NULL, W_ID INTEGER NOT NULL,
C_DELIVERY_CNT INTEGER NOT NULL, C_BANCE DECIMAL(12,2) NOT NULL,
C_YTD_PAYMENT DECIMAL(12,2) NOT NULL, C_PAYMENT_CNT INTEGER NOT NULL)

INDEX IN is_customer_069
INDEX IN is_customer_070
ORGANIZE BY KEY SEQUENCE
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 72489 ENDING AT 73554,
C_D_ID STARTING FROM 1 ENDING AT 10
ALLOW OVERFLOW;
CREATE TABLE CUSTOMER73
(
  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_LIMIT DECIMAL(12,2) NOT NULL,
  C_MIDDLE CHAR(2) NOT NULL,
  C_CREDIT_UTIL DECIMAL(12,2) 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_SINCE TIMESTAMP NOT NULL,
  C_CREDIT_LIMIT DECIMAL(12,2) 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_SINCE TIMESTAMP NOT NULL,
  C_CREDIT_LIMIT DECIMAL(12,2) 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_SINCE TIMESTAMP NOT NULL,
  C_CREDIT_LIMIT DECIMAL(12,2) 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_SINCE TIMESTAMP NOT NULL,
  C_CREDIT_LIMIT DECIMAL(12,2) 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_SINCE TIMESTAMP NOT NULL,
  C_CREDIT_LIMIT DECIMAL(12,2) 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_SINCE TIMESTAMP NOT NULL,
  C_CREDIT_LIMIT DECIMAL(12,2) 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_SINCE TIMESTAMP NOT NULL,
  C_CREDIT_LIMIT DECIMAL(12,2) 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_SINCE TIMESTAMP NOT NULL,
  C_CREDIT_LIMIT DECIMAL(12,2) 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_SINCE TIMESTAMP NOT NULL,
  C_CREDIT_LIMIT DECIMAL(12,2) 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_SINCE TIMESTAMP NOT NULL,
  C_CREDIT_LIMIT DECIMAL(12,2) 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_SINCE TIMESTAMP NOT NULL,
  C_CREDIT_LIMIT DECIMAL(12,2) 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_SINCE TIMESTAMP NOT NULL,
  C_CREDIT_LIMIT DECIMAL(12,2) 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_SINCE TIMESTAMP NOT NULL,
  C_CREDIT_LIMIT DECIMAL(12,2) 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_SINCE TIMESTAMP NOT NULL,
  C_CREDIT_LIMIT DECIMAL(12,2) 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_SINCE TIMESTAMP NOT NULL,
  C_CREDIT_LIMIT DECIMAL(12,2) 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_SINCE TIMESTAMP NOT NULL,
  C_CREDIT_LIMIT DECIMAL(12,2) 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_SINCE TIMESTAMP NOT NULL,
  C_CREDIT_LIMIT DECIMAL(12,2) 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_SINCE TIMESTAMP NOT NULL,
  C_CREDIT_LIMIT DECIMAL(12,2) 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_SINCE TIMESTAMP NOT NULL,
  C_CREDIT_LIMIT DECIMAL(12,2) 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_SINCE TIMESTAMP NOT NULL,
  C_CREDIT_LIMIT DECIMAL(12,2) 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_SINCE TIMESTAMP NOT NULL,
  C_CREDIT_LIMIT DECIMAL(12,2) 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_SINCE TIMESTAMP NOT NULL,
  C_CREDIT_LIMIT DECIMAL(12,2) 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_SINCE TIMESTAMP 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_087
INDEX IN is_customer_087
ORGANIZE BY KEY SEQUENCE
( C_W_ID STARTING FROM 91677 ENDING AT 92742,
C_D_ID STARTING FROM 1 ENDING AT 10
) ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER85;
CREATE TABLE CUSTOMER85
( 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 CHAR(2) NOT NULL,
C_MIDDLE CHAR(2) 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_YTD_PAYMENT DECIMAL(12,2) NOT NULL,
C_PAYMENT_CNT INTEGER NOT NULL
)
IN ts_customer_088
INDEX IN is_customer_088
ORGANIZE BY KEY SEQUENCE
( C_W_ID STARTING FROM 1 ENDING AT 3000,
C_D_ID STARTING FROM 1 ENDING AT 10
) ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER88;
CREATE TABLE CUSTOMER88
( 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 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_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_088
INDEX IN is_customer_088
ORGANIZE BY KEY SEQUENCE
( C_W_ID STARTING FROM 91677 ENDING AT 92742,
C_D_ID STARTING FROM 1 ENDING AT 10
) ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DRO
connect to TPCC in share mode; DROP TABLE CUSTOMER99;
CREATE TABLE CUSTOMER99
(                     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        CHAR(2)       NOT NULL,                     C_DISCOUNT      REAL          NOT NULL,                     C_DATA          VARCHAR(500)  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_099
connect to TPCC in share mode; DROP TABLE CUSTOMER97; CREATE TABLE CUSTOMER97
ALLOW OVERFLOW; connect reset; connect to TPCC in share mode; DROP TABLE CUSTOMER101;
CREATE TABLE CUSTOMER101                    (                     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        CHAR(2)       NOT NULL,                     C_DISCOUNT      REAL          NOT NULL,                     C_W_ID          INTEGER       NOT NULL,                     C_CREDIT_LIM    DECIMAL(12,2) NOT NULL,                     C_DELIVERY_CNTR INTEGER NOT NULL,                     C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,                     C_PAYMENT_CNTR  INTEGER NOT NULL                    )                    ALLOW OVERFLOW; connect reset; connect to TPCC in share mode; DROP TABLE CUSTOMER102;
CREATE TABLE CUSTOMER102                    (                     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        CHAR(2)       NOT NULL,                     C_DISCOUNT      REAL          NOT NULL,                     C_W_ID          INTEGER       NOT NULL,                     C_CREDIT_LIM    DECIMAL(12,2) NOT NULL,                     C_DELIVERY_CNTR INTEGER NOT NULL,                     C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,                     C_PAYMENT_CNTR  INTEGER NOT NULL                    )                    ALLOW OVERFLOW; connect reset; connect to TPCC in share mode; DROP TABLE CUSTOMER100;
CREATE TABLE CUSTOMER100                    (                     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        CHAR(2)       NOT NULL,                     C_DISCOUNT      REAL          NOT NULL,                     C_W_ID          INTEGER       NOT NULL,                     C_CREDIT_LIM    DECIMAL(12,2) NOT NULL,                     C_DELIVERY_CNTR INTEGER NOT NULL,                     C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,                     C_PAYMENT_CNTR  INTEGER NOT NULL                    )                    ALLOW OVERFLOW; connect reset; connect to TPCC in share mode; DROP TABLE CUSTOMER99; CREATE TABLE CUSTOMER99
(                     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        CHAR(2)       NOT NULL,                     C_DISCOUNT      REAL          NOT NULL,                     C_DATA          VARCHAR(500)  NOT NULL,                     C_BALANCE       DECIMAL(12,2) NOT NULL,                     C_YTD_PAYMENT   DECIMAL(12,2) NOT NULL,                     C_PAYMENT_CNTR  INTEGER NOT NULL                    )                    ALLOW OVERFLOW;
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 19189 ENDING AT 22386
 ALLOW OVERFLOW;

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 19189 ENDING AT 22386
 ALLOW OVERFLOW;

DROP TABLE DISTRICT4; CREATE TABLE DISTRICT4                    (                     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                    )                    INDEX IN ts_dist_004
 ORGANIZE BY KEY SEQUENCE (                     D_ID STARTING FROM 1 ENDING AT 10,
 D_W_ID STARTING FROM 9595 ENDING AT 12792
 ALLOW OVERFLOW;

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_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                    )                    INDEX IN ts_dist_005
 ORGANIZE BY KEY SEQUENCE (                     D_ID STARTING FROM 1 ENDING AT 10,
 ALLOW OVERFLOW;

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                    )                    INDEX IN ts_dist_009
 ORGANIZE BY KEY SEQUENCE (                     D_ID STARTING FROM 1 ENDING AT 10,
 ALLOW OVERFLOW;
```
CREATE TABLE DISTRICT16
D_NEXT_O_ID INTEGER 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 47971 ENDING AT 51168
)
ALLOW OVERFLOW;

connect to TPCC in share mode;
DROP TABLE DISTRICT17;
CREATE TABLE DISTRICT17
D_NEXT_O_ID INTEGER 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
D_W_ID STARTING FROM 51169 ENDING AT 54366
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT13;
CREATE TABLE DISTRICT13
D_NEXT_O_ID INTEGER 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 28763 ENDING AT 31980)
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_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_W_ID STARTING FROM 41575 ENDING AT 41574)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT10;
CONNECT TO TPCC IN SHARED 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 51169 ENDING AT 51168)
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 57564 ENDING AT 57564)
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 35178 ENDING AT 35178)
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 35178 ENDING AT 35178)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT18;
CREATE TABLE DISTRICT18
D_NEXT_O_ID INTEGER NOT NULL,
D_STREET_1 CHAR(20) NOT NULL,
D_ZIP CHAR(9) NOT NULL
)
IN ts_dist_018
INDEX IN ts_dist_018
ORGANIZE BY KEY SEQUENCE (D_W_ID STARTING FROM 54367 ENDING AT 54367)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT15;
CREATE TABLE DISTRICT15
D_NEXT_O_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 44773 ENDING AT 44773)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT10;
CONNECT TO TPCC IN SHARED 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 38377 ENDING AT 41574)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT07;
CONNECT TO TPCC IN SHARED MODE;
DROP TABLE DISTRICT09;
CONNECT TO TPCC IN SHARED MODE;
DROP TABLE DISTRICT11;
CONNECT TO TPCC IN SHARED MODE;
DROP TABLE DISTRICT08;
CONNECT TO TPCC IN SHARED MODE;
DROP TABLE DISTRICT10;
CONNECT TO TPCC IN SHARED MODE;
DROP TABLE DISTRICT06;
CONNECT TO TPCC IN SHARED MODE;
DROP TABLE DISTRICT04;
CONNECT TO TPCC IN SHARED MODE;
DROP TABLE DISTRICT02;
CONNECT TO TPCC IN SHARED MODE;
DROP TABLE DISTRICT01;
CONNECT TO TPCC IN SHARED MODE;
DROP TABLE DISTRICT12;
CONNECT TO TPCC IN SHARED MODE;
DROP TABLE DISTRICT03;
CONNECT TO TPCC IN SHARED MODE;
DROP TABLE DISTRICT05;
CONNECT TO TPCC IN SHARED MODE;
DROP TABLE DISTRICT018;
CONNECT TO TPCC IN SHARED MODE;
DROP TABLE DISTRICT017;
CONNECT TO TPCC IN SHARED MODE;
DROP TABLE DISTRICT016;
CONNECT TO TPCC IN SHARED MODE;
DROP TABLE DISTRICT015;
CONNECT TO TPCC IN SHARED MODE;
DROP TABLE DISTRICT014;
CONNECT TO TPCC IN SHARED MODE;
DROP TABLE DISTRICT013;
CONNECT TO TPCC IN SHARED MODE;
DROP TABLE DISTRICT012;
CONNECT TO TPCC IN SHARED MODE;
DROP TABLE DISTRICT011;
CONNECT TO TPCC IN SHARED MODE;
DROP TABLE DISTRICT010;
CONNECT TO TPCC IN SHARED MODE;
DROP TABLE DISTRICT09;
CONNECT TO TPCC IN SHARED MODE;
DROP TABLE DISTRICT08;
CONNECT TO TPCC IN SHARED MODE;
DROP TABLE DISTRICT07;
CONNECT TO TPCC IN SHARED MODE;
DROP TABLE DISTRICT06;
CONNECT TO TPCC IN SHARED MODE;
DROP TABLE DISTRICT05;
CONNECT TO TPCC IN SHARED MODE;
DROP TABLE DISTRICT04;
CONNECT TO TPCC IN SHARED MODE;
DROP TABLE DISTRICT03;
CONNECT TO TPCC IN SHARED MODE;
DROP TABLE DISTRICT02;
CONNECT TO TPCC IN SHARED MODE;
DROP TABLE DISTRICT01;
CONNECT TO TPCC IN SHARED MODE;
```
INDEX IN ts_neworda_005
ORGANIZE BY KEY-SEQUENCE :
NO_W_ID STARTING FROM 25783 ENDING AT 28980,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_D_ID STARTING FROM 1900 ENDING AT 3678
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 35179 ENDING AT 38376,                     NO_D_ID STARTING FROM 1 ENDING AT 10,                     NO_O_ID STARTING FROM 1900 ENDING AT 3678
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 35179 ENDING AT 38376,                     NO_D_ID STARTING FROM 1 ENDING AT 10,                     NO_O_ID STARTING FROM 1900 ENDING AT 3678
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 22387 ENDING AT 25584,                     NO_D_ID STARTING FROM 1 ENDING AT 10,                     NO_O_ID STARTING FROM 1900 ENDING AT 3678
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 25856 ENDING AT 28782,                     NO_D_ID STARTING FROM 1 ENDING AT 10,                     NO_O_ID STARTING FROM 1900 ENDING AT 3678
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 31986 ENDING AT 35183,                     NO_D_ID STARTING FROM 1 ENDING AT 10,                     NO_O_ID STARTING FROM 1900 ENDING AT 3678
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 35184 ENDING AT 38381,                     NO_D_ID STARTING FROM 1 ENDING AT 10,                     NO_O_ID STARTING FROM 1900 ENDING AT 3678
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 38382 ENDING AT 41579,                     NO_D_ID STARTING FROM 1 ENDING AT 10,                     NO_O_ID STARTING FROM 1900 ENDING AT 3678
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 41580 ENDING AT 44777,                     NO_D_ID STARTING FROM 1 ENDING AT 10,                     NO_O_ID STARTING FROM 1900 ENDING AT 3678
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 44778 ENDING AT 47975,                     NO_D_ID STARTING FROM 1 ENDING AT 10,                     NO_O_ID STARTING FROM 1900 ENDING AT 3678
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 47976 ENDING AT 51163,                     NO_D_ID STARTING FROM 1 ENDING AT 10,                     NO_O_ID STARTING FROM 1900 ENDING AT 3678
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 51164 ENDING AT 54361,                     NO_D_ID STARTING FROM 1 ENDING AT 10,                     NO_O_ID STARTING FROM 1900 ENDING AT 3678
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 54362 ENDING AT 57559,                     NO_D_ID STARTING FROM 1 ENDING AT 10,                     NO_O_ID STARTING FROM 1900 ENDING AT 3678
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 57560 ENDING AT 60757,                     NO_D_ID STARTING FROM 1 ENDING AT 10,                     NO_O_ID STARTING FROM 1900 ENDING AT 3678
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 60758 ENDING AT 63955,                     NO_D_ID STARTING FROM 1 ENDING AT 10,                     NO_O_ID STARTING FROM 1900 ENDING AT 3678
ALLOW OVERFLOW;
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
)
ORGANIZE BY KEY SEQUENCE (NO_W_ID STARTING FROM 83149 ENDING AT 86346, NO_D_ID STARTING FROM 1 ENDING AT 10, NO_O_ID STARTING FROM 1900 ENDING AT 3678)
INDEX IN ts_newordA_027
ORGANIZE BY KEY SEQUENCE (NO_W_ID STARTING FROM 86347 ENDING AT 89544, NO_D_ID STARTING FROM 1 ENDING AT 10, NO_O_ID STARTING FROM 1900 ENDING AT 3678)
ALLOW OVERFLOW;

connect reset;

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
)
ORGANIZE BY KEY SEQUENCE (NO_W_ID STARTING FROM 73555 ENDING AT 76752, NO_D_ID STARTING FROM 1 ENDING AT 10, NO_O_ID STARTING FROM 1900 ENDING AT 3678)
INDEX IN ts_newordA_028
ORGANIZE BY KEY SEQUENCE (NO_W_ID STARTING FROM 76753 ENDING AT 79950, NO_D_ID STARTING FROM 1 ENDING AT 10, NO_O_ID STARTING FROM 1900 ENDING AT 3678)
ALLOW OVERFLOW;

connect reset;

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
)
ORGANIZE BY KEY SEQUENCE (NO_W_ID STARTING FROM 60763 ENDING AT 63960, NO_D_ID STARTING FROM 1 ENDING AT 10, NO_O_ID STARTING FROM 1900 ENDING AT 3678)
INDEX IN ts_newordA_029
ORGANIZE BY KEY SEQUENCE (NO_W_ID STARTING FROM 63961 ENDING AT 67158, NO_D_ID STARTING FROM 1 ENDING AT 10, NO_O_ID STARTING FROM 1900 ENDING AT 3678)
ALLOW OVERFLOW;

connect reset;

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
)
ORGANIZE BY KEY SEQUENCE (NO_W_ID STARTING FROM 67159 ENDING AT 70356, NO_D_ID STARTING FROM 1 ENDING AT 10, NO_O_ID STARTING FROM 1900 ENDING AT 3678)
ALLOW OVERFLOW;

connect reset;
CREATE TABLE NEW_ORDERA36

NO_O_ID   INTEGER     NOT NULL,
NO_D_ID   SMALLINT    NOT NULL,
NO_W_ID   INTEGER     NOT NULL

INDEX IN ts_newordA_036
ORGANIZE BY KEY SEQUENCE (NO_W_ID STARTING FROM 108733 ENDING AT 111930, NO_O_ID STARTING FROM 1900 ENDING AT 3678)
ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA37;
CREATE TABLE NEW_ORDERA37

NO_O_ID   INTEGER     NOT NULL,
NO_D_ID   SMALLINT    NOT NULL,
NO_W_ID   INTEGER     NOT NULL

INDEX IN ts_newordA_037
ORGANIZE BY KEY SEQUENCE (NO_W_ID STARTING FROM 111931 ENDING AT 115128, NO_D_ID STARTING FROM 1 ENDING AT 10, NO_O_ID STARTING FROM 1900 ENDING AT 3678)
ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA38;
CREATE TABLE NEW_ORDERA38

NO_O_ID   INTEGER     NOT NULL,
NO_D_ID   SMALLINT    NOT NULL,
NO_W_ID   INTEGER     NOT NULL

INDEX IN ts_newordA_038
ORGANIZE BY KEY SEQUENCE (NO_W_ID STARTING FROM 118327 ENDING AT 121524, NO_D_ID STARTING FROM 1 ENDING AT 10, NO_O_ID STARTING FROM 1900 ENDING AT 3678)
ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA39;
CREATE TABLE NEW_ORDERA39

NO_O_ID   INTEGER     NOT NULL,
NO_D_ID   SMALLINT    NOT NULL,
NO_W_ID   INTEGER     NOT NULL

INDEX IN ts_newordA_039
ORGANIZE BY KEY SEQUENCE (NO_W_ID STARTING FROM 121525 ENDING AT 124722, NO_D_ID STARTING FROM 1 ENDING AT 10, NO_O_ID STARTING FROM 1900 ENDING AT 3678)
ALLOW OVERFLOW;
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
) ORGANIZE BY KEY SEQUENCE
(NO_D_ID STARTING FROM 38377 ENDING AT 41574,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 3679 ENDING AT 5457)
ALLOW OVERFLOW;

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
) ORGANIZE BY KEY SEQUENCE
(NO_D_ID STARTING FROM 22387 ENDING AT 25584,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 3679 ENDING AT 5457)
ALLOW OVERFLOW;

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
) ORGANIZE BY KEY SEQUENCE
(NO_W_ID STARTING FROM 28783 ENDING AT 31980,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 3679 ENDING AT 5457)
ALLOW OVERFLOW;

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
) ORGANIZE BY KEY SEQUENCE
(NO_D_ID STARTING FROM 31981 ENDING AT 35178,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 3679 ENDING AT 5457)
ALLOW OVERFLOW;

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
) ORGANIZE BY KEY SEQUENCE
(NO_W_ID STARTING FROM 15991 ENDING AT 19188,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 3679 ENDING AT 5457)
ALLOW OVERFLOW;

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
) ORGANIZE BY KEY SEQUENCE
(NO_W_ID STARTING FROM 19189 ENDING AT 22386,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 3679 ENDING AT 5457)
ALLOW OVERFLOW;

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
) ORGANIZE BY KEY SEQUENCE
(NO_W_ID STARTING FROM 47971 ENDING AT 51168,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 3679 ENDING AT 5457)
ALLOW OVERFLOW;
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 70357 ENDING AT 73554, NO_D_ID STARTING FROM 1 ENDING AT 10, NO_O_ID STARTING FROM 3679 ENDING AT 5457)
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 73555 ENDING AT 76752, NO_D_ID STARTING FROM 1 ENDING AT 10, NO_O_ID STARTING FROM 3679 ENDING AT 5457)
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 76753 ENDING AT 79950, NO_D_ID STARTING FROM 1 ENDING AT 10, NO_O_ID STARTING FROM 3679 ENDING AT 5457)
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 79951 ENDING AT 83148, NO_D_ID STARTING FROM 1 ENDING AT 10, NO_O_ID STARTING FROM 3679 ENDING AT 5457)
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 83149 ENDING AT 86346, NO_D_ID STARTING FROM 1 ENDING AT 10, NO_O_ID STARTING FROM 3679 ENDING AT 5457)
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 86347 ENDING AT 89544, NO_D_ID STARTING FROM 1 ENDING AT 10, NO_O_ID STARTING FROM 3679 ENDING AT 5457)
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 89545 ENDING AT 92742, NO_D_ID STARTING FROM 1 ENDING AT 10, NO_O_ID STARTING FROM 3679 ENDING AT 5457)
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 92743 ENDING AT 95940, NO_D_ID STARTING FROM 1 ENDING AT 10, NO_O_ID STARTING FROM 3679 ENDING AT 5457)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
<table>
<thead>
<tr>
<th>Table Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>new_orderb30</td>
<td>DROP TABLE NEW_ORDERB30; CREATE TABLE NEW_ORDERB30</td>
</tr>
<tr>
<td>new_orderb31</td>
<td>DROP TABLE NEW_ORDERB31; CREATE TABLE NEW_ORDERB31</td>
</tr>
<tr>
<td>new_orderb32</td>
<td>DROP TABLE NEW_ORDERB32; CREATE TABLE NEW_ORDERB32</td>
</tr>
<tr>
<td>new_orderb33</td>
<td>DROP TABLE NEW_ORDERB33; CREATE TABLE NEW_ORDERB33</td>
</tr>
<tr>
<td>new_orderb34</td>
<td>DROP TABLE NEW_ORDERB34; CREATE TABLE NEW_ORDERB34</td>
</tr>
<tr>
<td>new_orderb35</td>
<td>DROP TABLE NEW_ORDERB35; CREATE TABLE NEW_ORDERB35</td>
</tr>
<tr>
<td>new_orderb36</td>
<td>DROP TABLE NEW_ORDERB36; CREATE TABLE NEW_ORDERB36</td>
</tr>
<tr>
<td>new_orderb37</td>
<td>DROP TABLE NEW_ORDERB37; CREATE TABLE NEW_ORDERB37</td>
</tr>
<tr>
<td>orders1</td>
<td>DROP TABLE ORDERS1; CREATE TABLE ORDERS1</td>
</tr>
<tr>
<td>orders30</td>
<td>DROP TABLE ORDERS30; CREATE TABLE ORDERS30</td>
</tr>
<tr>
<td>orders31</td>
<td>DROP TABLE ORDERS31; CREATE TABLE ORDERS31</td>
</tr>
<tr>
<td>orders32</td>
<td>DROP TABLE ORDERS32; CREATE TABLE ORDERS32</td>
</tr>
<tr>
<td>orders33</td>
<td>DROP TABLE ORDERS33; CREATE TABLE ORDERS33</td>
</tr>
<tr>
<td>orders34</td>
<td>DROP TABLE ORDERS34; CREATE TABLE ORDERS34</td>
</tr>
<tr>
<td>orders35</td>
<td>DROP TABLE ORDERS35; CREATE TABLE ORDERS35</td>
</tr>
<tr>
<td>orders36</td>
<td>DROP TABLE ORDERS36; CREATE TABLE ORDERS36</td>
</tr>
<tr>
<td>orders37</td>
<td>DROP TABLE ORDERS37; CREATE TABLE ORDERS37</td>
</tr>
<tr>
<td>orders38</td>
<td>DROP TABLE ORDERS38; CREATE TABLE ORDERS38</td>
</tr>
<tr>
<td>orders39</td>
<td>DROP TABLE ORDERS39; CREATE TABLE ORDERS39</td>
</tr>
<tr>
<td>orders40</td>
<td>DROP TABLE ORDERS40; CREATE TABLE ORDERS40</td>
</tr>
</tbody>
</table>

DDL/CRTB_ORDERS.ddl

connect reset; connect to TPCC in share mode; DROP TABLE NEW_ORDERB34; CREATE TABLE NEW_ORDERB34
( NO_O_ID INTEGER NOT NULL, NO_D_ID SMALLINT NOT NULL, NO_W_ID INTEGER NOT NULL )
INDEX IN ts_newordb_034
ORGANIZE BY KEY SEQUENCE { NO_W_ID STARTING FROM 95941 ENDING AT 99138, NO_D_ID STARTING FROM 1 ENDING AT 10, NO_O_ID STARTING FROM 3679 ENDING AT 5457 }
ALLOW OVERFLOW;
connect reset; connect to TPCC in share mode; DROP TABLE NEW_ORDERB35; CREATE TABLE NEW_ORDERB35
( NO_O_ID INTEGER NOT NULL, NO_D_ID SMALLINT NOT NULL, NO_W_ID INTEGER NOT NULL )
INDEX IN ts_newordb_035
ORGANIZE BY KEY SEQUENCE { NO_W_ID STARTING FROM 105535 ENDING AT 108732, NO_D_ID STARTING FROM 1 ENDING AT 10, NO_O_ID STARTING FROM 3679 ENDING AT 5457 }
ALLOW OVERFLOW;
connect reset; connect to TPCC in share mode; DROP TABLE NEW_ORDERB36; CREATE TABLE NEW_ORDERB36
( NO_O_ID INTEGER NOT NULL, NO_D_ID SMALLINT NOT NULL, NO_W_ID INTEGER NOT NULL )
INDEX IN ts_newordb_036
ORGANIZE BY KEY SEQUENCE { NO_W_ID STARTING FROM 111931 ENDING AT 115128, NO_D_ID STARTING FROM 1 ENDING AT 10, NO_O_ID STARTING FROM 3679 ENDING AT 5457 }
ALLOW OVERFLOW;
connect reset; connect to TPCC in share mode; DROP TABLE NEW_ORDERB37; CREATE TABLE NEW_ORDERB37
( NO_O_ID INTEGER NOT NULL, NO_D_ID SMALLINT NOT NULL, NO_W_ID INTEGER NOT NULL )
INDEX IN ts_newordb_037
ORGANIZE BY KEY SEQUENCE { NO_W_ID STARTING FROM 115129 ENDING AT 118326, NO_D_ID STARTING FROM 1 ENDING AT 10, NO_O_ID STARTING FROM 3679 ENDING AT 5457 }
ALLOW OVERFLOW;
connect reset; connect to TPCC in share mode; DROP TABLE NEW_ORDERB38; CREATE TABLE NEW_ORDERB38
( NO_O_ID INTEGER NOT NULL, NO_D_ID SMALLINT NOT NULL, NO_W_ID INTEGER NOT NULL )
INDEX IN ts_newordb_038
ORGANIZE BY KEY SEQUENCE { NO_W_ID STARTING FROM 118327 ENDING AT 121524, NO_D_ID STARTING FROM 1 ENDING AT 10, NO_O_ID STARTING FROM 3679 ENDING AT 5457 }
ALLOW OVERFLOW;
connect reset; connect to TPCC in share mode; DROP TABLE ORDERS1; CREATE TABLE ORDERS1
( O_C_ID INTEGER NOT NULL, O_D_ID SMALLINT NOT NULL, O_W_ID INTEGER NOT NULL )
INDEX IN ts_order_001
ORGANIZE BY KEY SEQUENCE { O_W_ID STARTING FROM 1 ENDING AT 1066, O_D_ID STARTING FROM 1 ENDING AT 10, O_O_ID STARTING FROM 1 ENDING AT 3678 }
ALLOW OVERFLOW;
CREATE TABLE ORDERS9
(O_C_ID INTEGER 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 3678,
O_W_ID STARTING FROM 8529 ENDING AT 9594,
O_D_ID STARTING FROM 1 ENDING AT 10
ALLOW OVERFLOW;

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_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 3678,
O_W_ID STARTING FROM 5331 ENDING AT 6396,
O_D_ID STARTING FROM 1 ENDING AT 10
ALLOW OVERFLOW;

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_002
INDEX IN is_order_002
ORGANIZE BY KEY SEQUENCE
O_ID STARTING FROM 1 ENDING AT 3678,
O_W_ID STARTING FROM 1067 ENDING AT 2132,
O_D_ID STARTING FROM 1 ENDING AT 10
ALLOW OVERFLOW;

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 3678,
O_W_ID STARTING FROM 2133 ENDING AT 3198,
O_D_ID STARTING FROM 1 ENDING AT 10
ALLOW OVERFLOW;

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_004
INDEX IN is_order_004
ORGANIZE BY KEY SEQUENCE
O_ID STARTING FROM 1 ENDING AT 3678,
O_W_ID STARTING FROM 3199 ENDING AT 4264,
O_D_ID STARTING FROM 1 ENDING AT 10
ALLOW OVERFLOW;

DROP TABLE ORDERS2;
CREATE TABLE ORDERS2
(O_C_ID INTEGER 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)
INDEX IN is_order_002
ORGANIZE BY KEY SEQUENCE
O_ID STARTING FROM 1 ENDING AT 3678,
O_W_ID STARTING FROM 1067 ENDING AT 2132,
O_D_ID STARTING FROM 1 ENDING AT 10
ALLOW OVERFLOW;

DROP TABLE ORDERS11;
CREATE TABLE ORDERS11
(O_C_ID INTEGER NOT NULL,
O_ENTRY_D TIMESTAMP NOT NULL,
O_ID INTEGER NOT NULL)
IN ts_order_011
INDEX IN is_order_011
ORGANIZE BY KEY SEQUENCE
O_ID STARTING FROM 1 ENDING AT 3678,
O_W_ID STARTING FROM 8529 ENDING AT 9594,
O_D_ID STARTING FROM 1 ENDING AT 10
ALLOW OVERFLOW;
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_ALL_LOCAL SMALLINT NOT NULL,
  O_ID INTEGER NOT NULL,
  O_W_ID INTEGER NOT NULL,
  O_D_ID SMALLINT NOT NULL )
INDEX IN is_order_012
ORGANIZE BY KEY SEQUENCE ( O_ID STARTING FROM 1 ENDING AT 3678,
  O_W_ID STARTING FROM 12793 ENDING AT 13858,
  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_ALL_LOCAL SMALLINT NOT NULL,
  O_ID INTEGER NOT NULL,
  O_W_ID INTEGER NOT NULL,
  O_D_ID SMALLINT NOT NULL )
INDEX IN is_order_013
ORGANIZE BY KEY SEQUENCE ( O_ID STARTING FROM 1 ENDING AT 3678,
  O_W_ID STARTING FROM 13859 ENDING AT 14924,
  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_ALL_LOCAL SMALLINT NOT NULL,
  O_ID INTEGER NOT NULL,
  O_W_ID INTEGER NOT NULL,
  O_D_ID SMALLINT NOT NULL )
INDEX IN is_order_014
ORGANIZE BY KEY SEQUENCE ( O_ID STARTING FROM 1 ENDING AT 3678,
  O_W_ID STARTING FROM 14925 ENDING AT 16090,
  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_ALL_LOCAL SMALLINT NOT NULL,
  O_ID INTEGER NOT NULL,
  O_W_ID INTEGER NOT NULL,
  O_D_ID SMALLINT NOT NULL )
INDEX IN is_order_015
ORGANIZE BY KEY SEQUENCE ( O_ID STARTING FROM 1 ENDING AT 3678,
  O_W_ID STARTING FROM 16091 ENDING AT 17156,
  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_ALL_LOCAL SMALLINT NOT NULL,
  O_ID INTEGER NOT NULL,
  O_W_ID INTEGER NOT NULL,
  O_D_ID SMALLINT NOT NULL )
INDEX IN is_order_016
ORGANIZE BY KEY SEQUENCE ( O_ID STARTING FROM 1 ENDING AT 3678,
  O_W_ID STARTING FROM 17157 ENDING AT 18212,
  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_ALL_LOCAL SMALLINT NOT NULL,
  O_ID INTEGER NOT NULL,
  O_W_ID INTEGER NOT NULL,
  O_D_ID SMALLINT NOT NULL )
INDEX IN is_order_017
ORGANIZE BY KEY SEQUENCE ( O_ID STARTING FROM 1 ENDING AT 3678,
  O_W_ID STARTING FROM 18213 ENDING AT 19268,
  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_ALL_LOCAL SMALLINT NOT NULL,
  O_ID INTEGER NOT NULL,
  O_W_ID INTEGER NOT NULL,
  O_D_ID SMALLINT NOT NULL )
INDEX IN is_order_018
ORGANIZE BY KEY SEQUENCE ( O_ID STARTING FROM 1 ENDING AT 3678,
  O_W_ID STARTING FROM 19269 ENDING AT 20324,
  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_ALL_LOCAL SMALLINT NOT NULL,
  O_ID INTEGER NOT NULL,
  O_W_ID INTEGER NOT NULL,
  O_D_ID SMALLINT NOT NULL )
INDEX IN is_order_019
ORGANIZE BY KEY SEQUENCE ( O_ID STARTING FROM 1 ENDING AT 3678,
  O_W_ID STARTING FROM 20325 ENDING AT 21380,
  O_D_ID STARTING FROM 1 ENDING AT 10 )
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
INDEX IN ts_order_071
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678, O_W_ID STARTING FROM 75687 ENDING AT 76752, O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS72;
CREATE TABLE ORDERS72
(O_C_ID INTEGER NOT NULL, O_ENTRY_D TIMESTAMP NOT NULL, O_CARRIER_ID SMALLINT NOT NULL, O_W_ID INTEGER NOT NULL, O_D_ID SMALLINT NOT NULL)
IN ts_order_072
INDEX IN ts_order_072
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678, O_W_ID STARTING FROM 76753 ENDING AT 77818, O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS73;
CREATE TABLE ORDERS73
(O_C_ID INTEGER NOT NULL, O_ENTRY_D TIMESTAMP NOT NULL, O_CARRIER_ID SMALLINT NOT NULL, O_W_ID INTEGER NOT NULL, O_D_ID SMALLINT NOT NULL)
IN ts_order_073
INDEX IN ts_order_073
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678, O_W_ID STARTING FROM 77819 ENDING AT 78884, O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS74;
CREATE TABLE ORDERS74
(O_C_ID INTEGER NOT NULL, O_ENTRY_D TIMESTAMP NOT NULL, O_CARRIER_ID SMALLINT NOT NULL, O_W_ID INTEGER NOT NULL, O_D_ID SMALLINT NOT NULL)
IN ts_order_074
INDEX IN ts_order_074
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678, O_W_ID STARTING FROM 78885 ENDING AT 79950, O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS75;
CREATE TABLE ORDERS75
(O_C_ID INTEGER NOT NULL, O_ENTRY_D TIMESTAMP NOT NULL, O_CARRIER_ID SMALLINT NOT NULL, O_W_ID INTEGER NOT NULL, O_D_ID SMALLINT NOT NULL)
IN ts_order_075
INDEX IN ts_order_075
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678, O_W_ID STARTING FROM 82083 ENDING AT 83148, O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS76;
CREATE TABLE ORDERS76
(O_C_ID INTEGER NOT NULL, O_ENTRY_D TIMESTAMP NOT NULL, O_CARRIER_ID SMALLINT NOT NULL, O_W_ID INTEGER NOT NULL, O_D_ID SMALLINT NOT NULL)
IN ts_order_076
INDEX IN ts_order_076
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678, O_W_ID STARTING FROM 84215 ENDING AT 85280, O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS77;
CREATE TABLE ORDERS77
(O_C_ID INTEGER NOT NULL, O_ENTRY_D TIMESTAMP NOT NULL, O_CARRIER_ID SMALLINT NOT NULL, O_W_ID INTEGER NOT NULL, O_D_ID SMALLINT NOT NULL)
IN ts_order_077
INDEX IN ts_order_077
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678, O_W_ID STARTING FROM 86293 ENDING AT 87348, O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS78;
CREATE TABLE ORDERS78
(O_C_ID INTEGER NOT NULL, O_ENTRY_D TIMESTAMP NOT NULL, O_CARRIER_ID SMALLINT NOT NULL, O_W_ID INTEGER NOT NULL, O_D_ID SMALLINT NOT NULL)
IN ts_order_078
INDEX IN ts_order_078
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678, O_W_ID STARTING FROM 88057 ENDING AT 89112, O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS79;
CREATE TABLE ORDERS79
(O_C_ID INTEGER NOT NULL, O_ENTRY_D TIMESTAMP NOT NULL, O_CARRIER_ID SMALLINT NOT NULL, O_W_ID INTEGER NOT NULL, O_D_ID SMALLINT NOT NULL)
IN ts_order_079
INDEX IN ts_order_079
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678, O_W_ID STARTING FROM 90119 ENDING AT 91174, O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS80;
CREATE TABLE ORDERS80
(O_C_ID INTEGER NOT NULL, O_ENTRY_D TIMESTAMP NOT NULL, O_CARRIER_ID SMALLINT NOT NULL, O_W_ID INTEGER NOT NULL, O_D_ID SMALLINT NOT NULL)
IN ts_order_080
INDEX IN ts_order_080
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678, O_W_ID STARTING FROM 92177 ENDING AT 93232, O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS81;
CREATE TABLE ORDERS81
(O_C_ID INTEGER NOT NULL, O_ENTRY_D TIMESTAMP NOT NULL, O_CARRIER_ID SMALLINT NOT NULL, O_W_ID INTEGER NOT NULL, O_D_ID SMALLINT NOT NULL)
IN ts_order_081
INDEX IN ts_order_081
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678, O_W_ID STARTING FROM 94235 ENDING AT 95290, O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS82;
CREATE TABLE ORDERS82
(O_C_ID INTEGER NOT NULL, O_ENTRY_D TIMESTAMP NOT NULL, O_CARRIER_ID SMALLINT NOT NULL, O_W_ID INTEGER NOT NULL, O_D_ID SMALLINT NOT NULL)
<table>
<thead>
<tr>
<th>O_C_ID</th>
<th>O_ENTRY_D</th>
<th>O_CARRIER_ID</th>
<th>O_OL_CNT</th>
<th>O_ALL_LOCAL</th>
<th>O_ID</th>
<th>O_D_ID</th>
<th>NOT NULL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td></td>
</tr>
</tbody>
</table>

**ALLOW OVERFLOW;**
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS82,
CREATE TABLE ORDERS82
(  
  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_D_ID SMALLINT NOT NULL
)
INDEX IN ts_order_082
ORGANIZE BY KEY SEQUENCE
(  
  O_ID STARTING FROM 1 ENDING AT 3678,
  O_W_ID STARTING FROM 92743 ENDING AT 93808,
  O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS83,
CREATE TABLE ORDERS83
(  
  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_D_ID SMALLINT NOT NULL
)
INDEX IN ts_order_083
ORGANIZE BY KEY SEQUENCE
(  
  O_W_ID STARTING FROM 85281 ENDING AT 86346,
  O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS84,
CREATE TABLE ORDERS84
(  
  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_D_ID SMALLINT NOT NULL
)
INDEX IN ts_order_084
ORGANIZE BY KEY SEQUENCE
(  
  O_ID STARTING FROM 1 ENDING AT 3678,
  O_W_ID STARTING FROM 86347 ENDING AT 87412,
  O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS85,
CREATE TABLE ORDERS85
(  
  O_C_ID INTEGER 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
)
INDEX IN ts_order_085
ORGANIZE BY KEY SEQUENCE
(  
  O_ID STARTING FROM 1 ENDING AT 3678,
  O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS86,
CREATE TABLE ORDERS86
(  
  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_D_ID SMALLINT NOT NULL
)
INDEX IN ts_order_086
ORGANIZE BY KEY SEQUENCE
(  
  O_W_ID STARTING FROM 86347 ENDING AT 87412,
  O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS87,
CREATE TABLE ORDERS87
(  
  O_C_ID INTEGER 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
)
INDEX IN ts_order_087
ORGANIZE BY KEY SEQUENCE
(  
  O_ID STARTING FROM 1 ENDING AT 3678,
  O_W_ID STARTING FROM 87413 ENDING AT 88478,
  O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS88,
CREATE TABLE ORDERS88
(  
  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_D_ID SMALLINT NOT NULL
)
INDEX IN ts_order_088
ORGANIZE BY KEY SEQUENCE
(  
  O_W_ID STARTING FROM 88479 ENDING AT 89544,
  O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS89,
CREATE TABLE ORDERS89
(  
  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
)
INDEX IN ts_order_089
ORGANIZE BY KEY SEQUENCE
(  
  O_ID STARTING FROM 1 ENDING AT 3678,
  O_W_ID STARTING FROM 92743 ENDING AT 93808,
  O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS90,
CREATE TABLE ORDERS90
(  
  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
)
INDEX IN ts_order_090
ORGANIZE BY KEY SEQUENCE
(  
  O_W_ID STARTING FROM 93809 ENDING AT 94874,
  O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS91,
CREATE TABLE ORDERS91
(  
  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
)
INDEX IN ts_order_091
ORGANIZE BY KEY SEQUENCE
(  
  O_W_ID STARTING FROM 94875 ENDING AT 95940,
  O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS92,
CREATE TABLE ORDERS92
(  
  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
)
INDEX IN ts_order_092
ORGANIZE BY KEY SEQUENCE
(  
  O_ID STARTING FROM 1 ENDING AT 3678,
  O_W_ID STARTING FROM 96347 ENDING AT 97412,
  O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS93,
CREATE TABLE ORDERS93
(  
  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_D_ID SMALLINT NOT NULL
)
INDEX IN ts_order_093
ORGANIZE BY KEY SEQUENCE
(  
  O_ID STARTING FROM 1 ENDING AT 3678,
  O_W_ID STARTING FROM 97413 ENDING AT 98478,
  O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS94,
CREATE TABLE ORDERS94
(  
  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_D_ID SMALLINT NOT NULL
)
INDEX IN ts_order_094
ORGANIZE BY KEY SEQUENCE
(  
  O_ID STARTING FROM 1 ENDING AT 3678,
  O_W_ID STARTING FROM 98479 ENDING AT 99544,
  O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
CREATE TABLE ORDERS99

| 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 |

INDEX IN is_order_099
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678, O_W_ID STARTING FROM 104469 ENDING AT 105534, O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode; CREATE TABLE ORDERS100

| 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 |

INDEX IN is_order_099
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678, O_W_ID STARTING FROM 102337 ENDING AT 103402, O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode; CREATE TABLE ORDERS93

| 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 |

INDEX IN is_order_099
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678, O_W_ID STARTING FROM 98073 ENDING AT 99138, O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode; CREATE TABLE ORDERS94

| 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 |

INDEX IN is_order_099
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678, O_W_ID STARTING FROM 103403 ENDING AT 104468, O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode; DROP TABLE ORDERS93; CREATE TABLE ORDERS95

| 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 |

INDEX IN is_order_095
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678, O_W_ID STARTING FROM 95941 ENDING AT 97006, O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode; DROP TABLE ORDERS94; CREATE TABLE ORDERS96

| 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 |

INDEX IN is_order_096
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678, O_W_ID STARTING FROM 103403 ENDING AT 104468, O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode; DROP TABLE ORDERS95; CREATE TABLE ORDERS97

| 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 |

INDEX IN is_order_097
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678, O_W_ID STARTING FROM 103402 ENDING AT 103402, O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode; DROP TABLE ORDERS96; CREATE TABLE ORDERS98

| 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 |

INDEX IN is_order_098
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678, O_W_ID STARTING FROM 103403 ENDING AT 104468, O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode; DROP TABLE ORDERS97; CREATE TABLE ORDERS99

| 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 |

INDEX IN is_order_099
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678, O_W_ID STARTING FROM 103403 ENDING AT 104468, O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode;
```
CREATE TABLE ORDERS118
(O_C_ID          INTEGER     NOT NULL,
O_D_ID          SMALLINT    NOT NULL,
O_W_ID          INTEGER     NOT NULL,
O_ID            INTEGER     NOT NULL)
INDEX IN ts_order_118
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678,
O_W_ID STARTING FROM 121525 ENDING AT 122590,
O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;
```

```
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS119;
CREATE TABLE ORDERS119
(O_C_ID          INTEGER     NOT NULL,
O_W_ID          INTEGER     NOT NULL,
O_D_ID          SMALLINT    NOT NULL)
INDEX IN ts_order_119
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678,
O_W_ID STARTING FROM 120459 ENDING AT 121524,
O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;
```

```
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS120;
CREATE TABLE ORDERS120
(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)
INDEX IN ts_order_115
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678,
O_W_ID STARTING FROM 121525 ENDING AT 122590,
O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;
```

```
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS115;
CREATE TABLE ORDERS115
(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)
INDEX IN ts_order_111
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678,
O_W_ID STARTING FROM 117261 ENDING AT 118326,
O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;
```

```
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS114;
CREATE TABLE ORDERS114
(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)
INDEX IN ts_order_114
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678,
O_W_ID STARTING FROM 115832 ENDING AT 116907,
O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;
```

```
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS113;
CREATE TABLE ORDERS113
(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)
INDEX IN ts_order_113
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678,
O_W_ID STARTING FROM 119093 ENDING AT 120168,
O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;
```

```
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS112;
CREATE TABLE ORDERS112
(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)
INDEX IN ts_order_112
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678,
O_W_ID STARTING FROM 110967 ENDING AT 112042,
O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;
```

```
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS111;
CREATE TABLE ORDERS111
(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)
INDEX IN ts_order_111
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678,
O_W_ID STARTING FROM 107247 ENDING AT 108322,
O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;
```

```
connect reset;
connect to TPCC in share mode;
DROP TABLE DDL_CRTB_ORDER_LINE.ddl
```

```
CREATE TABLE ORDERS118
(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)
INDEX IN ts_order_118
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678,
O_W_ID STARTING FROM 124723 ENDING AT 125788,
O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;
```

```
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS119;
CREATE TABLE ORDERS119
(O_C_ID          INTEGER     NOT NULL,
O_W_ID          INTEGER     NOT NULL,
O_D_ID          SMALLINT    NOT NULL)
INDEX IN ts_order_119
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678,
O_W_ID STARTING FROM 125789 ENDING AT 126854,
O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;
```

```
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS120;
CREATE TABLE ORDERS120
(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)
INDEX IN ts_order_120
ORGANIZE BY KEY SEQUENCE (O_ID STARTING FROM 1 ENDING AT 3678,
O_W_ID STARTING FROM 126855 ENDING AT 127920,
O_D_ID STARTING FROM 1 ENDING AT 10)
ALLOW OVERFLOW;
```

```
connect reset;
connect to TPCC in share mode;
DROP TABLE DDL_CRTB_ORDER_LINE.ddl
```

```
CREATE TABLE DDL_CRTB_ORDER_LINE.ddl
connect reset;
connect to TPCC in share mode;
DROP TABLE DDL_CRTB_ORDER_LINE.ddl
```
<table>
<thead>
<tr>
<th>Column</th>
<th>Type</th>
<th>Nullability</th>
<th>Index</th>
<th>Organize by Key Sequence</th>
</tr>
</thead>
<tbody>
<tr>
<td>OL_DELIVERY_D</td>
<td>TIMESTAMP</td>
<td>NOT NULL</td>
<td></td>
<td>OL_W_ID STARTING FROM 6397 ENDING AT 7462, OL_O_ID STARTING FROM 1 ENDING AT 3678, OL_NUMBER STARTING FROM 1 ENDING AT 15</td>
</tr>
<tr>
<td>OL_AMOUNT</td>
<td>DECIMAL(6,2)</td>
<td>NOT NULL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OL_I_ID</td>
<td>INTEGER</td>
<td>NOT NULL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OL_SUPPLY_W_ID</td>
<td>INTEGER</td>
<td>NOT NULL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OL_QUANTITY</td>
<td>SMALLINT</td>
<td>NOT NULL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OL_DIST_INFO</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OL_O_ID</td>
<td>INTEGER</td>
<td>NOT NULL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OL_D_ID</td>
<td>SMALLINT</td>
<td>NOT NULL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OL_W_ID</td>
<td>INTEGER</td>
<td>NOT NULL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OL_NUMBER</td>
<td>SMALLINT</td>
<td>NOT NULL</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

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 INTEGER NOT NULL,
  OL_W_ID INTEGER NOT NULL,
  OL_NUMBER SMALLINT NOT NULL
)
INDEX IN Ts_orderline_007
ORGANIZE BY KEY SEQUENCE
OL_W_ID STARTING FROM 6397 ENDING AT 7462,
OL_D_ID STARTING FROM 1 ENDING AT 10,
OL_O_ID STARTING FROM 1 ENDING AT 3678,
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_LINE6
(
  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 INTEGER NOT NULL,
  OL_W_ID INTEGER NOT NULL,
  OL_NUMBER SMALLINT NOT NULL
)
INDEX IN Ts_orderline_006
ORGANIZE BY KEY SEQUENCE
OL_W_ID STARTING FROM 6397 ENDING AT 7462,
OL_D_ID STARTING FROM 1 ENDING AT 10,
OL_O_ID STARTING FROM 1 ENDING AT 3678,
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_LINE5
(
  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 INTEGER NOT NULL,
  OL_W_ID INTEGER NOT NULL,
  OL_NUMBER SMALLINT NOT NULL
)
INDEX IN Ts_orderline_005
ORGANIZE BY KEY SEQUENCE
OL_W_ID STARTING FROM 6397 ENDING AT 7462,
OL_D_ID STARTING FROM 1 ENDING AT 10,
OL_O_ID STARTING FROM 1 ENDING AT 3678,
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_LINE4
(
  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
)
INDEX IN Ts_orderline_004
ORGANIZE BY KEY SEQUENCE
OL_W_ID STARTING FROM 3199 ENDING AT 4264,
OL_D_ID STARTING FROM 1 ENDING AT 10,
OL_O_ID STARTING FROM 1 ENDING AT 3678,
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_LINE3
(
  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
)
INDEX IN Ts_orderline_003
ORGANIZE BY KEY SEQUENCE
OL_W_ID STARTING FROM 2133 ENDING AT 3198,
OL_D_ID STARTING FROM 1 ENDING AT 10,
OL_O_ID STARTING FROM 1 ENDING AT 3678,
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_LINE2
(
  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
)
INDEX IN Ts_orderline_002
ORGANIZE BY KEY SEQUENCE
OL_W_ID STARTING FROM 3198 ENDING AT 4263,
OL_D_ID STARTING FROM 1 ENDING AT 10,
OL_O_ID STARTING FROM 1 ENDING AT 3678,
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_LINE1
(
  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
)
INDEX IN Ts_orderline_001
ORGANIZE BY KEY SEQUENCE
OL_W_ID STARTING FROM 2132 ENDING AT 3197,
OL_D_ID STARTING FROM 1 ENDING AT 10,
OL_O_ID STARTING FROM 1 ENDING AT 3678,
OL_NUMBER STARTING FROM 1 ENDING AT 15
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE1;

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OL_NUMBER</td>
<td>SMALLINT</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>OL_W_ID</td>
<td>SMALLINT</td>
<td>STARTING FROM 24519 ENDING AT 25584, 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_D_ID INTEGER NOT NULL, OL_W_ID INTEGER NOT NULL, OL_NUMBER SMALLINT NOT NULL )</td>
</tr>
<tr>
<td>OL_D_ID</td>
<td>SMALLINT</td>
<td>STARTING FROM 1 ENDING AT 10, 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_D_ID INTEGER NOT NULL, OL_W_ID INTEGER NOT NULL, OL_NUMBER SMALLINT NOT NULL )</td>
</tr>
<tr>
<td>OL_O_ID</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>OL_W_ID</td>
<td>SMALLINT</td>
<td>STARTING FROM 25585 ENDING AT 26650, 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_D_ID INTEGER NOT NULL, OL_W_ID INTEGER NOT NULL, OL_NUMBER SMALLINT NOT NULL )</td>
</tr>
<tr>
<td>OL_D_ID</td>
<td>SMALLINT</td>
<td>STARTING FROM 1 ENDING AT 10, 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_D_ID INTEGER NOT NULL, OL_W_ID INTEGER NOT NULL, OL_NUMBER SMALLINT NOT NULL )</td>
</tr>
<tr>
<td>OL_O_ID</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>OL_D_ID</td>
<td>SMALLINT</td>
<td>STARTING FROM 1 ENDING AT 10, 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_D_ID INTEGER NOT NULL, OL_W_ID INTEGER NOT NULL, OL_NUMBER SMALLINT NOT NULL )</td>
</tr>
<tr>
<td>OL_W_ID</td>
<td>SMALLINT</td>
<td>STARTING FROM 18123 ENDING AT 19188, 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_D_ID INTEGER NOT NULL, OL_W_ID INTEGER NOT NULL, OL_NUMBER SMALLINT NOT NULL )</td>
</tr>
<tr>
<td>OL_D_ID</td>
<td>SMALLINT</td>
<td>STARTING FROM 1 ENDING AT 10, 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_D_ID INTEGER NOT NULL, OL_W_ID INTEGER NOT NULL, OL_NUMBER SMALLINT NOT NULL )</td>
</tr>
<tr>
<td>OL_O_ID</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>OL_D_ID</td>
<td>SMALLINT</td>
<td>STARTING FROM 1 ENDING AT 10, 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_D_ID INTEGER NOT NULL, OL_W_ID INTEGER NOT NULL, OL_NUMBER SMALLINT NOT NULL )</td>
</tr>
<tr>
<td>OL_W_ID</td>
<td>SMALLINT</td>
<td>STARTING FROM 20255 ENDING AT 21320, 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_D_ID INTEGER NOT NULL, OL_W_ID INTEGER NOT NULL, OL_NUMBER SMALLINT NOT NULL )</td>
</tr>
<tr>
<td>OL_D_ID</td>
<td>SMALLINT</td>
<td>STARTING FROM 1 ENDING AT 10, 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_D_ID INTEGER NOT NULL, OL_W_ID INTEGER NOT NULL, OL_NUMBER SMALLINT NOT NULL )</td>
</tr>
<tr>
<td>OL_O_ID</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>OL_D_ID</td>
<td>SMALLINT</td>
<td>STARTING FROM 1 ENDING AT 10, 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_D_ID INTEGER NOT NULL, OL_W_ID INTEGER NOT NULL, OL_NUMBER SMALLINT NOT NULL )</td>
</tr>
<tr>
<td>OL_W_ID</td>
<td>SMALLINT</td>
<td>STARTING FROM 21321 ENDING AT 22386, 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_D_ID INTEGER NOT NULL, OL_W_ID INTEGER NOT NULL, OL_NUMBER SMALLINT NOT NULL )</td>
</tr>
<tr>
<td>OL_D_ID</td>
<td>SMALLINT</td>
<td>STARTING FROM 1 ENDING AT 10, 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_D_ID INTEGER NOT NULL, OL_W_ID INTEGER NOT NULL, OL_NUMBER SMALLINT NOT NULL )</td>
</tr>
<tr>
<td>OL_O_ID</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>OL_D_ID</td>
<td>SMALLINT</td>
<td>STARTING FROM 1 ENDING AT 10, 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_D_ID INTEGER NOT NULL, OL_W_ID INTEGER NOT NULL, OL_NUMBER SMALLINT NOT NULL )</td>
</tr>
<tr>
<td>OL_W_ID</td>
<td>SMALLINT</td>
<td>STARTING FROM 22387 ENDING AT 23452, 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_D_ID INTEGER NOT NULL, OL_W_ID INTEGER NOT NULL, OL_NUMBER SMALLINT NOT NULL )</td>
</tr>
<tr>
<td>OL_D_ID</td>
<td>SMALLINT</td>
<td>STARTING FROM 1 ENDING AT 10, 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_D_ID INTEGER NOT NULL, OL_W_ID INTEGER NOT NULL, OL_NUMBER SMALLINT NOT NULL )</td>
</tr>
<tr>
<td>OL_O_ID</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>OL_D_ID</td>
<td>SMALLINT</td>
<td>STARTING FROM 1 ENDING AT 10, 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_D_ID INTEGER NOT NULL, OL_W_ID INTEGER NOT NULL, OL_NUMBER SMALLINT NOT NULL )</td>
</tr>
<tr>
<td>OL_W_ID</td>
<td>SMALLINT</td>
<td>STARTING FROM 23453 ENDING AT 24518, 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_D_ID INTEGER NOT NULL, OL_W_ID INTEGER NOT NULL, OL_NUMBER SMALLINT NOT NULL )</td>
</tr>
<tr>
<td>OL_D_ID</td>
<td>SMALLINT</td>
<td>STARTING FROM 1 ENDING AT 10, 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_D_ID INTEGER NOT NULL, OL_W_ID INTEGER NOT NULL, OL_NUMBER SMALLINT NOT NULL )</td>
</tr>
<tr>
<td>OL_O_ID</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>OL_D_ID</td>
<td>SMALLINT</td>
<td>STARTING FROM 1 ENDING AT 10, 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_D_ID INTEGER NOT NULL, OL_W_ID INTEGER NOT NULL, OL_NUMBER SMALLINT NOT NULL )</td>
</tr>
<tr>
<td>OL_W_ID</td>
<td>SMALLINT</td>
<td>STARTING FROM 24519 ENDING AT 25584, 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_D_ID INTEGER NOT NULL, OL_W_ID INTEGER NOT NULL, OL_NUMBER SMALLINT NOT NULL )</td>
</tr>
</tbody>
</table>
CREATE TABLE ORDER_LINE63
(
  OL_W_ID          INTEGER      NOT NULL,
  OL_NUMBER        SMALLINT     NOT NULL,
  OL_DELIVERY_D    TIMESTAMP    NOT NULL,
  OL_AMT          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,
)
INDEX IN ts_orderline_063
ORGANIZE BY KEY SEQUENCE (                     OL_W_ID STARTING FROM 66093 ENDING AT 67158,
OL_D_ID STARTING FROM 1 ENDING AT 10,                     OL_O_ID STARTING FROM 1 ENDING AT 3678,
OL_NUMBER STARTING FROM 1 ENDING AT 15                    )                    ALLOW OVERFLOW;

DROP TABLE ORDER_LINE64;
CREATE TABLE ORDER_LINE64
(
  OL_W_ID          INTEGER      NOT NULL,
  OL_NUMBER        SMALLINT     NOT NULL,                     OL_DELIVERY_D    TIMESTAMP    NOT NULL,                     OL_AMT          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,
)
INDEX IN ts_orderline_064
ORGANIZE BY KEY SEQUENCE (                     OL_W_ID STARTING FROM 67159 ENDING AT 68224,
OL_D_ID STARTING FROM 1 ENDING AT 10,                     OL_O_ID STARTING FROM 1 ENDING AT 3678,
OL_NUMBER STARTING FROM 1 ENDING AT 15                    )                    ALLOW OVERFLOW;

DROP TABLE ORDER_LINE65;
CREATE TABLE ORDER_LINE65
(
  OL_W_ID          INTEGER      NOT NULL,                     OL_DELIVERY_D    TIMESTAMP    NOT NULL,                     OL_AMT          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                      )
INDEX IN ts_orderline_065
ORGANIZE BY KEY SEQUENCE (                     OL_W_ID STARTING FROM 68225 ENDING AT 69290,
OL_D_ID STARTING FROM 1 ENDING AT 10,                     OL_O_ID STARTING FROM 1 ENDING AT 3678,
OL_NUMBER STARTING FROM 1 ENDING AT 15                    )                    ALLOW OVERFLOW;

DROP TABLE ORDER_LINE66;
CREATE TABLE ORDER_LINE66
(
  OL_W_ID          INTEGER      NOT NULL,                     OL_DELIVERY_D    TIMESTAMP    NOT NULL,                     OL_AMT          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                      )
INDEX IN ts_orderline_066
ORGANIZE BY KEY SEQUENCE (                     OL_W_ID STARTING FROM 69291 ENDING AT 70356,
OL_D_ID STARTING FROM 1 ENDING AT 10,                     OL_O_ID STARTING FROM 1 ENDING AT 3678,
OL_NUMBER STARTING FROM 1 ENDING AT 15                    )                    ALLOW OVERFLOW;

DROP TABLE ORDER_LINE67;
CREATE TABLE ORDER_LINE67
(
  OL_W_ID          INTEGER      NOT NULL,                     OL_DELIVERY_D    TIMESTAMP    NOT NULL,                     OL_AMT          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                      )
INDEX IN ts_orderline_067
ORGANIZE BY KEY SEQUENCE (                     OL_W_ID STARTING FROM 70357 ENDING AT 71422,
OL_D_ID STARTING FROM 1 ENDING AT 10,                     OL_O_ID STARTING FROM 1 ENDING AT 3678,
OL_NUMBER STARTING FROM 1 ENDING AT 15                    )                    ALLOW OVERFLOW;

DROP TABLE ORDER_LINE68;
CREATE TABLE ORDER_LINE68
(
  OL_W_ID          INTEGER      NOT NULL,                     OL_DELIVERY_D    TIMESTAMP    NOT NULL,                     OL_AMT          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                      )
INDEX IN ts_orderline_068
ORGANIZE BY KEY SEQUENCE (                     OL_W_ID STARTING FROM 71423 ENDING AT 72488,
OL_D_ID STARTING FROM 1 ENDING AT 10,                     OL_O_ID STARTING FROM 1 ENDING AT 3678,
OL_NUMBER STARTING FROM 1 ENDING AT 15                    )                    ALLOW OVERFLOW;

DROP TABLE ORDER_LINE69;
CREATE TABLE ORDER_LINE69
(
  OL_W_ID          INTEGER      NOT NULL,                     OL_DELIVERY_D    TIMESTAMP    NOT NULL,                     OL_AMT          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                      )
INDEX IN ts_orderline_069
ORGANIZE BY KEY SEQUENCE (                     OL_W_ID STARTING FROM 72489 ENDING AT 73554,
OL_D_ID STARTING FROM 1 ENDING AT 10,                     OL_O_ID STARTING FROM 1 ENDING AT 3678,
OL_NUMBER STARTING FROM 1 ENDING AT 15                    )                    ALLOW OVERFLOW;
CREATE TABLE ORDER_LINE77
(
    OL_DELIVERY_D    TIMESTAMP    NOT NULL,
    OL_DIST_INFO     CHAR(24)     NOT NULL,
    OL_O_ID          INTEGER      NOT NULL,
    OL_D_ID          SMALLINT     NOT NULL,
    OL_NUMBER        SMALLINT     NOT NULL
)
INDEX IN ts_orderline_077
ORGANIZE BY KEY SEQUENCE (OL_W_ID STARTING FROM 81017 ENDING AT 82082, OL_D_ID STARTING FROM 1 ENDING AT 10, OL_O_ID STARTING FROM 1 ENDING AT 3678, OL_NUMBER STARTING FROM 1 ENDING AT 15)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE77;
CREATE TABLE ORDER_LINE78
(
    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
)
INDEX IN ts_orderline_078
ORGANIZE BY KEY SEQUENCE (OL_W_ID STARTING FROM 82083 ENDING AT 83148, OL_D_ID STARTING FROM 1 ENDING AT 10, OL_O_ID STARTING FROM 1 ENDING AT 3678, OL_NUMBER STARTING FROM 1 ENDING AT 15)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE78;
CREATE TABLE ORDER_LINE79
(
    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_D_ID          SMALLINT     NOT NULL,
    OL_W_ID          INTEGER      NOT NULL,
    OL_NUMBER        SMALLINT     NOT NULL
)
INDEX IN ts_orderline_079
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE79;
connect to TPCC in share mode; DROP TABLE ORDER_LINE86; CREATE TABLE ORDER_LINE86 ( 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_W_ID INTEGER NOT NULL, OL_D_ID SMALLINT NOT NULL, OL_NUMBER SMALLINT NOT NULL ) INDEX IN ts_orderline_086
INDEX IN ts_orderline_086
ORGANIZE BY KEY SEQUENCE ( OL_W_ID STARTING FROM 90611 ENDING AT 91676, OL_D_ID STARTING FROM 1 ENDING AT 10, OL_O_ID STARTING FROM 1 ENDING AT 3678, OL_NUMBER STARTING FROM 1 ENDING AT 15 ) ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode; DROP TABLE ORDER_LINE82; CREATE TABLE ORDER_LINE82 ( 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_D_ID SMALLINT NOT NULL, OL_W_ID INTEGER NOT NULL, OL_NUMBER SMALLINT NOT NULL ) INDEX IN ts_orderline_082
INDEX IN ts_orderline_082
ORGANIZE BY KEY SEQUENCE ( OL_W_ID STARTING FROM 84215 ENDING AT 85280, OL_D_ID STARTING FROM 1 ENDING AT 10, OL_O_ID STARTING FROM 1 ENDING AT 3678, OL_NUMBER STARTING FROM 1 ENDING AT 15 ) ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode; DROP TABLE ORDER_LINE88; CREATE TABLE ORDER_LINE88 ( 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_D_ID SMALLINT NOT NULL, OL_W_ID INTEGER NOT NULL, OL_NUMBER SMALLINT NOT NULL ) INDEX IN ts_orderline_088
INDEX IN ts_orderline_088
ORGANIZE BY KEY SEQUENCE ( OL_W_ID STARTING FROM 92743 ENDING AT 93808, OL_D_ID STARTING FROM 1 ENDING AT 15, OL_O_ID STARTING FROM 1 ENDING AT 3678, OL_NUMBER STARTING FROM 1 ENDING AT 15 ) ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode; DROP TABLE ORDER_LINE80; CREATE TABLE ORDER_LINE80 ( OL_I_ID INTEGER 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 ) INDEX IN ts_orderline_080
INDEX IN ts_orderline_080
ORGANIZE BY KEY SEQUENCE ( OL_W_ID STARTING FROM 84215 ENDING AT 85280, OL_D_ID STARTING FROM 1 ENDING AT 10, OL_O_ID STARTING FROM 1 ENDING AT 3678, OL_NUMBER STARTING FROM 1 ENDING AT 15 ) ALLOW OVERFLOW; connect reset;
DROP TABLE ORDER_LINE115;
CREATE TABLE ORDER_LINE115
(
    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_W_ID          INTEGER      NOT NULL,
    OL_D_ID          SMALLINT     NOT NULL,
    OL_NUMBER        SMALLINT     NOT NULL
)                    IN ts_orderline_115
INDEX IN ts_orderline_115
ORGANIZE BY KEY SEQUENCE (                     OL_W_ID STARTING FROM 126855 ENDING AT 127920,                     OL_D_ID STARTING FROM 1 ENDING AT 10,                     OL_O_ID STARTING FROM 1 ENDING AT 3678,                     OL_NUMBER STARTING FROM 1 ENDING AT 15                    )                    ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE116; CREATE TABLE ORDER_LINE116
(
    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_W_ID          INTEGER      NOT NULL,
    OL_D_ID          SMALLINT     NOT NULL,
    OL_NUMBER        SMALLINT     NOT NULL
)                    IN ts_orderline_116
INDEX IN ts_orderline_116
ORGANIZE BY KEY SEQUENCE (                     OL_W_ID STARTING FROM 127921 ENDING AT 128900,                     OL_D_ID STARTING FROM 1 ENDING AT 10,                     OL_O_ID STARTING FROM 1 ENDING AT 3678,                     OL_NUMBER STARTING FROM 1 ENDING AT 15                    )                    ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE117; CREATE TABLE ORDER_LINE117
(
    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_W_ID          INTEGER      NOT NULL,
    OL_D_ID          SMALLINT     NOT NULL,
    OL_NUMBER        SMALLINT     NOT NULL
)                    IN ts_orderline_117
INDEX IN ts_orderline_117
ORGANIZE BY KEY SEQUENCE (                     OL_W_ID STARTING FROM 128901 ENDING AT 129870,                     OL_D_ID STARTING FROM 1 ENDING AT 10,                     OL_O_ID STARTING FROM 1 ENDING AT 3678,                     OL_NUMBER STARTING FROM 1 ENDING AT 15                    )                    ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE118; CREATE TABLE ORDER_LINE118
(
    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_W_ID          INTEGER      NOT NULL,
    OL_D_ID          SMALLINT     NOT NULL,
    OL_NUMBER        SMALLINT     NOT NULL
)                    IN ts_orderline_118
INDEX IN ts_orderline_118
ORGANIZE BY KEY SEQUENCE (                     OL_W_ID STARTING FROM 129871 ENDING AT 130840,                     OL_D_ID STARTING FROM 1 ENDING AT 10,                     OL_O_ID STARTING FROM 1 ENDING AT 3678,                     OL_NUMBER STARTING FROM 1 ENDING AT 15                    )                    ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE119; CREATE TABLE ORDER_LINE119
(
    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)
                    IN ts_orderline_119
INDEX IN ts_orderline_119
ORGANIZE BY KEY SEQUENCE (                     OL_W_ID STARTING FROM 130841 ENDING AT 131810,                     OL_D_ID STARTING FROM 1 ENDING AT 10,                     OL_O_ID STARTING FROM 1 ENDING AT 3678,                     OL_NUMBER STARTING FROM 1 ENDING AT 15                    )                    ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE120; CREATE TABLE ORDER_LINE120
(
    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)
                    IN ts_orderline_120
INDEX IN ts_orderline_120
ORGANIZE BY KEY SEQUENCE (                     OL_W_ID STARTING FROM 131811 ENDING AT 132780,                     OL_D_ID STARTING FROM 1 ENDING AT 10,                     OL_O_ID STARTING FROM 1 ENDING AT 3678,                     OL_NUMBER STARTING FROM 1 ENDING AT 15                    )                    ALLOW OVERFLOW;
connect reset;
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_DATAA         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 1066                    )                    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_DATAA         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 1 ENDING AT 1067                    )                    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_DATAA         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 1 ENDING AT 1067                    )                    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_DATAA         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,
DROP TABLE STOCK8;
CREATE TABLE STOCK8
(
S_REMOTE_CNT    INTEGER     NOT NULL,
S_QUANTITY      INTEGER     NOT NULL,
S_ORDER_CNT     INTEGER     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
)
INDEX IN ts_stock_008
ORGANIZE BY KEY SEQUENCE (                     S_I_ID STARTING FROM 1 ENDING AT 100000,                     S_W_ID STARTING FROM 7463 ENDING AT 8528
)
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_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
)
INDEX IN ts_stock_006
ORGANIZE BY KEY SEQUENCE (                     S_I_ID STARTING FROM 1 ENDING AT 100000,                     S_W_ID STARTING FROM 5331 ENDING AT 6396
)
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_DATA          VARCHAR(50) NOT NULL,
S_DIST_01       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
)
INDEX IN ts_stock_003
ORGANIZE BY KEY SEQUENCE (                     S_I_ID STARTING FROM 1 ENDING AT 100000,                     S_W_ID STARTING FROM 6397 ENDING AT 7462
)
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_DATA          VARCHAR(50) 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
)
INDEX IN ts_stock_009
ORGANIZE BY KEY SEQUENCE (                     S_I_ID STARTING FROM 1 ENDING AT 100000,                     S_W_ID STARTING FROM 6525 ENDING AT 7594
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
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 15991 ENDING AT 16956 )
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_011
INDEX IN ts_stock_011
ORGANIZE BY KEY SEQUENCE (                     S_I_ID STARTING FROM 1 ENDING AT 100000,                     S_W_ID STARTING FROM 11727 ENDING AT 12792 )
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 12797 ENDING AT 13762 )
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 12793 ENDING AT 13858 )
ALLOW OVERFLOW;
connect reset;
INDEX IN ts_stock_017
ORGANIZE BY KEY SEQUENCE (S_I_ID STARTING FROM 1 ENDING AT 100000, S_W_ID STARTING FROM 17057 ENDING AT 18122)
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 19189 ENDING AT 20254)
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 20255 ENDING AT 21320)
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 21321 ENDING AT 22386)
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 22387 ENDING AT 23452)
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 23453 ENDING AT 24518)
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 24519 ENDING AT 25584)
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 25585 ENDING AT 26650)
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 26651 ENDING AT 27716)
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 27717 ENDING AT 28782)
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 28783 ENDING AT 29848)
ALLOW OVERFLOW;
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_W_ID           INTEGER     NOT NULL,                     S_I_ID          INTEGER     NOT NULL,                     S_W_ID           INTEGER     NOT NULL
)
INDEX IN ts_stock_025
ORGANIZE BY KEY SEQUENCE (                     S_I_ID STARTING FROM 1 ENDING AT 10000,
S_W_ID STARTING FROM 25885 ENDING AT 26650
)
ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK26;
CREATE TABLE STOCK26
(    S_REMOTE_CNT    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
)
INDEX IN ts_stock_026
ORGANIZE BY KEY SEQUENCE (                     S_I_ID STARTING FROM 1 ENDING AT 10000,
S_W_ID STARTING FROM 26651 ENDING AT 27716
)
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)    NOTNull,
ORGANIZE BY KEY SEQUENCE |
S_ID INTEGER NOT NULL, 
S_DATA VARCHAR(50) NOT NULL, 
S_YTD INTEGER 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_W_ID INTEGER NOT NULL, 
S_I_ID INTEGER NOT NULL, 
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_W_ID INTEGER NOT NULL, 
S_I_ID INTEGER NOT NULL, 
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_W_ID INTEGER NOT NULL, 
S_I_ID INTEGER NOT NULL, 
S_REMOTE_CNT INTEGER NOT NULL, 
S_QUANTITY INTEGER NOT NULL, 
S_YTD INTEGER NOT NULL, 
S_DATA VARCHAR(50) NOT NULL, 
S_W_ID STARTING FROM 00000 ENDING AT 00000, 
S_W_ID STARTING FROM 33117 ENDING AT 33112, 
S_W_ID STARTING FROM 30000 ENDING AT 30000.
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK47;
CREATE TABLE STOCK47
INDEX IN ts_stock_047
INDEX IN ts_stock_048
ORGANIZE BY KEY SEQUENCE
S_I_ID STARTING FROM 1 ENDING AT 100000,
S_W_ID STARTING FROM 50103 ENDING AT 51168
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK49;
CREATE TABLE STOCK49
INDEX IN ts_stock_049
INDEX IN ts_stock_050
ORGANIZE BY KEY SEQUENCE
S_I_ID STARTING FROM 1 ENDING AT 100000,
S_W_ID STARTING FROM 51169 ENDING AT 52234
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK50;
CREATE TABLE STOCK50
INDEX IN ts_stock_051
INDEX IN ts_stock_052
ORGANIZE BY KEY SEQUENCE
S_I_ID STARTING FROM 1 ENDING AT 100000,
S_W_ID STARTING FROM 52235 ENDING AT 53300
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK52;
CREATE TABLE STOCK52
INDEX IN ts_stock_053
INDEX IN ts_stock_054
ORGANIZE BY KEY SEQUENCE
S_I_ID STARTING FROM 1 ENDING AT 100000,
S_W_ID STARTING FROM 53301 ENDING AT 54366
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK53;
CREATE TABLE STOCK53
INDEX IN ts_stock_055
INDEX IN ts_stock_056
ORGANIZE BY KEY SEQUENCE
S_I_ID STARTING FROM 1 ENDING AT 100000,
S_W_ID STARTING FROM 54367 ENDING AT 55432
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK54;
CREATE TABLE STOCK54
INDEX IN ts_stock_057
INDEX IN ts_stock_058
ORGANIZE BY KEY SEQUENCE
S_I_ID STARTING FROM 1 ENDING AT 100000,
S_W_ID STARTING FROM 55433 ENDING AT 56498
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK55;
CREATE TABLE STOCK55
<table>
<thead>
<tr>
<th>Column</th>
<th>Type</th>
<th>Constraints</th>
</tr>
</thead>
<tbody>
<tr>
<td>S_I_ID</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_W_ID</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_ORDER_CNT</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_YTD</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DATA</td>
<td>VARCHAR</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_REMOTE_CNT</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_QUANTITY</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_01</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_02</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_03</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_04</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_05</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_06</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_07</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_08</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_09</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_10</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
</tbody>
</table>

Connect reset; connect to TPCC in share mode; DROP TABLE STOCK68; CREATE TABLE STOCK68
<table>
<thead>
<tr>
<th>Column</th>
<th>Type</th>
<th>Constraints</th>
</tr>
</thead>
<tbody>
<tr>
<td>S_REMOTE_CNT</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_QUANTITY</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_ORDER_CNT</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_YTD</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DATA</td>
<td>VARCHAR</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_01</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_02</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_03</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_04</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_05</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_06</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_07</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_10</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
</tbody>
</table>

Connect reset; connect to TPCC in share mode; DROP TABLE STOCK67; CREATE TABLE STOCK67
<table>
<thead>
<tr>
<th>Column</th>
<th>Type</th>
<th>Constraints</th>
</tr>
</thead>
<tbody>
<tr>
<td>S_REMOTE_CNT</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_QUANTITY</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_ORDER_CNT</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_YTD</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DATA</td>
<td>VARCHAR</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_01</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_02</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_03</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_04</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_05</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_06</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_07</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_08</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_09</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_10</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
</tbody>
</table>

Connect reset; connect to TPCC in share mode; DROP TABLE STOCK66; CREATE TABLE STOCK66
<table>
<thead>
<tr>
<th>Column</th>
<th>Type</th>
<th>Constraints</th>
</tr>
</thead>
<tbody>
<tr>
<td>S_REMOTE_CNT</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_QUANTITY</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_ORDER_CNT</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_YTD</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DATA</td>
<td>VARCHAR</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_01</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_02</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_03</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_04</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_05</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_06</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_07</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_08</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_09</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_10</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
</tbody>
</table>

Connect reset; connect to TPCC in share mode; DROP TABLE STOCK65; CREATE TABLE STOCK65
<table>
<thead>
<tr>
<th>Column</th>
<th>Type</th>
<th>Constraints</th>
</tr>
</thead>
<tbody>
<tr>
<td>S_REMOTE_CNT</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_QUANTITY</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_ORDER_CNT</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_YTD</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DATA</td>
<td>VARCHAR</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_01</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_02</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_03</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_04</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_05</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_06</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_07</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_08</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_09</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_10</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
</tbody>
</table>

Connect reset; connect to TPCC in share mode; DROP TABLE STOCK64; CREATE TABLE STOCK64
<table>
<thead>
<tr>
<th>Column</th>
<th>Type</th>
<th>Constraints</th>
</tr>
</thead>
<tbody>
<tr>
<td>S_REMOTE_CNT</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_QUANTITY</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_ORDER_CNT</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_YTD</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DATA</td>
<td>VARCHAR</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_01</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_02</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_03</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_04</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_05</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_06</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_07</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_08</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_09</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_10</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
</tbody>
</table>

Connect reset; connect to TPCC in share mode; DROP TABLE STOCK63; CREATE TABLE STOCK63
<table>
<thead>
<tr>
<th>Column</th>
<th>Type</th>
<th>Constraints</th>
</tr>
</thead>
<tbody>
<tr>
<td>S_REMOTE_CNT</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_QUANTITY</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_ORDER_CNT</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_YTD</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DATA</td>
<td>VARCHAR</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_01</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_02</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_03</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_04</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_05</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_06</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_07</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>S_DIST_08</td>
<td>CHAR(24)</td>
<td>NOT NULL</td>
</tr>
</tbody>
</table>

Connect reset; connect to TPCC in share mode;
CREATE TABLE STOCK74 (                     S_REMOTE_CNT    INTEGER     NOT NULL,                     S_QUANTITY      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_I_ID          INTEGER     NOT NULL,                     S_W_ID          INTEGER     NOT NULL                    )
DROP TABLE STOCK81;
CREATE TABLE STOCK81
(
S_REMOTE_CNT    INTEGER     NOT NULL,                     S_QUANTITY      INTEGER     NOT NULL,                     S_ORDER_CNT     INTEGER     NOT NULL,
S_YTD           INTEGER     NOT NULL,
S_DIST_07       CHAR(24)    NOT NULL,                     S_I_ID          INTEGER     NOT NULL,                     S_W_ID          INTEGER     NOT NULL,
)
IN ts_stock_078
INDEX IN ts_stock_078
ORGANIZE BY KEY SEQUENCE | S_I_ID STARTING FROM 1 ENDING AT 100000,
S_W_ID STARTING FROM 82083 ENDING AT 83148
ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK82;
CREATE TABLE STOCK82
(
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_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_081
INDEX IN ts_stock_081
ORGANIZE BY KEY SEQUENCE | S_I_ID STARTING FROM 1 ENDING AT 100000,
S_W_ID STARTING FROM 86347 ENDING AT 87412
ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK80;
CREATE TABLE STOCK80
(
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_079
INDEX IN ts_stock_079
ORGANIZE BY KEY SEQUENCE | S_I_ID STARTING FROM 1 ENDING AT 100000,
S_W_ID STARTING FROM 86347 ENDING AT 87412
ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK77;
CREATE TABLE STOCK77
(
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_077
INDEX IN ts_stock_077
ORGANIZE BY KEY SEQUENCE | S_I_ID STARTING FROM 1 ENDING AT 100000,
S_W_ID STARTING FROM 82083 ENDING AT 83148
ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK83;
CREATE TABLE STOCK83
(
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_082
INDEX IN ts_stock_082
ORGANIZE BY KEY SEQUENCE | S_I_ID STARTING FROM 1 ENDING AT 100000,
S_W_ID STARTING FROM 8347 ENDING AT 8412
ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode;
CREATE TABLE STOCK103
(  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_I_ID       CHAR(24) NOT NULL,
  S_W_ID       INTEGER NOT NULL)
INDEX IN ts_stock_103
ORGANIZE BY KEY SEQUENCE (  S_I_ID STARTING FROM 1 ENDING AT 100000,
  S_W_ID STARTING FROM 103403 ENDING AT 104468)
ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK104;
CREATE TABLE STOCK104
(  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)
INDEX IN ts_stock_104
ORGANIZE BY KEY SEQUENCE (  S_I_ID STARTING FROM 1 ENDING AT 100000,
  S_W_ID STARTING FROM 106601 ENDING AT 107666)
ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK105;
CREATE TABLE STOCK105
(  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)
INDEX IN ts_stock_105
ORGANIZE BY KEY SEQUENCE (  S_I_ID STARTING FROM 1 ENDING AT 100000,
  S_W_ID STARTING FROM 109799 ENDING AT 110864)
ALLOW OVERFLOW;
S_DIST_01 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 STARTING FROM 1 ENDING AT 100000,
S_W_ID INTEGER NOT NULL,
)

INDEX IN ts_stock_105

ORGANIZE BY KEY SEQUENCE
S_I_ID STARTING FROM 1 ENDING AT 100000,
S_W_ID STARTING FROM 110956 ENDING AT 111900

ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK105;
CREATE TABLE STOCK105
(
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,
)

INDEX IN ts_stock_105

ORGANIZE BY KEY SEQUENCE
S_I_ID STARTING FROM 1 ENDING AT 100000,
S_W_ID STARTING FROM 111901 ENDING AT 112956

ALLOW OVERFLOW;

connect reset;
connect to TPCC in shared mode;
DROP TABLE STOCK105;
CREATE TABLE STOCK105
(
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,
)

INDEX IN ts_stock_105

ORGANIZE BY KEY SEQUENCE
S_I_ID STARTING FROM 1 ENDING AT 100000,
S_W_ID STARTING FROM 112957 ENDING AT 113912

ALLOW OVERFLOW;

connect reset;
connect to TPCC in shared mode;
DROP TABLE STOCK105;
CREATE TABLE STOCK105
(
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,
)

INDEX IN ts_stock_105

ORGANIZE BY KEY SEQUENCE
S_I_ID STARTING FROM 1 ENDING AT 100000,
S_W_ID STARTING FROM 113913 ENDING AT 114868

ALLOW OVERFLOW;

connect reset;
connect to TPCC in shared mode;
DROP TABLE STOCK106;
CREATE TABLE STOCK106
(
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,
)

INDEX IN ts_stock_106

ORGANIZE BY KEY SEQUENCE
S_I_ID STARTING FROM 1 ENDING AT 100000,
S_W_ID STARTING FROM 114869 ENDING AT 115824

ALLOW OVERFLOW;

connect reset;
connect to TPCC in shared mode;
DROP TABLE STOCK106;
CREATE TABLE STOCK106
(
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,
)

INDEX IN ts_stock_106

ORGANIZE BY KEY SEQUENCE
S_I_ID STARTING FROM 1 ENDING AT 100000,
S_W_ID STARTING FROM 115825 ENDING AT 116780

ALLOW OVERFLOW;

connect reset;
connect to TPCC in shared mode;
DROP TABLE STOCK106;
CREATE TABLE STOCK106
(
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,
)

INDEX IN ts_stock_106

ORGANIZE BY KEY SEQUENCE
S_I_ID STARTING FROM 1 ENDING AT 100000,
S_W_ID STARTING FROM 116781 ENDING AT 117736

ALLOW OVERFLOW;

connect reset;
connect to TPCC in shared mode;
DROP TABLE STOCK106;
CREATE TABLE STOCK106
(
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_DIST_08 | CHAR(24) | NOT NULL |
| S_DIST_09 | CHAR(24) | NOT NULL |
| S_DIST_10 | CHAR(24) | NOT NULL |

INDEX ON `ts_stock_117`

ORGANIZE BY KEY SEQUENCE {
S_I_ID STARTING FROM 1 ENDING AT 10000,
S_W_ID STARTING FROM 123657 ENDING AT 124722
}

ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK112;
CREATE TABLE STOCK113
( S_REMOTE_CNT INTEGER NOT NULL,
S_QUANTITY INTEGER NOT NULL,
S_ORDER_CNT 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,
)

INDEX ON `ts_stock_113`

ORGANIZE BY KEY SEQUENCE {
S_I_ID STARTING FROM 1 ENDING AT 10000,
S_W_ID STARTING FROM 119393 ENDING AT 120458
}

ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK114;
CREATE TABLE STOCK115
( S_REMOTE_CNT INTEGER NOT NULL,
S_QUANTITY INTEGER NOT NULL,
S_ORDER_CNT 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,
)

INDEX ON `ts_stock_115`

ORGANIZE BY KEY SEQUENCE {
S_I_ID STARTING FROM 1 ENDING AT 10000,
S_W_ID STARTING FROM 120459 ENDING AT 121524
}

ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK116;
CREATE TABLE STOCK117
( S_REMOTE_CNT INTEGER NOT NULL,
S_QUANTITY INTEGER NOT NULL,
S_ORDER_CNT 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,
)

INDEX ON `ts_stock_117`

ORGANIZE BY KEY SEQUENCE {
S_I_ID STARTING FROM 1 ENDING AT 10000,
S_W_ID STARTING FROM 121525 ENDING AT 122590
}

ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK118;
CREATE TABLE STOCK119
( S_REMOTE_CNT INTEGER NOT NULL,
S_QUANTITY INTEGER NOT NULL,
S_ORDER_CNT 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,
)

INDEX ON `ts_stock_119`

ORGANIZE BY KEY SEQUENCE {
S_I_ID STARTING FROM 1 ENDING AT 10000,
S_W_ID STARTING FROM 122591 ENDING AT 123656
}

ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK120;
CREATE TABLE STOCK121
( S_REMOTE_CNT INTEGER NOT NULL,
S_QUANTITY INTEGER NOT NULL,
S_ORDER_CNT 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,
)

INDEX ON `ts_stock_121`

ORGANIZE BY KEY SEQUENCE {
S_I_ID STARTING FROM 1 ENDING AT 10000,
S_W_ID STARTING FROM 123657 ENDING AT 124722
}

ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK122;
CREATE TABLE STOCK123
( S_REMOTE_CNT INTEGER NOT NULL,
S_QUANTITY INTEGER NOT NULL,
S_ORDER_CNT 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,
)

INDEX ON `ts_stock_123`

ORGANIZE BY KEY SEQUENCE {
S_I_ID STARTING FROM 1 ENDING AT 10000,
S_W_ID STARTING FROM 124723 ENDING AT 125788
}

ALLOW OVERFLOW;

connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK124;
CREATE TABLE STOCK125
( S_REMOTE_CNT INTEGER NOT NULL,
S_QUANTITY INTEGER NOT NULL,
S_ORDER_CNT 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,
)

INDEX ON `ts_stock_125`

ORGANIZE BY KEY SEQUENCE {
S_I_ID STARTING FROM 1 ENDING AT 10000,
S_W_ID STARTING FROM 125789 ENDING AT 126854
}

ALLOW OVERFLOW;
INDEX IN ts_ware_009
ORGANIZE BY KEY SEQUENCE (
W_ID STARTING FROM 25595 ENDING AT 28782
)
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 28763 ENDING AT 31980
)
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 31981 ENDING AT 35178
)
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 35179 ENDING AT 38376
)
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 38377 ENDING AT 41574
)
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 41575 ENDING AT 44772
)
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_TAX       REAL            NOT NULL,
W_ID        INTEGER         NOT NULL)
IN ts_ware_015
INDEX IN ts_ware_015
ORGANIZE BY KEY SEQUENCE (
W_ID STARTING FROM 44773 ENDING AT 47970
)
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 47971 ENDING AT 51188
)
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 51189 ENDING AT 54386
)
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 54387 ENDING AT 57564
)
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_YYTD      DECIMAL(12,2)   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 57565 ENDING AT 60762
)
ALLOW OVERFLOW;
CREATE TABLE WAREHOUSE27
(
  W_STREET_1 CHAR(20) NOT NULL,
  W_STREET_2 CHAR(20) NOT NULL,
  W_CITY CHAR(20) 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 70575 ENDING AT 73554)
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_ID INTEGER NOT NULL
) IN ts_ware_028
INDEX IN ts_ware_028
ORGANIZE BY KEY SEQUENCE (W_ID STARTING FROM 86347 ENDING AT 89544)
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 70357 ENDING AT 73552)
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 86953 ENDING AT 89950)
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 89545 ENDING AT 92742)
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 89553 ENDING AT 92749)
ALLOW OVERFLOW;
<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>W_ID</td>
<td>INTEGER</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_NAME</td>
<td>CHAR(10)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STREET_1</td>
<td>CHAR(20)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STREET_2</td>
<td>CHAR(20)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_CITY</td>
<td>CHAR(20)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STATE</td>
<td>CHAR(2)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_ZIP</td>
<td>CHAR(9)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_TAX</td>
<td>REAL</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_YTD</td>
<td>DECIMAL(12,2)</td>
<td>NOT NULL</td>
</tr>
</tbody>
</table>

**CREATE TABLE WAREHOUSE31**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>W_NAME</td>
<td>CHAR(10)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STREET_1</td>
<td>CHAR(20)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STREET_2</td>
<td>CHAR(20)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STATE</td>
<td>CHAR(2)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_ZIP</td>
<td>CHAR(9)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_TAX</td>
<td>REAL</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_YTD</td>
<td>DECIMAL(12,2)</td>
<td>NOT NULL</td>
</tr>
</tbody>
</table>

**CREATE TABLE WAREHOUSE32**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>W_NAME</td>
<td>CHAR(10)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STREET_1</td>
<td>CHAR(20)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STREET_2</td>
<td>CHAR(20)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STATE</td>
<td>CHAR(2)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_ZIP</td>
<td>CHAR(9)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_TAX</td>
<td>REAL</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_YTD</td>
<td>DECIMAL(12,2)</td>
<td>NOT NULL</td>
</tr>
</tbody>
</table>

**CREATE TABLE WAREHOUSE33**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>W_NAME</td>
<td>CHAR(10)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STREET_1</td>
<td>CHAR(20)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STREET_2</td>
<td>CHAR(20)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STATE</td>
<td>CHAR(2)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_ZIP</td>
<td>CHAR(9)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_TAX</td>
<td>REAL</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_YTD</td>
<td>DECIMAL(12,2)</td>
<td>NOT NULL</td>
</tr>
</tbody>
</table>

**CREATE TABLE WAREHOUSE34**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>W_NAME</td>
<td>CHAR(10)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STREET_1</td>
<td>CHAR(20)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STREET_2</td>
<td>CHAR(20)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STATE</td>
<td>CHAR(2)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_ZIP</td>
<td>CHAR(9)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_TAX</td>
<td>REAL</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_YTD</td>
<td>DECIMAL(12,2)</td>
<td>NOT NULL</td>
</tr>
</tbody>
</table>

**CREATE TABLE WAREHOUSE35**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>W_NAME</td>
<td>CHAR(10)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STREET_1</td>
<td>CHAR(20)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STREET_2</td>
<td>CHAR(20)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STATE</td>
<td>CHAR(2)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_ZIP</td>
<td>CHAR(9)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_TAX</td>
<td>REAL</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_YTD</td>
<td>DECIMAL(12,2)</td>
<td>NOT NULL</td>
</tr>
</tbody>
</table>

**CREATE TABLE WAREHOUSE36**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>W_NAME</td>
<td>CHAR(10)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STREET_1</td>
<td>CHAR(20)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STREET_2</td>
<td>CHAR(20)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_CITY</td>
<td>CHAR(20)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STATE</td>
<td>CHAR(2)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_ZIP</td>
<td>CHAR(9)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_TAX</td>
<td>REAL</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_YTD</td>
<td>DECIMAL(12,2)</td>
<td>NOT NULL</td>
</tr>
</tbody>
</table>

**CREATE TABLE WAREHOUSE37**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>W_NAME</td>
<td>CHAR(10)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STREET_1</td>
<td>CHAR(20)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STREET_2</td>
<td>CHAR(20)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STATE</td>
<td>CHAR(2)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_ZIP</td>
<td>CHAR(9)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_TAX</td>
<td>REAL</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_YTD</td>
<td>DECIMAL(12,2)</td>
<td>NOT NULL</td>
</tr>
</tbody>
</table>

**CREATE TABLE WAREHOUSE38**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>W_NAME</td>
<td>CHAR(10)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STREET_1</td>
<td>CHAR(20)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STREET_2</td>
<td>CHAR(20)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STATE</td>
<td>CHAR(2)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_ZIP</td>
<td>CHAR(9)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_TAX</td>
<td>REAL</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_YTD</td>
<td>DECIMAL(12,2)</td>
<td>NOT NULL</td>
</tr>
</tbody>
</table>

**CREATE TABLE WAREHOUSE39**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>W_NAME</td>
<td>CHAR(10)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STREET_1</td>
<td>CHAR(20)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STREET_2</td>
<td>CHAR(20)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STATE</td>
<td>CHAR(2)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_ZIP</td>
<td>CHAR(9)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_TAX</td>
<td>REAL</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_YTD</td>
<td>DECIMAL(12,2)</td>
<td>NOT NULL</td>
</tr>
</tbody>
</table>

**CREATE TABLE WAREHOUSE40**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>W_NAME</td>
<td>CHAR(10)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STREET_1</td>
<td>CHAR(20)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STREET_2</td>
<td>CHAR(20)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STATE</td>
<td>CHAR(2)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_ZIP</td>
<td>CHAR(9)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_TAX</td>
<td>REAL</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_YTD</td>
<td>DECIMAL(12,2)</td>
<td>NOT NULL</td>
</tr>
</tbody>
</table>

**CREATE TABLE WAREHOUSE41**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>W_NAME</td>
<td>CHAR(10)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STREET_1</td>
<td>CHAR(20)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STREET_2</td>
<td>CHAR(20)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_STATE</td>
<td>CHAR(2)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_ZIP</td>
<td>CHAR(9)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_TAX</td>
<td>REAL</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>W_YTD</td>
<td>DECIMAL(12,2)</td>
<td>NOT NULL</td>
</tr>
</tbody>
</table>
SELECT * FROM CUSTOMER120
COMMIT WORK;
connect reset;

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 UNION ALL
SELECT * FROM CUSTOMER65 UNION ALL
SELECT * FROM CUSTOMER66 UNION ALL
SELECT * FROM CUSTOMER67 UNION ALL
SELECT * FROM CUSTOMER68 UNION ALL
SELECT * FROM CUSTOMER69 UNION ALL
SELECT * FROM CUSTOMER70 UNION ALL
SELECT * FROM CUSTOMER71 UNION ALL
SELECT * FROM CUSTOMER72 UNION ALL
SELECT * FROM CUSTOMER73 UNION ALL
SELECT * FROM CUSTOMER74 UNION ALL
SELECT * FROM CUSTOMER75 UNION ALL
SELECT * FROM CUSTOMER76 UNION ALL
SELECT * FROM CUSTOMER78 UNION ALL
SELECT * FROM CUSTOMER79 UNION ALL
SELECT * FROM CUSTOMER80 UNION ALL
SELECT * FROM CUSTOMER81 UNION ALL
SELECT * FROM CUSTOMER82 UNION ALL
SELECT * FROM CUSTOMER83 UNION ALL
SELECT * FROM CUSTOMER84 UNION ALL
SELECT * FROM CUSTOMER86 UNION ALL
SELECT * FROM CUSTOMER87 UNION ALL
SELECT * FROM CUSTOMER88 UNION ALL
SELECT * FROM CUSTOMER93 UNION ALL
SELECT * FROM CUSTOMER94 UNION ALL
SELECT * FROM CUSTOMER95 UNION ALL
SELECT * FROM CUSTOMER96 UNION ALL
SELECT * FROM CUSTOMER97 UNION ALL
SELECT * FROM CUSTOMER98 UNION ALL
SELECT * FROM CUSTOMER99 UNION ALL
SELECT * FROM CUSTOMER100 UNION ALL
SELECT * FROM CUSTOMER101 UNION ALL
SELECT * FROM CUSTOMER105 UNION ALL
SELECT * FROM CUSTOMER106 UNION ALL
SELECT * FROM CUSTOMER107 UNION ALL
SELECT * FROM CUSTOMER108 UNION ALL
SELECT * FROM CUSTOMER110 UNION ALL
SELECT * FROM CUSTOMER111 UNION ALL
SELECT * FROM CUSTOMER112 UNION ALL
SELECT * FROM CUSTOMER113 UNION ALL
SELECT * FROM CUSTOMER114 UNION ALL
SELECT * FROM CUSTOMER115 UNION ALL
SELECT * FROM CUSTOMER117 UNION ALL
SELECT * FROM CUSTOMER118 UNION ALL
SELECT * FROM CUSTOMER119 UNION ALL
SELECT * FROM CUSTOMER112 UNION ALL
SELECT * FROM CUSTOMER113 UNION ALL
SELECT * FROM CUSTOMER114 UNION ALL
SELECT * FROM CUSTOMER115 UNION ALL
SELECT * FROM CUSTOMER116 UNION ALL
SELECT * FROM CUSTOMER117 UNION ALL
SELECT * FROM CUSTOMER118 UNION ALL
SELECT * FROM CUSTOMER119 UNION ALL
SELECT * FROM CUSTOMER120 WITH ROW MOVEMENT;
commit reset;

DDL/CRVW_CUSTOMER.ddl
connect to TPCC in share mode;
 DROP VIEW Customer;
 CREATE VIEW Customer
 (C_ID,
 C_NAME,
 C_ADDR,
 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

DDL/CRVW_DISTRICT.ddl
connect to TPCC in share mode;
 DROP VIEW District;
 CREATE VIEW District
 (D_NEXT_O_ID,
 D_TAX,
 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 UNION ALL
SELECT * FROM DISTRICT33 UNION ALL
SELECT * FROM DISTRICT34 UNION ALL
SELECT * FROM DISTRICT35 UNION ALL
SELECT * FROM DISTRICT36 UNION ALL
SELECT * FROM DISTRICT37 UNION ALL
SELECT * FROM DISTRICT38 UNION ALL
SELECT * FROM DISTRICT39 UNION ALL
SELECT * FROM DISTRICT40 WITH ROW MOVEMENT;
commit reset;

DDL/CRVW_HISTORY.ddl
connect to TPCC in share mode;
 DROP VIEW History;
 CREATE VIEW History;

TPC Benchmark™ C Full Disclosure Report - IBM System p 570 Model 9117-MMA
<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CL_AMOUNT, CL_I_ID, CL_SUPPLY_W_ID, CL_QUANTITY, CL_DIST_INFO, CL_O_ID, CL_D_ID, CL_W_ID, CL_NUMBER</td>
<td></td>
</tr>
<tr>
<td>V AS SELECT * FROM ORDER_LINE1 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE2 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE3 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE4 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE5 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE6 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE7 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE8 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE9 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE10 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE11 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE12 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE13 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE14 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE15 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE16 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE17 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE18 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE19 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE20 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE21 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE22 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE23 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE24 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE25 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE26 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE27 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE28 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE29 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE30 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE31 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE32 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE33 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE34 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE35 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE36 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE37 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE38 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE39 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE40 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE41 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE42 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE43 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE44 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE45 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE46 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE47 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE48 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE49 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE50 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE51 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE52 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE53 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE54 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE55 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE56 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE57 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE58 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE59 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE60 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE61 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE62 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE63 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE64 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE65 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE66 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE67 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE68 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE69 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE70 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE71 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE72 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE73 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE74 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE75 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE76 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE77 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE78 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE79 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE80 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE81 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE82 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE83 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE84 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE85 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE86 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE87 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE88 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE89 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE90 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE91 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE92 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE93 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE94 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE95 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE96 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE97 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE98 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE99 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE100 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE101 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE102 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE103 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE104 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE105 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE106 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE107 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE108 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE109 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE110 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE111 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE112 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE113 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE114 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE115 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE116 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE117 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE118 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE119 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>SELECT * FROM ORDER_LINE120 UNION ALL</td>
<td></td>
</tr>
<tr>
<td>WITH ROW MOVEMENT; COMMIT WORK;</td>
<td></td>
</tr>
<tr>
<td>connect reset;</td>
<td></td>
</tr>
</tbody>
</table>

**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) WITH ROW MOVEMENT; COMMIT WORK; connect reset;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER 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
31980000 INSERT INTO CUSTOMER10;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER 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
31980000 INSERT INTO CUSTOMER11;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER 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
31980000 INSERT INTO CUSTOMER12;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER 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
31980000 INSERT INTO CUSTOMER13;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER 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
31980000 INSERT INTO CUSTOMER14;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER 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
31980000 INSERT INTO CUSTOMER15;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER 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
31980000 INSERT INTO CUSTOMER16;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER 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
31980000 INSERT INTO CUSTOMER17;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER 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
31980000 INSERT INTO CUSTOMER18;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER 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
31980000 INSERT INTO CUSTOMER19;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER 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
31980000 INSERT INTO CUSTOMER20;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER 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
31980000 INSERT INTO CUSTOMER21;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER 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
31980000 INSERT INTO CUSTOMER22;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER 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
31980000 INSERT INTO CUSTOMER23;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER 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
31980000 INSERT INTO CUSTOMER24;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER 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
31980000 INSERT INTO CUSTOMER25;
IMPORT FROM /flats/F1_064/customer_064_1.dat OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPACT=50 COMMITCOUNT 31980000 INSERT INTO CUSTOMER65;
COMMIT WORK;
CONNECT RESET;
CONNECTION TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER65 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_065/customer_065_1.dat OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPACT=50 COMMITCOUNT 31980000 INSERT INTO CUSTOMER65;
COMMIT WORK;
CONNECT RESET;
CONNECTION TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER65 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_067/customer_067_1.dat OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPACT=50 COMMITCOUNT 31980000 INSERT INTO CUSTOMER67;
COMMIT WORK;
CONNECT RESET;
CONNECTION TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER67 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_068/customer_068_1.dat OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPACT=50 COMMITCOUNT 31980000 INSERT INTO CUSTOMER68;
COMMIT WORK;
CONNECT RESET;
CONNECTION TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER68 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_070/customer_070_1.dat OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPACT=50 COMMITCOUNT 31980000 INSERT INTO CUSTOMER70;
COMMIT WORK;
CONNECT RESET;
CONNECTION TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER70 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_071/customer_071_1.dat OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPACT=50 COMMITCOUNT 31980000 INSERT INTO CUSTOMER71;
COMMIT WORK;
CONNECT RESET;
CONNECTION TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER71 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_072/customer_072_1.dat OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPACT=50 COMMITCOUNT 31980000 INSERT INTO CUSTOMER72;
COMMIT WORK;
CONNECT RESET;
CONNECTION TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER72 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_073/customer_073_1.dat OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPACT=50 COMMITCOUNT 31980000 INSERT INTO CUSTOMER73;
COMMIT WORK;
CONNECT RESET;
CONNECTION TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER73 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_074/customer_074_1.dat OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPACT=50 COMMITCOUNT 31980000 INSERT INTO CUSTOMER74;
COMMIT WORK;
CONNECT RESET;
CONNECTION TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER74 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_075/customer_075_1.dat OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPACT=50 COMMITCOUNT 31980000 INSERT INTO CUSTOMER75;
COMMIT WORK;
CONNECT RESET;
CONNECTION TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER75 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_076/customer_076_1.dat OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPACT=50 COMMITCOUNT 31980000 INSERT INTO CUSTOMER76;
COMMIT WORK;
CONNECT RESET;
CONNECTION TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER76 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_077/customer_077_1.dat OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPACT=50 COMMITCOUNT 31980000 INSERT INTO CUSTOMER77;
COMMIT WORK;
CONNECT RESET;
CONNECTION TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER77 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_078/customer_078_1.dat OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPACT=50 COMMITCOUNT 31980000 INSERT INTO CUSTOMER78;
COMMIT WORK;
CONNECT RESET;
CONNECTION TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER78 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_079/customer_079_1.dat OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPACT=50 COMMITCOUNT 31980000 INSERT INTO CUSTOMER79;
COMMIT WORK;
CONNECT RESET;
CONNECTION TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER79 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_080/customer_080_1.dat OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPACT=50 COMMITCOUNT 31980000 INSERT INTO CUSTOMER80;
COMMIT WORK;
CONNECT RESET;
CONNECTION TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER80 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_081/customer_081_1.dat OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPACT=50 COMMITCOUNT 31980000 INSERT INTO CUSTOMER81;
COMMIT WORK;
CONNECT RESET;
CONNECTION TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER81 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_082/customer_082_1.dat OF DEL MODIFIED BY COLDEL| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPACT=50 COMMITCOUNT 31980000 INSERT INTO CUSTOMER82;
COMMIT WORK;
CONNECT RESET;
CONNECTION TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER82 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_100/orders_100_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPONDB=50 COMMITCOUNT
31980000 INSERT INTO ORDERS100;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS100 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_100/orders_100_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPONDB=50 COMMITCOUNT
31980000 INSERT INTO ORDERS100;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS100 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_100/orders_100_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPONDB=50 COMMITCOUNT
31980000 INSERT INTO ORDERS100;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS100 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_100/orders_100_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPONDB=50 COMMITCOUNT
31980000 INSERT INTO ORDERS100;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS100 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_100/orders_100_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPONDB=50 COMMITCOUNT
31980000 INSERT INTO ORDERS100;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS100 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_100/orders_100_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPONDB=50 COMMITCOUNT
31980000 INSERT INTO ORDERS100;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS100 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_100/orders_100_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPONDB=50 COMMITCOUNT
31980000 INSERT INTO ORDERS100;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS100 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_100/orders_100_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPONDB=50 COMMITCOUNT
31980000 INSERT INTO ORDERS100;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS100 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_100/orders_100_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPONDB=50 COMMITCOUNT
31980000 INSERT INTO ORDERS100;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS100 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_100/orders_100_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPONDB=50 COMMITCOUNT
31980000 INSERT INTO ORDERS100;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS100 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_100/orders_100_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPONDB=50 COMMITCOUNT
31980000 INSERT INTO ORDERS100;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS100 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_100/orders_100_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPONDB=50 COMMITCOUNT
31980000 INSERT INTO ORDERS100;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS100 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_100/orders_100_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPONDB=50 COMMITCOUNT
31980000 INSERT INTO ORDERS100;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS100 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_100/orders_100_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPONDB=50 COMMITCOUNT
31980000 INSERT INTO ORDERS100;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS100 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_100/orders_100_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPONDB=50 COMMITCOUNT
31980000 INSERT INTO ORDERS100;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS100 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_100/orders_100_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPONDB=50 COMMITCOUNT
31980000 INSERT INTO ORDERS100;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS100 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_100/orders_100_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPONDB=50 COMMITCOUNT
31980000 INSERT INTO ORDERS100;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS100 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_100/orders_100_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPONDB=50 COMMITCOUNT
31980000 INSERT INTO ORDERS100;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS100 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_100/orders_100_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPONDB=50 COMMITCOUNT
31980000 INSERT INTO ORDERS100;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS100 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM /flats/F1_100/orders_100_1.dat OF DEL MODIFIED BY COLDEL|
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPONDB=50 COMMITCOUNT
31980000 INSERT INTO ORDERS100;
COMMIT WORK; CONNECT TO TPCC IN SHARE MODE; IMPORT FROM flata_F1_01/Warehouse_01.dat OF DEL MODIFIED BY DELMOD|
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 flata_F1_02/Warehouse_02.dat OF DEL MODIFIED BY DELMOD| 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 flata_F1_03/Warehouse_03.dat OF DEL MODIFIED BY DELMOD| 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 flata_F1_04/Warehouse_04.dat OF DEL MODIFIED BY DELMOD| 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 flata_F1_05/Warehouse_05.dat OF DEL MODIFIED BY DELMOD| 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 flata_F1_06/Warehouse_06.dat OF DEL MODIFIED BY DELMOD| 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 flata_F1_07/Warehouse_07.dat OF DEL MODIFIED BY DELMOD| 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 flata_F1_08/Warehouse_08.dat OF DEL MODIFIED BY DELMOD| 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 flata_F1_09/Warehouse_09.dat OF DEL MODIFIED BY DELMOD| 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 flata_F1_10/Warehouse_10.dat OF DEL MODIFIED BY DELMOD| 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 flata_F1_11/Warehouse_11.dat OF DEL MODIFIED BY DELMOD| 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 flata_F1_12/Warehouse_12.dat OF DEL MODIFIED BY DELMOD| TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE12; COMMIT WORK;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_094/warehouse_094_1.dat OF DEL MODIFIED BY COLDEL
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPRESS=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE37;
COMMIT WORK;
CONNECT RESSET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_095/warehouse_095_1.dat OF DEL MODIFIED BY COLDEL
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPRESS=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE38;
COMMIT WORK;
CONNECT RESSET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_096/warehouse_096_1.dat OF DEL MODIFIED BY COLDEL
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPRESS=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE39;
COMMIT WORK;
CONNECT RESSET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM /flats/F1_097/warehouse_097_1.dat OF DEL MODIFIED BY COLDEL
TIMESTAMPFORMAT="YYYY-MM-DD HH:MM:SS" KEEPBLANKS COMPRESS=50 ALLOW WRITE
ACCESS COMMITCOUNT 1000 INSERT INTO WAREHOUSE40;
COMMIT WORK;
CONNECT RESSET;

DDL/RNST_CUSTOMER.ddl

connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER1 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER2 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER3 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER4 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER5 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER6 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER7 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER8 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER9 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER10 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER11 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER12 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER13 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER14 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER15 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER16 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER17 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER18 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER19 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER20 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER21 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER22 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER23 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER24 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER25 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER26 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER27 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER28 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER29 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER30 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER31 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER32 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER33 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER34 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER35 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER36 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER37 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER38 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER39 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER40 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER41 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER42 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER43 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER44 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER45 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER46 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER47 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcCUSTOMER48 AND INDEXES ALL;
<table>
<thead>
<tr>
<th>Table Name</th>
<th>Indexes</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>CUSTOMER103</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>CUSTOMER104</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT1</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT2</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT3</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT4</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT5</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT6</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT7</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT8</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT9</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT10</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT11</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT12</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT13</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT14</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT15</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT16</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT17</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT18</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT19</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT20</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT21</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT22</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT23</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT24</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT25</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT26</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT27</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT28</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT29</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT30</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT31</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT32</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT33</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT34</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT35</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
<tr>
<td>DISTRICT36</td>
<td>ALL</td>
<td>RUNSTATS</td>
</tr>
</tbody>
</table>

**DDL/RNST DISTRICT.ddl**

```sql
CONNECT TO TPCC IN SHARE MODE;
RUNSTATS ON TABLE tpcc.DISTRICT1 AND INDEXES ALL;
CONNECT TO TPCC IN SHARE MODE;
RUNSTATS ON TABLE tpcc.DISTRICT2 AND INDEXES ALL;
CONNECT TO TPCC IN SHARE MODE;
RUNSTATS ON TABLE tpcc.DISTRICT3 AND INDEXES ALL;
CONNECT TO TPCC IN SHARE MODE;
RUNSTATS ON TABLE tpcc.DISTRICT4 AND INDEXES ALL;
CONNECT TO TPCC IN SHARE MODE;
RUNSTATS ON TABLE tpcc.DISTRICT5 AND INDEXES ALL;
CONNECT TO TPCC IN SHARE MODE;
RUNSTATS ON TABLE tpcc.DISTRICT6 AND INDEXES ALL;
CONNECT TO TPCC IN SHARE MODE;
RUNSTATS ON TABLE tpcc.DISTRICT7 AND INDEXES ALL;
CONNECT TO TPCC IN SHARE MODE;
RUNSTATS ON TABLE tpcc.DISTRICT8 AND INDEXES ALL;
CONNECT TO TPCC IN SHARE MODE;
RUNSTATS ON TABLE tpcc.DISTRICT9 AND INDEXES ALL;
CONNECT TO TPCC IN SHARE MODE;
RUNSTATS ON TABLE tpcc.DISTRICT10 AND INDEXES ALL;
CONNECT TO TPCC IN SHARE MODE;
RUNSTATS ON TABLE tpcc.DISTRICT11 AND INDEXES ALL;
CONNECT TO TPCC IN SHARE MODE;
RUNSTATS ON TABLE tpcc.DISTRICT12 AND INDEXES ALL;
CONNECT TO TPCC IN SHARE MODE;
RUNSTATS ON TABLE tpcc.DISTRICT13 AND INDEXES ALL;
CONNECT TO TPCC IN SHARE MODE;
RUNSTATS ON TABLE tpcc.DISTRICT14 AND INDEXES ALL;
CONNECT TO TPCC IN SHARE MODE;
RUNSTATS ON TABLE tpcc.DISTRICT15 AND INDEXES ALL;
CONNECT TO TPCC IN SHARE MODE;
RUNSTATS ON TABLE tpcc.DISTRICT16 AND INDEXES ALL;
CONNECT TO TPCC IN SHARE MODE;
RUNSTATS ON TABLE tpcc.DISTRICT17 AND INDEXES ALL;
CONNECT TO TPCC IN SHARE MODE;
RUNSTATS ON TABLE tpcc.DISTRICT18 AND INDEXES ALL;
```
RUNSTATS ON TABLE tpc:NEW_ORDERA25 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERA26 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERA27 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERA28 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERA29 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERA30 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERA31 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERA32 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERA33 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERA34 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERA35 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERA36 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERA37 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERA38 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERA39 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERA40 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERB2 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERB3 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERB4 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERB5 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERB6 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERB7 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERB8 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERB9 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERB10 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERB11 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERB12 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERB13 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERB14 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERB15 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERB16 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERB17 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERB18 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERB19 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode; 
RUNSTATS ON TABLE tpc:NEW_ORDERB20 AND INDEXES ALL; COMMIT WORK; 
connect reset; 
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS52 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS53 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS54 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS55 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS56 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS57 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS58 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS59 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS60 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS61 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS62 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS63 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS64 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS65 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS66 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS67 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS68 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS69 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS70 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS71 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS72 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS73 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS74 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS75 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS76 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS77 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS78 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS79 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS80 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS81 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS82 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS83 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS84 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS85 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS86 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS87 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS88 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS89 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS90 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS91 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS92 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS93 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS94 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS95 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS96 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS97 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS98 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS99 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS100 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS101 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS102 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS103 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS104 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS105 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS106 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS107 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS108 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS109 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS110 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS111 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS112 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode; RUNSTATS ON TABLE tpcc.ORDERS113 AND INDEXES ALL; COMMIT WORK; connect reset; connect to TPCC in share mode;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS89 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS90 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS91 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS92 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS93 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS94 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS95 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS96 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS97 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS98 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS99 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS100 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS101 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS102 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS103 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS104 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS105 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS106 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS107 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS108 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS109 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS110 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS111 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS112 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS113 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS114 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS115 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS116 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS117 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS118 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS119 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS120 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS121 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS122 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS123 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS124 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS125 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS126 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS127 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS128 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS129 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS130 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS131 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS132 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS133 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS134 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS135 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS136 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS137 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS138 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS139 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS140 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS141 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS142 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS143 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS144 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS145 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS146 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS147 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS148 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS149 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE tpcc.ORDERS150 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK11 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK12 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK13 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK14 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK15 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK16 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK17 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK18 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK19 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK20 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK21 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK22 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK23 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK24 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK25 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK26 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK27 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK28 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK29 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK30 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK31 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK32 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK33 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK34 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK35 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK36 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK37 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK38 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK39 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK40 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK41 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK42 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK43 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK44 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK45 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK46 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK47 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK48 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK49 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK50 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK51 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK52 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK53 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK54 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK55 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK56 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK57 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK58 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK59 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK60 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK61 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK62 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK63 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;  
connect to TPCC in share mode;  
RUNSTATS ON TABLE tpcc.STOCK64 AND INDEXES ALL;  
COMMIT WORK;  
connect reset;
ALTER TABLESPACE is_customer_046 BUFFERPOOL CST4;
ALTER TABLESPACE is_customer_047 BUFFERPOOL CST4;
ALTER TABLESPACE is_customer_048 BUFFERPOOL CST4;
ALTER TABLESPACE is_customer_049 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_050 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_051 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_052 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_053 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_054 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_055 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_056 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_057 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_058 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_059 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_060 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_061 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_062 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_063 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_064 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_065 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_066 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_067 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_068 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_069 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_070 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_071 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_072 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_073 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_074 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_075 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_076 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_077 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_078 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_079 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_080 BUFFERPOOL CST5;
ALTER TABLESPACE is_customer_081 BufferPool CST5;
ALTER TABLESPACE is_customer_082 BufferPool CST5;
ALTER TABLESPACE is_customer_083 BufferPool CST5;
ALTER TABLESPACE is_customer_084 BufferPool CST5;
ALTER TABLESPACE is_customer_085 BufferPool CST5;
ALTER TABLESPACE is_customer_086 BufferPool CST5;
ALTER TABLESPACE is_customer_087 BufferPool CST5;
ALTER TABLESPACE is_customer_088 BufferPool CST5;
ALTER TABLESPACE is_customer_089 BufferPool CST5;
ALTER TABLESPACE is_customer_090 BufferPool CST5;
ALTER TABLESPACE is_customer_091 BufferPool CST5;
ALTER TABLESPACE is_customer_092 BufferPool CST5;
ALTER TABLESPACE is_customer_093 BufferPool CST5;
ALTER TABLESPACE is_customer_094 BufferPool CST5;
ALTER TABLESPACE is_customer_095 BufferPool CST5;
ALTER TABLESPACE is_customer_096 BufferPool CST5;
ALTER TABLESPACE is_customer_097 BufferPool CST5;
ALTER TABLESPACE is_customer_098 BufferPool CST5;
ALTER TABLESPACE is_customer_099 BufferPool CST5;
ALTER TABLESPACE is_customer_100 BufferPool CST5;
ALTER TABLESPACE is_customer_101 BufferPool CST5;
ALTER TABLESPACE is_customer_102 BufferPool CST5;
ALTER TABLESPACE is_customer_103 BufferPool CST5;
ALTER TABLESPACE is_customer_104 BufferPool CST5;
ALTER TABLESPACE is_customer_105 BufferPool CST5;
ALTER TABLESPACE is_customer_106 BufferPool CST5;
ALTER TABLESPACE is_customer_107 BufferPool CST5;
ALTER TABLESPACE is_customer_108 BufferPool CST5;
ALTER TABLESPACE is_customer_109 BufferPool CST5;
ALTER TABLESPACE is_customer_110 BufferPool CST5;
ALTER TABLESPACE is_customer_111 BufferPool CST5;
ALTER TABLESPACE is_customer_112 BufferPool CST5;
ALTER TABLESPACE is_customer_113 BufferPool CST5;
ALTER TABLESPACE is_customer_114 BufferPool CST5;
ALTER TABLESPACE is_customer_115 BufferPool CST5;
ALTER TABLESPACE is_customer_116 BufferPool CST5;
ALTER TABLESPACE is_customer_117 BufferPool CST5;
ALTER TABLESPACE is_customer_118 BufferPool CST5;
ALTER TABLESPACE is_customer_119 BufferPool CST5;
ALTER TABLESPACE is_customer_120 BufferPool CST5;
ALTER TABLESPACE is_customer_121 BufferPool CST5;
ALTER TABLESPACE is_customer_122 BufferPool CST5;
ALTER TABLESPACE ts_order_063 BUFFERPOOL ORD6;
ALTER TABLESPACE ts_order_075 BUFFERPOOL ORD7;
ALTER TABLESPACE ts_order_079 BUFFERPOOL ORD7;
ALTER TABLESPACE ts_order_080 BUFFERPOOL ORD7; 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_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_023 BUFFERPOOL ORD_I2;
ALTER TABLESPACE is_order_024 BUFFERPOOL ORD_I2; ALTER TABLESPACE is_order_025 BUFFERPOOL ORD_I3;
ALTER TABLESPACE is_order_030 BUFFERPOOL ORD_I3;
ALTER TABLESPACE ts_order_030 BUFFERPOOL ORD_I3;
ALTER TABLESPACE ts_orderline_058 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_073 BUFFERPOOL OLN7;
ALTER TABLESPACE ts_orderline_077 BUFFERPOOL OLN7; ALTER TABLESPACE ts_orderline_078 BUFFERPOOL OLN7; ALTER TABLESPACE ts_orderline_080 BUFFERPOOL OLN7;
ALTER TABLESPACE ts_orderline_095 BUFFERPOOL ORD8;
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_030 BUFFERPOOL ORD_I3;
ALTER TABLESPACE ts_orderline_058 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_073 BUFFERPOOL OLN7;
ALTER TABLESPACE ts_orderline_077 BUFFERPOOL OLN7; ALTER TABLESPACE ts_orderline_078 BUFFERPOOL OLN7; ALTER TABLESPACE ts_orderline_080 BUFFERPOOL OLN7;
ALTER TABLESPACE ts_orderline_095 BUFFERPOOL ORD8;
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_030 BUFFERPOOL ORD_I3;
ALTER TABLESPACE ts_orderline_058 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_073 BUFFERPOOL OLN7;
ALTER TABLESPACE ts_orderline_077 BUFFERPOOL OLN7; ALTER TABLESPACE ts_orderline_078 BUFFERPOOL OLN7; ALTER TABLESPACE ts_orderline_080 BUFFERPOOL OLN7;
ALTER TABLESPACE ts_orderline_095 BUFFERPOOL ORD8;
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_030 BUFFERPOOL ORD_I3;
ALTER TABLESPACE ts_orderline_058 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_073 BUFFERPOOL OLN7;
ALTER TABLESPACE ts_orderline_077 BUFFERPOOL OLN7; ALTER TABLESPACE ts_orderline_078 BUFFERPOOL OLN7; ALTER TABLESPACE ts_orderline_080 BUFFERPOOL OLN7;
ALTER TABLESPACE ts_orderline_095 BUFFERPOOL ORD8;
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_030 BUFFERPOOL ORD_I3;
ALTER TABLESPACE ts_orderline_058 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_073 BUFFERPOOL OLN7;
ALTER TABLESPACE ts_orderline_077 BUFFERPOOL OLN7; ALTER TABLESPACE ts_orderline_078 BUFFERPOOL OLN7; ALTER TABLESPACE ts_orderline_080 BUFFERPOOL OLN7;
ALTER TABLESPACE ts_orderline_095 BUFFERPOOL ORD8;
-- 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/rD1F01V6CSTI' 222720     )
  EXENTSIZE 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     )
  EXENTSIZE 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     )
  EXENTSIZE 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     )
  EXENTSIZE 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     )
  EXENTSIZE 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     )
  EXENTSIZE 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     )
  EXENTSIZE 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     )
  EXENTSIZE 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     )
  EXENTSIZE 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     )
  EXENTSIZE 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     )
  EXENTSIZE 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     )
  EXENTSIZE 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     )
  EXENTSIZE 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/rD1F05V4CSTI' 222720     )
  EXENTSIZE 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     )
  EXENTSIZE 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     )
  EXENTSIZE 256
  PREFETCHSIZE 4096
  BUFFERPOOL ibmdefaultbp8K;
COMMIT;
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/rdsk/c0t0d0s1' 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/rdsk/c0t0d0s2' 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/rdsk/c0t0d0s3' 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/rdsk/c0t0d0s4' 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/rdsk/c0t0d0s5' 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/rdsk/c0t0d0s6' 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/rdsk/c0t0d0s7' 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/rdsk/c0t0d0s8' 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/rdsk/c0t0d0s9' 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/rdsk/c0t0d0s10' 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
preferedsize 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_052 of D1

drop tablespace is_customer_052;
create regular tablespace is_customer_052 pagesize 8K
managed by database
using
  (  
    device '/dev/rdsk/c1t1d0s0' 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/rdsk/c1t1d0s1' 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/rdsk/c1t1d0s2' 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/rdsk/c1t1d0s3' 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/rdsk/c1t1d0s4' 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/rdsk/c1t1d0s5' 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/rdsk/c1t1d0s6' 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/rdsk/c1t1d0s7' 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/rdsk/c1t1d0s8' 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/rdsk/c1t1d0s9' 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/rdsk/c1t1d0s10' 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/rdsk/c1t1d0s11' 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/rdsk/c1t1d0s12' 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/rdsk/c1t1d0s13' 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/rdsk/c1t1d0s14' 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/rdsk/c1t1d0s15' 222720  
  )
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;
-- now creating TS for is_customer_068 of D1
create regular tablespace is_customer_068 pagesize 8K
managed by database using
  (    device '/dev/rD1F12V2CSTI' 222720
     )
exentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_069 of D1
create regular tablespace is_customer_069 pagesize 8K
managed by database using
  (    device '/dev/rD1F12V3CSTI' 222720    )
exentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_070 of D1
create regular tablespace is_customer_070 pagesize 8K
managed by database using
  (    device '/dev/rD1F12V4CSTI' 222720
     )
exentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_071 of D1
create regular tablespace is_customer_071 pagesize 8K
managed by database using
  (    device '/dev/rD1F12V5CSTI' 222720
     )
exentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_072 of D1
create regular tablespace is_customer_072 pagesize 8K
managed by database using
  (    device '/dev/rD1F12V6CSTI' 222720
     )
exentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_073 of D1
create regular tablespace is_customer_073 pagesize 8K
managed by database using
  (    device '/dev/rD1F13V1CSTI' 222720
     )
exentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_074 of D1
create regular tablespace is_customer_074 pagesize 8K
managed by database using
  (    device '/dev/rD1F13V2CSTI' 222720
     )
exentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_075 of D1
create regular tablespace is_customer_075 pagesize 8K
managed by database using
  (    device '/dev/rD1F13V3CSTI' 222720
     )
exentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_076 of D1
create regular tablespace is_customer_076 pagesize 8K
managed by database using
  (    device '/dev/rD1F13V4CSTI' 222720    )
exentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_077 of D1
create regular tablespace is_customer_077 pagesize 8K
managed by database using
  (    device '/dev/rD1F14V1CSTI' 222720
     )
exentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_078 of D1
create regular tablespace is_customer_078 pagesize 8K
managed by database using
  (    device '/dev/rD1F14V2CSTI' 222720    )
exentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_079 of D1
create regular tablespace is_customer_079 pagesize 8K
managed by database using
  (    device '/dev/rD1F14V3CSTI' 222720    )
exentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_080 of D1
create regular tablespace is_customer_080 pagesize 8K
managed by database using
  (    device '/dev/rD1F14V4CSTI' 222720    )
exentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_081 of D1
create regular tablespace is_customer_081 pagesize 8K
managed by database using
  (    device '/dev/rD1F14V5CSTI' 222720    )
exentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for is_customer_082 of D1
create regular tablespace is_customer_082 pagesize 8K
managed by database using
  (    device '/dev/rD1F14V6CSTI' 222720    )
exentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;
create regular tablespace is_customer_083 pagesize 8K
managed by database
using
{
    device '/dev/rD1F15V5CSTI' 222720
}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;
-- now creating TS for is_customer_084 of D1
create regular tablespace is_customer_084 pagesize 8K
managed by database
using
{
    device '/dev/rD1F15V6CSTI' 222720
}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;
-- now creating TS for is_customer_085 of D1
create regular tablespace is_customer_085 pagesize 8K
managed by database
using
{
    device '/dev/rD1F15V7CSTI' 222720
}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;
-- now creating TS for is_customer_086 of D1
create regular tablespace is_customer_086 pagesize 8K
managed by database
using
{
    device '/dev/rD1F15V8CSTI' 222720
}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;
-- now creating TS for is_customer_087 of D1
create regular tablespace is_customer_087 pagesize 8K
managed by database
using
{
    device '/dev/rD1F15V9CSTI' 222720
}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;
-- now creating TS for is_customer_088 of D1
create regular tablespace is_customer_088 pagesize 8K
managed by database
using
{
    device '/dev/rD1F15V10CSTI' 222720
}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;
-- now creating TS for is_customer_089 of D1
create regular tablespace is_customer_089 pagesize 8K
managed by database
using
{
    device '/dev/rD1F15V11CSTI' 222720
}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;
-- now creating TS for is_customer_090 of D1
create regular tablespace is_customer_090 pagesize 8K
managed by database
using
{
    device '/dev/rD1F15V12CSTI' 222720
}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;
-- now creating TS for is_customer_091 of D1
create regular tablespace is_customer_091 pagesize 8K
managed by database
using
{
    device '/dev/rD1F15V13CSTI' 222720
}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;
-- now creating TS for is_customer_092 of D1
create regular tablespace is_customer_092 pagesize 8K
managed by database
using
{
    device '/dev/rD1F15V14CSTI' 222720
}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;
-- now creating TS for is_customer_093 of D1
create regular tablespace is_customer_093 pagesize 8K
managed by database
using
{
    device '/dev/rD1F15V15CSTI' 222720
}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;
-- now creating TS for is_customer_094 of D1
create regular tablespace is_customer_094 pagesize 8K
managed by database
using
{
    device '/dev/rD1F15V16CSTI' 222720
}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;
-- now creating TS for is_customer_095 of D1
create regular tablespace is_customer_095 pagesize 8K
managed by database
using
{
    device '/dev/rD1F15V17CSTI' 222720
}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;
-- now creating TS for is_customer_096 of D1
create regular tablespace is_customer_096 pagesize 8K
managed by database
using
{
    device '/dev/rD1F15V18CSTI' 222720
}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;
-- now creating TS for is_customer_097 of D1
create regular tablespace is_customer_097 pagesize 8K
managed by database
using
{
    device '/dev/rD1F15V19CSTI' 222720
}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;
-- now creating TS for is_customer_098 of D1
create regular tablespace is_customer_098 pagesize 8K
managed by database
using
{
    device '/dev/rD1F15V20CSTI' 222720
}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;
-- now creating TS for is_customer_099 of D1
create regular tablespace is_customer_099 pagesize 8K
managed by database
using
{
    device '/dev/rD1F15V21CSTI' 222720
}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;
-- now creating TS for is_customer_100 of D1
create regular tablespace is_customer_100 pagesize 8K
managed by database
using
{
    device '/dev/rD1F15V22CSTI' 222720
}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;
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/0F17V6CSTI' 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/0F17V7CSTI' 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/0F17V8CSTI' 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/0F17V9CSTI' 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/0F17V10CSTI' 222720
}
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/0F17V11CSTI' 222720
}
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  
  )
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  
  )
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  
  )
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/rD1F02V5ORDI' 191744  
  )
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/rD1F02V6ORDI' 191744  
  )
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/rD1F03V1ORDI' 191744  
  )
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/rD1F03V2ORDI' 191744  
  )
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/rD1F03V3ORDI' 191744  
  )
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/rD1F03V4ORDI' 191744  
  )
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/rD1F03V5ORDI' 191744  
  )
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/rD1F04V1ORDI' 191744  
  )
bufferpool ibmdefaultbp8K;
create regular tablespace is_order_025 pagesize 8K
managed by database using
  (  device '/dev/rdsk/c0t0d0s17' 191744
  )
  extentsize 256
  preferredsize 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/rdsk/c0t0d0s18' 191744
  )
  extentsize 256
  preferredsize 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/rdsk/c0t0d0s19' 191744
  )
  extentsize 256
  preferredsize 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/rdsk/c0t0d0s20' 191744
  )
  extentsize 256
  preferredsize 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/rdsk/c0t0d0s21' 191744
  )
  extentsize 256
  preferredsize 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/rdsk/c0t0d0s22' 191744
  )
  extentsize 256
  preferredsize 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/rdsk/c0t0d0s23' 191744
  )
  extentsize 256
  preferredsize 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/rdsk/c0t0d0s24' 191744
  )
  extentsize 256
  preferredsize 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/rdsk/c0t0d0s25' 191744
  )
  extentsize 256
  preferredsize 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/rdsk/c0t0d0s26' 191744
  )
  extentsize 256
  preferredsize 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/rdsk/c0t0d0s27' 191744
  )
  extentsize 256
  preferredsize 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/rdsk/c0t0d0s28' 191744
  )
  extentsize 256
  preferredsize 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/rdsk/c0t0d0s29' 191744
  )
  extentsize 256
  preferredsize 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/rdsk/c0t0d0s30' 191744
  )
  extentsize 256
  preferredsize 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/rdsk/c0t0d0s31' 191744
  )
  extentsize 256
  preferredsize 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/rdsk/c0t0d0s32' 191744
  )
  extentsize 256
  preferredsize 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
   prefetched 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/rD1F10V7ORDI' 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/rD1F10V8ORDI' 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/rD1F10V9ORDI' 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/rD1F10V10ORDI' 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/rD1F10V11ORDI' 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/rD1F10V12ORDI' 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/rD1F10V2ORDI' 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;
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;
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/rD1F15V7ORDI' 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/rD1F15V8ORDI' 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/rD1F15V9ORDI' 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/rD1F15V10ORDI' 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/rD1F15V11ORDI' 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/rD1F15V12ORDI' 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/rD1F15V13ORDI' 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/rD1F15V14ORDI' 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/rD1F15V15ORDI' 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/rD1F15V16ORDI' 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/rD1F15V17ORDI' 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/rD1F15V18ORDI' 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/rD1F15V19ORDI' 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/rD1F18V1ORDI' 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/rD1F19V1ORDI' 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/rD1F19V6ORDI' 191744 )
   extentsize 256
   prefetchsize 4096
   bufferpool ibmdefaultbp8K;
   commit;
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;

-- 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/rD1F02V1CST' 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/rD1F02V2CST' 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/rD1F02V3CST' 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/rD1F02V4CST' 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/rD1F02V5CST' 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/rD1F02V6CST' 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/rD1F03V1CST' 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/rD1F03V2CST' 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/rD1F03V3CST' 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/rD1F03V4CST' 6717440 )
extentsize 256
prefetchsize 4096;
commit;
drop tablespace ts_customer_015;
create regular tablespace ts_customer_015 pagesize 4K managed by database using (device '/dev/01F03V3CST' 6717440)
extenze size 256; prefdhreshize 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/01F03V4CST' 6717440)
extenze size 256; prefdhreshize 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/01F03V5CST' 6717440)
extenze size 256; prefdhreshize 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/01F03V6CST' 6717440)
extenze size 256; prefdhreshize 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/01F04V1CST' 6717440)
extenze size 256; prefdhreshize 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/01F04V2CST' 6717440)
extenze size 256; prefdhreshize 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/01F04V3CST' 6717440)
extenze size 256; prefdhreshize 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/01F04V4CST' 6717440)
extenze size 256; prefdhreshize 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/01F04V5CST' 6717440)
extenze size 256; prefdhreshize 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/01F04V6CST' 6717440)
extenze size 256; prefdhreshize 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/01F05V1CST' 6717440)
extenze size 256; prefdhreshize 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/01F05V2CST' 6717440)
extenze size 256; prefdhreshize 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/01F05V3CST' 6717440)
extenze size 256; prefdhreshize 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/01F05V4CST' 6717440)
extenze size 256; prefdhreshize 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/01F05V5CST' 6717440)
extenze size 256; prefdhreshize 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/01F05V6CST' 6717440)
extenze size 256; prefdhreshize 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/01F06V1CST' 6717440)
extenze size 256; prefdhreshize 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/rd/f0/vcst' 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/rd/f0/vcst' 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/rd/f0/vcst' 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/rd/f0/vcst' 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/rd/f0/vcst' 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/rd/f0/vcst' 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/rd/f0/vcst' 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/rd/f0/vcst' 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/rd/f0/vcst' 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/rd/f0/vcst' 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/rd/f0/vcst' 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/rd/f0/vcst' 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/rd/f0/vcst' 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/rd/f0/vcst' 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/rd/f0/vcst' 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/rd/f0/vcst' 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/rd/f0/vcst' 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/rdsk/c0t0d0' 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/rdsk/c0t0d1' 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/rdsk/c0t0d2' 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/rdsk/c0t0d3' 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/rdsk/c0t0d4' 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/rdsk/c0t0d5' 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/rdsk/c0t0d6' 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/rdsk/c0t0d7' 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/rdsk/c0t0d8' 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/rdsk/c0t0d9' 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/rdsk/c0t0da' 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/rdsk/c0t0db' 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/rdsk/c0t0dc' 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/rdsk/c0t0dd' 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/rdsk/c0t0de' 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/rdsk/c0t0df' 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/rdsk/c0t0e0' 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/rdsk/c0t0e1' 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/rdsk/c0t0e2' 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/rdsk/c0t0e3' 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/rdsk/c0t0e4' 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/rdsk/c0t0e5' 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/01F11V6CST' 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/01F12V1CST' 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/01F12V2CST' 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/01F12V3CST' 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/01F12V4CST' 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/01F12V5CST' 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/01F12V6CST' 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/01F12V7CST' 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/01F12V8CST' 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/01F12V9CST' 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/01F12V10CST' 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/01F12V11CST' 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/01F12V12CST' 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/01F12V13CST' 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/01F12V14CST' 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/01F12V15CST' 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/01F12V16CST' 6717440  
  )  
extentsize 256 prefetchsize 4096; commit;
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;
using (device '/dev/rD1F17V3CST' 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/rD1F17V4CST' 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/rD1F17V5CST' 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/rD1F17V6CST' 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/rD1F17V7CST' 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/rD1F19V7CST' 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/rD1F19V8CST' 6717440)
  extentsize 256
  prefetchsize 4096;
create regular tablespace ts_customer_116 pagesize 4K managed by database using  
  (  
    device '/dev/01F020V2CST' 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/01F020V3CST' 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/01F020V4CST' 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/01F020V5CST' 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/01F020V6CST' 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/01F01V2DIST' 512,  
    device '/dev/01F01V3DIST' 512,  
    device '/dev/01F01V4DIST' 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/01F01V5DIST' 512,  
    device '/dev/01F01V6DIST' 512,  
    device '/dev/01F01V7DIST' 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/01F02V1DIST' 512,  
    device '/dev/01F02V2DIST' 512,  
    device '/dev/01F02V3DIST' 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/01F02V4DIST' 512,  
    device '/dev/01F02V5DIST' 512,  
    device '/dev/01F02V6DIST' 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/01F03V1DIST' 512,  
    device '/dev/01F03V2DIST' 512,  
    device '/dev/01F03V3DIST' 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/01F03V4DIST' 512,  
    device '/dev/01F03V5DIST' 512,  
    device '/dev/01F03V6DIST' 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/01F04V1DIST' 512,  
    device '/dev/01F04V2DIST' 512,  
    device '/dev/01F04V3DIST' 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/01F04V4DIST' 512,  
    device '/dev/01F04V5DIST' 512,  
    device '/dev/01F04V6DIST' 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/01F05V1DIST' 512,  
    device '/dev/01F05V2DIST' 512,  
    device '/dev/01F05V3DIST' 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/01F05V4DIST' 512,  
    device '/dev/01F05V5DIST' 512,  
    device '/dev/01F05V6DIST' 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 )
exentsize 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 )
exentsize 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 )
exentsize 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 )
exentsize 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 )
exentsize 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 )
exentsize 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 )
exentsize 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 )
exentsize 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 )
exentsize 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 )
exentsize 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 )
exentsize 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 )
exentsize 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 )
exentsize 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 )
exentsize 64
prefetchsize 4096;
commit;
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 tpc;
-- 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/rD1F02V1HIST' 160768,
    device '/dev/rD1F02V2HIST' 160768,
    device '/dev/rD1F02V3HIST' 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/rD1F03V1HIST' 160768,
    device '/dev/rD1F03V2HIST' 160768,
    device '/dev/rD1F03V3HIST' 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/rD1F04V1HIST' 160768,
    device '/dev/rD1F04V2HIST' 160768,
    device '/dev/rD1F04V3HIST' 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/rD1F05V1HIST' 160768,
    device '/dev/rD1F05V2HIST' 160768,
    device '/dev/rD1F05V3HIST' 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/rD1F06V1HIST' 160768,
    device '/dev/rD1F06V2HIST' 160768,
    device '/dev/rD1F06V3HIST' 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/rD1F07V1HIST' 160768,
    device '/dev/rD1F07V2HIST' 160768,
    device '/dev/rD1F07V3HIST' 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/rD1F08V1HIST' 160768,
    device '/dev/rD1F08V2HIST' 160768,
    device '/dev/rD1F08V3HIST' 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/rD1F09V1HIST' 160768,
    device '/dev/rD1F09V2HIST' 160768,
    device '/dev/rD1F09V3HIST' 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
    (    /dev/rD1F07V1HIST 160768,
    /dev/rD1F07V2HIST 160768,
    /dev/rD1F07V3HIST 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
    (    /dev/rD1F08V1HIST 160768,
    /dev/rD1F08V2HIST 160768,
    /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
    (    /dev/rD1F09V1HIST 160768,
    /dev/rD1F09V2HIST 160768,
    /dev/rD1F09V3HIST 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
    (    /dev/rD1F10V1HIST 160768,
    /dev/rD1F10V2HIST 160768,
    /dev/rD1F10V3HIST 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
    (    /dev/rD1F11V1HIST 160768,
    /dev/rD1F11V2HIST 160768,
    /dev/rD1F11V3HIST 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
    (    /dev/rD1F12V1HIST 160768,
    /dev/rD1F12V2HIST 160768,
    /dev/rD1F12V3HIST 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
    (    /dev/rD1F13V1HIST 160768,
    /dev/rD1F13V2HIST 160768,
    /dev/rD1F13V3HIST 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
    (    /dev/rD1F14V1HIST 160768,
    /dev/rD1F14V2HIST 160768,
    /dev/rD1F14V3HIST 160768    )
EXTENTSIZE 256
PREFETCHSIZE 4096
BUFFERPOOL ibmdefaultbp16K;
COMMIT;
drop tablespace ts_history_027;
create regular tablespace ts_history_027 pagesize 16K
managed by database
using 
(  
device '/dev/01F16V4HIST' 160768,  
device '/dev/01F16V5HIST' 160768,  
device '/dev/01F16V6HIST' 160768  
);  
extensize 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/01F16V4HIST' 160768,  
device '/dev/01F16V5HIST' 160768,  
device '/dev/01F16V6HIST' 160768  
);  
extensize 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/01F17V1HIST' 160768,  
device '/dev/01F17V2HIST' 160768,  
device '/dev/01F17V3HIST' 160768  
);  
extensize 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/01F17V4HIST' 160768,  
device '/dev/01F17V5HIST' 160768,  
device '/dev/01F17V6HIST' 160768  
);  
extensize 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/01F18V1HIST' 160768,  
device '/dev/01F18V2HIST' 160768,  
device '/dev/01F18V3HIST' 160768  
);  
extensize 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/01F18V4HIST' 160768,  
device '/dev/01F18V5HIST' 160768,  
device '/dev/01F18V6HIST' 160768  
);  
extensize 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/01F19V1HIST' 160768,  
device '/dev/01F19V2HIST' 160768,  
device '/dev/01F19V3HIST' 160768  
);  
extensize 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/01F19V4HIST' 160768,  
device '/dev/01F19V5HIST' 160768,  
device '/dev/01F19V6HIST' 160768  
);  
extensize 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/01F20V1HIST' 160768,  
device '/dev/01F20V2HIST' 160768,  
device '/dev/01F20V3HIST' 160768  
);  
extensize 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/01F20V4HIST' 160768,  
device '/dev/01F20V5HIST' 160768,  
device '/dev/01F20V6HIST' 160768  
);  
extensize 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/01F21V1HIST' 160768,  
device '/dev/01F21V2HIST' 160768,  
device '/dev/01F21V3HIST' 160768  
);  
extensize 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/01F21V4HIST' 160768,  
device '/dev/01F21V5HIST' 160768,  
device '/dev/01F21V6HIST' 160768  
);  
extensize 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/01F22V1HIST' 160768,  
device '/dev/01F22V2HIST' 160768,  
device '/dev/01F22V3HIST' 160768  
);  
extensize 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/01F22V4HIST' 160768,  
device '/dev/01F22V5HIST' 160768,  
device '/dev/01F22V6HIST' 160768  
);  
extensize 256  
prefetchsize 4096  
bufferpool ibmdefaultbp16K;  
commit;
ts/crts_item.ddl

-- 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

-- now creating TS for ts_item_002 of D1
drop tablespace ts_item_002;
cREATE Regular tablespace ts_item_002 pagesize 4K managed by database

-- now creating TS for ts_item_003 of D1
drop tablespace ts_item_003;
cREATE Regular tablespace ts_item_003 pagesize 4K managed by database

-- now creating TS for ts_item_004 of D1
drop tablespace ts_item_004;
cREATE Regular tablespace ts_item_004 pagesize 4K managed by database

-- now creating TS for ts_item_005 of D1
drop tablespace ts_item_005;
cREATE Regular tablespace ts_item_005 pagesize 4K managed by database

TPC Benchmark™ C Full Disclosure Report - IBM System p 570 Model 9117-MMA Page 271 of 318
drop tablespace ts_neworda_006; create regular tablespace ts_neworda_006 pagesize 4K managed by database using
  
  
  
  
  
  }

extend 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
  
  
  
  
  
  }

extend 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
  
  
  
  
  
  }

extend 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
  
  
  
  
  
  }

extend 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
  
  
  
  
  
  }

extend 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
  
  
  
  
  
  }

extend 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/rdisk1/V4NORA' 201472,
    device '/dev/rdisk1/V5NORA' 201472,
    device '/dev/rdisk1/V6NORA' 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/rdisk1/V1NORA' 201472,
    device '/dev/rdisk1/V2NORA' 201472,
    device '/dev/rdisk1/V3NORA' 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/rdisk1/V4NORA' 201472,
    device '/dev/rdisk1/V5NORA' 201472,
    device '/dev/rdisk1/V6NORA' 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/rdisk1/V1NORA' 201472,
    device '/dev/rdisk1/V2NORA' 201472,
    device '/dev/rdisk1/V3NORA' 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/rdisk1/V4NORA' 201472,
    device '/dev/rdisk1/V5NORA' 201472,
    device '/dev/rdisk1/V6NORA' 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/rdisk1/V4NORA' 201472,
    device '/dev/rdisk1/V5NORA' 201472,
    device '/dev/rdisk1/V6NORA' 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/rdisk1/V4NORA' 201472,
    device '/dev/rdisk1/V5NORA' 201472,
    device '/dev/rdisk1/V6NORA' 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/rdisk1/V4NORA' 201472,
    device '/dev/rdisk1/V5NORA' 201472,
    device '/dev/rdisk1/V6NORA' 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/rdisk1/V4NORA' 201472,
    device '/dev/rdisk1/V5NORA' 201472,
    device '/dev/rdisk1/V6NORA' 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/rdisk1/V4NORA' 201472,
    device '/dev/rdisk1/V5NORA' 201472,
    device '/dev/rdisk1/V6NORA' 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/rdisk1/V4NORA' 201472,
    device '/dev/rdisk1/V5NORA' 201472,
    device '/dev/rdisk1/V6NORA' 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/rdisk1/V4NORA' 201472,
    device '/dev/rdisk1/V5NORA' 201472,
    device '/dev/rdisk1/V6NORA' 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/rdisk1/V4NORA' 201472,
    device '/dev/rdisk1/V5NORA' 201472,
    device '/dev/rdisk1/V6NORA' 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/rdisk1/V4NORA' 201472,
    device '/dev/rdisk1/V5NORA' 201472,
    device '/dev/rdisk1/V6NORA' 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/rdisk1/V4NORA' 201472,
    device '/dev/rdisk1/V5NORA' 201472,
    device '/dev/rdisk1/V6NORA' 201472
  )
  extentsize 256
  prefetchsize 4096;
commit;
device '/dev/rD1F17V6NORA' 201472
) 
exitextize 256
preferSize 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
)
eextentsize 256
prefetchsize 4096;
commit;
-- 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
)
create regular tablespace `ts_newordb_009` of D1

drop tablespace `ts_newordb_009`;
create regular tablespace `ts_newordb_009` pagesize 4K managed by database using
(  
  device '/dev/rdisk0s1N0RB' 201472,
  device '/dev/rdisk0s2N0RB' 201472,
  device '/dev/rdisk0s3N0RB' 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/rdisk0s4N0RB' 201472,
  device '/dev/rdisk0s5N0RB' 201472,
  device '/dev/rdisk0s6N0RB' 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/rdisk0s7N0RB' 201472,
  device '/dev/rdisk0s8N0RB' 201472,
  device '/dev/rdisk0s9N0RB' 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/rdisk0s10N0RB' 201472,
  device '/dev/rdisk0s11N0RB' 201472,
  device '/dev/rdisk0s12N0RB' 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/rdisk0s13N0RB' 201472,
  device '/dev/rdisk0s14N0RB' 201472,
  device '/dev/rdisk0s15N0RB' 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/rdisk0s16N0RB' 201472,
  device '/dev/rdisk0s17N0RB' 201472,
  device '/dev/rdisk0s18N0RB' 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/rdisk0s19N0RB' 201472,
  device '/dev/rdisk0s20N0RB' 201472,
  device '/dev/rdisk0s21N0RB' 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/rdisk0s22N0RB' 201472,
  device '/dev/rdisk0s23N0RB' 201472,
  device '/dev/rdisk0s24N0RB' 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/rdisk0s25N0RB' 201472,
  device '/dev/rdisk0s26N0RB' 201472,
  device '/dev/rdisk0s27N0RB' 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/rdisk0s28N0RB' 201472,
  device '/dev/rdisk0s29N0RB' 201472,
  device '/dev/rdisk0s30N0RB' 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/rdisk0s31N0RB' 201472,
  device '/dev/rdisk0s32N0RB' 201472,
  device '/dev/rdisk0s33N0RB' 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/rdisk0s34N0RB' 201472,
  device '/dev/rdisk0s35N0RB' 201472,
  device '/dev/rdisk0s36N0RB' 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/rdisk0s37N0RB' 201472,
  device '/dev/rdisk0s38N0RB' 201472,
  device '/dev/rdisk0s39N0RB' 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/rdisk0s40N0RB' 201472,
  device '/dev/rdisk0s41N0RB' 201472,
  device '/dev/rdisk0s42N0RB' 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/rdisk0s43N0RB' 201472,
  device '/dev/rdisk0s44N0RB' 201472,
  device '/dev/rdisk0s45N0RB' 201472
)
  extentsize 256
  prefetchsize 4096;
  commit;
managed by database
using
(device '/dev/rdsk/c0t1l0d0s0' 201472,
 device '/dev/rdsk/c0t1l0d0s1' 201472,
 device '/dev/rdsk/c0t1l0d0s2' 201472)
extent 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/rdsk/c0t1l0d0s0' 201472,
 device '/dev/rdsk/c0t1l0d0s1' 201472,
 device '/dev/rdsk/c0t1l0d0s2' 201472)
extent 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/rdsk/c0t1l0d0s0' 201472,
 device '/dev/rdsk/c0t1l0d0s1' 201472,
 device '/dev/rdsk/c0t1l0d0s2' 201472)
extent 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/rdsk/c0t1l0d0s0' 201472,
 device '/dev/rdsk/c0t1l0d0s1' 201472,
 device '/dev/rdsk/c0t1l0d0s2' 201472)
extent 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/rdsk/c0t1l0d0s0' 201472,
 device '/dev/rdsk/c0t1l0d0s1' 201472,
 device '/dev/rdsk/c0t1l0d0s2' 201472)
extent 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/rdsk/c0t1l0d0s0' 201472,
 device '/dev/rdsk/c0t1l0d0s1' 201472,
 device '/dev/rdsk/c0t1l0d0s2' 201472)
extent 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/rdsk/c0t1l0d0s0' 201472,
 device '/dev/rdsk/c0t1l0d0s1' 201472,
 device '/dev/rdsk/c0t1l0d0s2' 201472)
extent 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/rdsk/c0t1l0d0s0' 201472,
 device '/dev/rdsk/c0t1l0d0s1' 201472,
 device '/dev/rdsk/c0t1l0d0s2' 201472)
extent 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/rdsk/c0t1l0d0s0' 201472,
 device '/dev/rdsk/c0t1l0d0s1' 201472,
 device '/dev/rdsk/c0t1l0d0s2' 201472)
extent 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/rdsk/c0t1l0d0s0' 201472,
 device '/dev/rdsk/c0t1l0d0s1' 201472,
 device '/dev/rdsk/c0t1l0d0s2' 201472)
extent 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/rdsk/c0t1l0d0s0' 201472,
 device '/dev/rdsk/c0t1l0d0s1' 201472,
 device '/dev/rdsk/c0t1l0d0s2' 201472)
extent 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/rdsk/c0t1l0d0s0' 201472,
 device '/dev/rdsk/c0t1l0d0s1' 201472,
 device '/dev/rdsk/c0t1l0d0s2' 201472)
extent 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/rdsk/c0t1l0d0s0' 201472,
 device '/dev/rdsk/c0t1l0d0s1' 201472,
 device '/dev/rdsk/c0t1l0d0s2' 201472)
extent 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/rdsk/c0t1l0d0s0' 201472,
 device '/dev/rdsk/c0t1l0d0s1' 201472,
 device '/dev/rdsk/c0t1l0d0s2' 201472)
extent 256
prefetchsize 4096;
-- now creating TS for ts_reworkdb_038 of D1
drop tablespace ts_reworkdb_038;
create regular tablespace ts_reworkdb_038 pagesize 4K
managed by database
using
{|device '/dev/01F19V4NORB' 201472|
device '/dev/01F19V5NORB' 201472|
device '/dev/01F19V6NORB' 201472|}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultpool;
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/01F20V1NORB' 201472|
device '/dev/01F20V2NORB' 201472|
device '/dev/01F20V3NORB' 201472|}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultpool;
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/01F20V4NORB' 201472|
device '/dev/01F20V5NORB' 201472|
device '/dev/01F20V6NORB' 201472|}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultpool;
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/01F01V1ORD' 205312|}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultpool;
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/01F01V2ORD' 205312|}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultpool;
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/01F01V3ORD' 205312|}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultpool;
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/01F01V4ORD' 205312|}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultpool;
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/01F01V5ORD' 205312|}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultpool;
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/01F01V6ORD' 205312|}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultpool;
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/01F02V1ORD' 205312|}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultpool;
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/01F02V2ORD' 205312|}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultpool;
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/01F02V3ORD' 205312|}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultpool;
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/01F02V4ORD' 205312|}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultpool;
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/01F02V5ORD' 205312|}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultpool;
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/01F02V6ORD' 205312|}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultpool;
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/rD1F03V7ORD' 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/rD1F03V8ORD' 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/rD1F03V9ORD' 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/rD1F03V10ORD' 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/rD1F04V1ORD' 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/rD1F04V2ORD' 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/rD1F04V3ORD' 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/rD1F04V4ORD' 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/rD1F04V5ORD' 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/rD1F04V6ORD' 205312  
  )
  extentsize 256
  prefetchsize 4096
  bufferpool ibmdefaultbp8K;
commit;
using
(device '/dev/01F05V4ORD' 205312)
exentsize 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/01F05V5ORD' 205312)
exentsize 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/01F05V6ORD' 205312)
exentsize 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/01F06V1ORD' 205312)
exentsize 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/01F06V2ORD' 205312)
exentsize 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/01F06V3ORD' 205312)
exentsize 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/01F06V4ORD' 205312)
exentsize 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/01F06V5ORD' 205312)
exentsize 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/01F06V6ORD' 205312)
exentsize 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/01F07V1ORD' 205312)
exentsize 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/01F07V2ORD' 205312)
exentsize 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/01F07V3ORD' 205312)
exentsize 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/01F07V4ORD' 205312)
exentsize 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/01F07V5ORD' 205312)
exentsize 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/01F07V6ORD' 205312)
exentsize 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/01F08V1ORD' 205312)
exentsize 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/rD1F08V0ORD' 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/rD1F08V1ORD' 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/rD1F08V2ORD' 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/rD1F08V3ORD' 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/rD1F08V4ORD' 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/rD1F08V5ORD' 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/rD1F08V6ORD' 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/rD1F09V1ORD' 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/rD1F09V2ORD' 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/rD1F09V3ORD' 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/rD1F09V4ORD' 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/rD1F09V5ORD' 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/rD1F09V6ORD' 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/rD1F10V1ORD' 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/rD1F10V2ORD' 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/rD1F10V3ORD' 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/rD1F10V4ORD' 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/rD1F10V5ORD' 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/rD1F10V6ORD' 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/rD1F11V0ORD' 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/rD1F11V1ORD' 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/rD1F11V2ORD' 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/rD1F11V3ORD' 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/rD1F11V4ORD' 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/rD1F11V5ORD' 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/rD1F11V6ORD' 205312
}
extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;
drop tablespace ts_order_060;
cREATE REGULAR TABLESPACE ts_order_060 PAGESIZE 8K MANAGED BY DATABASE
USING
  (  
  device '/dev/rD1F11V0ORD' 205312  
  )
exteinsize 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/rD1F12V0ORD' 205312  
  )
exteinsize 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/rD1F12V1ORD' 205312  
  )
exteinsize 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/rD1F12V2ORD' 205312  
  )
exteinsize 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/rD1F12V3ORD' 205312  
  )
exteinsize 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/rD1F13V0ORD' 205312  
  )
exteinsize 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/rD1F13V1ORD' 205312  
  )
exteinsize 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/rD1F13V2ORD' 205312  
  )
exteinsize 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/rD1F13V3ORD' 205312  
  )
exteinsize 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/rD1F13V4ORD' 205312  
  )
exteinsize 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/rD1F14V0ORD' 205312  
  )
exteinsize 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/rD1F14V1ORD' 205312  
  )
exteinsize 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/rD1F14V2ORD' 205312  
  )
exteinsize 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/rD1F14V3ORD' 205312  
  )
exteinsize 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/rD1F14V4ORD' 205312  
  )
exteinsize 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/rD1F15V0ORD' 205312  
  )
exteinsize 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/rD1F15V1ORD' 205312  
  )
exteinsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;
managed by database using
  (  
   device '/dev/rdisk4/VORID' 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/rdisk4/VORID' 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/rdisk4/VORID' 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/rdisk4/VORID' 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/rdisk4/VORID' 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/rdisk4/VORID' 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/01F18V1ORD' 205312
}
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/01F18V2ORD' 205312
}
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/01F18V3ORD' 205312
}
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/01F18V4ORD' 205312
}
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/01F18V5ORD' 205312
}
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/01F18V6ORD' 205312
}
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/01F18V7ORD' 205312
}
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/01F18V8ORD' 205312
}
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/01F18V9ORD' 205312
}
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/01F18V10ORD' 205312
}
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/01F18V11ORD' 205312
}
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/01F18V12ORD' 205312
}
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/01F18V13ORD' 205312
}
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/01F18V14ORD' 205312
}
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/01F18V15ORD' 205312
}
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/01F18V16ORD' 205312
}
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/01F18V17ORD' 205312
}
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/01F18V18ORD' 205312
}
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/01F18V19ORD' 205312
}
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/01F18V20ORD' 205312
}
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/01F18V21ORD' 205312
}
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/01F18V22ORD' 205312
}
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/01F18V23ORD' 205312
}
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/01F18V24ORD' 205312
}
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/01F18V25ORD' 205312
}
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/01F18V26ORD' 205312
}
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;
drop tablespace ts_orderline_002; create regular tablespace ts_orderline_002 pagesize 8K managed by database using
  (  device '/dev/rdisk/V2/ORL' 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
  (  device '/dev/rdisk/V3/ORL' 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
  (  device '/dev/rdisk/V4/ORL' 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
  (  device '/dev/rdisk/V5/ORL' 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
  (  device '/dev/rdisk/V0/ORL' 5348864  )
  extentsize 256
  prefetchsize 4096
  bufferpool ibmdefaultbp8K;
commit;
drop tablespace ts_orderline_018;
create regular tablespace ts_orderline_018 pagesize 8K
managed by database
using
(device '/dev/rD1F03V5ORL' 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/rD1F04V4ORL' 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/rD1F05V3ORL' 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/rD1F06V2ORL' 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/rD1F06V5ORL' 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;
drop tablespace ts_orderline_049;
cREATE REGULAR TABLESPACE ts_orderline_049 PAGESIZE 8K MANAGED BY DATABASE USING (device '/dev/rD1F09V0ORL' 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/rD1F09V1ORL' 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/rD1F09V2ORL' 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/rD1F09V3ORL' 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/rD1F09V4ORL' 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/rD1F09V5ORL' 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/rD1F09V6ORL' 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/rD1F09V7ORL' 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/rD1F09V8ORL' 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/rD1F09V9ORL' 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/rD1F09V10ORL' 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/rD1F09V11ORL' 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/rD1F09V12ORL' 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/rD1F09V13ORL' 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/rD1F09V14ORL' 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/rD1F09V15ORL' 5348864);
EXTENTSIZE 256 PREFETCHSIZE 4096 BUFFERPOOL ibmdefaultbp8K;
COMMIT;
create regular tablespace ts_orderline_064 pagesize 8K managed by database using
{
  device '/dev/01F11V4ORL' 5348864
  extentsize 256
  prefercache 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/01F11V5ORL' 5348864
  extentsize 256
  prefercache 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/01F11V6ORL' 5348864
  extentsize 256
  prefercache 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/01F12V1ORL' 5348864
  extentsize 256
  prefercache 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/01F12V2ORL' 5348864
  extentsize 256
  prefercache 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/01F12V3ORL' 5348864
  extentsize 256
  prefercache 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/01F12V4ORL' 5348864
  extentsize 256
  prefercache 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/01F12V5ORL' 5348864
  extentsize 256
  prefercache 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/01F12V6ORL' 5348864
  extentsize 256
  prefercache 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/01F13V1ORL' 5348864
  extentsize 256
  prefercache 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/01F13V2ORL' 5348864
  extentsize 256
  prefercache 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/01F13V3ORL' 5348864
  extentsize 256
  prefercache 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/01F13V4ORL' 5348864
  extentsize 256
  prefercache 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/01F13V5ORL' 5348864
  extentsize 256
  prefercache 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/01F13V6ORL' 5348864
  extentsize 256
  prefercache 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/01F14V1ORL' 5348864
  extentsize 256
  prefercache 4096
  bufferpool ibmdefaultbp8K;
} commit;
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/rD1F15V0ORL' 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/rD1F15V1ORL' 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/rD1F15V2ORL' 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/rD1F15V3ORL' 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/rD1F15V4ORL' 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/rD1F15V5ORL' 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/rD1F15V6ORL' 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/rD1F15V7ORL' 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/rD1F15V8ORL' 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/rD1F15V9ORL' 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/rD1F15V10ORL' 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/rD1F15V11ORL' 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/rD1F15V12ORL' 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/rD1F15V13ORL' 5348864
      extentsize 256
    prefetchsize 4096
      bufferpool ibmdefaultbp8K;

commit;
drop tablespace ts_orderline_096;
create regular tablespace ts_orderline_096 pagesize 8K
managed by database using
  
  device '/dev/rD1F1V5ORL' 5348864
  
  extentsize 256
  
  prefetchsize 4096
  
  bufferpool ibmdefaultbp8K;
commit;
-- now creating TS for ts_orderline_096 of D1

drop tablespace ts_orderline_097;
create regular tablespace ts_orderline_097 pagesize 8K
managed by database using
  
  device '/dev/rD1F1V6ORL' 5348864
  
  extentsize 256
  
  prefetchsize 4096
  
  bufferpool ibmdefaultbp8K;
commit;
-- now creating TS for ts_orderline_097 of D1

drop tablespace ts_orderline_098;
create regular tablespace ts_orderline_098 pagesize 8K
managed by database using
  
  device '/dev/rD1F1V7ORL' 5348864
  
  extentsize 256
  
  prefetchsize 4096
  
  bufferpool ibmdefaultbp8K;
commit;
-- now creating TS for ts_orderline_098 of D1

drop tablespace ts_orderline_099;
create regular tablespace ts_orderline_099 pagesize 8K
managed by database using
  
  device '/dev/rD1F1V8ORL' 5348864
  
  extentsize 256
  
  prefetchsize 4096
  
  bufferpool ibmdefaultbp8K;
commit;
-- now creating TS for ts_orderline_099 of D1

drop tablespace ts_orderline_100;
create regular tablespace ts_orderline_100 pagesize 8K
managed by database using
  
  device '/dev/rD1F1V9ORL' 5348864
  
  extentsize 256
  
  prefetchsize 4096
  
  bufferpool ibmdefaultbp8K;
commit;
-- now creating TS for ts_orderline_100 of D1

-- 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/rD1F1V5ORL' 5348864
  
  extentsize 256
  
  prefetchsize 4096
  
  bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_101 of D1

-- 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/rD1F1V6ORL' 5348864
  
  extentsize 256
  
  prefetchsize 4096
  
  bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_102 of D1

-- 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/rD1F1V7ORL' 5348864
  
  extentsize 256
  
  prefetchsize 4096
  
  bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_103 of D1

-- 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/rD1F1V8ORL' 5348864
  
  extentsize 256
  
  prefetchsize 4096
  
  bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_104 of D1

-- 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/rD1F1V9ORL' 5348864
  
  extentsize 256
  
  prefetchsize 4096
  
  bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_105 of D1

-- 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/rD1F1V5ORL' 5348864
  
  extentsize 256
  
  prefetchsize 4096
  
  bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_106 of D1

-- 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/rD1F1V6ORL' 5348864
  
  extentsize 256
  
  prefetchsize 4096
  
  bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_107 of D1

-- 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/rD1F1V7ORL' 5348864
  
  extentsize 256
  
  prefetchsize 4096
  
  bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_108 of D1

-- 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/rD1F1V8ORL' 5348864
  
  extentsize 256
  
  prefetchsize 4096
  
  bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_109 of D1

-- 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/rD1F1V9ORL' 5348864
  
  extentsize 256
  
  prefetchsize 4096
  
  bufferpool ibmdefaultbp8K;
commit;

-- now creating TS for ts_orderline_110 of D1

-- 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/rD1F1V5ORL' 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/rD0F19V0STK' 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/rD0F19V1STK' 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/rD0F19V2STK' 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/rD0F19V3STK' 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/rD0F19V4STK' 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/rD0F19V5STK' 5348864
) extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;
connect reset;

---BEGIN---

-- 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/rD0F19V0STK' 9329152
) extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
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/rD0F19V1STK' 9329152
) extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
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/rD0F19V2STK' 9329152
) extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
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/rD0F19V3STK' 9329152
) extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
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/rD0F19V4STK' 9329152
) extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
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/rD0F19V5STK' 9329152
) 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/rD1F20V0STK' 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/rD1F20V1STK' 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/rD1F20V2STK' 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/rD1F20V3STK' 5348864
) extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
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/rD1F20V0STK' 9329152
) extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
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/rD1F20V1STK' 9329152
) extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
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/rD1F20V2STK' 9329152
) extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
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/rD1F20V3STK' 9329152
) extentsize 256
prefetchsize 4096
bufferpool ibmdefaultbp8K;
commit;
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/rd01f02v1stk' 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/rd01f02v6stk' 9329152  
     extentsize 256  
     prefetchsize 4096  
   )
commit;
drop tablespace ts_stock_029;
create regular tablespace ts_stock_029 pagesize 4K
managed by database using

  ( device '/dev/rD1F04V5STK' 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/rD1F05V1STK' 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/rD1F05V2STK' 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/rD1F05V3STK' 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/rD1F05V4STK' 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/rD1F05V5STK' 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/rD1F05V6STK' 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/rD1F06V1STK' 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/rD1F06V2STK' 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/rD1F06V3STK' 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/rD1F06V4STK' 9329152 )
  extentsize 256
  prefetchsize 4096;
commit;
-- now creating TS for ts_stock_040 of D1


drop tablespace ts_stock_040;
managed by database
using
(  device '/dev/rD1F07V4STK' 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/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;
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;
drop tablespace ts_stock_074;
create regular tablespace ts_stock_074 pagesize 4K
managed by database
using
(device '/dev/rD1F13V5STK' 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/rD1F14V1STK' 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/rD1F14V2STK' 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/rD1F14V3STK' 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/rD1F14V4STK' 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/rD1F14V5STK' 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/rD1F14V6STK' 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/rD1F15V1STK' 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/rD1F15V2STK' 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/rD1F15V3STK' 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/rD1F15V4STK' 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/rD1F15V5STK' 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/rD1F15V6STK' 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/rD1F15V7STK' 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/rD1F15V8STK' 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/01F16V1STK 9329152
}
extentsize 256;
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/01F16V2STK 9329152
}
extentsize 256;
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/01F16V3STK 9329152
}
extentsize 256;
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/01F16V4STK 9329152
}
extentsize 256;
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/01F16V5STK 9329152
}
extentsize 256;
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/01F16V6STK 9329152
}
extentsize 256;
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/01F17V1STK 9329152
}
extentsize 256;
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/01F17V2STK 9329152
}
extentsize 256;
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/01F17V3STK 9329152
}
extentsize 256;
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/01F18V1STK 9329152
}
extentsize 256;
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/01F18V2STK 9329152
}
extentsize 256;
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/01F18V3STK 9329152
}
extentsize 256;
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/01F18V4STK 9329152
}
extentsize 256;
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/01F18V5STK 9329152
}
extentsize 256;
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/01F18V6STK 9329152
}
extentsize 256;
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/01F18V7STK 9329152
}
extentsize 256;
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/01F18V8STK 9329152
}
extentsize 256;
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;

-- 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/rD1F01V7WARE' 128,  
    device '/dev/rD1F01V8WARE' 128,  
    device '/dev/rD1F01V9WARE' 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/rD1F01V10WARE' 128,  
    device '/dev/rD1F01V11WARE' 128,  
    device '/dev/rD1F01V12WARE' 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/rD1F02V2WARE' 128,
    device '/dev/rD1F02V3WARE' 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/rD1F04V1WARE' 128,
    device '/dev/rD1F04V2WARE' 128,
    device '/dev/rD1F04V3WARE' 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/rD1F05V1WARE' 128,
    device '/dev/rD1F05V2WARE' 128,
    device '/dev/rD1F05V3WARE' 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/rD1F06V1WARE' 128,
    device '/dev/rD1F06V2WARE' 128,
    device '/dev/rD1F06V3WARE' 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/rD1F07V1WARE' 128,
    device '/dev/rD1F07V2WARE' 128,
    device '/dev/rD1F07V3WARE' 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/rD1F08V1WARE' 128,
    device '/dev/rD1F08V2WARE' 128,
    device '/dev/rD1F08V3WARE' 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/rD1F09V1WARE' 128,
    device '/dev/rD1F09V2WARE' 128,
    device '/dev/rD1F09V3WARE' 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/rD1F0AWARE' 128,
    device '/dev/rD1F0BWARE' 128,
    device '/dev/rD1F0CWARE' 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/rD1F0DWARE' 128,
    device '/dev/rD1F0EWARE' 128,
    device '/dev/rD1F0FWARE' 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/rD1F01WARE' 128,
    device '/dev/rD1F02WARE' 128,
    device '/dev/rD1F03WARE' 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/rD1F04WARE' 128,
    device '/dev/rD1F05WARE' 128,
    device '/dev/rD1F06WARE' 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/rD1F07WARE' 128,
    device '/dev/rD1F08WARE' 128,
    device '/dev/rD1F09WARE' 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/rD1F0AWARE' 128,
    device '/dev/rD1F0BWARE' 128,
    device '/dev/rD1F0CWARE' 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/rD1F0DWARE' 128,
    device '/dev/rD1F0EWARE' 128,
    device '/dev/rD1F0FWARE' 128,
  )
extentsize 32
prefetchsize 4096;
commit;
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/rD1F11V7WARE' 128,
    device '/dev/rD1F11V8WARE' 128,
    device '/dev/rD1F11V9WARE' 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/rD1F12V1WARE' 128,
    device '/dev/rD1F12V2WARE' 128,
    device '/dev/rD1F12V3WARE' 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/rD1F12V4WARE' 128,
    device '/dev/rD1F12V5WARE' 128,
    device '/dev/rD1F12V6WARE' 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/rD1F13V1WARE' 128,
    device '/dev/rD1F13V2WARE' 128,
    device '/dev/rD1F13V3WARE' 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/rD1F13V4WARE' 128,
    device '/dev/rD1F13V5WARE' 128,
    device '/dev/rD1F13V6WARE' 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/rD1F14V1WARE' 128,
    device '/dev/rD1F14V2WARE' 128,
    device '/dev/rD1F14V3WARE' 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/rD1F14V4WARE' 128,
    device '/dev/rD1F14V5WARE' 128,
    device '/dev/rD1F14V6WARE' 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/rD1F15V1WARE' 128,
    device '/dev/rD1F15V2WARE' 128,
    device '/dev/rD1F15V3WARE' 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/rD1F15V4WARE' 128,
    device '/dev/rD1F15V5WARE' 128,
    device '/dev/rD1F15V6WARE' 128
  )
extentsize 32
prefetchsize 4096;
commit;  -- now creating TS for ts_ware_032 of D1
-- 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/rD1F17VWARE' 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

Src.Common/Makefile
Src.Common/tpccmisc.c

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

double current_time_ms(void) {
    return (t.tv_sec + (double)t.tv_usec/(1000*1000)); }

double current_time(void) {
    return(time(NULL)); }
/* ***************************************************************** */
#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|%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|%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
*
*/

/* 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;
}

/* 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 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();

/* ***************************************************************** */
/* Validate Command Line Arguments
*/

TPC Benchmark™ C Full Disclosure Report - IBM System p 570 Model 9117-MMA

Page 304 of 318


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 10:
/* 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);
            numBytes = sprintf(Buffer, fmtStock,
                s_order_cnt,                          stock_quant,                          s_remote_cnt,                          s_ytd,                          stock_dist_03,                          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; }
        }
    }
    rc = GenericClose(&hnd);
}
/* 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_a_string_with_original(item_name, "item", item_num);
        /* create price field */
        item_price = rand_decimal(100, 10000, 2);
        /* create ORIGINAL field */
        create_a_string_with_original(item_data, "item", item_num);
        numBytes = sprintf(Buffer, fmtItem,
            item_im_id,                          item_num,                          item_name,                          item_data,                          item_price,                          item_im_id,                          item_num);
        rc = GenericWrite(&hnd, Buffer, numBytes);
        if (rc != 0) { goto item_done; }
    }
    rc = GenericClose(&hnd);
}
/* 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 item_done; }
    for (ware_num = ware_start; ware_num <= ware_end; ware_num++)
    {
        /* create image id field */
        ware_im_id = rand_integer(1, 10000);
        /* create warehouse name field */
        create_a_string_with_original(ware_name, "ware", ware_num);
        /* create price field */
        ware_price = rand_decimal(100, 10000, 2);
        /* create ORIGINAL field */
        create_a_string_with_original(ware_data, "ware", ware_num);
        numBytes = sprintf(Buffer, fmtWare,
            ware_name,                          ware_im_id,                          ware_num,                          ware_price,                          ware_price);
```
if (rc != 0) { goto neword_done; }

* characteristics. We also create a btree index in W/D/O 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.

nu_ord_id++)
{       if (!quiet_mode) {          fprintf(stdout, "NEW_ORDER for Customer #%d:
", nu_ord_id);
          fflush(stdout);

}       for (ware_num = ware_start; ware_num <= ware_end; ware_num++)
{             numBytes = sprintf(Buffer, fmtNewOrd,

            cust_num,                                dist_num,                                ware_num,                                h_date,
            cust_credit,                                cust_discount,                                cust_first,                                cust_middle,                                cust_credit_lim,                                cust_since,                                cust_street_1,                                cust_street_2,                                cust_city,                                ware_num,                                h_date, 10.00,                                hist_data);
             rc = GenericWrite(&hnd, Buffer, numBytes);             if (rc != 0) { goto cust_done; }
}

```
# tpccrnd.c - Random generation functions for TPC-C

```c
#include "db2tpcc.h"
#include "tpccmisc.h"
#include "lval.h"

static char tbl_cust[CUSTOMERS_PER_DISTRICT];
static char *last_name_parts[] = {    "BAR",    "AHL",    "PRI",    "PRES",    "ESE",    "ANTI",    "CALLY",    "ATION",    "EING"};

int rand_integer ( int val_lo, int val_hi ) {
  return((random()%(val_hi-val_lo+1))+val_lo);
}
```

**Governed under the terms of the International License Agreement for Non-Warranted Sample Code.**

**All Rights Reserved.**

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

```c
/**
 * tpcmd c - Random generation functions for TPC-C
 */

#include <stdio.h>
#include <string.h>
#include "db2tpcc.h"
#include "tpccmisc.h"
#include "tval.h"

static char tbl_ci Cust[CUSTOMERS_PER_DISTRICT];
static char tbl_alnum[] = "0123456789abcdefghijklmnopqrstuvwxyzABCDEFHIJKLMNOPQRSTUVWXYZ";
static char *last_name_parts[] = { "BAR", "AHL", "PRI", "PRES", "ESE", "ANTI", "CALLY", "ATION", "EING"};

/**
 * @brief...
 */

```

TPC Benchmark™ C Full Disclosure Report - IBM System p 570 Model 9117-MMA Page 308 of 318
**create_random_n_string**

```c
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**

```c
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);
```

**initialize_random**

```c
void initialize_random(void) {
    int t = current_time();
srand(t);
srandom(t);
}
```

**create_random_a_string**

```c
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); }
```

**rand_decimal**

```c
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**

```c
void seed_1_3000( void ) {   int i;
    for (i = 0; i < CUSTOMERS_PER_DISTRICT; i++) {     tbl_cust[i] = 0;   }
}
```

**random_1_3000**

```c
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[i] == 0) {
            tbl_cust[i] = 1;
            return(i+1);         }
    x++;
    if (x == CUSTOMERS_PER_DISTRICT)    x=0;
}
```

**create_random_a_string_with_original**

```c
int create_random_a_string_with_original( 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); }
```
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_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);
}

typedef __DB2TPCC_H

#include <sys/types.h>
#include "lval.h"

int create_random_last_name(char *out_buffer, int cust_num) {
    int random_num;    
    #define INVALID_ITEM_ID (2 * ITEMS) + 1  
    #define UNUSED_ITEM_ID 0

    else
    {
        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)); }

int create_random_item(char *out_buffer, int cust_num) {
    int random_num;    
    #define INVALID_ITEM_ID (2 * ITEMS) + 1  
    #define UNUSED_ITEM_ID 0

    else
    {
        random_num %= 100;
        strcat(out_buffer, item_items[random_num / 10]);  
        random_num %= 10;  
        strcat(out_buffer, item_items[random_num]);
    }
    return(strlen(out_buffer)); }

#include/db2tpcc.h
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*);

/* Generic Macros */
/* *************************************************************************/
#define GEN_ERRCODE errno
/* *************************************************************************/
/* *************************************************************************/
/* 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) {                      
    char     s_C_LAST[17];                               
} item[15];                                              
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);}

/* UNIX Semaphore Macros                                                  */
#include <sys/types.h>  
#include <sys/ipc.h>  
#include <sys/sem.h>
struct in_delivery_struct {
    int16_t  s_W_ID;
    int16_t  s_O_CARRIER_ID;
    int16_t  s_ol_cnt;
    int16_t  s_OL_QUANTITY;
    int16_t  pad1[2];
    char     s_O_ENTRY_D_time[27];
    int32_t  s_C_ID;
    int16_t  transtatus;
    int16_t  deadlocks;
    char     s_C_FIRST[17];
    char     s_C_MIDDLE[3];
    char     s_C_LAST[17];
    char     s_O_CARRIER_ID[17];
    char     s_O_ENTRY_D_time[27];
    int32_t  s_threshold;
    int16_t  len;   
    int16_t  pad[SPGENERAL_PAD];   
};

struct out_delivery_struct {
    int16_t  s_ol_cnt;   
    int16_t  s_OL_QUANTITY;   
    int16_t  s_OL_DELIVERY_D_time;   
    int16_t  pad2;   
    int32_t  s_C_ID;   
    int32_t  s_C_Balance;   
    int32_t  s_C_LAST;
    int16_t  transtatus;
    int16_t  deadlocks;
};

struct out_stocklev_struct {
    int16_t  s_W_ID;
    int16_t  s_SUPPLY_W_ID;
    int16_t  s_REGULAR_SUPPLY_W_ID;
    int16_t  s_AUTHORITY_SUPPLY_W_ID;
    int16_t  s_LOW_STOCK_W_ID;
    int16_t  s_STOCKLEVEL_W_ID;
    int16_t  s_SHOPW_ID;
    int16_t  s_DISTRICTS_WITH_STOCKLEVEL;
    int16_t  s_THRESHOLD;
    int32_t  s_C_ID;
    int32_t  s_LOW_STOCK;
    int16_t  pad[SPGENERAL_PAD];   
    int32_t  s_THRESHOLD;
    int16_t  len;   
    int16_t  pad[SPGENERAL_PAD];   
};

/* Platform Macros */
/* *************************************************************************/
/* Platform Isolation Layer */
/* *************************************************************************/
/* 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 */

/* Platform Prototypes */
/* ***************************************************************************/
/* ***************************************************************************/
/* Platform Prototypes */
/* ***************************************************************************/

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);
#endif

ifdef __cplusplus

define IOH_OPEN(hnd)                                    
if (hnd->type == IOH_FILE_APPEND) {                      
    char     s_C_LAST[17];                               
} item[15];                                              
if (hnd->fd == -1) {                                      
    rc = -1;                                              
} else {                                                  
    rc = 0;                                                
}
#endif // __DB2TPCC_H
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=v8

# The schema name is typically the SQL authorization 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; if 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=${HOME}/tpccdata
export TPCC_FENCED=NO
## Appendix - D: Pricing Information

![CDW Shopping Cart](image)

**3Com Baseline Switch 2824 24-port unmanaged Gigabit**

<table>
<thead>
<tr>
<th>Quantity</th>
<th>Product Details</th>
<th>CDW Code</th>
<th>Availability</th>
<th>Price</th>
<th>Ext. Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>3Com Baseline Switch 2824 24-port unmanaged Gigabit</td>
<td>512294</td>
<td>In Stock</td>
<td>$289.99</td>
<td>$1,739.94</td>
</tr>
</tbody>
</table>

Click **to remove an item from your cart**

Sub-Total: **$1,739.94**

**800.750.4239**
IBM Corporation  
Tony Petrossian  
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 ($).

<table>
<thead>
<tr>
<th>Part Number</th>
<th>Description</th>
<th>Unit Price</th>
<th>Quantity</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>P73-00295</td>
<td>Windows Server 2003, Standard x86 Edition</td>
<td>$719</td>
<td>64</td>
<td>$46,016</td>
</tr>
<tr>
<td></td>
<td>Server License Only - No CALs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Discount Schedule: Open Program - No Level</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Unit Price reflects a 28% discount from the retail unit price of $999.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Full License</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>No Discounts Applied</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>N/A</td>
<td>Microsoft Problem Resolution Services</td>
<td>$245</td>
<td>1</td>
<td>$245</td>
</tr>
<tr>
<td></td>
<td>Professional Support (1 Incident)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

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: PCTpPe07051600061426. Please include this Reference ID in any correspondence regarding this price quote.
The requested quote for the System p 570 TPC-C benchmark using DB2 9 and IBM System Storage DS4800

<table>
<thead>
<tr>
<th>Description</th>
<th>Part No.</th>
<th>Unit Price</th>
<th>Qty</th>
<th>Ext Price</th>
<th>Maint Price</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Server Hardware</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Server 1:9117 Model MMA</td>
<td>9117-MMA</td>
<td>10,195</td>
<td>1</td>
<td>10,195</td>
<td>7,146</td>
</tr>
<tr>
<td>Op Panel (MMA)</td>
<td>1845</td>
<td>199</td>
<td>1</td>
<td>199</td>
<td></td>
</tr>
<tr>
<td>P6 Processor Power Regulator</td>
<td>5625</td>
<td>1,500</td>
<td>12</td>
<td>18,000</td>
<td></td>
</tr>
<tr>
<td>System CEC Enclosure with Bezel</td>
<td>5626</td>
<td>500</td>
<td>4</td>
<td>2,000</td>
<td></td>
</tr>
<tr>
<td>AC Power Supply, 200-240v, 1500 Watt</td>
<td>5628</td>
<td>1,502</td>
<td>8</td>
<td>12,016</td>
<td></td>
</tr>
<tr>
<td>Media Enclosure and Backplane</td>
<td>5629</td>
<td>185</td>
<td>1</td>
<td>185</td>
<td></td>
</tr>
<tr>
<td>Service Processor Interface</td>
<td>5648</td>
<td>1,000</td>
<td>4</td>
<td>4,000</td>
<td></td>
</tr>
<tr>
<td>Processor Enclosure and Backplane</td>
<td>5663</td>
<td>2,000</td>
<td>4</td>
<td>8,000</td>
<td></td>
</tr>
<tr>
<td>I/O Backplane</td>
<td>5666</td>
<td>4,500</td>
<td>4</td>
<td>18,000</td>
<td></td>
</tr>
<tr>
<td>System Midplane</td>
<td>5667</td>
<td>1,000</td>
<td>4</td>
<td>4,000</td>
<td></td>
</tr>
<tr>
<td>SAS DASD Backplane, 6-pk</td>
<td>5668</td>
<td>1,051</td>
<td>4</td>
<td>4,204</td>
<td></td>
</tr>
<tr>
<td>Line Cord, DRWR TO IBM PDU, 14', 200-240V/10A,</td>
<td>6458</td>
<td>19</td>
<td>8</td>
<td>152</td>
<td></td>
</tr>
<tr>
<td>Rack Mount kit for IBM 19&quot; rack</td>
<td>7164</td>
<td>222</td>
<td>4</td>
<td>888</td>
<td></td>
</tr>
<tr>
<td>Power Distribution Backplane</td>
<td>7870</td>
<td>265</td>
<td>4</td>
<td>1,060</td>
<td></td>
</tr>
<tr>
<td>P6 P6 SMP Fabric Cable, DRWR/DRWR</td>
<td>3660</td>
<td>2,000</td>
<td>3</td>
<td>6,000</td>
<td></td>
</tr>
<tr>
<td>P6 P6 SMP Fabric Cable, DRWR/NC/DRWR</td>
<td>3664</td>
<td>4,000</td>
<td>2</td>
<td>8,000</td>
<td></td>
</tr>
<tr>
<td>P6 SMP Fabric Cable DRWR/NC/NC/DRWR</td>
<td>3665</td>
<td>8,000</td>
<td>1</td>
<td>8,000</td>
<td></td>
</tr>
<tr>
<td>Enhanced FSP cable, 4 enclosures</td>
<td>5660</td>
<td>8,000</td>
<td>1</td>
<td>8,000</td>
<td></td>
</tr>
<tr>
<td>IDE Slimline DVD-ROM Drive</td>
<td>5756</td>
<td>275</td>
<td>1</td>
<td>275</td>
<td></td>
</tr>
<tr>
<td>4.7GHz POWER6 -2 Core Processor Card, 0-core active</td>
<td>7380</td>
<td>11,500</td>
<td>8</td>
<td>92,000</td>
<td>34,368</td>
</tr>
<tr>
<td>One Processor Activation for Processor Feature #7380</td>
<td>5403</td>
<td>23,000</td>
<td>16</td>
<td>368,000</td>
<td>63,360</td>
</tr>
<tr>
<td>256GB Memory (8x32GB) DDR2 POWER6 memory</td>
<td>8129</td>
<td>97,126</td>
<td>3</td>
<td>291,378</td>
<td></td>
</tr>
<tr>
<td>Activation of 256 GB DDR2 POWER6 Memory</td>
<td>5861</td>
<td>387,840</td>
<td>3</td>
<td>1,183,520</td>
<td></td>
</tr>
<tr>
<td>I/O Riser, 2x Serial, 2x p5I02C E'net (Evans)</td>
<td>5636</td>
<td>399</td>
<td>4</td>
<td>1,596</td>
<td></td>
</tr>
<tr>
<td>73.4GB SAS DASD, 15K RPM</td>
<td>3646</td>
<td>659</td>
<td>2</td>
<td>1,318</td>
<td></td>
</tr>
<tr>
<td>4 Gigabit Fibre Channel PCI-X Adapter</td>
<td>5759</td>
<td>3,308</td>
<td>21</td>
<td>69,468</td>
<td></td>
</tr>
<tr>
<td>IBM 10/100/1000 Base-TX Ethernet PCI-X Adapter</td>
<td>5701</td>
<td>699</td>
<td>4</td>
<td>2,796</td>
<td></td>
</tr>
<tr>
<td>DX Dual Port- 12X Channel Attach</td>
<td>1802</td>
<td>1,499</td>
<td>4</td>
<td>5,996</td>
<td></td>
</tr>
<tr>
<td>IO Drawer 7314-G30</td>
<td>7314-G30</td>
<td>2,850</td>
<td>4</td>
<td>11,400</td>
<td>20,324</td>
</tr>
<tr>
<td>Planer and Tray Assembly</td>
<td>6590</td>
<td>1,300</td>
<td>4</td>
<td>5,200</td>
<td></td>
</tr>
<tr>
<td>I/O Drawer Mounting Enclosure</td>
<td>7314</td>
<td>525</td>
<td>2</td>
<td>1,050</td>
<td></td>
</tr>
<tr>
<td>AC Power Supply 300 Watt</td>
<td>6270</td>
<td>300</td>
<td>8</td>
<td>2,400</td>
<td></td>
</tr>
<tr>
<td>1.5M 12X ENHANCED IB CABLE</td>
<td>1830</td>
<td>400</td>
<td>8</td>
<td>3,200</td>
<td></td>
</tr>
<tr>
<td>Power Controll SPCN</td>
<td>6631</td>
<td>250</td>
<td>4</td>
<td>1,000</td>
<td></td>
</tr>
<tr>
<td>Line cord</td>
<td>6458</td>
<td>14</td>
<td>8</td>
<td>112</td>
<td></td>
</tr>
<tr>
<td>Dual Port 12X Channel Adapter</td>
<td>6446</td>
<td>575</td>
<td>4</td>
<td>2,300</td>
<td></td>
</tr>
<tr>
<td>POWER CONTROL CABLE, 3M, (SPCN)</td>
<td>6006</td>
<td>40</td>
<td>8</td>
<td>320</td>
<td></td>
</tr>
<tr>
<td>Rack Model T00</td>
<td>7014-T00</td>
<td>2,920</td>
<td>1</td>
<td>2,920</td>
<td>768</td>
</tr>
<tr>
<td>Front Trim Kit For 1.8 Meter Rack (Black)</td>
<td>6246</td>
<td>158</td>
<td>1</td>
<td>158</td>
<td></td>
</tr>
<tr>
<td>Side Panel (Black)</td>
<td>6098</td>
<td>150</td>
<td>2</td>
<td>300</td>
<td></td>
</tr>
<tr>
<td>PDU to 14', 200-240V/24A, UTG0247, PT#12</td>
<td>6654</td>
<td>240</td>
<td>1</td>
<td>240</td>
<td></td>
</tr>
<tr>
<td>HMC 1:7310-C05 Desktop Hardw.Mgmt.Console</td>
<td>7310-C05</td>
<td>1,830</td>
<td>1</td>
<td>1,830</td>
<td>1,344</td>
</tr>
<tr>
<td>IBM ThinkVision C170 17-inch Color Monitor</td>
<td>3631</td>
<td>250</td>
<td>1</td>
<td>250</td>
<td></td>
</tr>
<tr>
<td>Power Cord (6-foot), To Wall Plug Type #4</td>
<td>6470</td>
<td>18</td>
<td>2</td>
<td>36</td>
<td></td>
</tr>
<tr>
<td>Ethernet Cable, 6M, HMC to System Unit</td>
<td>7801</td>
<td>15</td>
<td>1</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>Keyboard - English, #103P</td>
<td>8800</td>
<td>104</td>
<td>1</td>
<td>104</td>
<td></td>
</tr>
<tr>
<td>Mouse - Attachment Cable</td>
<td>8841</td>
<td>78</td>
<td>1</td>
<td>78</td>
<td></td>
</tr>
</tbody>
</table>

Subtotal: 2,142,159 127,310
## Storage

<table>
<thead>
<tr>
<th>Description</th>
<th>Model</th>
<th>Quantity</th>
<th>Price</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>DS4800 Disk System Model 82</td>
<td>1815-82A</td>
<td>53,995</td>
<td>21</td>
<td>1,133,895</td>
</tr>
<tr>
<td>DS4800 8-Storage Partitions</td>
<td>8870</td>
<td>10,000</td>
<td>21</td>
<td>210,000</td>
</tr>
<tr>
<td>(22R4255) DS4800 AIX Host Kit</td>
<td>7711</td>
<td>7,000</td>
<td>21</td>
<td>147,000</td>
</tr>
<tr>
<td>DS4000 EXP810 Enclosure</td>
<td>1812-81A</td>
<td>6,000</td>
<td>248</td>
<td>1,488,000</td>
</tr>
<tr>
<td>72GB/15K Drive 4Gb FC disks</td>
<td>5413</td>
<td>1,679</td>
<td>21</td>
<td>282,072</td>
</tr>
<tr>
<td>36GB/15K Drive 4Gb FC disks</td>
<td>5412</td>
<td>892</td>
<td>3,312</td>
<td>2,954,304</td>
</tr>
<tr>
<td>Short Wave SFP</td>
<td>2410</td>
<td>998</td>
<td>185</td>
<td>184,630</td>
</tr>
<tr>
<td>Fiber Cable 25m</td>
<td>5625</td>
<td>189</td>
<td>42</td>
<td>7,938</td>
</tr>
<tr>
<td>Fiber Cable 1m</td>
<td>5601</td>
<td>79</td>
<td>496</td>
<td>39,184</td>
</tr>
<tr>
<td>3 Year Warranty Service Upgrade 1812-81A 24x7x4</td>
<td>960</td>
<td>248</td>
<td></td>
<td>238,080</td>
</tr>
<tr>
<td>3 Year Warranty Service Upgrade 1815-82A 24x7x4</td>
<td>3,200</td>
<td>21</td>
<td></td>
<td>67,200</td>
</tr>
<tr>
<td><strong>Subtotal</strong></td>
<td></td>
<td></td>
<td></td>
<td><strong>6,447,023</strong></td>
</tr>
<tr>
<td><strong>Server Software</strong></td>
<td></td>
<td></td>
<td></td>
<td><strong>305,280</strong></td>
</tr>
<tr>
<td>AIX 5.3 (media only)</td>
<td>5692-A5L</td>
<td>50</td>
<td>1</td>
<td>50</td>
</tr>
<tr>
<td>AIX Software per Processor</td>
<td>5765-G03</td>
<td>1,225</td>
<td>16</td>
<td>19,600</td>
</tr>
<tr>
<td>Software Maintenance for AIX, 3 Year</td>
<td>5773-SM3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F5 3 yr SWMA for AIX per Processor</td>
<td>466</td>
<td>1,958</td>
<td>16</td>
<td>31,328</td>
</tr>
<tr>
<td>F5 3 yr Services 7x24 Support per Processor</td>
<td>468</td>
<td>496</td>
<td>16</td>
<td>7,936</td>
</tr>
<tr>
<td>Partition Load Manager SW Maint: 3 year</td>
<td>5773-PLM</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F5 3 yr SWMA for AIX per Processor</td>
<td>627</td>
<td>55</td>
<td>16</td>
<td>880</td>
</tr>
<tr>
<td>F5 3 yr Services 7x24 Support per Processor</td>
<td>628</td>
<td>14</td>
<td>16</td>
<td>224</td>
</tr>
<tr>
<td>VIO Software Maintenance (3Y)</td>
<td>5773-VIO</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Per Processor F5 VIO 3 Yr Maintenance</td>
<td>573</td>
<td>245</td>
<td>16</td>
<td>3,920</td>
</tr>
<tr>
<td>Per Processor F5 VIO 3 Yr Maint 24x7 Support</td>
<td>574</td>
<td>64</td>
<td>16</td>
<td>1,024</td>
</tr>
<tr>
<td>Initial Software Support 3 Year</td>
<td>5773-RS3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Per Processor Software Support 3 Year</td>
<td>569</td>
<td>675</td>
<td>1</td>
<td>675</td>
</tr>
<tr>
<td>Per Processor 24x7 Software Support 3 Year</td>
<td>570</td>
<td>236</td>
<td>1</td>
<td>236</td>
</tr>
<tr>
<td>C for AIX user Lic+SW maint 12 MO</td>
<td>D5A1DLL</td>
<td>515</td>
<td>1</td>
<td>515</td>
</tr>
<tr>
<td>C for AIX user annual SW maint renewal</td>
<td>E1A1FLL</td>
<td>103</td>
<td>2</td>
<td>206</td>
</tr>
<tr>
<td>DB2 Enterprise Proc 9 Lic/1 year Maintenance</td>
<td>271</td>
<td>1,920</td>
<td>1</td>
<td>519,533</td>
</tr>
<tr>
<td>DB2 9 Enterprise Edition Proc Maint Renew</td>
<td>13</td>
<td>3,840</td>
<td>1</td>
<td>49,498</td>
</tr>
<tr>
<td><strong>Subtotal</strong></td>
<td></td>
<td></td>
<td></td>
<td><strong>539,698</strong></td>
</tr>
<tr>
<td><strong>System x Servers</strong></td>
<td></td>
<td></td>
<td></td>
<td><strong>95,927</strong></td>
</tr>
<tr>
<td>xSeries 346 Express Model</td>
<td>884025U</td>
<td>2,175</td>
<td>64</td>
<td>139,200</td>
</tr>
<tr>
<td>3.2GHz 800MHz 2MB L2 Cache Xeon Processor</td>
<td>40K2505</td>
<td>499</td>
<td>64</td>
<td>31,936</td>
</tr>
<tr>
<td>1GB (2x12MB Kit) PC2-3200</td>
<td>39M5818</td>
<td>238</td>
<td>64</td>
<td>15,232</td>
</tr>
<tr>
<td>36GB 15K Hot Swap SCSI</td>
<td>40K1026</td>
<td>269</td>
<td>64</td>
<td>17,216</td>
</tr>
<tr>
<td>NetBAY S2 42U Standard Rack Cabinet</td>
<td>93074RX</td>
<td>1,489</td>
<td>26</td>
<td>38,714</td>
</tr>
<tr>
<td>Optical 3-Button Mouse - USB</td>
<td>40K9201</td>
<td>19</td>
<td>1</td>
<td>19</td>
</tr>
<tr>
<td>Preferred Pro Full Size PS/2 Keyboard</td>
<td>40K9584</td>
<td>29</td>
<td>1</td>
<td>29</td>
</tr>
<tr>
<td>IBM C117 17&quot; CRT Monitor</td>
<td>49387NU</td>
<td>149</td>
<td>1</td>
<td>149</td>
</tr>
<tr>
<td><strong>Subtotal</strong></td>
<td></td>
<td></td>
<td></td>
<td><strong>242,495</strong></td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td></td>
<td></td>
<td><strong>9,371,375</strong></td>
</tr>
<tr>
<td><strong>Total IBM Discounts</strong></td>
<td></td>
<td></td>
<td></td>
<td><strong>566,917</strong></td>
</tr>
<tr>
<td><strong>Three-Year Cost of Ownership</strong></td>
<td></td>
<td></td>
<td></td>
<td><strong>5,664,826</strong></td>
</tr>
</tbody>
</table>

For more information on:
- For System x products: [http://www-03.ibm.com/systems/x/](http://www-03.ibm.com/systems/x/)

For additional information, please contact me directly:
Daniel Shea
IBM Sales & Distribution, STG Sales
1-781-895-2244