

TPC Benchmark™ H Full Disclosure Report
for
IBM® @server™ xSeries® 365
using
IBM DB2® Universal Database 8.1

Submitted for Review

May 18, 2004



First Edition - May 2004

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 the 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.

This publication was produced in the United States. IBM may not offer the products, services, or features discussed in this document in other countries, and the information is subject to change without notice. Consult your local IBM representative for information on products and services available in your area.

© Copyright International Business Machines Corporation 2004. All rights reserved.

Permission is hereby granted to reproduce this document in whole or in part, provided the copyright notice as printed above is set forth in full text on the title page of each item reproduced.

U.S. 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.

Trademarks

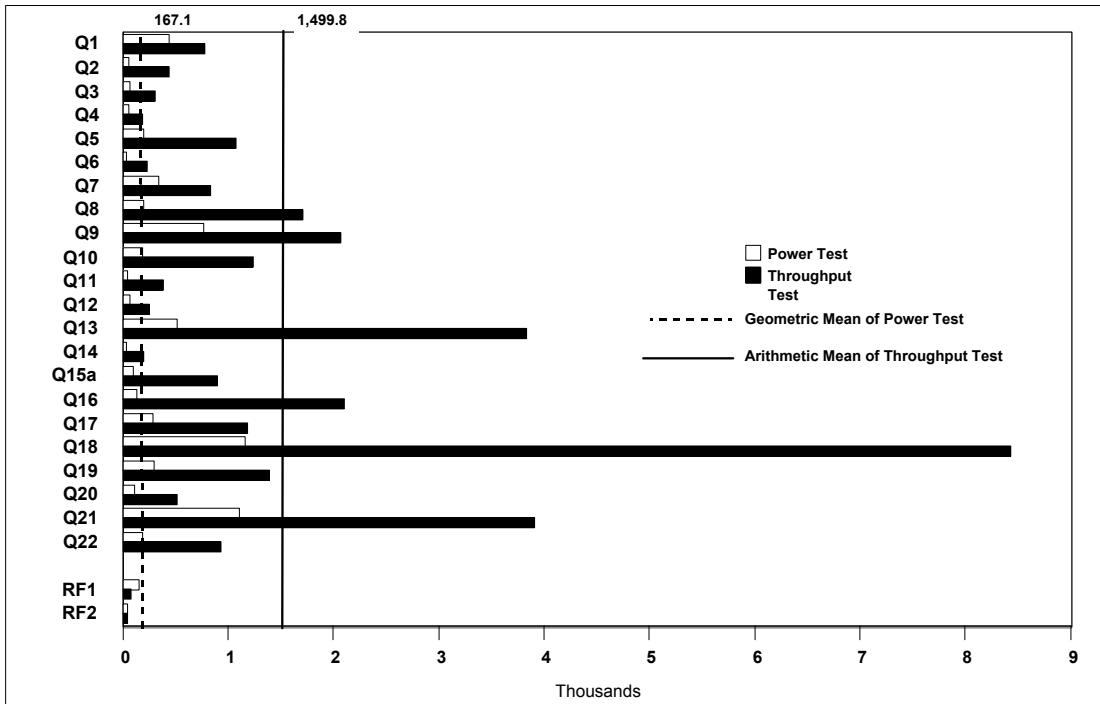
IBM, the IBM e-business logo, DB2, DB2 Universal Database, and xSeries are trademarks or registered trademarks of International Business Machines Corporation.

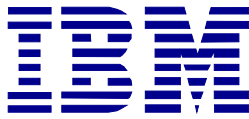
The following terms used in this publication are trademarks of other companies as follows: TPC Benchmark, TPC-H, QppH QthH and QphH are trademarks of Transaction Processing Performance Council; Intel and Xeon are trademarks or registered trademarks of Intel Corporation; Microsoft and Windows are trademarks or registered trademarks of Microsoft Corporation. Other company, product, or service names, which may be denoted by two asterisks (**), may be trademarks or service marks of others.

Notes

¹ GHz only measures microprocessor internal clock speed, not application performance. Many factors affect application performance.

² When referring to hard disk capacity, one GB equals one billion bytes. Total user-accessible capacity may be less.

		IBM® @server® xSeries® 365 with IBM DB2® UDB 8.1		TPC-H Rev 2.1.0
				Report Date: May 18, 2004
Total System Cost		Composite Query-per-Hour Metric		Price/Performance
\$248,778		5003.1 QphH @ 300GB		\$50 per QphH @ 300GB
Database Size	Database Manager	Operating System	Other Software	Availability Date
300GB	IBM DB2 UDB 8.1	Microsoft® Windows® Server 2003 Enterprise Edition w/QFE KB834628	Microsoft Visual C++ .NET Standard	May 18, 2004
				
Database Load Time: 05:33:00		Load Included Backup: Y	Total Data Storage / Database Size: 12.82	
RAID (Base Tables Only): N		RAID (Base Tables and Auxiliary Data Structures): N	RAID (All): N	
Configuration				
Processors/Cores/Threads	4/4/8	Intel Xeon MP 3.0GHz / 4MB ECC L3 Cache		
Memory	16	1GB PC2100 DDR ECC SDRAM RDIMM		
Disk Controllers	4	ServeRAID-6M Ultra320 SCSI Adapter		
Disk Drives	112	36.4GB 15K Ultra320 SCSI Drive		
	2	36.4GB 15K Ultra320 SCSI Drive		
Total Disk Storage		3864.6GB		

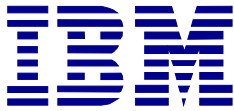


IBM @server xSeries 365 with IBM DB2 UDB 8.1

TPC-H 2.1.0 Executive Summary

Report Date: May 18, 2004

Description	Part Number	Third Party Brand	Pricing	Unit Price	Quantity	Extended Price	3-Yr. Maint. Price
Server Hardware							
IBM e(logo)server xSeries 365 w/3.0GHz Xeon MP w/2 x 3.0GHz/4MB Xeon MP Processors w/4 x 512MB ECC SDRAM RDIMMs	8862-6RX	IBM		1	15,599	1	1,000
xSeries 3.0GHz/4MB Processor	13N0710	IBM		1	5,749	2	11,498
1GB PC2100 DDR ECC SDRAM RDIMM	73P2031	IBM		1	659	16	10,544
Memory Upgrade Card	73P9710	IBM		1	999	1	999
36.4GB 15K Ultra320 SCSI Hot-Swap Drive	32P0734	IBM		1	495	2	990
ServeRAID-6M Ultra320 SCSI Contoller	32P0033	IBM		1	999	4	3,996
APC Smart-UPS Model 1400	32P1020	IBM		1	855	1	855
Netfinity 4.2M Ultra2 SCSI Cable	03K9311	IBM		1	105	8	840
IBM Preferred Pro Full-Size Keyboard	31P7415	IBM		1	29	1	29
IBM Sleek 2-Button Mouse	28L3673	IBM		1	15	1	15
E54 15" (13.8" Viewable) Color Monitor	633147N	IBM		1	119	1	90
NetBAY42S Enterprise Rack	9306421	IBM		1	1,439	1	168
					Subtotal	46,923	1,258
Server Storage							
IBM EXP400 Rack Storage Exp. Enclosure	17331RU	IBM		1	3,099	8	6,080
36.4GB 15K Ultra320 SCSI Hot-Swap Drive	32P0734	IBM		1	495	112	55,440
					Subtotal	80,232	6,080
Server Software							
DB2 UDB ESE 8.1 SW Lic. & Maintenance 12 Months	D518GLL	IBM		1	21,425	4	85,700
SW Maintenance Renewal - 1 Year	E00BILL	IBM		1	1,020	8	8,160
DPF SW License & Maintenance 12 Months	D518JLL	IBM		1	6,436	4	25,744
DPF SW Maintenance Renewal - 1 Year	E00BILL	IBM		1	306	8	2,448
Microsoft Windows Server 2003 Enterprise Edition	P7200264	Microsoft		2	2,399	1	Incl Below
Operating System Support Package	PRO-PRORS-16U-01	Microsoft		2	1,950	3	5,850
Microsoft Visual C++ .NET Standard		Microsoft		3	90	1	90
					Subtotal	113,933	16,458
Compsat Technology discount					Discount	16,106	
					Total	224,982	23,796
Three-Year Cost of Ownership:						248,778	
Pricing: 1 - Compsat Technology; 2 - Microsoft; 3 - Amazon.com				QphH@300GB: 5,003.1			
Warranty and Maintenance: The standard warranty has been upgraded to 3 years of 24x7x4 coverage.				\$/QphH@300GB: \$50			
Audited by Francois Raab, InfoSizing, Inc.							
Prices used in TPC benchmarks reflect the actual prices a customer would pay for a one-time purchase of the stated components. Individually negotiated discounts are not permitted. Special prices based on assumptions about past or future purchases are not permitted. All discounts reflect standard pricing policies for the listed components. For complete details, see the pricing sections of the TPC benchmark specifications. If you find that stated prices are not available according to these terms, please inform the TPC at pricing@tpc.org. Thank you.							



**IBM @server xSeries 445
with
IBM DB2 UDB 8.1**

TPC-H Rev 2.1.0

Report Date: May 18, 2004

Measurement Results:

Database Scale Factor	300
Total Data Storage/Database Size	12:82
Start of Database Load	14:38:37
End of Database Load	20:11:05
Database Load Time	5:33
Query Streams for Throughput Test	6
TPC-H Power	6,463.9
TPC-H Throughput	3,872.4
TPC-H Composite Query-per-Hour (QphH@300GB)	5,003.1
Total System Price over 3 Years	\$248,778
TPC-H Price/Performance Metric (\$/QphH@300GB)	\$50

Measurement Interval:

Measurement Interval in Throughput Test (Ts) = 36,814

Duration of Stream Execution:

	Seed	Query Start Date/Time Query End Date/Time	RF1 Start Date/Time RF1 End Date/Time	RF2 Start Date/Time RF2 End Date/Time	Duration
Stream 00	514201105	05/14/04 21:15:29 05/14/04 23:07:05	05/14/04 21:15:29 05/14/04 21:18:03	05/14/04 23:06:13 05/14/04 23:07:05	01:51:36
Stream 01	514201106	05/14/04 23:07:08 05/15/04 09:07:24	05/14/04 23:07:08 05/15/04 09:08:48	05/15/04 09:08:48 05/15/04 09:09:52	10:00:16
Stream 02	514201107	05/14/04 23:07:08 05/15/04 06:37:40	05/15/04 09:09:52 05/15/04 09:11:02	05/15/04 09:11:02 05/15/04 09:11:51	07:30:32
Stream 03	514201108	05/14/04 23:07:08 05/15/04 08:55:05	05/15/04 09:11:51 05/15/04 09:13:10	05/15/04 09:13:10 05/15/04 09:14:06	09:47:57
Stream 04	514201109	05/14/04 23:07:08 05/15/04 07:35:18	05/15/04 09:14:06 05/15/04 09:15:28	05/15/04 09:15:28 05/15/04 09:16:19	08:28:10
Stream 05	514201110	05/14/04 23:07:08 05/15/04 09:07:14	05/15/04 09:16:19 05/15/04 09:17:44	05/15/04 09:17:44 05/15/04 09:18:39	10:00:06
Stream 06	514201111	05/14/04 23:07:08 05/15/04 08:19:37	05/15/04 09:18:39 05/15/04 09:19:50	05/15/04 09:19:50 05/15/04 09:20:42	09:12:29



IBM @server xSeries 445
with
IBM DB2 UDB 8.1

TPC-H Rev 2.1.0

Report Date: May 18, 2004

TPC-H Timing Intervals (in seconds):

Query	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12
Stream 00	447.2	61.2	72.8	58.3	206.9	38.2	340.6	206.6	775.3	192.5	50.6	75.6
Stream 01	724.6	331.3	227.4	84.1	432.1	67.5	349.3	1,916.4	1,172.1	1,249.5	240.2	385.0
Stream 02	568.8	652.1	246.2	148.5	1,025.0	398.0	709.4	1,028.8	2,093.7	1,144.9	603.6	171.5
Stream 03	1,051.8	492.9	521.9	366.2	1,748.5	317.0	1,087.1	2,300.8	2,237.5	1,225.7	465.4	265.8
Stream 04	870.3	361.8	200.9	191.5	1,162.7	234.6	882.1	1,739.3	2,388.1	844.8	237.6	305.1
Stream 05	1,003.5	351.8	220.8	194.9	640.9	169.3	1,203.5	1,723.3	1,661.7	731.9	499.3	188.2
Stream 06	491.4	460.6	425.2	172.2	1,491.7	256.8	794.9	1,559.6	2,866.7	2,271.1	310.1	248.6
Minimum	447.2	61.2	72.8	58.3	206.9	38.2	340.6	206.6	775.3	192.5	50.6	75.6
Average	785.1	441.8	307.1	192.9	1,083.5	240.6	837.7	1,711.4	2,070.0	1,244.7	392.7	260.1
Maximum	1,051.8	652.1	521.9	366.2	1,748.5	398.0	1,203.5	2,300.8	2,866.7	2,271.1	603.6	385.0

Stream ID	Q13	Q14	Q15a	Q16	Q17	Q18	Q19	Q20	Q21	Q22	RF1	RF2
Stream 00	517.5	43.2	101.0	134.7	286.5	1,163.4	298.3	119.8	1,112.3	188.2	154.3	51.9
Stream 01	2,635.8	141.7	718.4	2,698.9	1,363.3	13,154.1	925.8	592.9	5,893.0	713.3	84.0	64.0
Stream 02	1,733.0	224.0	930.7	3,187.5	1,805.8	5,215.1	1,648.5	430.8	2,282.0	784.2	69.4	49.3
Stream 03	3,290.5	125.9	846.1	1,046.5	1,686.2	11,099.8	1,660.6	618.8	2,086.9	734.8	78.6	56.4
Stream 04	6,717.9	188.0	789.3	802.2	1,015.7	3,745.9	1,344.0	435.8	5,045.1	991.0	82.1	51.1
Stream 05	4,968.7	340.7	703.6	2,903.0	631.2	8,261.5	1,506.8	546.7	6,263.9	1,290.9	84.6	55.1
Stream 06	3,709.7	174.4	1,408.6	1,989.6	624.5	9,110.2	1,266.7	525.6	1,882.9	1,107.6	71.6	51.2
Minimum	517.5	43.2	101.0	134.7	286.5	1,163.4	298.3	119.8	1,112.3	188.2	69.4	49.3
Average	3,842.6	199.1	899.5	2,104.6	1,187.8	8,431.1	1,392.1	525.1	3,909.0	937.0	78.4	54.5
Maximum	6,717.9	340.7	1,408.6	3,187.5	1,805.8	13,154.1	1,660.6	618.8	6,263.9	1,290.9	154.3	64.0

Table of Contents

Preface	11
1 General Items	13
1.1 Benchmark Sponsor	13
1.2 Parameter Settings	13
1.3 Configuration Diagrams	13
1.3.1 Priced and Measured Configurations	14
2 Clause 1: Logical Database Design Related Items	15
2.1 Database Table Definitions	15
2.2 Database Organization	15
2.3 Horizontal Partitioning	15
2.4 Replication	15
3 Clause 2: Queries and Update Functions Related Items	16
3.1 Query Language	16
3.2 Random Number Generation	16
3.3 Substitution Parameters Generation	16
3.4 Query Text and Output Data from Database	16
3.5 Query Substitution Parameters and Seeds Used	16
3.6 Query Isolation Level	16
3.7 Refresh Function Implementation	17
4 Clause 3: Database System Properties Related Items	18
4.1 Atomicity Requirements	18
4.1.1 Atomicity of Completed Transactions	18
4.1.2 Atomicity of Aborted Transactions	18
4.2 Consistency Requirements	18
4.2.1 Consistency Condition	18
4.2.2 Consistency Tests	19
4.3 Isolation Requirements	19
4.3.1 Isolation Test 1	19
4.3.2 Isolation Test 2	19
4.3.3 Isolation Test 3	20
4.3.4 Isolation Test 4	20
4.3.5 Isolation Test 5	20
4.3.6 Isolation Test 6	21
4.4 Durability Requirements	21
4.4.1 Failure of a Durable Medium	21
4.4.2 Loss of Log and System Crash	21
4.4.3 System Crash	22
4.4.4 Memory Failure	22
5 Clause 4: Scaling and Database Population Related Items	23
5.1 Cardinality of Tables	23
5.2 Distribution of Tables and Logs	23
5.3 Database Partition / Replication Mapping	28
5.4 RAID Implementation	28
5.5 DBGEN Modifications	29
5.6 Database Load Time	29
5.7 Data Storage Ratio	29
5.8 Database Load Mechanism Details and Illustration	29
5.9 Qualification Database Configuration	30
6 Clause 5: Performance Metrics and Execution Rules Related Items	31
6.1 System Activity between Load and Performance Tests	31
6.2 Steps in the Power Test	31
6.3 Timing Intervals for Each Query and Refresh Function	31

6.4 Number of Streams for the Throughput Test	31
6.5 Start and End Date/Times for Each Query Stream	31
6.6 Total Elapsed Time for the Measurement Interval	31
6.7 Refresh Function Start Date/Time and Finish Date/Time	31
6.8 Timing Intervals for Each Query and Each Refresh Function for Each Stream	32
6.9 Performance Metrics	32
6.10 Performance Metric and Numerical Quantities from Both Runs	32
6.11 System Activity between Tests	32
7 Clause 6: SUT and Driver Implementation Related Items	33
7.1 Driver	33
7.2 Implementation-Specific Layer	33
7.3 Profile-Directed Optimization	33
8 Clause 7: Pricing Related Items	34
8.1 Hardware and Software Components	34
8.2 Three-Year Cost of System Configuration	34
8.3 Availability Dates	34
8.4 Country-Specific Pricing	34
Clause 9: Audit Related Items	35
9.1 Auditor's Report	35
Appendix A: Tunable Parameters and System Configuration	38
DB2 UDB 8.1 Database and Database Manager Configuration	38
DB2 Version	43
DB2 Registry Variables	43
Microsoft Windows Server 2003 Enterprise Edition	43
<i>Configuration Parameters</i>	43
<i>SUT Hardware Information Report</i>	43
Appendix B: Database Build Scripts	76
affinity_8mln_4P	76
backuptestdb	76
buildtpcd	76
clrlogs.bat	84
create_bufferpools	84
create_indexes	84
create_nodegroups	85
create_tables	85
create_tablespaces	86
createUFtbls	87
dss.runstats	88
load.db2set_8mln.bat	88
load.dbcfg_8mln	88
load_dbmcfg_8mln	88
load_all.sql	89
loadcfg.sql	89
load_8mln.bat	90
run.db2set_8mln.bat	90
run.dbcfg_8mln	90
run.dbmcfg_8mln	90
runstats_UF.bat	90
scattered_read	90
setlogs.bat	90
temp_4K.bat	91
tpcd.setup	91
Qualification Database	93
2mln_qual_load.sql	93
affinity_2mln.bat	94

<i>bkupqual2.bat</i>	94
<i>clrqlogs.bat</i>	94
<i>create_mln_instance_qual.bat</i>	94
<i>create_tablespace_2mln_qual</i>	94
<i>go_qual.bat</i>	95
<i>load_2mln_qual.bat</i>	95
<i>load_dbcfg_qual</i>	95
<i>load_qual.bat</i>	95
<i>qual_untag.bat</i>	95
<i>run.db2set.qual.bat</i>	95
<i>run.dbcfg_qual</i>	95
<i>run.dbmcfg_qual</i>	96
<i>setlogs_qual.bat</i>	96
<i>temp_4K.bat</i>	96
<i>tpcd.setup</i>	96
Appendix C: Qualification Query Output	99
Qualification Queries	99
<i>Query 1</i>	99
<i>Query 2</i>	99
<i>Query 3</i>	100
<i>Query 4</i>	100
<i>Query 5</i>	101
<i>Query 6</i>	101
<i>Query 7</i>	101
<i>Query 8</i>	102
<i>Query 9</i>	102
<i>Query 10</i>	103
<i>Query 11</i>	104
<i>Query 12</i>	105
<i>Query 13</i>	105
<i>Query 14</i>	106
<i>Query 15a</i>	106
<i>Query 16</i>	106
<i>Query 17</i>	107
<i>Query 18</i>	107
<i>Query 19</i>	108
<i>Query 20</i>	108
<i>Query 21</i>	109
<i>Query 22</i>	109
First 10 Rows of the Database	110
Query Substitution Parameters	113
Appendix D: Driver Source Code	117
<i>doufload_v8.bat</i>	117
<i>load_UF1_data_V8</i>	117
<i>load_UF2_data_V8</i>	118
<i>loadSampleUFData</i>	118
<i>runpower</i>	119
<i>runthroughput</i>	122
<i>tpcd_cl.bat</i>	124
<i>tpcdbatch.h</i>	125
<i>tpcdbatch.sqc</i>	126
<i>tpcdUF.sqc</i>	152
Appendix E: ACID Transaction Source Code	158
<i>acid.h</i>	158
<i>acid.sqc</i>	158

makefile	168
Appendix F: Price Quotations	169

Preface

TPC Benchmark H Standard Specification was developed by the Transaction Processing Performance Council (TPC). It was released on February 26, 1999, and most recently revised (Revision 2.0) October 29, 2002. This is the full disclosure report for benchmark testing of the IBM *@server* xSeries 365 according to the TPC Benchmark H Standard Specification.

The TPC Benchmark H is a decision support benchmark. It consists of a suite of business-oriented ad hoc queries and concurrent data modifications. The queries and the data populating the database have been chosen to have broad industrywide relevance while maintaining a sufficient degree of ease of implementation. This benchmark illustrates decision support systems that:

- Examine large volumes of data;
- Execute queries with a high degree of complexity;
- Give answers to critical business questions.

TPC-H evaluates the performance of various decision support systems by the execution of set of queries against a standard database under controlled conditions. The TPC-H queries:

- Give answers to real-world business questions;
- Simulate generated ad-hoc queries (e.g., via a point-and-click GUI interface);
- Are far more complex than most OLTP transactions;
- Include a rich breadth of operators and selectivity constraints;
- Generate intensive activity on the part of the database server component of the system under test;
- Are executed against a database complying with specific population and scaling requirements;
- Are implemented with constraints derived from staying closely synchronized with an on-line production database.

The TPC-H operations are modeled as follows:

- The database is continuously available 24 hours a day, 7 days a week, for ad-hoc queries from multiple end users and data modifications against all tables, except possibly during infrequent (e.g., once a month) maintenance sessions.
- The TPC-H database tracks, possibly with some delay, the state of the OLTP database through ongoing refresh functions, which batch together a number of modifications impacting some part of the decision support database.
- Due to the worldwide nature of the business data stored in the TPC-H database, the queries and the refresh functions may be executed against the database at any time, especially in relation to each other. In addition, this mix of queries and refresh functions is subject to specific ACIDity requirements, since queries and refresh functions may execute concurrently.
- To achieve the optimal compromise between performance and operational requirements, the database administrator can set, once and for all, the locking levels and the concurrent scheduling rules for queries and refresh functions.

The minimum database required to run the benchmark holds business data from 10,000 suppliers. It contains almost 10 million rows representing a raw storage capacity of about 1 gigabyte. Compliant benchmark implementations may also use one of the larger permissible database populations (e.g., 100 gigabytes), as defined in Clause 4.1.3).

The performance metrics reported by TPC-H is called the TPC-H Composite Query-per-Hour Performance Metric (QphH@Size), and reflects multiple aspects of the capability of the system to process queries. These aspects include the selected database size against which the queries are executed, the query processing power when queries are submitted by a single stream, and the query throughput when queries are submitted by multiple concurrent users. The TPC-H Price/Performance metric is expressed as \$/QphH@Size. To be compliant with the TPC-H standard, all references to TPC-H results for a given configuration must include all required reporting components (see Clause 5.4.6). The TPC believes that comparisons of TPC-H results measured against different database sizes are misleading and discourages such comparisons.

The TPC-H database must be implemented using a commercially available database management system (DBMS), and the queries executed via an interface using dynamic SQL. The specification provides for variants of SQL, as implementers are not required to have implemented a specific SQL standard in full.

Benchmarks 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-H should not be used as a substitute for specific customer application benchmarking when critical capacity planning and/or product evaluation decisions are contemplated.

1 General Items

1.1 Benchmark Sponsor

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

IBM Corporation sponsored this TPC-H benchmark.

1.2 Parameter Settings

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

- Database tuning options
- Optimizer/Query execution options
- Query Processing tool/language configuration parameters
- Recovery/commit options
- Consistency/locking options
- Operating system and configuration parameters
- Configuration parameters and options for any other software component incorporated into the pricing structure
- Compiler optimization options.

Appendix A, “Tunable Parameters,” contains a list of all DB2 parameters and operating system parameters. Session initialization parameters can be set during or immediately after establishing the connection to the database within the tpcdbatch program documented in Appendix D, “Driver Source Code.” This result uses the default session initialization parameters established during preprocessing/binding of the tpcdbatch program.

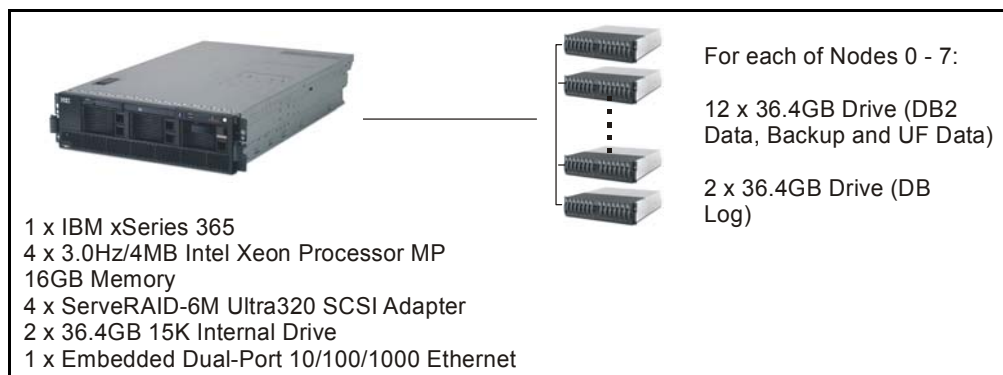
1.3 Configuration Diagrams

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 and type of disk units (and controllers, if applicable)
- Number and type of disk units (and controllers, if applicable)
- Number of channels or bus connections to disk units, including their protocol type
- Number of LAN (e.g., Ethernet) connections, including routers, workstations, terminals, etc., that were physically used in the test or are incorporated into the pricing structure
- Type and run-time execution location of software components (e.g., DBMS, query processing tools/languages, middleware components, software drivers, etc.).

The configuration diagram for the tested and priced system is provided on the following page.

1.3.1 Priced and Measured Configurations



The priced configuration for the xSeries 365 contained:

- Four Intel Xeon MP 3.0GHz processors, each with 4MB of ECC L3 cache
- 16GB of memory
- One embedded dual-port 10/100/1000 Ethernet interface
- Four ServeRAID-6M Ultra320 SCSI adapters
- One hundred twelve (112) 36.4GB 15K Ultra320 SCSI disk drives
- Eight (8) EXP400 Storage Expansion Enclosures
- Two (2) 36.4GB 15K Ultra320 SCSI Disk Drives

The measured configuration was different in that it contained one additional EXP300 Storage Expansion unit, which held fourteen 36.4GB 15K Ultra320 SCSI drives that held raw data files. A fifth ServeRAID adapter was used to attach these drives. This hardware was not priced.

2 Clause 1: Logical Database Design Related Items

2.1 Database Table Definitions

Listings must be provided for all table definition statements and all other statements used to set up the test and qualification databases. (8.1.2.1)

Appendix B contains the scripts that were used to set up the TPC-H test and qualification databases.

2.2 Database Organization

The physical organization of tables and indexes within the test and qualification databases must be disclosed. If the column ordering of any table is different from that specified in Clause 1.4, it must be noted.

Appendix B contains the scripts that were used to create the indexes on the test and qualification databases.

2.3 Horizontal Partitioning

Horizontal partitioning of tables and rows in the test and qualification databases must be disclosed (see Clause 1.5.4).

Horizontal partitioning was used for all tables except for the nation and region tables. See Appendix B, “Database Build Scripts.”

2.4 Replication

Any replication of physical objects must be disclosed and must conform to the requirements of Clause 1.5.6).

Replication was not used.

3 Clause 2: Queries and Update Functions Related Items

3.1 Query Language

The query language used to implement the queries must be identified.

SQL was the query language used.

3.2 Random Number Generation

The method of verification for the random number generation must be described unless the supplied DBGEN and QGEN were used.

The TPC-supplied DBGEN version 1.3.0 and QGEN version 1.3.0 were used to generate all database populations.

3.3 Substitution Parameters Generation

The method used to generate values for substitution parameters must be disclosed. If QGEN is not used for this purpose, then the source code of any non-commercial tool used must be disclosed. If QGEN is used, the version number, release number, modification number and patch level of QGEN must be disclosed.

The supplied QGEN version 1.3.0 was used to generate the substitution parameters.

3.4 Query Text and Output Data from Database

The executable query text used for query validation must be disclosed along with the corresponding output data generated during the execution of the query text against the qualification database. If minor modifications (see Clause 2.2.3) have been applied to any functional query definitions or approved variants in order to obtain executable query text, these modifications must be disclosed and justified. The justification for a particular minor query modification can apply collectively to all queries for which it has been used. The output data for the power and throughput tests must be made available electronically upon request.

Appendix C contains the output for each of the qualification queries. The functional query definitions and variants used in this disclosure use the following minor query modifications:

- Table names and view names are fully qualified. For example, the nation table is referred to as "TPCD.NATION."
- The standard IBM SQL date syntax is used for date arithmetic. For example, DATE('1996-01-01')+3 MONTHS.
- The semicolon (;) is used as a command delimiter.

3.5 Query Substitution Parameters and Seeds Used

All query substitution parameters used for all performance tests must be disclosed in tabular format, along with the seeds used to generate these parameters.

Appendix C contains the seed and query substitution parameters used.

3.6 Query Isolation Level

The isolation level used to run the queries must be disclosed. If the isolation level does not map closely to one of the isolation levels defined in Clause 3.4, additional descriptive detail must be provided.

The isolation level used to run the queries was "repeatable read."

3.7 Refresh Function Implementation

The details of how the refresh functions were implemented must be disclosed (including source code of any non-commercial program used).

The refresh functions are part of the implementation-specific layer/driver code included in Appendix D, “Driver Source Code.”

4 Clause 3: Database System Properties Related Items

The results of the ACID tests must be disclosed, along with a description of how the ACID requirements were met. This includes disclosing the code written to implement the ACID Transaction and Query.

All ACID tests were conducted according to specifications. The Atomicity, Isolation, Consistency and Durability tests were performed on the xSeries 365 server. Appendix E contains the ACID transaction source code.

4.1 Atomicity Requirements

The system under test must guarantee that 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.

4.1.1 Atomicity of Completed Transactions

Perform the ACID transactions for a randomly selected set of input data and verify that the appropriate rows have been changed in the ORDER, LINEITEM and HISTORY tables.

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

1. The total price from the ORDER table and the extended price from the LINEITEM table were retrieved for a randomly selected order key. The number of records in the HISTORY table was also retrieved.
2. The ACID Transaction T1 was executed for the order key used in step 1.
3. The total price and extended price were retrieved for the same order key used in step 1 and step 2. It was verified that:
$$T1.EXTENDEDPRICE=OLD.EXTENDEDPRICE+((T1.DELTA)*$$
$$(OLD.EXTENDEDPRICE/OLD.QUANTITY)), T1.TOTALPRICE=OLD.TOTALPRICE+$$
$$((T1.EXTENDEDPRICE-OLD.EXTENDEDPRICE)*(1-DISCOUNT))*(1+TAX)),$$
 and that the number of records in the History table had increased by 1.

4.1.2 Atomicity of Aborted Transactions

Perform the ACID transactions for a randomly selected set of input data, and verify that the appropriate rows have been changed in the ORDER, LINEITEM and HISTORY tables.

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

1. The ACID application is passed a parameter to execute a rollback of the transaction instead of performing the commit.
2. The total price from the ORDER table and the extended price from the LINEITEM table were retrieved for a random order key. The number of records in the HISTORY table was also retrieved.
3. The ACID transaction was executed for the orderkey used in step 2. The transaction was rolled back.
4. The total price and the extended price were retrieved for the same orderkey used in step 2 and step 3. It was verified that the extended price and the total price were the same as in step 2.

4.2 Consistency Requirements

Consistency is the property of the application that requires any execution of transactions to take the database from one consistent state to another.

4.2.1 Consistency Condition

A consistent state for the TPC-H database is defined to exist when:

$$O_TOTALPRICE=SUM(L_EXTENDEDPRICE*(1-L_DISCOUNT)*(1+L_TAX))$$

for each ORDER and LINEITEM defined by (O_ORDERKEY=L_ORDERKEY)

The following queries were executed before and after a measurement to show that the database was always in a consistent state both initially and after a measurement.

```
SELECT DECIMAL(SUM(DECIMAL(INTEGER(INTEGER(DECIMAL
(INTEGER(100*DECIMAL(L_EXTENDEDPRICE,20,2)),20,3)*
(1-L_DISCOUNT))*(1+L_TAX)),20,3)/100.0),20,3)
FROM TPCD.LINEITEM WHERE L_ORDEYKEY=okey
SELECT DECIMAL(SUM(O_TOTALPRICE,20,3)) from TPCD.ORDERS WHERE O_ORDERKEY =
okey
```

4.2.2 Consistency Tests

Verify that the ORDER and LINEITEM tables are initially consistent as defined in Clause 3.3.2.1, based on a random sample of at least 10 distinct values of O_ORDERKEY.

The queries defined in 4.2.1, “Consistency Condition,” were run after initial database build and prior to executing the ACID transaction. The queries showed that the database was in a consistent condition.

After executing 7 streams of 100 ACID transactions each, the queries defined in 4.2.1, “Consistency Condition,” were run again. The queries showed that the database was still in a consistent state.

4.3 Isolation Requirements

4.3.1 Isolation Test 1

This test demonstrates isolation for the read-write conflict of a read-write transaction and a read-only transaction when the read-write transaction is committed.

The following steps were performed to satisfy the test of isolation for a read-only and a read-write committed transaction:

1. First session: Start an ACID transaction with a randomly selected O_KEY, L_KEY and DELTA. The transaction is delayed for 60 seconds just prior to the Commit.
2. Second session: Start an ACID query for the same O_KEY as in the ACID transaction.
3. Second session: The ACID query attempts to read the file but is locked out by the ACID transaction waiting to complete.
4. First session: The ACID transaction is released and the Commit is executed releasing the record. With the LINEITEM record now released, the ACID query can now complete.
5. Second session: Verify that the ACID query delays for approximately 60 seconds and that the results displayed for the ACID query match the input for the ACID transaction.

4.3.2 Isolation Test 2

This test demonstrates isolation for the read-write conflict of read-write transaction and read-only transaction when the read-write transaction is rolled back.

The following steps were performed to satisfy the test of isolation for read-only and a rolled back read-write transaction:

1. First session: Perform the ACID transaction for a random O_KEY, L_KEY and DELTA. The transaction is delayed for 60 seconds just prior to the Rollback.
2. Second session: Start an ACID query for the same O_KEY as in the ACID transaction. The ACID query attempts to read the LINEITEM table but is locked out by the ACID transaction.
3. First session: The ACID transaction is released and the Rollback is executed, releasing the read.

4. Second session: With the LINEITEM record now released, the ACID query completes.

4.3.3 Isolation Test 3

This test demonstrates isolation for the write-write conflict of two refresh transactions when the first transaction is committed.

The following steps were performed to verify isolation of two refresh transactions:

1. First session: Start an ACID transaction T1 for a randomly selected O_KEY, L_KEY and DELTA. The transaction is delayed for 60 seconds just prior to the COMMIT.
2. Second session: Start a second ACID transaction T2 for the same O_KEY, L_KEY, and for a randomly selected DELTA2. This transaction is forced to wait while the 1st session holds a lock on the LINEITEM record requested by the second session.
3. First session: The ACID transaction T1 is released and the Commit is executed, releasing the record. With the LINEITEM record now released, the ACID transaction T2 can now complete.
4. Verify that:

$$T2.L_EXTENDEDPRICE = T1.L_EXTENDEDPRICE + DELTA * \\ (T1.L_EXTENDEDPRICE) / T1.L_QUANTITY)$$

4.3.4 Isolation Test 4

This test demonstrates isolation for write-write conflict of two ACID transactions when the first transaction is rolled back.

The following steps were performed to verify the isolation of two ACID transactions after the first one is rolled back:

1. First session: Start an ACID transaction T1 for a randomly selected O_KEY, L_KEY, and DELTA. The transaction is delayed for 60 seconds just prior to the rollback.
2. Second session: Start a second ACID transaction T2 for the same O_KEY, L_KEY used by the 1st session. This transaction is forced to wait while the 1st session holds a lock on the LINEITEM record requested by the second session.
3. First session: Rollback the ACID transaction T1. With the LINEITEM record now released, the ACID transaction T2 completes.
4. Verify that $T2.L_EXTENDEDPRICE = T1.L_EXTENDEDPRICE$

4.3.5 Isolation Test 5

This test demonstrates the ability of read and write transactions affecting different database tables to make progress concurrently.

1. First session: Start an ACID transaction, T1, for a randomly selected O_KEY, L_KEY and DELTA. The ACID transaction was suspended prior to COMMIT.
2. First session: Start a second ACID transaction, T2, which selects random values of PS_PARTKEY and PS_SUPPKEY and returns all columns of the PARTSUPP table for which PS_PARTKEY and PS_SUPPKEY are equal to the selected values.
3. T2 completed.
4. T1 was allowed to complete.
5. It was verified that the appropriate rows in the ORDERS, LINEITEM and HISTORY tables have been changed.

4.3.6 Isolation Test 6

This test demonstrates that the continuous submission of arbitrary (read-only) queries against one or more tables of the database does not indefinitely delay refresh transactions affecting those tables from making progress.

1. First session: A transaction T1, which executes modified TPC-H query 1 with DELTA=0, was started.
2. Second session: Before T1 completed, an ACID transaction T2, with randomly selected values of O_KEY, L_KEY and DELTA, was started.
3. Third session: Before T1 completed, a transaction T3, which executes modified TPC-H query 1 with a randomly selected value of DELTA (not equal to 0), was started.
4. T1 completed.
5. T2 completed.
6. T3 completed.
7. It was verified that the appropriate rows in the ORDERS, LINEITEM and HISTORY tables were changed.

4.4 Durability Requirements

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

4.4.1 Failure of a Durable Medium

Guarantee the database and committed updates are preserved across a permanent irrecoverable failure of any single durable medium containing TPC-H database tables or recovery log tables.

The database log was stored on RAID-1 protected storage. The tables for the database were stored on RAID-0 storage, with the exception of the Nation and Region tables, which were stored on an internal drive. A backup of the database was taken to a separate array for drives than those used for the database.

The tests were conducted on the qualification database. The steps performed are shown below.

1. The complete database was backed up once . The backup of the data was not on the same drive as the data itself.
2. Seven streams of ACID transactions were started. Each stream executed a minimum of 100 transactions.
3. One physical drive of a RAID-0 data volume was removed.
4. The seven streams of ACID transactions failed and recorded their number of committed transactions in success files.
5. The failed disk was replaced with a new drive. The database data partitions containing the failed disk were deleted and recreated. The junction points were recreated.
6. A database restore was issued using the backup taken at the beginning of this test.
7. A command was issued causing the database to run through its roll-forward recovery.
8. The counts in the success files and the HISTORY table count were compared and were found to match.

4.4.2 Loss of Log and System Crash

Guarantee the database and committed updates are preserved across a permanent irrecoverable failure of any single durable medium containing TPC-H database tables or recovery log tables.

1. Seven streams of ACID transactions were started. Each stream executed a minimum of 100 transactions.
2. While the test was running, one of the disks from the database RAID-1 log on Node 0 was removed.
3. The test continued running for approximately an additional 30 transactions per stream.
4. Then the system was powered off.
5. When the power was restored, the system rebooted and the database was restarted.
6. The database went through a recovery period.

7. The success file and the HISTORY table counts were compared. One more transaction was in the History table than was in the success file. This is known as an “in-flight” transaction.
8. The database log disk was replaced and a rebuild function was initiated to restore the RAID-1 log array to its protected status. The rebuild completed successfully.

4.4.3 System Crash

Guarantee the database and committed updates are preserved across an instantaneous interruption (system crash/system hang) in processing which requires the system to reboot to recover.

This test was combined with the Loss of Log test. See the previous section.

4.4.4 Memory Failure

Guarantee the database and committed updates are preserved across failure of all or part of memory (loss of contents).

See the previous section.

5 Clause 4: Scaling and Database Population Related Items

5.1 Cardinality of Tables

The cardinality (e.g., the number of rows) of each table of the test database, as it existed at the completion of the database load (see Clause 4.2.5), must be disclosed.

Table Name	Rows
Order	450,000,000
Lineitem	1,799,989,091
Customer	45,000,000
Part	60,000,000
Supplier	3,000,000
Partsupp	240,000,000
Nation	25
Region	5

5.2 Distribution of Tables and Logs

The distribution of tables and logs across all media must be explicitly described.

The following series of tables shows the distribution of tables and logs across all media.

Controller	Drives	Partition*	Size	Use
Internal ServeRAID - 1	1 - 36.4GB	Physical Drive 0	33.9GB	NTFS C:\ Compiler, Operating System
	1 - 36.4GB	Physical Drive 1	33.9GB	NTFS D:\DB2, TPC-H Kit Small Tables
	6 - 36.4GB RAID-0	Physical Drive 2; Logical Node 4	18000MB	Lineitem Data
			3000MB	Lineitem Indexes
			8000MB	Other Tables Data
			2000MB	Other Indexes
			35000MB	Temp Tables
			35000MB	Temp Tables
			78.13GB	O: Stripeset Backups RF Raw Data
			26.65GB	Free Space
	6 - 36.4GB RAID-0	Physical Drive 3; Logical Node 5	18000MB	Lineitem Data
			3000MB	Lineitem Indexes
			8000MB	Other Tables Data
			2000MB	Other Indexes
			35000MB	Temp Tables
			35000MB	Temp Tables
			78.13GB	O: Stripeset Backups RF Raw Data
			26.65GB	Free Space
	2 - 36.4GB RAID-1	Physical Drive 4; Logical Node 4	33.9GB	NTFS E:\logs
	6 - 36.4GB RAID-0	Physical Drive 5; Logical Node 6	18000MB	Lineitem Data
			3000MB	Lineitem Indexes
			8000MB	Other Tables Data
			2000MB	Other Indexes
			35000MB	Temp Tables
			35000MB	Temp Tables
			78.13GB	O: Stripeset Backups RF Raw Data
			26.65GB	Free Space
	6 - 36.4GB RAID-0	Physical Drive 6; Logical Node 7	18000MB	Lineitem Data
			3000MB	Lineitem Indexes
			8000MB	Other Tables Data
			2000MB	Other Indexes
			35000MB	Temp Tables
			35000MB	Temp Tables
			78.13GB	O: Stripeset Backups RF Raw Data
			26.65GB	Free Space
	2 - 36.4GB RAID-1	Physical Drive 7; Logical Node 5	33.9GB	NTFS F:\logs

Controller	Drives	Partition*	Size	Use
ServeRAID - 2	14 -36.4GB RAID-5	Physical Drive 8;	1000MB 439.74GB	Unused NTFS W: Raw Data Files
ServeRAID - 3	6 - 36.4GB RAID-0	Physical Drive 9; Logical Node 4	18000MB 3000MB 8000MB 2000MB 35000MB 35000MB 78.13GB	Lineitem Data Lineitem Indexes Other Tables Data Other Indexes Temp Tables Temp Tables O:\Stripeset Backups RF Raw Data Free Space
			26.65GB	
	6 - 36.4GB RAID-0	Physical Drive 10; Logical Node 5	18000MB 3000MB 8000B 2000MB 35000MB 35000MB	Lineitem Data Lineitem Indexes Other Table Data Other Table Indexes Temp Tables Temp Tables
			26.65GB	Free Space
	2 - 36.4GB RAID-1	Physical Drive 11; Logical Node 6	33.9GB	NTFS G:\logs
	6 - 36.4GB RAID-0	Physical Drive 12; Logical Node 6	18000MB 3000MB 8000MB 2000MB 35000MB 35000MB	Lineitem Data Lineitem Indexes Other Tables Data Other Indexes Temp Tables Temp Tables
			104.77GB	Free Space
	6 - 36.4GB RAID-0	Physical Drive 13; Logical Node 7	18000MB 3000MB 8000MB 2000MB 35000MB 35000MB	Lineitem Data Lineitem Indexes Other Tables Data Other Indexes Temp Tables Temp Tables
			104.77GB	Free Space
	2 - 36.4GB RAID-1	Physical Drive 14; Logical Node 7	33.9GB	NTFS H:\logs

Controller	Drives	Partition*	Size	Use
ServeRAID - 4	6 - 36.4GB RAID-0	Physical Drive 15; Logical Node 0	18000MB	Lineitem Data
			3000MB	Lineitem Indexes
			8000MB	Other Tables Data
			2000MB	Other Indexes
			35000MB	Temp Tables
			35000MB	Temp Tables
			78.13GB	P: Stripeset
				Backups
	6 - 36.4GB RAID-0	Physical Drive 16; Logical Node 1		RF Raw Data
			1000MB	Lineitem Data (Qual)
			400MB	Lineitem Indexes (Qual)
			800MB	Other Tables Data (Qual)
			300MB	Other Indexes (Qual)
			2450MB	Temp Tables (Qual)
			2450MB	Temp Tables (Qual)
			19.32GB	Free Space
	2 - 36.4GB RAID-1	Physical Drive 17; Logical Node 0	18000MB	Lineitem Data
			3000MB	Lineitem Indexes
			8000MB	Other Tables Data
			2000MB	Other Indexes
			35000MB	Temp Tables
			35000MB	Temp Tables
			78.13GB	O: Stripeset
				Backups
	6 - 36.4GB RAID-0	Physical Drive 18; Logical Node 2		RF Raw Data
			1000MB	Lineitem Data (Qual)
			400MB	Lineitem Indexes (Qual)
			800MB	Other Tables Data (Qual)
			300MB	Other Indexes (Qual)
			2450MB	Temp Tables (Qual)
			2450MB	Temp Tables (Qual)
			19.32GB	Free Space
	6 - 36.4GB RAID-0	Physical Drive 19; Logical Node 3	33.9GB	NTFS I:\Logs
			18000MB	Lineitem Data
			3000MB	Lineitem Indexes
			8000MB	Other Tables Data
			2000MB	Other Indexes
			35000MB	Temp Tables
			35000MB	Temp Tables
			78.13GB	O: Stripeset
	2 - 36.4GB RAID-1	Physical Drive 20; Logical Node 1		Backups
				RF Raw Data
			26.65GB	Free Space
			18000MB	Lineitem Data
			3000MB	Lineitem Indexes
			8000MB	Other Table Data
			2000MB	Other Table Indexes
			35000MB	Temp Tables
	6 - 36.4GB RAID-0	Physical Drive 19; Logical Node 3	35000MB	Temp Tables
			78.13GB	O: Stripeset
				Backup
				RF Raw Data
			26.65GB	Free Space
			33.9GB	NTFS J:\Logs

Controller	Drives	Partition*	Size	Use
ServeRAID - 5	6 - 36.4GB RAID-0	Physical Drive 21; Logical Node 0	18000MB	Lineitem Data
			3000MB	Lineitem Indexes
			8000MB	Other Tables Data
			2000MB	Other Indexes
			35000MB	Temp Tables
			35000MB	Temp Tables
			78.13GB	P: Stripeset
				Backups
			1000MB	Lineitem Data (Qual)
			400MB	Lineitem Indexes (Qual)
			800MB	Other Tables Data (Qual)
			300MB	Other Indexes (Qual)
	6 - 36.4GB RAID-0	Physical Drive 22; Logical Node 1	2450MB	Temp Tables (Qual)
			2450MB	Temp Tables (Qual)
			19.32GB	Free Space
			18000MB	Lineitem Data
			3000MB	Lineitem Indexes
			8000MB	Other Table Data
			2000MB	Other Table Indexes
			35000MB	Temp Tables
			35000MB	Temp Tables
			78.13GB	P: Stripeset
				Backups
			1000MB	Lineitem Data (Qual)
	2 - 36.4GB RAID-1	Physical Drive 23	400MB	Lineitem Indexes (Qual)
			800MB	Other Tables Data (Qual)
			300MB	Other Indexes (Qual)
			2450MB	Temp Tables (Qual)
			2450MB	Temp Tables (Qual)
			19.32GB	Free Space
			33.9GB	NTFS K\Logs
	6 - 36.4GB RAID-0	Physical Drive 24; Logical Node 2	18000MB	Lineitem Data
			3000MB	Lineitem Indexes
			8000MB	Other Tables Data
			2000MB	Other Indexes
			35000MB	Temp Tables
			35000MB	Temp Tables
			104.77GB	Free Space
			18000MB	Lineitem Data
			3000MB	Lineitem Indexes
			8000MB	Other Tables Data
			2000MB	Other Indexes
			35000MB	Temp Tables
	6 - 36.4GB RAID-0	Physical Drive 25; Logical Node 3	35000MB	Temp Tables
			104.77GB	Free Space
			18000MB	Lineitem Data
			3000MB	Lineitem Indexes
			8000MB	Other Tables Data
			2000MB	Other Indexes
			35000MB	Temp Tables
			35000MB	Temp Tables
			104.77GB	Free Space
			18000MB	Lineitem Data
			3000MB	Lineitem Indexes
			8000MB	Other Tables Data
	2 - 36.4GB RAID-1	Physical Drive 26; Logical Node 3	2000MB	Other Indexes
			35000MB	Temp Tables
			35000MB	Temp Tables
			104.77GB	Free Space
			33.9GB	NTFS L:\Logs

* The physical drives that do not have drive letters are assigned Windows junction points

The priced configuration used 114 disks. An additional 14 disks were used to store the 300GB database files and were not priced.

5.3 Database Partition / Replication Mapping

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

The database was not replicated. The database was logically partitioned into eight logical nodes.

5.4 RAID Implementation

Implementations may use some form of RAID to ensure high availability. If used for data, auxiliary storage (e.g., indexes) or temporary space, the level of RAID must be disclosed for each device.

RAID-1 was used for log disks. RAID-0 was used for all other database disks and the temporary tablespace. The Nation and Region tables were placed on the second (D:\) internal SCSI drive .

5.5 DBGEN Modifications

Any modifications to the DBGEN (see Clause 4.2.1) source code must be disclosed. In the event that a program other than DBGEN was used to populate the database, it must be disclosed in its entirety.

The standard distribution DBGEN version 1.3.0 was used for database population. No modifications were made.

5.6 Database Load Time

The database load time for the test database (see Clause 4.3) must be disclosed.

See the Executive Summary at the beginning of this report.

5.7 Data Storage Ratio

The data storage ratio must be disclosed. It is computed as the ratio between the total amount of priced disk space and the chosen test database size as defined in Clause 4.1.3.

The calculation of the data storage ratio is shown in the following table.

Disk Type	Number of Disks	Formatted Space per Disk	Total Disk Space	Scale Factor	Storage Ratio
36.4GB 15K Ultra160 SCSI Drive	112	33.9GB	3796.8GB		
36.4GB 15K Ultra320 SCSI Drive	2	33.9GB	67.8GB		
Total			3864.6GB	300GB	12.82

The data storage ratio is 12.82, derived by dividing 3864.6GB by the database size of 300GB.

5.8 Database Load Mechanism Details and Illustration

The details of the database load must be disclosed, including a block diagram illustrating the overall process. Disclosure of the load procedure includes all steps, scripts, input and configuration files required to completely reproduce the test and qualification databases.

Flat files for each of the tables were created using DBGEN.

The NATION and REGION tables were created on D:\ and then loaded from dbgen output. The other tables were loaded on the eight logical nodes.

The tables were loaded as depicted in Figure 4-1.

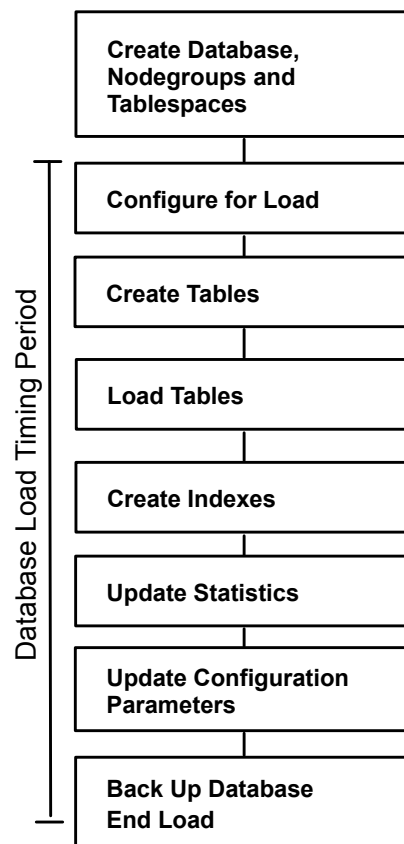


Figure 4-1. Database Load Procedure

5.9 Qualification Database Configuration

Any differences between the configuration of the qualification database and the test database must be disclosed.

The qualification database used identical scripts and disk structure to create and load the data with adjustments for size difference, fewer logical nodes (2), and fewer logical drives were used. Adjustments were also made so that the test database and the qualification database could exist concurrently. The command “SET DB2INSTANCE=QUAL” was used to change from the test database to the qualification database. This setting persisted for the DB2 command session. See Section 5.2 for details.

6 Clause 5: Performance Metrics and Execution Rules Related Items

6.1 System Activity between Load and Performance Tests

Any system activity on the SUT that takes place between the conclusion of the load test and the beginning of the performance test must be fully disclosed.

The auditor requested that queries be run against the database to verify the correctness of the database load.

6.2 Steps in the Power Test

The details of the steps followed to implement the power test (e.g., system reboot, database restart) must be disclosed.

The following steps were used to implement the power test:

1. RF1 Refresh Transaction
2. Stream 00 Execution
3. RF2 Refresh Transaction

6.3 Timing Intervals for Each Query and Refresh Function

The timing intervals for each query of the measured set and for both update functions must be reported for the power test.

See the Numerical Quantities Summary in the Executive Summary at the beginning of this report.

6.4 Number of Streams for the Throughput Test

The number of execution streams used for the throughput test must be disclosed.

Six streams were used for the throughput test.

6.5 Start and End Date/Times for Each Query Stream

The start time and finish time for each query execution stream must be reported for the throughput test.

See the Numerical Quantities Summary in the Executive Summary at the beginning of this report.

6.6 Total Elapsed Time for the Measurement Interval

The total elapsed time for the measurement interval must be reported for the throughput test.

See the Numerical Quantities Summary in the Executive Summary at the beginning of this report..

6.7 Refresh Function Start Date/Time and Finish Date/Time

The start time and finish time for each update function in the update stream must be reported for the throughput test.

See the Numerical Quantities Summary in the Executive Summary at the beginning of this report.

6.8 Timing Intervals for Each Query and Each Refresh Function for Each Stream

The timing intervals for each query of each stream and for each update function must be reported for the throughput test.

See the Numerical Quantities Summary in the Executive Summary at the beginning of this report.

6.9 Performance Metrics

The computed performance metrics, related numerical quantities, and the price/performance metric must be reported.

See the Numerical Quantities Summary in the Executive Summary at the beginning of this report.

6.10 Performance Metric and Numerical Quantities from Both Runs

The performance metric and numerical quantities from both runs must be disclosed.

Two consecutive runs of the TPC-H benchmark were performed. The following table contains the results for both runs.

	QppH @ 300GB	QthH @ 300GB	QphH @ 300GB
Run1	6,463.9	3,872.4	5,003.1
Run2	6,571.8	3,813.9	5,006.4

6.11 System Activity between Tests

Any activity on the SUT that takes place between the conclusion of Run1 and the beginning of Run2 must be disclosed.

DB2 was restarted between runs.

7 Clause 6: SUT and Driver Implementation Related Items

7.1 Driver

A detailed textual description of how the driver performs its functions, how its various components interact and any product functionality or environmental setting on which it relies must be provided. All related source code, scripts and configurations must be disclosed. The information provided should be sufficient for an independent reconstruction of the driver.

Appendix D, “Driver Source Code,” contains the source code used for the driver and all scripts used in connection with it.

The Power test is invoked by calling tpcdbatch with the stream number 0 specified, an indication that the refresh functions must be run, and the SQL file that contains the power stream queries.

The Throughput test is invoked by initiating a call to tpcdbatch for every query stream that will be run. Tpcdbatch gets the stream number for each of the streams, and the SQL file specific to that stream number as the queries to execute. The refresh function is initiated as a separate call to tpcdbatch with the SQL script for the refresh functions and the total number of query streams specified.

7.2 Implementation-Specific Layer

If an implementation-specific layer is used, then a detailed description of how it performs its functions must be supplied, including any related source code or scripts. This description should allow an independent reconstruction of the implementation-specific layer.

The implementation specific layer is a single executable SQL application that uses embedded dynamic SQL to process the EQT generated by QGEN. The application is called tpcdbatch to indicate that it processes a batch of TPC-H queries, although it is completely capable of processing any arbitrary SQL statement (both DML and DDL).

A separate instance of tpcdbatch is invoked for each stream. Each instance establishes a distinct connection to the database server through which the EQT is transmitted to the database and the results are returned through the implementation specific layer to the driver. When an instance of tpcdbatch is invoked, it is provided with a context of whether it is running a power test, query stream or refresh stream, as well as an input file containing the 22 queries and/or refresh functions. tpcdbatch then connects to the database, performs any session initialization as well as preparing output files required by the auditor. Then it proceeds to read from the input file and processes each query or refresh function in turn.

For queries, each query is prepared, described, and a cursor is opened and used to fetch the required number of rows. After the last row has been retrieved a commit is issued. For the refresh functions, during the database build all data is first split for each node using the db2split utility. For RF1, the data for each node is further split into n equal portions for both the lineitem and orders tables taking care that the records for the same orderkey remain in the same set. For RF2, the data for each node is further split into m equal portions. During the run, when tpcdbatch encounters a call to execute RF1, it first calls a shell script which loads these n sets of data into n sets of temporary tables (one each for lineitem and orders). Then tpcdbatch forks off n children to do an insert with subselect into the original lineitem and orders tables. When tpcdbatch encounters a call to execute RF2, it calls a shell script that loads these data into a single staging table. Then tpcdbatch forks off p children (where $p * x = m$) to do x sets of deletes from the orders and lineitem tables with a subselect from the staging table.

7.3 Profile-Directed Optimization

Profile-directed optimization was not used.

8 Clause 7: Pricing Related Items

8.1 Hardware and Software Components

A detailed list of the hardware and software used in the priced system must be reported. Each item must have a 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.

A detailed list of all hardware and software, including the 3-year price, is provided in the Executive Summary at the front of this report. The price quotations are included in Appendix F.

8.2 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 must be disclosed.

A detailed list of all hardware and software, including the 3-year price, is provided in the Executive Summary at the front of this report. The price quotations are included in Appendix F.

8.3 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, availability date reported on the Executive Summary must be the date by which all components are committed to being available. The Full Disclosure Report must report availability dates individually for at least each of the categories for which a pricing subtotal must be provided (see Clause 7.3.1.3).

The total solution as priced is currently available.

8.4 Country-Specific Pricing

Additional Clause 7 related items may be included in the Full Disclosure Report for each country-specific priced configuration. Country-specific pricing is subject to Clause 7.1.7.

The configuration is priced for the United States of America.

Clause 9: Audit Related Items

9.1 Auditor's Report

The auditor's agency name, address, phone number, and Attestation letter with a brief audit summary report indicating compliance must be included in the Full Disclosure Report. A statement should be included specifying who to contact in order to obtain further information regarding the audit process.

This implementation of the TPC Benchmark H was audited by Francois Raab of InfoSizing, Inc. For a copy of this disclosure, go to www.tpc.org.

Benchmark Sponsor:	Haider Rizvi Manager, DB2 Data Warehouse Performance IBM Canada Ltd; 8200 Warden Avenue Markham, Ontario L6G 1C7	Kamran Amini Manager, xSeries Performance IBM Systems and Technology Group 3039 Cornwallis Road Research Triangle Park, NC 27709
--------------------	---	---

May 17, 2004

I verified the TPC Benchmark™ H performance of the following configuration:

Platform: **IBM @server xSeries 365**
 Database Manager: **IBM DB2 UDB 8.1**
 Operating System: **Microsoft Windows Server 2003 Enterprise Edition
 with QFE KB834628**

The results were:

CPU (Speed)	Memory	Disks	QphH@300GB
IBM @server xSeries 365			
4 x Intel Xeon MP (3.0 GHz)	4 MB L3 Cache 16 GB Main	112 x 36.4 GB Ultra320 ext. 2 x 36.4 GB Ultra320 int.	5,003.1

In my opinion, this performance result was produced in compliance with the TPC's requirements for the benchmark. The following verification items were given special attention:

- The database records were defined with the proper layout and size
- The database population was generated using DBGEN
- The database was properly scaled to 300GB and populated accordingly
- The compliance of the database auxiliary data structures was verified

- The database load time was correctly measured and reported
- The required ACID properties were verified and met
- The query input variables were generated by QGEN
- The query text was produced using minor modifications and an approved query variant
- The execution of the queries against the SF1 database produced compliant answers
- A compliant implementation specific layer was used to drive the tests
- The throughput tests involved 6 query streams
- The ratio between the longest and the shortest query was such that no query timing was adjusted
- The execution times for queries and refresh functions were correctly measured and reported
- The repeatability of the measured results was verified
- The required amount of database log was configured
- The system pricing was verified for major components and maintenance
- The major pages from the FDR were verified for accuracy

Additional Audit Notes:

None.

Respectfully Yours,

A handwritten signature in black ink, appearing to read "François Raab", with a long horizontal flourish extending to the right.

François Raab
President

Appendix A: Tunable Parameters and System Configuration

DB2 UDB 8.1 Database and Database Manager Configuration

"DB2 Configuration for Node0"
get db cfg for tpcd

Database Configuration for Database tpcd

Database configuration release level = 0x0a00
Database release level = 0x0a00
Database territory = US
Database code page = 1252
Database code set = IBM-1252
Database country/region code = 1
Dynamic SQL Query management (DYN_QUERY_MGMT) = DISABLE
Discovery support for this database (DISCOVER_DB) = ENABLE
Default query optimization class (DFT_QUERYOPT) = 7
Degree of parallelism (DFT_DEGREE) = 1
Continue upon arithmetic exceptions (DFT_SQLMATHWARN) = NO
Default refresh age (DFT_REFRESH_AGE) = 0
Number of frequent values retained (NUM_FREQVALUES) = 0
Number of quantiles retained (NUM_QUANTILES) = 300
Backup pending = NO
Database is consistent = YES
Rollforward pending = NO
Restore pending = NO
Multi-page file allocation enabled = NO
Log retain for recovery status = RECOVERY
User exit for logging status = NO
Data Links Token Expiry Interval (sec) (DL_EXPINT) = 60
Data Links Write Token Init Expiry Intvl(DL_WT_IEXPINT) = 60
Data Links Number of Copies (DL_NUM_COPIES) = 1
Data Links Time after Drop (days) (DL_TIME_DROP) = 1
Data Links Token in Uppercase (DL_UPPER) = NO
Data Links Token Algorithm (DL_TOKEN) = MAC0
Database heap (4KB) (DBHEAP) = 20000
Size of database shared memory (4KB) (DATABASE_MEMORY) = AUTOMATIC
Catalog cache size (4KB) (CATALOGCACHE_SZ) = 386
Log buffer size (4KB) (LOGBUFSZ) = 2048
Utilities heap size (4KB) (UTIL_HEAP_SZ) = 40000
Buffer pool size (pages) (BUFFPAGE) = 50000
Extended storage segments size (4KB) (ESTORE_SEG_SZ) = 16000
Number of extended storage segments (NUM_ESTORE_SEGS) = 0
Max storage for lock list (4KB) (LOCKLIST) = 16384
Max size of appl. group mem set (4KB) (APPGROUP_MEM_SZ) = 40000
Percent of mem for appl. group heap (GROUPEAP_RATIO) = 70
Max appl. control heap size (4KB) (APP_CTL_HEAP_SZ) = 2048
Sort heap thres for shared sorts (4KB) (SHEAPTHRES_SHR) = (SHEAPTHRES)
Sort list heap (4KB) (SORTHEAP) = 6400
SQL statement heap (4KB) (STMTHEAP) = 10000
Default application heap (4KB) (APPLHEAPSZ) = 16000
Package cache size (4KB) (PCKCACHESZ) = 640
Statistics heap size (4KB) (STAT_HEAP_SZ) = 10000
Interval for checking deadlock (ms) (DLCHKTIME) = 5000
Percent. of lock lists per application (MAXLOCKS) = 5
Lock timeout (sec) (LOCKTIMEOUT) = -1
Changed pages threshold (CHNGPGS_THRESH) = 15
Number of asynchronous page cleaners (NUM_IOCLEANERS) = 4
Number of I/O servers (NUM_IOSERVERS) = 4
Index sort flag (INDEXSORT) = YES
Sequential detect flag (SEQDETECT) = YES
Default prefetch size (pages) (DFT_PREFETCH_SZ) = 16
Track modified pages (TRACKMOD) = OFF
Default number of containers = 1
Default tablespace extentsize (pages) (DFT_EXTENT_SZ) = 32
Max number of active applications (MAXAPPLS) = 40
Average number of active applications (AVG_APPLS) = 1
Max DB files open per application (MAXFILOP) = 1024

Log file size (4KB) (LOGFILSIZ) = 16384
Number of primary log files (LOGPRIMARY) = 20
Number of secondary log files (LOGSECOND) = 2
Changed path to log files (NEWLOGPATH) =
Path to log files = i:\logs\NODE0000\
Overflow log path (OVERFLOWLOGPATH) =
Mirror log path (MIRRORLOGPATH) =
First active log file = S0000036.LOG
Block log on disk full (BLK_LOG_DSK_FUL) = NO
Percent of max active log space by transaction(MAX_LOG) = 0
Num. of active log files for 1 active UOW(NUM_LOG_SPAN) = 0
Group commit count (MINCOMMIT) = 1
Percent log file reclaimed before soft ckpt (SOFTMAX) = 600
Log retain for recovery enabled (LOGRETAIN) = RECOVERY
User exit for logging enabled (USEREXIT) = OFF
Auto restart enabled (AUTORESTART) = ON
Index re-creation time (INDEXREC) = SYSTEM (ACCESS)
Default number of loadrec sessions (DFT_LOADREC_SES) = 1
Number of database backups to retain (NUM_DB_BACKUPS) = 12
Recovery history retention (days) (REC_HIS_RETENTN) = 366
TSM management class (TSM_MGMTCLASS) =
TSM node name (TSM_NODENAME) =
TSM owner (TSM_OWNER) =
TSM password (TSM_PASSWORD) =

"DB2 Configuration for Node1"
get db cfg for tpcd

Database Configuration for Database tpcd

Database configuration release level = 0x0a00
Database release level = 0x0a00
Database territory = US
Database code page = 1252
Database code set = IBM-1252
Database country/region code = 1
Dynamic SQL Query management (DYN_QUERY_MGMT) = DISABLE
Discovery support for this database (DISCOVER_DB) = ENABLE
Default query optimization class (DFT_QUERYOPT) = 7
Degree of parallelism (DFT_DEGREE) = 1
Continue upon arithmetic exceptions (DFT_SQLMATHWARN) = NO
Default refresh age (DFT_REFRESH_AGE) = 0
Number of frequent values retained (NUM_FREQVALUES) = 0
Number of quantiles retained (NUM_QUANTILES) = 300
Backup pending = NO
Database is consistent = YES
Rollforward pending = NO
Restore pending = NO
Multi-page file allocation enabled = NO
Log retain for recovery status = RECOVERY
User exit for logging status = NO
Data Links Token Expiry Interval (sec) (DL_EXPINT) = 60
Data Links Write Token Init Expiry Intvl(DL_WT_IEXPINT) = 60
Data Links Number of Copies (DL_NUM_COPIES) = 1
Data Links Time after Drop (days) (DL_TIME_DROP) = 1
Data Links Token in Uppercase (DL_UPPER) = NO
Data Links Token Algorithm (DL_TOKEN) = MAC0
Database heap (4KB) (DBHEAP) = 20000
Size of database shared memory (4KB) (DATABASE_MEMORY) = AUTOMATIC
Catalog cache size (4KB) (CATALOGCACHE_SZ) = 386
Log buffer size (4KB) (LOGBUFSZ) = 2048
Utilities heap size (4KB) (UTIL_HEAP_SZ) = 40000
Buffer pool size (pages) (BUFFPAGE) = 50000
Extended storage segments size (4KB) (ESTORE_SEG_SZ) = 16000
Number of extended storage segments (NUM_ESTORE_SEGS) = 0
Max storage for lock list (4KB) (LOCKLIST) = 16384
Max size of appl. group mem set (4KB) (APPGROUP_MEM_SZ) = 40000
Percent of mem for appl. group heap (GROUPEAP_RATIO) = 70
Max appl. control heap size (4KB) (APP_CTL_HEAP_SZ) = 2048
Sort heap thres for shared sorts (4KB) (SHEAPTHRES_SHR) = (SHEAPTHRES)
Sort list heap (4KB) (SORTHEAP) = 6400
SQL statement heap (4KB) (STMTHEAP) = 10000
Default application heap (4KB) (APPLHEAPSZ) = 16000
Package cache size (4KB) (PCKCACHESZ) = 640

Statistics heap size (4KB) (STAT_HEAP_SZ) = 10000
 Interval for checking deadlock (ms) (DLCHKTIME) = 5000
 Percent. of lock lists per application (MAXLOCKS) = 5
 Lock timeout (sec) (LOCKTIMEOUT) = -1
 Changed pages threshold (CHNGPGS_THRESH) = 15
 Number of asynchronous page cleaners (NUM_IOCLEANERS) = 4
 Number of I/O servers (NUM_IOSERVERS) = 4
 Index sort flag (INDEXSORT) = YES
 Sequential detect flag (SEQDETECT) = YES
 Default prefetch size (pages) (DFT_PREFETCH_SZ) = 16
 Track modified pages (TRACKMOD) = OFF
 Default number of containers = 1
 Default tablespace extentsize (pages) (DFT_EXTENT_SZ) = 32
 Max number of active applications (MAXAPPLS) = 40
 Average number of active applications (AVG_APPLS) = 1
 Max DB files open per application (MAXFILOP) = 1024
 Log file size (4KB) (LOGFILSIZ) = 16384
 Number of primary log files (LOGPRIMARY) = 20
 Number of secondary log files (LOGSECOND) = 2
 Changed path to log files (NEWLOGPATH) =
 Path to log files = j:\logs\NODE0001\
 Overflow log path (OVERFLOWLOGPATH) =
 Mirror log path (MIRRORLOGPATH) =
 First active log file = S0000032.LOG
 Block log on disk full (BLK_LOG_DSK_FUL) = NO
 Percent of max active log space by transaction (MAX_LOG) = 0
 Num. of active log files for 1 active UOW (NUM_LOG_SPAN) = 0
 Group commit count (MINCOMMIT) = 1
 Percent log file reclaimed before soft ckcpt (SOFTMAX) = 600
 Log retain for recovery enabled (LOGRETAIN) = RECOVERY
 User exit for logging enabled (USEREXIT) = OFF
 Auto restart enabled (AUTORESTART) = ON
 Index re-creation time (INDEXREC) = SYSTEM (ACCESS)
 Default number of loadrec sessions (DFT_LOADREC_SES) = 1
 Number of database backups to retain (NUM_DB_BACKUPS) = 12
 Recovery history retention (days) (REC_HIS_RETENTN) = 366
 TSM management class (TSM_MGMTCLASS) =
 TSM node name (TSM_NODENAME) =
 TSM owner (TSM_OWNER) =
 TSM password (TSM_PASSWORD) =

"DB2 Configuration for Node2"

get db cfg for tpcd

Database Configuration for Database tpcd

Database configuration release level = 0x0a00
 Database release level = 0x0a00
 Database territory = US
 Database code page = 1252
 Database code set = IBM-1252
 Database country/region code = 1
 Dynamic SQL Query management (DYN_QUERY_MGMT) = DISABLE
 Discovery support for this database (DISCOVER_DB) = ENABLE
 Default query optimization class (DFT_QUERYOPT) = 7
 Degree of parallelism (DFT_DEGREE) = 1
 Continue upon arithmetic exceptions (DFT_SQLMATHWARN) = NO
 Default refresh age (DFT_REFRESH_AGE) = 0
 Number of frequent values retained (NUM_FREQVALUES) = 0
 Number of quantiles retained (NUM_QUANTILES) = 300
 Backup pending = NO
 Database is consistent = YES
 Rollforward pending = NO
 Restore pending = NO
 Multi-page file allocation enabled = NO
 Log retain for recovery status = RECOVERY
 User exit for logging status = NO
 Data Links Token Expiry Interval (sec) (DL_EXPINT) = 60
 Data Links Write Token Init Expiry Intvl (DL_WT_IEXPINT) = 60
 Data Links Number of Copies (DL_NUM_COPIES) = 1
 Data Links Time after Drop (days) (DL_TIME_DROP) = 1
 Data Links Token in Uppercase (DL_UPPER) = NO
 Data Links Token Algorithm (DL_TOKEN) = MAC0
 Database heap (4KB) (DBHEAP) = 20000

Size of database shared memory (4KB) (DATABASE_MEMORY) = AUTOMATIC
 Catalog cache size (4KB) (CATALOGCACHE_SZ) = 386
 Log buffer size (4KB) (LOGBUFSZ) = 2048
 Utilities heap size (4KB) (UTIL_HEAP_SZ) = 40000
 Buffer pool size (pages) (BUFFPAGE) = 50000
 Extended storage segments size (4KB) (ESTORE_SEG_SZ) = 16000
 Number of extended storage segments (NUM_ESTORE_SEGS) = 0
 Max storage for lock list (4KB) (LOCKLIST) = 16384
 Max size of appl. group mem set (4KB) (APPGROUP_MEM_SZ) = 40000
 Percent of mem for appl. group heap (GROUPHEAP_RATIO) = 70
 Max appl. control heap size (4KB) (APP_CTL_HEAP_SZ) = 2048
 Sort heap thres for shared sorts (4KB) (SHEAPTHRES_SHR) = (SHEAPTHRES)
 Sort list heap (4KB) (SORTHEAP) = 6400
 SQL statement heap (4KB) (STMTHEAP) = 10000
 Default application heap (4KB) (APPLHEAPSZ) = 16000
 Package cache size (4KB) (PCKCACHESZ) = 640
 Statistics heap size (4KB) (STAT_HEAP_SZ) = 10000
 Interval for checking deadlock (ms) (DLCHKTIME) = 5000
 Percent. of lock lists per application (MAXLOCKS) = 5
 Lock timeout (sec) (LOCKTIMEOUT) = -1
 Changed pages threshold (CHNGPGS_THRESH) = 15
 Number of asynchronous page cleaners (NUM_IOCLEANERS) = 4
 Number of I/O servers (NUM_IOSERVERS) = 4
 Index sort flag (INDEXSORT) = YES
 Sequential detect flag (SEQDETECT) = YES
 Default prefetch size (pages) (DFT_PREFETCH_SZ) = 16
 Track modified pages (TRACKMOD) = OFF
 Default number of containers = 1
 Default tablespace extentsize (pages) (DFT_EXTENT_SZ) = 32
 Max number of active applications (MAXAPPLS) = 40
 Average number of active applications (AVG_APPLS) = 1
 Max DB files open per application (MAXFILOP) = 1024
 Log file size (4KB) (LOGFILSIZ) = 16384
 Number of primary log files (LOGPRIMARY) = 20
 Number of secondary log files (LOGSECOND) = 2
 Changed path to log files (NEWLOGPATH) =
 Path to log files = k:\logs\NODE0002\
 Overflow log path (OVERFLOWLOGPATH) =
 Mirror log path (MIRRORLOGPATH) =
 First active log file = S0000032.LOG
 Block log on disk full (BLK_LOG_DSK_FUL) = NO
 Percent of max active log space by transaction (MAX_LOG) = 0
 Num. of active log files for 1 active UOW (NUM_LOG_SPAN) = 0
 Group commit count (MINCOMMIT) = 1
 Percent log file reclaimed before soft ckcpt (SOFTMAX) = 600
 Log retain for recovery enabled (LOGRETAIN) = RECOVERY
 User exit for logging enabled (USEREXIT) = OFF
 Auto restart enabled (AUTORESTART) = ON
 Index re-creation time (INDEXREC) = SYSTEM (ACCESS)
 Default number of loadrec sessions (DFT_LOADREC_SES) = 1
 Number of database backups to retain (NUM_DB_BACKUPS) = 12
 Recovery history retention (days) (REC_HIS_RETENTN) = 366
 TSM management class (TSM_MGMTCLASS) =
 TSM node name (TSM_NODENAME) =
 TSM owner (TSM_OWNER) =
 TSM password (TSM_PASSWORD) =

"DB2 Configuration for Node3"

get db cfg for tpcd

Database Configuration for Database tpcd

Database configuration release level = 0x0a00
 Database release level = 0x0a00
 Database territory = US
 Database code page = 1252
 Database code set = IBM-1252
 Database country/region code = 1
 Dynamic SQL Query management (DYN_QUERY_MGMT) = DISABLE
 Discovery support for this database (DISCOVER_DB) = ENABLE
 Default query optimization class (DFT_QUERYOPT) = 7
 Degree of parallelism (DFT_DEGREE) = 1
 Continue upon arithmetic exceptions (DFT_SQLMATHWARN) = NO
 Default refresh age (DFT_REFRESH_AGE) = 0

Number of frequent values retained (NUM_FREQVALUES) = 0
 Number of quantiles retained (NUM_QUANTILES) = 300
 Backup pending = NO
 Database is consistent = YES
 Rollforward pending = NO
 Restore pending = NO
 Multi-page file allocation enabled = NO
 Log retain for recovery status = RECOVERY
 User exit for logging status = NO
 Data Links Token Expiry Interval (sec) (DL_EXPINT) = 60
 Data Links Write Token Init Expiry Intvl(DL_WT_IEXPINT) = 60
 Data Links Number of Copies (DL_NUM_COPIES) = 1
 Data Links Time after Drop (days) (DL_TIME_DROP) = 1
 Data Links Token in Uppercase (DL_UPPER) = NO
 Data Links Token Algorithm (DL_TOKEN) = MAC0
 Database heap (4KB) (DBHEAP) = 20000
 Size of database shared memory (4KB) (DATABASE_MEMORY) = AUTOMATIC
 Catalog cache size (4KB) (CATALOGCACHE_SZ) = 386
 Log buffer size (4KB) (LOGBUFSZ) = 2048
 Utilities heap size (4KB) (UTIL_HEAP_SZ) = 40000
 Buffer pool size (pages) (BUFFPAGE) = 50000
 Extended storage segments size (4KB) (ESTORE_SEG_SZ) = 16000
 Number of extended storage segments (NUM_ESTORE_SEGS) = 0
 Max storage for lock list (4KB) (LOCKLIST) = 16384
 Max size of appl. group mem set (4KB) (APPGROUP_MEM_SZ) = 40000
 Percent of mem for appl. group heap (GROUPHEAP_RATIO) = 70
 Max appl. control heap size (4KB) (APP_CTL_HEAP_SZ) = 2048
 Sort heap thres for shared sorts (4KB) (SHEAPTHRES_SHR) = (SHEAPTHRES)
 Sort list heap (4KB) (SORTHEAP) = 6400
 SQL statement heap (4KB) (STMTHEAP) = 10000
 Default application heap (4KB) (APPLHEAPSZ) = 16000
 Package cache size (4KB) (PCKCACHESZ) = 640
 Statistics heap size (4KB) (STAT_HEAP_SZ) = 10000
 Interval for checking deadlock (ms) (DLCHKTIME) = 5000
 Percent. of lock lists per application (MAXLOCKS) = 5
 Lock timeout (sec) (LOCKTIMEOUT) = -1
 Changed pages threshold (CHNGPGS_THRESH) = 15
 Number of asynchronous page cleaners (NUM_IOCLEANERS) = 4
 Number of I/O servers (NUM_IOSERVERS) = 4
 Index sort flag (INDEXSORT) = YES
 Sequential detect flag (SEQDETECT) = YES
 Default prefetch size (pages) (DFT_PREFETCH_SZ) = 16
 Track modified pages (TRACKMOD) = OFF
 Default number of containers = 1
 Default tablespace extentsize (pages) (DFT_EXTENT_SZ) = 32
 Max number of active applications (MAXAPPLS) = 40
 Average number of active applications (AVG_APPLS) = 1
 Max DB files open per application (MAXFILOP) = 1024
 Log file size (4KB) (LOGFILSIZ) = 16384
 Number of primary log files (LOGPRIMARY) = 20
 Number of secondary log files (LOGSECOND) = 2
 Changed path to log files (NEWLOGPATH) =
 Path to log files = I:\logs\NODE0003\
 Overflow log path (OVERFLOWLOGPATH) =
 Mirror log path (MIRRORLOGPATH) =
 First active log file = S0000033.LOG
 Block log on disk full (BLK_LOG_DSK_FUL) = NO
 Percent of max active log space by transaction(MAX_LOG) = 0
 Num. of active log files for 1 active UOW(NUM_LOG_SPAN) = 0
 Group commit count (MINCOMMIT) = 1
 Percent log file reclaimed before soft ckcpt (SOFTMAX) = 600
 Log retain for recovery enabled (LOGRETAIN) = RECOVERY
 User exit for logging enabled (USEREXIT) = OFF
 Auto restart enabled (AUTORESTART) = ON
 Index re-creation time (INDEXREC) = SYSTEM (ACCESS)
 Default number of loadrec sessions (DFT_LOADREC_SES) = 1
 Number of database backups to retain (NUM_DB_BACKUPS) = 12
 Recovery history retention (days) (REC_HIS_RETENTN) = 366
 TSM management class (TSM_MGMTCLASS) =
 TSM node name (TSM_NODENAME) =
 TSM owner (TSM_OWNER) =
 TSM password (TSM_PASSWORD) =

"DB2 Configuration for Node4"

get db cfg for tpcd

Database Configuration for Database tpcd

Database configuration release level = 0x0a00
 Database release level = 0x0a00
 Database territory = US
 Database code page = 1252
 Database code set = IBM-1252
 Database country/region code = 1
 Dynamic SQL Query management (DYN_QUERY_MGMT) = DISABLE
 Discovery support for this database (DISCOVER_DB) = ENABLE
 Default query optimization class (DFT_QUERYOPT) = 7
 Degree of parallelism (DFT_DEGREE) = 1
 Continue upon arithmetic exceptions (DFT_SQLMATHWARN) = NO
 Default refresh age (DFT_REFRESH_AGE) = 0
 Number of frequent values retained (NUM_FREQVALUES) = 0
 Number of quantiles retained (NUM_QUANTILES) = 300
 Backup pending = NO
 Database is consistent = YES
 Rollforward pending = NO
 Restore pending = NO
 Multi-page file allocation enabled = NO
 Log retain for recovery status = RECOVERY
 User exit for logging status = NO
 Data Links Token Expiry Interval (sec) (DL_EXPINT) = 60
 Data Links Write Token Init Expiry Intvl(DL_WT_IEXPINT) = 60
 Data Links Number of Copies (DL_NUM_COPIES) = 1
 Data Links Time after Drop (days) (DL_TIME_DROP) = 1
 Data Links Token in Uppercase (DL_UPPER) = NO
 Data Links Token Algorithm (DL_TOKEN) = MAC0
 Database heap (4KB) (DBHEAP) = 20000
 Size of database shared memory (4KB) (DATABASE_MEMORY) = AUTOMATIC
 Catalog cache size (4KB) (CATALOGCACHE_SZ) = 386
 Log buffer size (4KB) (LOGBUFSZ) = 2048
 Utilities heap size (4KB) (UTIL_HEAP_SZ) = 40000
 Buffer pool size (pages) (BUFFPAGE) = 50000
 Extended storage segments size (4KB) (ESTORE_SEG_SZ) = 16000
 Number of extended storage segments (NUM_ESTORE_SEGS) = 0
 Max storage for lock list (4KB) (LOCKLIST) = 16384
 Max size of appl. group mem set (4KB) (APPGROUP_MEM_SZ) = 40000
 Percent of mem for appl. group heap (GROUPHEAP_RATIO) = 70
 Max appl. control heap size (4KB) (APP_CTL_HEAP_SZ) = 2048
 Sort heap thres for shared sorts (4KB) (SHEAPTHRES_SHR) = (SHEAPTHRES)
 Sort list heap (4KB) (SORTHEAP) = 6400
 SQL statement heap (4KB) (STMTHEAP) = 10000
 Default application heap (4KB) (APPLHEAPSZ) = 16000
 Package cache size (4KB) (PCKCACHESZ) = 640
 Statistics heap size (4KB) (STAT_HEAP_SZ) = 10000
 Interval for checking deadlock (ms) (DLCHKTIME) = 5000
 Percent. of lock lists per application (MAXLOCKS) = 5
 Lock timeout (sec) (LOCKTIMEOUT) = -1
 Changed pages threshold (CHNGPGS_THRESH) = 15
 Number of asynchronous page cleaners (NUM_IOCLEANERS) = 4
 Number of I/O servers (NUM_IOSERVERS) = 4
 Index sort flag (INDEXSORT) = YES
 Sequential detect flag (SEQDETECT) = YES
 Default prefetch size (pages) (DFT_PREFETCH_SZ) = 16
 Track modified pages (TRACKMOD) = OFF
 Default number of containers = 1
 Default tablespace extentsize (pages) (DFT_EXTENT_SZ) = 32
 Max number of active applications (MAXAPPLS) = 40
 Average number of active applications (AVG_APPLS) = 1
 Max DB files open per application (MAXFILOP) = 1024
 Log file size (4KB) (LOGFILSIZ) = 16384
 Number of primary log files (LOGPRIMARY) = 20
 Number of secondary log files (LOGSECOND) = 2
 Changed path to log files (NEWLOGPATH) =
 Path to log files = e:\logs\NODE0004\
 Overflow log path (OVERFLOWLOGPATH) =
 Mirror log path (MIRRORLOGPATH) =
 First active log file = S0000032.LOG
 Block log on disk full (BLK_LOG_DSK_FUL) = NO
 Percent of max active log space by transaction(MAX_LOG) = 0

Num. of active log files for 1 active UOW(NUM_LOG_SPAN)= 0
 Group commit count (MINCOMMIT)= 1
 Percent log file reclaimed before soft ckpt (SOFTMAX)= 600
 Log retain for recovery enabled (LOGRETAIN)= RECOVERY
 User exit for logging enabled (USEREXIT)= OFF
 Auto restart enabled (AUTORESTART)= ON
 Index re-creation time (INDEXREC)= SYSTEM (ACCESS)
 Default number of loadrec sessions (DFT_LOADREC_SES)= 1
 Number of database backups to retain (NUM_DB_BACKUPS)= 12
 Recovery history retention (days) (REC_HIS_RETENTN)= 366
 TSM management class (TSM_MGMTCLASS)=
 TSM node name (TSM_NODENAME)=
 TSM owner (TSM_OWNER)=
 TSM password (TSM_PASSWORD)=

"DB2 Configuration for Node5"

get db cfg for tpcd

Database Configuration for Database tpcd

Database configuration release level = 0x0a00
 Database release level = 0x0a00
 Database territory = US
 Database code page = 1252
 Database code set = IBM-1252
 Database country/region code = 1
 Dynamic SQL Query management (DYN_QUERY_MGMT)= DISABLE
 Discovery support for this database (DISCOVER_DB)= ENABLE
 Default query optimization class (DFT_QUERYOPT)= 7
 Degree of parallelism (DFT_DEGREE)= 1
 Continue upon arithmetic exceptions (DFT_SQLMATHWARN)= NO
 Default refresh age (DFT_REFRESH_AGE)= 0
 Number of frequent values retained (NUM_FREQVALUES)= 0
 Number of quantiles retained (NUM_QUANTILES)= 300
 Backup pending = NO
 Database is consistent = YES
 Rollforward pending = NO
 Restore pending = NO
 Multi-page file allocation enabled = NO
 Log retain for recovery status = RECOVERY
 User exit for logging status = NO
 Data Links Token Expiry Interval (sec) (DL_EXPINT)= 60
 Data Links Write Token Init Expiry Intvl(DL_WT_IEXPINT)= 60
 Data Links Number of Copies (DL_NUM_COPIES)= 1
 Data Links Time after Drop (days) (DL_TIME_DROP)= 1
 Data Links Token in Uppercase (DL_UPPER)= NO
 Data Links Token Algorithm (DL_TOKEN)= MAC0
 Database heap (4KB) (DBHEAP)= 20000
 Size of database shared memory (4KB) (DATABASE_MEMORY)= AUTOMATIC
 Catalog cache size (4KB) (CATALOGCACHE_SZ)= 386
 Log buffer size (4KB) (LOGBUFSZ)= 2048
 Utilities heap size (4KB) (UTIL_HEAP_SZ)= 40000
 Buffer pool size (pages) (BUFFPAGE)= 50000
 Extended storage segments size (4KB) (ESTORE_SEG_SZ)= 16000
 Number of extended storage segments (NUM_ESTORE_SEGS)= 0
 Max storage for lock list (4KB) (LOCKLIST)= 16384
 Max size of appl. group mem set (4KB) (APPGROUP_MEM_SZ)= 40000
 Percent of mem for appl. group heap (GROUPHEAP_RATIO)= 70
 Max appl. control heap size (4KB) (APP_CTL_HEAP_SZ)= 2048
 Sort heap thres for shared sorts (4KB) (SHEAPTHRES_SHR)= (SHEAPTHRES)
 Sort list heap (4KB) (SORTHEAP)= 6400
 SQL statement heap (4KB) (STMTHEAP)= 10000
 Default application heap (4KB) (APPLHEAPSZ)= 16000
 Package cache size (4KB) (PCKCACHESZ)= 640
 Statistics heap size (4KB) (STAT_HEAP_SZ)= 10000
 Interval for checking deadlock (ms) (DLCHKTIME)= 5000
 Percent. of lock lists per application (MAXLOCKS)= 5
 Lock timeout (sec) (LOCKTIMEOUT)= -1
 Changed pages threshold (CHNGPGS_THRESH)= 15
 Number of asynchronous page cleaners (NUM_IOCLEANERS)= 4
 Number of I/O servers (NUM_IOSERVERS)= 4
 Index sort flag (INDEXSORT)= YES
 Sequential detect flag (SEQDETECT)= YES
 Default prefetch size (pages) (DFT_PREFETCH_SZ)= 16

Track modified pages (TRACKMOD)= OFF
 Default number of containers = 1
 Default tablespace extentsize (pages) (DFT_EXTENT_SZ)= 32
 Max number of active applications (MAXAPPLS)= 40
 Average number of active applications (AVG_APPLS)= 1
 Max DB files open per application (MAXFILOP)= 1024
 Log file size (4KB) (LOGFILSIZ)= 16384
 Number of primary log files (LOGPRIMARY)= 20
 Number of secondary log files (LOGSECOND)= 2
 Changed path to log files (NEWLOGPATH)=
 Path to log files = f:\logs\NODE0005\
 Overflow log path (OVERFLOWLOGPATH)=
 Mirror log path (MIRRORLOGPATH)=
 First active log file = S0000033.LOG
 Block log on disk full (BLK_LOG_DSK_FUL)= NO
 Percent of max active log space by transaction(MAX_LOG)= 0
 Num. of active log files for 1 active UOW(NUM_LOG_SPAN)= 0
 Group commit count (MINCOMMIT)= 1
 Percent log file reclaimed before soft ckpt (SOFTMAX)= 600
 Log retain for recovery enabled (LOGRETAIN)= RECOVERY
 User exit for logging enabled (USEREXIT)= OFF
 Auto restart enabled (AUTORESTART)= ON
 Index re-creation time (INDEXREC)= SYSTEM (ACCESS)
 Default number of loadrec sessions (DFT_LOADREC_SES)= 1
 Number of database backups to retain (NUM_DB_BACKUPS)= 12
 Recovery history retention (days) (REC_HIS_RETENTN)= 366
 TSM management class (TSM_MGMTCLASS)=
 TSM node name (TSM_NODENAME)=
 TSM owner (TSM_OWNER)=
 TSM password (TSM_PASSWORD)=

"DB2 Configuration for Node6"

get db cfg for tpcd

Database Configuration for Database tpcd

Database configuration release level = 0x0a00
 Database release level = 0x0a00
 Database territory = US
 Database code page = 1252
 Database code set = IBM-1252
 Database country/region code = 1
 Dynamic SQL Query management (DYN_QUERY_MGMT)= DISABLE
 Discovery support for this database (DISCOVER_DB)= ENABLE
 Default query optimization class (DFT_QUERYOPT)= 7
 Degree of parallelism (DFT_DEGREE)= 1
 Continue upon arithmetic exceptions (DFT_SQLMATHWARN)= NO
 Default refresh age (DFT_REFRESH_AGE)= 0
 Number of frequent values retained (NUM_FREQVALUES)= 0
 Number of quantiles retained (NUM_QUANTILES)= 300
 Backup pending = NO
 Database is consistent = YES
 Rollforward pending = NO
 Restore pending = NO
 Multi-page file allocation enabled = NO
 Log retain for recovery status = RECOVERY
 User exit for logging status = NO
 Data Links Token Expiry Interval (sec) (DL_EXPINT)= 60
 Data Links Write Token Init Expiry Intvl(DL_WT_IEXPINT)= 60
 Data Links Number of Copies (DL_NUM_COPIES)= 1
 Data Links Time after Drop (days) (DL_TIME_DROP)= 1
 Data Links Token in Uppercase (DL_UPPER)= NO
 Data Links Token Algorithm (DL_TOKEN)= MAC0
 Database heap (4KB) (DBHEAP)= 20000
 Size of database shared memory (4KB) (DATABASE_MEMORY)= AUTOMATIC
 Catalog cache size (4KB) (CATALOGCACHE_SZ)= 386
 Log buffer size (4KB) (LOGBUFSZ)= 2048
 Utilities heap size (4KB) (UTIL_HEAP_SZ)= 40000
 Buffer pool size (pages) (BUFFPAGE)= 50000
 Extended storage segments size (4KB) (ESTORE_SEG_SZ)= 16000
 Number of extended storage segments (NUM_ESTORE_SEGS)= 0
 Max storage for lock list (4KB) (LOCKLIST)= 16384
 Max size of appl. group mem set (4KB) (APPGROUP_MEM_SZ)= 40000
 Percent of mem for appl. group heap (GROUPHEAP_RATIO)= 70

Max appl. control heap size (4KB) (APP_CTL_HEAP_SZ) = 2048
 Sort heap thres for shared sorts (4KB) (SHEAPTHRES_SHR) = (SHEAPTHRES)
 Sort list heap (4KB) (SORTHEAP) = 6400
 SQL statement heap (4KB) (STMTHAP) = 10000
 Default application heap (4KB) (APPLHEAPSZ) = 16000
 Package cache size (4KB) (PCKCACHESZ) = 640
 Statistics heap size (4KB) (STAT_HEAP_SZ) = 10000
 Interval for checking deadlock (ms) (DLCHKTIME) = 5000
 Percent. of lock lists per application (MAXLOCKS) = 5
 Lock timeout (sec) (LOCKTIMEOUT) = -1
 Changed pages threshold (CHNGPGS_THRESH) = 15
 Number of asynchronous page cleaners (NUM_IOCLEANERS) = 4
 Number of I/O servers (NUM_IOSERVERS) = 4
 Index sort flag (INDEXSORT) = YES
 Sequential detect flag (SEQDETECT) = YES
 Default prefetch size (pages) (DFT_PREFETCH_SZ) = 16
 Track modified pages (TRACKMOD) = OFF
 Default number of containers = 1
 Default tablespace extentsize (pages) (DFT_EXTENT_SZ) = 32
 Max number of active applications (MAXAPPLS) = 40
 Average number of active applications (AVG_APPLS) = 1
 Max DB files open per application (MAXFILOP) = 1024
 Log file size (4KB) (LOGFILSIZ) = 16384
 Number of primary log files (LOGPRIMARY) = 20
 Number of secondary log files (LOGSECOND) = 2
 Changed path to log files (NEWLOGPATH) =
 Path to log files = g:\logs\NODE0006\
 Overflow log path (OVERFLOWLOGPATH) =
 Mirror log path (MIRRORLOGPATH) =
 First active log file = S0000033.LOG
 Block log on disk full (BLK_LOG_DSK_FUL) = NO
 Percent of max active log space by transaction (MAX_LOG) = 0
 Num. of active log files for 1 active UOW (NUM_LOG_SPAN) = 0
 Group commit count (MINCOMMIT) = 1
 Percent log file reclaimed before soft ckpt (SOFTMAX) = 600
 Log retain for recovery enabled (LOGRETAIN) = RECOVERY
 User exit for logging enabled (USEREXIT) = OFF
 Auto restart enabled (AUTORESTART) = ON
 Index re-creation time (INDEXREC) = SYSTEM (ACCESS)
 Default number of loadrec sessions (DFT_LOADREC_SES) = 1
 Number of database backups to retain (NUM_DB_BACKUPS) = 12
 Recovery history retention (days) (REC_HIS_RETENTN) = 366
 TSM management class (TSM_MGMTCLASS) =
 TSM node name (TSM_NODENAME) =
 TSM owner (TSM_OWNER) =
 TSM password (TSM_PASSWORD) =

"DB2 Configuration for Node7"
 get db cfg for tpcd

Database Configuration for Database tpcd

Database configuration release level = 0x0a00
 Database release level = 0x0a00
 Database territory = US
 Database code page = 1252
 Database code set = IBM-1252
 Database country/region code = 1
 Dynamic SQL Query management (DYN_QUERY_MGMT) = DISABLE
 Discovery support for this database (DISCOVER_DB) = ENABLE
 Default query optimization class (DFT_QUERYOPT) = 7
 Degree of parallelism (DFT_DEGREE) = 1
 Continue upon arithmetic exceptions (DFT_SQLMATHWARN) = NO
 Default refresh age (DFT_REFRESH_AGE) = 0
 Number of frequent values retained (NUM_FREQVALUES) = 0
 Number of quantiles retained (NUM_QUANTILES) = 300
 Backup pending = NO
 Database is consistent = YES
 Rollforward pending = NO
 Restore pending = NO
 Multi-page file allocation enabled = NO
 Log retain for recovery status = RECOVERY
 User exit for logging status = NO
 Data Links Token Expiry Interval (sec) (DL_EXPINT) = 60

Data Links Write Token Init Expiry Intvl (DL_WT_IEXPINT) = 60
 Data Links Number of Copies (DL_NUM_COPIES) = 1
 Data Links Time after Drop (days) (DL_TIME_DROP) = 1
 Data Links Token in Uppercase (DL_UPPER) = NO
 Data Links Token Algorithm (DL_TOKEN) = MAC0
 Database heap (4KB) (DBHEAP) = 20000
 Size of database shared memory (4KB) (DATABASE_MEMORY) = AUTOMATIC
 Catalog cache size (4KB) (CATALOGCACHE_SZ) = 386
 Log buffer size (4KB) (LOGBUFSZ) = 2048
 Utilities heap size (4KB) (UTIL_HEAP_SZ) = 40000
 Buffer pool size (pages) (BUFFPAGE) = 50000
 Extended storage segments size (4KB) (ESTORE_SEG_SZ) = 16000
 Number of extended storage segments (NUM_ESTORE_SEGS) = 0
 Max storage for lock list (4KB) (LOCKLIST) = 16384
 Max size of appl. group mem set (4KB) (APPGROUP_MEM_SZ) = 40000
 Percent of mem for appl. group heap (GROUPHEAP_RATIO) = 70
 Max appl. control heap size (4KB) (APP_CTL_HEAP_SZ) = 2048
 Sort heap thres for shared sorts (4KB) (SHEAPTHRES_SHR) = (SHEAPTHRES)
 Sort list heap (4KB) (SORTHEAP) = 6400
 SQL statement heap (4KB) (STMTHAP) = 10000
 Default application heap (4KB) (APPLHEAPSZ) = 16000
 Package cache size (4KB) (PCKCACHESZ) = 640
 Statistics heap size (4KB) (STAT_HEAP_SZ) = 10000
 Interval for checking deadlock (ms) (DLCHKTIME) = 5000
 Percent. of lock lists per application (MAXLOCKS) = 5
 Lock timeout (sec) (LOCKTIMEOUT) = -1
 Changed pages threshold (CHNGPGS_THRESH) = 15
 Number of asynchronous page cleaners (NUM_IOCLEANERS) = 4
 Number of I/O servers (NUM_IOSERVERS) = 4
 Index sort flag (INDEXSORT) = YES
 Sequential detect flag (SEQDETECT) = YES
 Default prefetch size (pages) (DFT_PREFETCH_SZ) = 16
 Track modified pages (TRACKMOD) = OFF
 Default number of containers = 1
 Default tablespace extentsize (pages) (DFT_EXTENT_SZ) = 32
 Max number of active applications (MAXAPPLS) = 40
 Average number of active applications (AVG_APPLS) = 1
 Max DB files open per application (MAXFILOP) = 1024
 Log file size (4KB) (LOGFILSIZ) = 16384
 Number of primary log files (LOGPRIMARY) = 20
 Number of secondary log files (LOGSECOND) = 2
 Changed path to log files (NEWLOGPATH) =
 Path to log files = h:\logs\NODE0007\
 Overflow log path (OVERFLOWLOGPATH) =
 Mirror log path (MIRRORLOGPATH) =
 First active log file = S0000033.LOG
 Block log on disk full (BLK_LOG_DSK_FUL) = NO
 Percent of max active log space by transaction (MAX_LOG) = 0
 Num. of active log files for 1 active UOW (NUM_LOG_SPAN) = 0
 Group commit count (MINCOMMIT) = 1
 Percent log file reclaimed before soft ckpt (SOFTMAX) = 600
 Log retain for recovery enabled (LOGRETAIN) = RECOVERY
 User exit for logging enabled (USEREXIT) = OFF
 Auto restart enabled (AUTORESTART) = ON
 Index re-creation time (INDEXREC) = SYSTEM (ACCESS)
 Default number of loadrec sessions (DFT_LOADREC_SES) = 1
 Number of database backups to retain (NUM_DB_BACKUPS) = 12
 Recovery history retention (days) (REC_HIS_RETENTN) = 366
 TSM management class (TSM_MGMTCLASS) =
 TSM node name (TSM_NODENAME) =
 TSM owner (TSM_OWNER) =
 TSM password (TSM_PASSWORD) =

"DB2 DBM Configuration for System"
 get dbm cfg

Database Manager Configuration

Node type = Enterprise Server Edition with local and remote clients
 Database manager configuration release level = 0x0a00
 Maximum total of files open (MAXTOTFILOP) = 16000
 CPU speed (millisec/instruction) (CPUSPEED) = 7.951129e-007
 Communications bandwidth (MB/sec) (COMM_BANDWIDTH) = 7.000000e+000

Max number of concurrently active databases (NUMDB) = 1
Data Links support (DATA LINKS) = NO
Federated Database System Support (FEDERATED) = NO
Transaction processor monitor name (TP_MON_NAME) =
Default charge-back account (DFT_ACCOUNT_STR) =
Java Development Kit installation path (JDK_PATH) =
Diagnostic error capture level (DIAGLEVEL) = 0
Notify Level (NOTIFYLEVEL) = 3
Diagnostic data directory path (DIAGPATH) =
Default database monitor switches
Buffer pool (DFT_MON_BUFPOOL) = OFF
Lock (DFT_MON_LOCK) = OFF
Sort (DFT_MON_SORT) = OFF
Statement (DFT_MON_STMT) = OFF
Table (DFT_MON_TABLE) = OFF
Timestamp (DFT_MON_TIMESTAMP) = ON
Unit of work (DFT_MON_UOW) = OFF
Monitor health of instance and databases (HEALTH_MON) = OFF
SYSADM group name (SYSADM_GROUP) =
SYSCTRL group name (SYSCTRL_GROUP) =
SYSMAINT group name (SYSMAINT_GROUP) =
SYSMON group name (SYSMON_GROUP) =
Database manager authentication (AUTHENTICATION) = SERVER
Cataloging allowed without authority (CATALOG_NOAUTH) = NO
Trust all clients (TRUST_ALLCLNTS) = YES
Trusted client authentication (TRUST_CLNTAUTH) = CLIENT
Bypass federated authentication (FED_NOAUTH) = NO
Default database path (DFTDBPATH) = D:
Database monitor heap size (4KB) (MON_HEAP_SZ) = 46
Java Virtual Machine heap size (4KB) (JAVA_HEAP_SZ) = 512
Audit buffer size (4KB) (AUDIT_BUF_SZ) = 0
Size of instance shared memory (4KB) (INSTANCE_MEMORY) = AUTOMATIC
Backup buffer default size (4KB) (BACKBUFSZ) = 1024
Restore buffer default size (4KB) (RESTBUFSZ) = 1024
Agent stack size (AGENT_STACK_SZ) = 16
Minimum committed private memory (4KB) (MIN_PRIV_MEM) = 32
Private memory threshold (4KB) (PRIV_MEM_THRESH) = 20000
Sort heap threshold (4KB) (SHEAPTHRES) = 250000
Directory cache support (DIR_CACHE) = YES
Application support layer heap size (4KB) (ASLHEAPSZ) = 15
Max requester I/O block size (bytes) (RQRIOLBK) = 32767
DOS requester I/O block size (bytes) (DOS_RQRIOLBK) = 4096
Query heap size (4KB) (QUERY_HEAP_SZ) = 1000
DRDA services heap size (4KB) (DRDA_HEAP_SZ) = 128
Workload impact by throttled utilities (UTIL_IMPACT_LIM) = 100
Priority of agents (AGENTPRI) = SYSTEM
Max number of existing agents (MAXAGENTS) = 3600
Agent pool size (NUM_POOLAGENTS) = 64
Initial number of agents in pool (NUM_INITAGENTS) = 4
Max number of coordinating agents (MAX_COORDAGENTS) = (MAXAGENTS - NUM_INITAGENTS)
Max no. of concurrent coordinating agents (MAXCAGENTS) = MAX_COORDAGENTS
Max number of client connections (MAX_CONNECTIONS) = MAX_COORDAGENTS
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 (INDEXREC) = ACCESS
Transaction manager database name (TM_DATABASE) = 1ST_CONN
Transaction resync interval (sec) (RESYNC_INTERVAL) = 180
SPM name (SPM_NAME) =
SPM log size (SPM_LOG_FILE_SZ) = 256
SPM resync agent limit (SPM_MAX_RESYNC) = 20
SPM log path (SPM_LOG_PATH) =
NetBIOS Workstation name (NNAME) =
TCP/IP Service name (SVCENAME) = DB2_TPCH_END
Discovery mode (DISCOVER) = SEARCH
Discover server instance (DISCOVER_INST) = ENABLE
Maximum query degree of parallelism (MAX_QUERYDEGREE) = ANY
Enable intra-partition parallelism (INTRA_PARALLEL) = NO
No. of int. communication buffers (4KB) (FCM_NUM_BUFFERS) = 30000
Number of FCM request blocks (FCM_NUM_RQB) = AUTOMATIC

Number of FCM connection entries (FCM_NUM_CONNECT) = AUTOMATIC
Number of FCM message anchors (FCM_NUM_ANCHORS) = AUTOMATIC
Node connection elapse time (sec) (CONN_ELAPSE) = 10
Max number of node connection retries (MAX_CONNRETRIES) = 5
Max time difference between nodes (min) (MAX_TIME_DIFF) = 60
db2start/db2stop timeout (min) (START_STOP_TIME) = 10

DB2 Version

"DB2 Version Information"
DB21085I Instance "TPCH" uses "32" bits and DB2 code release "SQL08015" with level identifier "02060106".

Informational tokens are "DB2 v8.1.5.449", "s040212", "WR21334", and FixPak "5".

Product is installed at "D:\SQLLIB".

DB2 Registry Variables

DB2_EXTENDED_OPTIMIZATION=YES
DB2_ANTIJOIN=Y
DB2BPVARS=d:\tpch\ddl\scattered_read
DB2ACCOUNTNAME=RELENTLESS1\tpch
DB2INSTOWNER=RELENTLESS1
DB2PORTRANGE=80000:80008
DB2MEMMAXFREE=1000000000
DB2OPTIONS=-t -v +c
DB2NTNOCACHE=ON
DB2INSTPROF=D:\INST
DB2_PARALLEL_IO=*

Microsoft Windows Server 2003 Enterprise Edition

Configuration Parameters

The following is the contents of boot.ini
The first line was executed on reboot and in effect for the audit.

[boot loader]
timeout=10
default=multi(0)disk(0)rdisk(0)partition(1)\WINDOWS
[operating systems]
multi(0)disk(0)rdisk(0)partition(1)\WINDOWS="Windows Server 2003, Enterprise /PAE" /fastdetect /PAE
multi(0)disk(0)rdisk(0)partition(1)\WINDOWS="Windows Server 2003, Enterprise" /fastdetect

SUT Hardware Information Report

System Information report written at: 05/12/04 11:36:02
System Name: RELENTLESS1
[System Summary]
Item Value
OS Name Microsoft(R) Windows(R) Server 2003, Enterprise Edition
Version 5.2.3790 Build 3790
OS Manufacturer Microsoft Corporation
System Name RELENTLESS1
System Manufacturer IBM
System Model eServer xSeries 365 -[88612RZ]-
System Type X86-based PC
Processor x86 Family 15 Model 2 Stepping 6 GenuineIntel ~3000 Mhz
Processor x86 Family 15 Model 2 Stepping 6 GenuineIntel ~3000 Mhz
Processor x86 Family 15 Model 2 Stepping 6 GenuineIntel ~3000 Mhz
Processor x86 Family 15 Model 2 Stepping 6 GenuineIntel ~3000 Mhz
Processor x86 Family 15 Model 2 Stepping 6 GenuineIntel ~3000 Mhz
Processor x86 Family 15 Model 2 Stepping 6 GenuineIntel ~3000 Mhz

Processor x86 Family 15 Model 2 Stepping 6 GenuineIntel ~3000 Mhz
 Processor x86 Family 15 Model 2 Stepping 6 GenuineIntel ~3000 Mhz
 BIOS Version/Date IBM-[RDJT24AUS-1.04]-, 2/10/2004
 SMBIOS Version 2.3
 Windows Directory C:\WINDOWS
 System Directory C:\WINDOWS\system32
 Boot Device \Device\HarddiskVolume1
 Locale United States
 Hardware Abstraction Layer Version = "5.2.3790.0 (srv03_rtm.030324-2048)"
 User NameRELENTLESS1\tpch
 Time ZoneEastern Daylight Time
 Total Physical Memory 16,384.00 MB
 Available Physical Memory 15.59 GB
 Total Virtual Memory35.71 GB
 Available Virtual Memory 35.13 GB
 Page File Space 19.71 GB
 Page File C:\pagefile.sys

[Hardware Resources]

[Conflicts/Sharing]

Resource Device
 Memory Address 0xF0000000-0xF7FFFFFFPCI bus
 Memory Address 0xF0000000-0xF7FFFFFFRADEON 7000 RADEON VE Family (Microsoft Corporation)
 I/O Port 0x00000000-0x00001FFF PCI bus
 I/O Port 0x00000000-0x00001FFF Direct memory access controller
 I/O Port 0x00002000-0x00003FFF PCI bus
 I/O Port 0x00002000-0x00003FFF LSI Logic 1020/1030 Ultra320 SCSI Adapter
 [DMA]

Resource Device Status
 Channel 2 Standard floppy disk controller OK
 Channel 4 Direct memory access controller OK
 [Forced Hardware]
 Device PNP Device ID

[I/O]

Resource	Device	Status
0x00000000-0x00001FFF	PCI bus	OK
0x00000000-0x00001FFF	Direct memory access controller	OK
0x00001800-0x000018FF	RADEON 7000 RADEON VE Family (Microsoft Corporation)	OK
0x000003B0-0x000003BB	RADEON 7000 RADEON VE Family (Microsoft Corporation)	OK
0x000003C0-0x000003DF	RADEON 7000 RADEON VE Family (Microsoft Corporation)	OK
0x00000060-0x00000060	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard	OK
0x00000064-0x00000064	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard	OK
0x000003F0-0x000003F5	Standard floppy disk controller	OK
0x000003F7-0x000003F7	Standard floppy disk controller	OK
0x000003F8-0x000003FF	Communications Port (COM1)	OK
0x00000020-0x00000021	Advanced programmable interrupt controller	OK
0x000000A0-0x000000A1	Advanced programmable interrupt controller	OK
0x00000080-0x0000008F	Direct memory access controller	OK
0x000000C0-0x000000DF	Direct memory access controller	OK
0x00000040-0x00000043	System timer	OK
0x00000070-0x00000073	System CMOS/real time clock	OK
0x00000061-0x00000061	System speaker	OK
0x000000F0-0x000000FF	Numeric data processor	OK
0x0000002E-0x0000002F	Motherboard resources	OK
0x00000092-0x00000092	Motherboard resources	OK
0x00000093-0x0000009F	Motherboard resources	OK
0x000000A8-0x000000A9	Motherboard resources	OK
0x00000400-0x000004FE	Motherboard resources	OK
0x00000500-0x0000053F	Motherboard resources	OK
0x00000600-0x00000600	Motherboard resources	OK
0x00000800-0x00000803	Motherboard resources	OK
0x00000C00-0x00000CFE	Motherboard resources	OK
0x00000F50-0x00000F5F	Motherboard resources	OK

Resource	Device	Status
0x00000700-0x0000070F	Standard Dual Channel PCI IDE Controller	OK
0x000001F0-0x000001F7	Primary IDE Channel	OK
0x000003F6-0x000003F6	Primary IDE Channel	OK
0x00000170-0x00000177	Secondary IDE Channel	OK
0x00000376-0x00000376	Secondary IDE Channel	OK
0x00000A79-0x00000A79	ISAPNP Read Data Port	OK
0x00000279-0x00000279	ISAPNP Read Data Port	OK
0x00000274-0x00000277	ISAPNP Read Data Port	OK
0x00002000-0x00003FFF	PCI bus	OK
0x00002000-0x00003FFF	LSI Logic 1020/1030 Ultra320 SCSI Adapter	OK
0x00002100-0x000021FF	LSI Logic 1020/1030 Ultra320 SCSI Adapter	OK
0x00004000-0x00004FFF	PCI bus	OK
0x00007000-0x00007FFF	PCI bus	OK
0x00006000-0x00006FFF	PCI bus	OK
0x00005000-0x00005FFF	PCI bus	OK

[IRQs]

Resource	Device	Status
IRQ 9	Microsoft ACPI-Compliant System	OK
IRQ 22	RADEON 7000 RADEON VE Family (Microsoft Corporation)	OK
IRQ 1	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard	OK
IRQ 12	PS/2 Compatible Mouse	OK
IRQ 6	Standard floppy disk controller	OK
IRQ 4	Communications Port (COM1)	OK
IRQ 0	System timer	OK
IRQ 8	System CMOS/real time clock	OK
IRQ 13	Numeric data processor	OK
IRQ 14	Primary IDE Channel	OK
IRQ 16	Standard OpenHCD USB Host Controller	OK
IRQ 20	Broadcom NetXtreme Gigabit Ethernet	OK
IRQ 21	Broadcom NetXtreme Gigabit Ethernet #2	OK
IRQ 18	LSI Logic 1020/1030 Ultra320 SCSI Adapter	OK
IRQ 19	LSI Logic 1020/1030 Ultra320 SCSI Adapter	OK
IRQ 26	IBM ServeRAID 6M Controller	OK
IRQ 32	IBM ServeRAID 6M Controller	OK
IRQ 68	IBM ServeRAID 6M Controller	OK
IRQ 53	IBM ServeRAID 6M Controller	OK
IRQ 36	IBM ServeRAID 6M Controller	OK

[Memory]

Resource	Device	Status
0xF0000000-0xF7FFFFFF	PCI bus	OK
0xF0000000-0xF7FFFFFF	RADEON 7000 RADEON VE Family (Microsoft Corporation)	OK
0xF8000000-0xFBFFFFFF	PCI bus	OK
0xFBFF0000-0xFBFFFFFF	RADEON 7000 RADEON VE Family (Microsoft Corporation)	OK
0xA0000-0xBFFFF	RADEON 7000 RADEON VE Family (Microsoft Corporation)	OK
0xFBFEF000-0xFBFEFFFF	Standard OpenHCD USB Host Controller	OK
0xEE000000-0xEEFFFFFF	PCI bus	OK
0xEFFF0000-0xEFFFFFFF	Broadcom NetXtreme Gigabit Ethernet	OK
0xEFFE0000-0xEFFEFFFF	Broadcom NetXtreme Gigabit Ethernet	OK
0xEFFD0000-0xEFFDFFFF	Broadcom NetXtreme Gigabit Ethernet #2	OK
0xEFFC0000-0xEFFCFFFF	Broadcom NetXtreme Gigabit Ethernet #2	OK
0xEFFB0000-0xEFFBFFFF	LSI Logic 1020/1030 Ultra320 SCSI Adapter	OK
0xEFFA0000-0xEFFAFFFF	LSI Logic 1020/1030 Ultra320 SCSI Adapter	OK
0xEFF90000-0xEFF9FFFF	LSI Logic 1020/1030 Ultra320 SCSI Adapter	OK
0xEFF80000-0xEFF8FFFF	LSI Logic 1020/1030 Ultra320 SCSI Adapter	OK
0xEF800000-0xEFCFFFFF	PCI standard PCI-to-PCI bridge	OK
0xEFC00000-0xEFC0FFFF	IBM ServeRAID 6M Controller	OK
0xEC000000-0xEDFFFFFF	PCI bus	OK
0xED800000-0xEDCFFFFF	PCI standard PCI-to-PCI bridge	OK
0xEDC00000-0xEDC0FFFF	IBM ServeRAID 6M Controller	OK
0xE6000000-0xE7FFFFFF	PCI bus	OK
0xE7800000-0xE7CFFFFF	PCI standard PCI-to-PCI bridge	OK

0xE7C00000-0xE7C00FFF IBM ServeRAID 6M Controller OK
0xE8000000-0xE9FFFFFF PCI bus OK
0xE9800000-0xE9CFFFFF PCI standard PCI-to-PCI bridge OK
0xE9C00000-0xE9C00FFF IBM ServeRAID 6M Controller OK
0xEA000000-0xEBFFFFFF PCI bus OK
0xEB800000-0xEBCFFFFF PCI standard PCI-to-PCI bridge OK
0xEBC00000-0xEBC00FFF IBM ServeRAID 6M Controller OK
0x0400-0x04FF System board OK
0x9B400-0x9FFFF System board OK
0xC0000-0xDFFFF Motherboard resources OK
0xE0000-0xFFFF Motherboard resources OK
0xFE700000-0xFFFFFFFF Motherboard resources OK
0x1000-0x9B3FF Memory Module OK
0x100000-0xDFFBF7FF Memory Module OK

[Components]

[Multimedia]

[Audio Codecs]

CODEC	Manufacturer	Description	Status	File	Version
Size	Creation Date				
c:\windows\system32\msg723.acm	Microsoft Corporation				
OK	C:\WINDOWS\system32\MSG723.ACM	4.4.4000	116.00 KB		
(118,784 bytes)	4/1/2004 2:34 PM				
c:\windows\system32\imaadp32.acm	Microsoft Corporation				
OK	C:\WINDOWS\system32\IMAADP32.ACM	5.2.3790.0			
(srv03_rtm.030324-2048)	15.50 KB (15,872 bytes)		3/25/2003 7:00 AM		
c:\windows\system32\msadp32.acm	Microsoft Corporation				
OK	C:\WINDOWS\system32\MSADP32.ACM	5.2.3790.0			
(srv03_rtm.030324-2048)	14.50 KB (14,848 bytes)		3/25/2003 7:00 AM		
c:\windows\system32\msgsm32.acm	Microsoft Corporation				
OK	C:\WINDOWS\system32\MSGSM32.ACM	5.2.3790.0			
(srv03_rtm.030324-2048)	20.50 KB (20,992 bytes)		3/25/2003 7:00 AM		
c:\windows\system32\msaud32.acm	Microsoft Corporation				
Windows Media Audio Codec	OK				
C:\WINDOWS\system32\MSAUD32.ACM	8.00.00.4487		288.00 KB		
(294,912 bytes)	3/25/2003 7:00 AM				
c:\windows\system32\msg711.acm	Microsoft Corporation				
OK	C:\WINDOWS\system32\MSG711.ACM	5.2.3790.0			
(srv03_rtm.030324-2048)	10.00 KB (10,240 bytes)		3/25/2003 7:00 AM		
c:\windows\system32\tssoft32.acm	DSP GROUP, INC.		OK		
C:\WINDOWS\system32\TSSOFT32.ACM	1.01	9.50 KB (9,728 bytes)			
3/25/2003 7:00 AM					
c:\windows\system32\sl_anet.acm	Sipro Lab Telecom Inc.		Sipro		
Lab Telecom Audio Codec	OK				
C:\WINDOWS\system32\SL_ANET.ACM	3.02	84.00 KB (86,016 bytes)			
3/25/2003 7:00 AM					
c:\windows\system32\l3codeca.acm	Fraunhofer Institut Integrierte				
Schaltungen IIS	Fraunhofer IIS MPEG Layer-3 Codec	OK			
C:\WINDOWS\system32\L3CODECA.ACM	1, 9, 0, 0305		284.00		
KB (290,816 bytes)	3/25/2003 7:00 AM				

[Video Codecs]

CODEC	Manufacturer	Description	Status	File	Version
Size	Creation Date				
c:\windows\system32\tscopyuv.dll	Microsoft Corporation		OK		
C:\WINDOWS\system32\TSBYUV.DLL	5.2.3790.0 (srv03_rtm.030324-2048)				
8.00 KB (8,192 bytes)	3/24/2003 8:50 PM				
c:\windows\system32\msyuv.dll	Microsoft Corporation		OK		
C:\WINDOWS\system32\MSYUV.DLL	5.2.3790.0 (srv03_rtm.030324-2048)				
16.50 KB (16,896 bytes)	3/24/2003 8:49 PM				
c:\windows\system32\msrle32.dll	Microsoft Corporation		OK		
C:\WINDOWS\system32\MSRLE32.DLL	5.2.3790.0 (srv03_rtm.030324-2048)				
10.50 KB (10,752 bytes)	3/25/2003 7:00 AM				
c:\windows\system32\iyuv_32.dll	Microsoft Corporation				
OK	C:\WINDOWS\system32\IYUV_32.DLL	5.2.3790.0			
(srv03_rtm.030324-2048)	45.00 KB (46,080 bytes)		3/24/2003 8:49 PM		

c:\windows\system32\msvidc32.dll Microsoft Corporation
OK C:\WINDOWS\system32\MSVIDC32.DLL 5.2.3790.0
(srv03_rtm.030324-2048) 26.50 KB (27,136 bytes) 3/25/2003 7:00 AM
c:\windows\system32\msh261.drv Microsoft Corporation
OK C:\WINDOWS\system32\MSH261.DRV 4.4.4000 180.00 KB
(184,320 bytes) 4/1/2004 2:34 PM
c:\windows\system32\msh263.drv Microsoft Corporation
OK C:\WINDOWS\system32\MSH263.DRV 4.4.4000 284.00 KB
(290,816 bytes) 3/24/2003 8:46 PM

[CD-ROM]

Item	Value
Drive	Z:
Description	CD-ROM Drive
Media Loaded	No
Media Type	CD-ROM
Name	TEAC CD-224E
Manufacturer	(Standard CD-ROM drives)
Status	OK
Transfer Rate	Not Available
SCSI Target ID	0
PNP Device ID	
IDE\CDROMTEAC_CD-224E	2.9B \5&B130548&0&0.0.0
Driver	c:\windows\system32\drivers\cdrom.sys (5.2.3790.0 (srv03_rtm.030324-2048), 49.50 KB (50,688 bytes), 3/25/2003 7:00 AM)

[Sound Device]

Item	Value
------	-------

[Display]

Item	Value
Name	RADEON 7000 RADEON VE Family (Microsoft Corporation)
PNP Device ID	
PCI\VEN_1002&DEV_5159&SUBSYS_029A1014&REV_00\3&267A616A&0&40	
Adapter Type	Radeon 7000 (QY), ATI Technologies Inc. compatible
Adapter Description	RADEON 7000 RADEON VE Family (Microsoft Corporation)
Adapter RAM	8.00 MB (8,388,608 bytes)
Installed Drivers	ati2dvag.dll
Driver Version	5.1.2600.0
INF File	atiixpag.inf (ati2mtag_RV100 section)
Color Planes	1
Color Table Entries	4294967296
Resolution	1024 x 768 x 75 hertz
Bits/Pixel	32
Memory Address	0xF0000000-0xF7FFFFFF
I/O Port	0x00001800-0x000018FF
Memory Address	0xFBFF0000-0xFBFFFFFF
IRQ Channel	IRQ 22
I/O Port	0x000003B0-0x000003BB
I/O Port	0x000003C0-0x000003DF
Memory Address	0xA0000-0xBFFFF
Driver	c:\windows\system32\drivers\ati2mtag.sys (6.13.10.6153, 439.63 KB (450,176 bytes), 4/1/2004 9:07 AM)

[Infrared]

Item	Value
------	-------

[Input]

[Keyboard]

Item	Value
Description	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard
Name	Enhanced (101- or 102-key)
Layout	00000409
PNP Device ID	ACPI\PNP0303\4&39D9D85A&0
Number of Function Keys	12

I/O Port 0x00000060-0x00000060
 I/O Port 0x00000064-0x00000064
 IRQ Channel IRQ 1
 Driver c:\windows\system32\drivers\i8042prt.sys (5.2.3790.0
 (srv03_rtm.030324-2048), 68.50 KB (70,144 bytes), 3/25/2003 7:00 AM)
 Description USB Human Interface Device
 Name Enhanced (101- or 102-key)
 Layout 00000409
 PNP Device ID USB\VID_04B3&PID_4001&MI_00\6&2061A23&0&0000
 Number of Function Keys 12
 Driver c:\windows\system32\drivers\hidusb.sys (5.2.3790.0
 (srv03_rtm.030324-2048), 11.50 KB (11,776 bytes), 3/25/2003 7:00 AM)

[Pointing Device]

Item	Value
Hardware Type	PS/2 Compatible Mouse
Number of Buttons	5
Status	OK
PNP Device ID	ACPI\PNP0F13\4&39D9D85A&0
Power Management Supported	No
Double Click Threshold	6
Handedness	Right Handed Operation
IRQ Channel	IRQ 12
Driver	c:\windows\system32\drivers\i8042prt.sys (5.2.3790.0 (srv03_rtm.030324-2048), 68.50 KB (70,144 bytes), 3/25/2003 7:00 AM)
Hardware Type	USB Human Interface Device
Number of Buttons	5
Status	OK
PNP Device ID	USB\VID_04B3&PID_4001&MI_01\6&2061A23&0&0001
Power Management Supported	No
Double Click Threshold	6
Handedness	Right Handed Operation
Driver	c:\windows\system32\drivers\hidusb.sys (5.2.3790.0 (srv03_rtm.030324-2048), 11.50 KB (11,776 bytes), 3/25/2003 7:00 AM)

[Modem]

Item	Value
------	-------

[Network]

[Adapter]

Item	Value
Name	[00000001] RAS Async Adapter
Adapter Type	Not Available
Product Type	RAS Async Adapter
Installed	Yes
PNP Device ID	Not Available
Last Reset	5/12/2004 7:23 AM
Index	1
Service Name	AsyncMac
IP Address	Not Available
IP Subnet	Not Available
Default IP Gateway	Not Available
DHCP Enabled	No
DHCP Server	Not Available
DHCP Lease Expires	Not Available
DHCP Lease Obtained	Not Available
MAC Address	Not Available
Name	[00000002] WAN Miniport (L2TP)
Adapter Type	Not Available
Product Type	WAN Miniport (L2TP)
Installed	Yes
PNP Device ID	ROOT\MS_L2TPMINIPORT\0000
Last Reset	5/12/2004 7:23 AM
Index	2
Service Name	Rasl2tp
IP Address	Not Available
IP Subnet	Not Available
Default IP Gateway	Not Available
DHCP Enabled	No
DHCP Server	Not Available
DHCP Lease Expires	Not Available

DHCP Lease Obtained	Not Available
MAC Address	Not Available
Driver	c:\windows\system32\drivers\rasl2tp.sys (5.2.3790.0 (srv03_rtm.030324-2048), 77.00 KB (78,848 bytes), 3/25/2003 7:00 AM)
Name	[00000003] WAN Miniport (PPTP)
Adapter Type	Wide Area Network (WAN)
Product Type	WAN Miniport (PPTP)
Installed	Yes
PNP Device ID	ROOT\MS_PPTPMINIPORT\0000
Last Reset	5/12/2004 7:23 AM
Index	3
Service Name	PptpMiniport
IP Address	Not Available
IP Subnet	Not Available
Default IP Gateway	Not Available
DHCP Enabled	No
DHCP Server	Not Available
DHCP Lease Expires	Not Available
DHCP Lease Obtained	Not Available
MAC Address	50:50:54:50:30:30
Driver	c:\windows\system32\drivers\raspppt.sys (5.2.3790.0 (srv03_rtm.030324-2048), 70.50 KB (72,192 bytes), 3/25/2003 7:00 AM)
Name	[00000004] WAN Miniport (PPPOE)
Adapter Type	Wide Area Network (WAN)
Product Type	WAN Miniport (PPPOE)
Installed	Yes
PNP Device ID	ROOT\MS_PPPOEMINIPORT\0000
Last Reset	5/12/2004 7:23 AM
Index	4
Service Name	RasPppoe
IP Address	Not Available
IP Subnet	Not Available
Default IP Gateway	Not Available
DHCP Enabled	No
DHCP Server	Not Available
DHCP Lease Expires	Not Available
DHCP Lease Obtained	Not Available
MAC Address	33:50:6F:45:30:30
Driver	c:\windows\system32\drivers\raspppoe.sys (5.2.3790.0 (srv03_rtm.030324-2048), 38.00 KB (38,912 bytes), 3/25/2003 7:00 AM)
Name	[00000005] Direct Parallel
Adapter Type	Not Available
Product Type	Direct Parallel
Installed	Yes
PNP Device ID	ROOT\MS_PTMINIPORT\0000
Last Reset	5/12/2004 7:23 AM
Index	5
Service Name	Raspti
IP Address	Not Available
IP Subnet	Not Available
Default IP Gateway	Not Available
DHCP Enabled	No
DHCP Server	Not Available
DHCP Lease Expires	Not Available
DHCP Lease Obtained	Not Available
MAC Address	Not Available
Driver	c:\windows\system32\drivers\raspti.sys (5.2.3790.0 (srv03_rtm.030324-2048), 18.50 KB (18,944 bytes), 3/25/2003 7:00 AM)
Name	[00000006] WAN Miniport (IP)
Adapter Type	Not Available
Product Type	WAN Miniport (IP)
Installed	Yes
PNP Device ID	ROOT\MS_NDISWANIP\0000
Last Reset	5/12/2004 7:23 AM
Index	6
Service Name	NdisWan
IP Address	Not Available
IP Subnet	Not Available
Default IP Gateway	Not Available
DHCP Enabled	No
DHCP Server	Not Available
DHCP Lease Expires	Not Available
DHCP Lease Obtained	Not Available
MAC Address	Not Available

Driver c:\windows\system32\drivers\ndiswan.sys (5.2.3790.0
(srv03_rtm.030324-2048), 96.50 KB (98,816 bytes), 3/25/2003 7:00 AM)
Name [00000007] Broadcom NetXtreme Gigabit Ethernet
Adapter Type Ethernet 802.3
Product Type Broadcom NetXtreme Gigabit Ethernet
Installed Yes
PNP Device ID
PCI\VEN_14E4&DEV_1648&SUBSYS_02B41014&REV_02\3&13C0B0C5&0&08
Last Reset 5/12/2004 7:23 AM
Index 7
Service Name b57w2k
IP Address 0.0.0.0
IP Subnet 0.0.0.0
Default IP Gateway Not Available
DHCP Enabled No
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address 00:09:6B:F5:01:94
Memory Address 0xEFFF0000-0xEFFFFFFF
Memory Address 0xEFFE0000-0xEFFFFFFF
IRQ Channel IRQ 20

Driver c:\windows\system32\drivers\b57xp32.sys (7.33.0.0 built by: WinDDK,
110.63 KB (113,280 bytes), 5/5/2004 11:26 PM)
Name [00000008] Broadcom NetXtreme Gigabit Ethernet
Adapter Type Ethernet 802.3
Product Type Broadcom NetXtreme Gigabit Ethernet
Installed Yes
PNP Device ID
PCI\VEN_14E4&DEV_1648&SUBSYS_02B41014&REV_02\3&13C0B0C5&0&09
Last Reset 5/12/2004 7:23 AM
Index 8
Service Name b57w2k
IP Address 192.168.30.200
IP Subnet 255.255.255.0
Default IP Gateway 192.168.30.100
DHCP Enabled No
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address 00:09:6B:F5:01:95
Memory Address 0xEFFD0000-0xEFFDFFFF
Memory Address 0xEFFC0000-0xEFFCFFFF
IRQ Channel IRQ 21
Driver c:\windows\system32\drivers\b57xp32.sys (7.33.0.0 built by: WinDDK,
110.63 KB (113,280 bytes), 5/5/2004 11:26 PM)

[Protocol]

Item	Value
Name	MSAFD Tcpip [TCP/IP]
Connectionless Service	No
Guarantees Delivery	Yes
Guarantees Sequencing	Yes
Maximum Address Size	16 bytes
Maximum Message Size	0 bytes
Message Oriented	No
Minimum Address Size	16 bytes
Pseudo Stream Oriented	No
Supports Broadcasting	No
Supports Connect Data	No
Supports Disconnect Data	No
Supports Encryption	No
Supports Expedited Data	Yes
Supports Graceful Closing	Yes
Supports Guaranteed Bandwidth	No
Supports Multicasting	No
Name	MSAFD Tcpip [UDP/IP]
Connectionless Service	Yes
Guarantees Delivery	No
Guarantees Sequencing	No
Maximum Address Size	16 bytes
Maximum Message Size	63.93 KB (65,467 bytes)
Message Oriented	Yes

Minimum Address Size	16 bytes
Pseudo Stream Oriented	No
Supports Broadcasting	Yes
Supports Connect Data	No
Supports Disconnect Data	No
Supports Encryption	No
Supports Expedited Data	No
Supports Graceful Closing	No
Supports Guaranteed Bandwidth	No
Supports Multicasting	Yes
Name	RSVP UDP Service Provider
Connectionless Service	Yes
Guarantees Delivery	No
Guarantees Sequencing	No
Maximum Address Size	16 bytes
Maximum Message Size	63.93 KB (65,467 bytes)
Message Oriented	Yes
Minimum Address Size	16 bytes
Pseudo Stream Oriented	No
Supports Broadcasting	Yes
Supports Connect Data	No
Supports Disconnect Data	No
Supports Encryption	Yes
Supports Expedited Data	No
Supports Graceful Closing	No
Supports Guaranteed Bandwidth	No
Supports Multicasting	Yes
Name	RSVP TCP Service Provider
Connectionless Service	No
Guarantees Delivery	Yes
Guarantees Sequencing	Yes
Maximum Address Size	16 bytes
Maximum Message Size	0 bytes
Message Oriented	No
Minimum Address Size	16 bytes
Pseudo Stream Oriented	No
Supports Broadcasting	No
Supports Connect Data	No
Supports Disconnect Data	No
Supports Encryption	Yes
Supports Expedited Data	Yes
Supports Graceful Closing	Yes
Supports Guaranteed Bandwidth	No
Supports Multicasting	No
Name	MSAFD NetBIOS

[\\Device\\NetBT_Tcpip_{0A57A295-E826-4293-9F1A-F6306EA9707D}]
SEQPACKET 3
Connectionless Service No
Guarantees Delivery Yes
Guarantees Sequencing Yes
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting No
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No
Name MSAFD NetBIOS
[\\Device\\NetBT_Tcpip_{0A57A295-E826-4293-9F1A-F6306EA9707D}]
DATAGRAM 3
Connectionless Service Yes
Guarantees Delivery No
Guarantees Sequencing No
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No

Supports Broadcasting	Yes
Supports Connect Data	No
Supports Disconnect Data	No
Supports Encryption	No
Supports Expedited Data	No
Supports Graceful Closing	No
Supports Guaranteed Bandwidth	No
Supports Multicasting	No
Name	MSAFD NetBIOS
[\Device\NetBT_Tcpip_{075708C0-41E4-425C-8F25-6255EB31F2DE}]	
SEQPACKET 0	
Connectionless Service	No
Guarantees Delivery	Yes
Guarantees Sequencing	Yes
Maximum Address Size	20 bytes
Maximum Message Size	62.50 KB (64,000 bytes)
Message Oriented	Yes
Minimum Address Size	20 bytes
Pseudo Stream Oriented	No
Supports Broadcasting	No
Supports Connect Data	No
Supports Disconnect Data	No
Supports Encryption	No
Supports Expedited Data	No
Supports Graceful Closing	No
Supports Guaranteed Bandwidth	No
Supports Multicasting	No
Name	MSAFD NetBIOS
[\Device\NetBT_Tcpip_{075708C0-41E4-425C-8F25-6255EB31F2DE}]	
DATAGRAM 0	
Connectionless Service	Yes
Guarantees Delivery	No
Guarantees Sequencing	No
Maximum Address Size	20 bytes
Maximum Message Size	62.50 KB (64,000 bytes)
Message Oriented	Yes
Minimum Address Size	20 bytes
Pseudo Stream Oriented	No
Supports Broadcasting	Yes
Supports Connect Data	No
Supports Disconnect Data	No
Supports Encryption	No
Supports Expedited Data	No
Supports Graceful Closing	No
Supports Guaranteed Bandwidth	No
Supports Multicasting	No
Name	MSAFD NetBIOS
[\Device\NetBT_Tcpip_{934889F5-5D62-4C21-BF27-3278D09E226F}]	
SEQPACKET 1	
Connectionless Service	No
Guarantees Delivery	Yes
Guarantees Sequencing	Yes
Maximum Address Size	20 bytes
Maximum Message Size	62.50 KB (64,000 bytes)
Message Oriented	Yes
Minimum Address Size	20 bytes
Pseudo Stream Oriented	No
Supports Broadcasting	No
Supports Connect Data	No
Supports Disconnect Data	No
Supports Encryption	No
Supports Expedited Data	No
Supports Graceful Closing	No
Supports Guaranteed Bandwidth	No
Supports Multicasting	No
Name	MSAFD NetBIOS
[\Device\NetBT_Tcpip_{934889F5-5D62-4C21-BF27-3278D09E226F}]	
DATAGRAM 1	
Connectionless Service	Yes
Guarantees Delivery	No
Guarantees Sequencing	No
Maximum Address Size	20 bytes
Maximum Message Size	62.50 KB (64,000 bytes)
Message Oriented	Yes

Minimum Address Size	20 bytes
Pseudo Stream Oriented	No
Supports Broadcasting	Yes
Supports Connect Data	No
Supports Disconnect Data	No
Supports Encryption	No
Supports Expedited Data	No
Supports Graceful Closing	No
Supports Guaranteed Bandwidth	No
Supports Multicasting	No
Name	MSAFD NetBIOS
[\Device\NetBT_Tcpip_{B05A3288-07EF-4526-8919-63F84FE8590C}]	
SEQPACKET 2	
Connectionless Service	No
Guarantees Delivery	Yes
Guarantees Sequencing	Yes
Maximum Address Size	20 bytes
Maximum Message Size	62.50 KB (64,000 bytes)
Message Oriented	Yes
Minimum Address Size	20 bytes
Pseudo Stream Oriented	No
Supports Broadcasting	No
Supports Connect Data	No
Supports Disconnect Data	No
Supports Encryption	No
Supports Expedited Data	No
Supports Graceful Closing	No
Supports Guaranteed Bandwidth	No
Supports Multicasting	No
Name	MSAFD NetBIOS
[\Device\NetBT_Tcpip_{B05A3288-07EF-4526-8919-63F84FE8590C}]	
DATAGRAM 2	
Connectionless Service	Yes
Guarantees Delivery	No
Guarantees Sequencing	No
Maximum Address Size	20 bytes
Maximum Message Size	62.50 KB (64,000 bytes)
Message Oriented	Yes
Minimum Address Size	20 bytes
Pseudo Stream Oriented	No
Supports Broadcasting	Yes
Supports Connect Data	No
Supports Disconnect Data	No
Supports Encryption	No
Supports Expedited Data	No
Supports Graceful Closing	No
Supports Guaranteed Bandwidth	No
Supports Multicasting	No

[WinSock]

Item	Value
File	c:\windows\system32\winsock.dll
Size	2.80 KB (2,864 bytes)
Version	3.10
File	c:\windows\system32\wsock32.dll
Size	22.00 KB (22,528 bytes)
Version	5.2.3790.0 (srv03_rtm.030324-2048)

[Ports]

[Serial]

Item	Value
Name	Communications Port (COM1)
Status	OK
PNP Device ID	ACPI\PNP0501\1
Maximum Input Buffer Size	0
Maximum Output Buffer Size	No
Settable Baud Rate	Yes
Settable Data Bits	Yes
Settable Flow Control	Yes
Settable Parity	Yes
Settable Parity Check	Yes
Settable Stop Bits	Yes

Settable RLSD Yes
 Supports RLSD Yes
 Supports 16 Bit Mode No
 Supports Special Characters No
 Baud Rate 9600
 Bits/Byte 8
 Stop Bits 1
 Parity None
 Busy No
 Abort Read/Write on Error No
 Binary Mode Enabled Yes
 Continue XMit on XOff No
 CTS Outflow Control No
 Discard NULL Bytes No
 DSR Outflow Control 0
 DSR Sensitivity 0
 DTR Flow Control Type Enable
 EOF Character 0
 Error Replace Character 0
 Error Replacement Enabled No
 Event Character 0
 Parity Check Enabled No
 RTS Flow Control Type Enable
 XOff Character 19
 XOffXMit Threshold 512
 XOn Character 17
 XOnXMit Threshold 2048
 XOnXOff InFlow Control 0
 XOnXOff OutFlow Control 0
 I/O Port 0x000003F8-0x000003FF
 IRQ Channel IRQ 4
 Driver c:\windows\system32\drivers\serial.sys (5.2.3790.0
 (srv03_rtm.030324-2048), 76.00 KB (77,824 bytes), 3/25/2003 7:00 AM)

[Parallel]

Item Value
 [Storage]

[Drives]

Item Value
 Drive A:
 Description 3 1/2 Inch Floppy Drive
 Drive C:
 Description Local Fixed Disk
 Compressed No
 File System NTFS
 Size 33.89 GB (36,388,605,952 bytes)
 Free Space 29.82 GB (32,013,955,072 bytes)
 Volume Name
 Volume Serial Number 5CF10E19
 Drive D:
 Description Local Fixed Disk
 Compressed No
 File System NTFS
 Size 33.90 GB (36,396,830,720 bytes)
 Free Space 25.96 GB (27,876,397,056 bytes)
 Volume Name New Volume
 Volume Serial Number 18A14372

Drive E:
 Description Local Fixed Disk
 Compressed No
 File System NTFS
 Size 33.90 GB (36,396,068,864 bytes)
 Free Space 32.58 GB (34,984,943,616 bytes)
 Volume Name Node 4
 Volume Serial Number B0EE0F09

Drive F:
 Description Local Fixed Disk
 Compressed No
 File System NTFS
 Size 33.90 GB (36,396,068,864 bytes)

Free Space 32.58 GB (34,984,939,520 bytes)
 Volume Name Node 5
 Volume Serial Number E8FCEC2A
 Drive G:
 Description Local Fixed Disk
 Compressed No
 File System NTFS
 Size 33.90 GB (36,396,068,864 bytes)
 Free Space 32.58 GB (34,984,947,712 bytes)
 Volume Name Node 6
 Volume Serial Number 1C0E2BA2
 Drive H:
 Description Local Fixed Disk
 Compressed No
 File System NTFS
 Size 33.90 GB (36,396,068,864 bytes)
 Free Space 32.58 GB (34,984,947,712 bytes)
 Volume Name Node 7
 Volume Serial Number 5C1AB97E
 Drive I:
 Description Local Fixed Disk
 Compressed No
 File System NTFS
 Size 33.90 GB (36,396,068,864 bytes)
 Free Space 31.15 GB (33,450,913,792 bytes)
 Volume Name node 0
 Volume Serial Number BC2E9E41
 Drive J:
 Description Local Fixed Disk
 Compressed No
 File System NTFS
 Size 33.90 GB (36,396,068,864 bytes)
 Free Space 31.39 GB (33,709,502,464 bytes)
 Volume Name node 1
 Volume Serial Number 888739AB
 Drive K:
 Description Local Fixed Disk
 Compressed No
 File System NTFS
 Size 33.90 GB (36,396,068,864 bytes)
 Free Space 32.58 GB (34,984,947,712 bytes)
 Volume Name Node 2
 Volume Serial Number 4047E4AC
 Drive L:
 Description Local Fixed Disk
 Compressed No
 File System NTFS
 Size 33.90 GB (36,396,068,864 bytes)
 Free Space 32.58 GB (34,984,939,520 bytes)
 Volume Name Node 3
 Volume Serial Number D44FC109
 Drive O:
 Description Local Fixed Disk
 Compressed No
 File System NTFS
 Size 468.75 GB (503,316,475,904 bytes)
 Free Space 234.52 GB (251,816,132,608 bytes)
 Volume Name Backup 1
 Volume Serial Number 98A19DE9
 Drive P:
 Description Local Fixed Disk
 Compressed No
 File System NTFS
 Size 468.75 GB (503,316,475,904 bytes)
 Free Space 272.14 GB (292,212,678,656 bytes)
 Volume Name Backup 2

Volume Serial Number D0DDAD8

Drive W:
Description Local Fixed Disk
Compressed No
File System NTFS
Size 439.74 GB (472,166,428,672 bytes)
Free Space 61.46 GB (65,987,022,848 bytes)
Volume Name FlatFiles
Volume Serial Number D0F14DA9

Drive Z:
Description CD-ROM Disc
[Disks]

Item	Value
Description	Disk drive
Manufacturer	(Standard disk drives)
Model	IBM ServeRAID SCSI Disk Device
Bytes/Sector	512
Media Loaded	Yes
Media Type	Fixed hard disk
Partitions	1
SCSI Bus	0
SCSI Logical Unit	0
SCSI Port	5
SCSI Target ID	0
Sectors/Track	32
Size	440.72 GB (473,216,057,344 bytes)
Total Cylinders	225,647
Total Sectors	924,250,112
Total Tracks	28,882,816
Tracks/Cylinder	128
Partition	Disk #8, Partition #0
Partition Size	440.72 GB (473,216,776,704 bytes)
Partition Starting Offset	32,256 bytes
Description	Disk drive
Manufacturer	(Standard disk drives)
Model	IBM ServeRAID SCSI Disk Device
Bytes/Sector	512
Media Loaded	Yes
Media Type	Fixed hard disk
Partitions	1
SCSI Bus	0
SCSI Logical Unit	0
SCSI Port	8
SCSI Target ID	0
Sectors/Track	32
Size	203.41 GB (218,405,797,888 bytes)
Total Cylinders	104,144
Total Sectors	426,573,824
Total Tracks	13,330,432
Tracks/Cylinder	128
Partition	Disk #21, Partition #0
Partition Size	203.41 GB (218,407,878,656 bytes)
Partition Starting Offset	16,384 bytes
Description	Disk drive
Manufacturer	(Standard disk drives)
Model	IBM ServeRAID SCSI Disk Device
Bytes/Sector	512
Media Loaded	Yes
Media Type	Fixed hard disk
Partitions	1
SCSI Bus	0
SCSI Logical Unit	0
SCSI Port	8
SCSI Target ID	1
Sectors/Track	32
Size	203.41 GB (218,405,797,888 bytes)
Total Cylinders	104,144
Total Sectors	426,573,824
Total Tracks	13,330,432
Tracks/Cylinder	128
Partition	Disk #22, Partition #0

Partition Size	203.41 GB (218,407,878,656 bytes)
Partition Starting Offset	16,384 bytes
Description	Disk drive
Manufacturer	(Standard disk drives)
Model	IBM ServeRAID SCSI Disk Device
Bytes/Sector	512
Media Loaded	Yes
Media Type	Fixed hard disk
Partitions	1
SCSI Bus	0
SCSI Logical Unit	0
SCSI Port	8
SCSI Target ID	2
Sectors/Track	32
Size	33.90 GB (36,396,072,960 bytes)
Total Cylinders	17,355
Total Sectors	71,086,080
Total Tracks	2,221,440
Tracks/Cylinder	128
Partition	Disk #23, Partition #0
Partition Size	33.90 GB (36,396,831,744 bytes)
Partition Starting Offset	32,256 bytes
Description	Disk drive
Manufacturer	(Standard disk drives)
Model	IBM ServeRAID SCSI Disk Device
Bytes/Sector	512
Media Loaded	Yes
Media Type	Fixed hard disk
Partitions	1
SCSI Bus	0
SCSI Logical Unit	0
SCSI Port	8
SCSI Target ID	3
Sectors/Track	32
Size	203.41 GB (218,405,797,888 bytes)
Total Cylinders	104,144
Total Sectors	426,573,824
Total Tracks	13,330,432
Tracks/Cylinder	128
Partition	Disk #24, Partition #0
Partition Size	203.41 GB (218,407,878,656 bytes)
Partition Starting Offset	16,384 bytes
Description	Disk drive
Manufacturer	(Standard disk drives)
Model	IBM ServeRAID SCSI Disk Device
Bytes/Sector	512
Media Loaded	Yes
Media Type	Fixed hard disk
Partitions	1
SCSI Bus	0
SCSI Logical Unit	0
SCSI Port	8
SCSI Target ID	4
Sectors/Track	32
Size	203.41 GB (218,405,797,888 bytes)
Total Cylinders	104,144
Total Sectors	426,573,824
Total Tracks	13,330,432
Tracks/Cylinder	128
Partition	Disk #25, Partition #0
Partition Size	203.41 GB (218,407,878,656 bytes)
Partition Starting Offset	16,384 bytes
Description	Disk drive
Manufacturer	(Standard disk drives)
Model	IBM ServeRAID SCSI Disk Device
Bytes/Sector	512
Media Loaded	Yes
Media Type	Fixed hard disk
Partitions	1
SCSI Bus	0
SCSI Logical Unit	0
SCSI Port	8
SCSI Target ID	5
Sectors/Track	32

Size 33.90 GB (36,396,072,960 bytes)
 Total Cylinders 17,355
 Total Sectors 71,086,080
 Total Tracks 2,221,440
 Tracks/Cylinder 128
 Partition Disk #26, Partition #0
 Partition Size 33.90 GB (36,400,250,880 bytes)
 Partition Starting Offset 16,384 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM ServeRAID SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 1
 SCSI Bus 0
 SCSI Logical Unit 0
 SCSI Port 6
 SCSI Target ID 0
 Sectors/Track 32
 Size 203.41 GB (218,405,797,888 bytes)
 Total Cylinders 104,144
 Total Sectors 426,573,824
 Total Tracks 13,330,432
 Tracks/Cylinder 128
 Partition Disk #9, Partition #0
 Partition Size 203.41 GB (218,407,878,656 bytes)
 Partition Starting Offset 16,384 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM ServeRAID SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 1
 SCSI Bus 0
 SCSI Logical Unit 0
 SCSI Port 6
 SCSI Target ID 1
 Sectors/Track 32
 Size 203.41 GB (218,405,797,888 bytes)
 Total Cylinders 104,144
 Total Sectors 426,573,824
 Total Tracks 13,330,432
 Tracks/Cylinder 128
 Partition Disk #10, Partition #0
 Partition Size 203.41 GB (218,407,878,656 bytes)
 Partition Starting Offset 16,384 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM ServeRAID SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 1
 SCSI Bus 0
 SCSI Logical Unit 0
 SCSI Port 6
 SCSI Target ID 2
 Sectors/Track 32
 Size 33.90 GB (36,396,072,960 bytes)
 Total Cylinders 17,355
 Total Sectors 71,086,080
 Total Tracks 2,221,440
 Tracks/Cylinder 128
 Partition Disk #11, Partition #0
 Partition Size 33.90 GB (36,400,250,880 bytes)
 Partition Starting Offset 16,384 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM ServeRAID SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk

Partitions 1
 SCSI Bus 0
 SCSI Logical Unit 0
 SCSI Port 6
 SCSI Target ID 3
 Sectors/Track 32
 Size 203.41 GB (218,405,797,888 bytes)
 Total Cylinders 104,144
 Total Sectors 426,573,824
 Total Tracks 13,330,432
 Tracks/Cylinder 128
 Partition Disk #12, Partition #0
 Partition Size 203.41 GB (218,407,878,656 bytes)
 Partition Starting Offset 16,384 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM ServeRAID SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 1
 SCSI Bus 0
 SCSI Logical Unit 0
 SCSI Port 6
 SCSI Target ID 4
 Sectors/Track 32
 Size 203.41 GB (218,405,797,888 bytes)
 Total Cylinders 104,144
 Total Sectors 426,573,824
 Total Tracks 13,330,432
 Tracks/Cylinder 128
 Partition Disk #13, Partition #0
 Partition Size 203.41 GB (218,407,878,656 bytes)
 Partition Starting Offset 16,384 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM ServeRAID SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 1
 SCSI Bus 0
 SCSI Logical Unit 0
 SCSI Port 6
 SCSI Target ID 5
 Sectors/Track 32
 Size 33.90 GB (36,396,072,960 bytes)
 Total Cylinders 17,355
 Total Sectors 71,086,080
 Total Tracks 2,221,440
 Tracks/Cylinder 128
 Partition Disk #14, Partition #0
 Partition Size 33.90 GB (36,400,250,880 bytes)
 Partition Starting Offset 16,384 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM ServeRAID SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 1
 SCSI Bus 0
 SCSI Logical Unit 0
 SCSI Port 4
 SCSI Target ID 0
 Sectors/Track 32
 Size 203.41 GB (218,407,895,040 bytes)
 Total Cylinders 104,145
 Total Sectors 426,577,920
 Total Tracks 13,330,560
 Tracks/Cylinder 128
 Partition Disk #2, Partition #0
 Partition Size 203.41 GB (218,407,878,656 bytes)
 Partition Starting Offset 16,384 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM ServeRAID SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 1
 SCSI Bus 0
 SCSI Logical Unit 0
 SCSI Port 4
 SCSI Target ID 1
 Sectors/Track 32
 Size 203.41 GB (218,407,895,040 bytes)
 Total Cylinders 104,145
 Total Sectors 426,577,920
 Total Tracks 13,330,560
 Tracks/Cylinder 128
 Partition Disk #3, Partition #0
 Partition Size 203.41 GB (218,407,878,656 bytes)
 Partition Starting Offset 16,384 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM ServeRAID SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 1
 SCSI Bus 0
 SCSI Logical Unit 0
 SCSI Port 4
 SCSI Target ID 2
 Sectors/Track 32
 Size 33.90 GB (36,400,267,264 bytes)
 Total Cylinders 17,357
 Total Sectors 71,094,272
 Total Tracks 2,221,696
 Tracks/Cylinder 128
 Partition Disk #4, Partition #0
 Partition Size 33.90 GB (36,400,250,880 bytes)
 Partition Starting Offset 16,384 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM ServeRAID SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 1
 SCSI Bus 0
 SCSI Logical Unit 0
 SCSI Port 4
 SCSI Target ID 3
 Sectors/Track 32
 Size 203.41 GB (218,407,895,040 bytes)
 Total Cylinders 104,145
 Total Sectors 426,577,920
 Total Tracks 13,330,560
 Tracks/Cylinder 128
 Partition Disk #5, Partition #0
 Partition Size 203.41 GB (218,407,878,656 bytes)
 Partition Starting Offset 16,384 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM ServeRAID SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 1
 SCSI Bus 0
 SCSI Logical Unit 0
 SCSI Port 4
 SCSI Target ID 4

Sectors/Track 32
 Size 203.41 GB (218,407,895,040 bytes)
 Total Cylinders 104,145
 Total Sectors 426,577,920
 Total Tracks 13,330,560
 Tracks/Cylinder 128
 Partition Disk #6, Partition #0
 Partition Size 203.41 GB (218,407,878,656 bytes)
 Partition Starting Offset 16,384 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM ServeRAID SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 1
 SCSI Bus 0
 SCSI Logical Unit 0
 SCSI Port 4
 SCSI Target ID 5
 Sectors/Track 32
 Size 33.90 GB (36,400,267,264 bytes)
 Total Cylinders 17,357
 Total Sectors 71,094,272
 Total Tracks 2,221,696
 Tracks/Cylinder 128
 Partition Disk #7, Partition #0
 Partition Size 33.90 GB (36,400,250,880 bytes)
 Partition Starting Offset 16,384 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM ServeRAID SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 1
 SCSI Bus 0
 SCSI Logical Unit 0
 SCSI Port 7
 SCSI Target ID 0
 Sectors/Track 32
 Size 203.41 GB (218,405,797,888 bytes)
 Total Cylinders 104,144
 Total Sectors 426,573,824
 Total Tracks 13,330,432
 Tracks/Cylinder 128
 Partition Disk #15, Partition #0
 Partition Size 203.41 GB (218,407,878,656 bytes)
 Partition Starting Offset 16,384 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM ServeRAID SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 1
 SCSI Bus 0
 SCSI Logical Unit 0
 SCSI Port 7
 SCSI Target ID 1
 Sectors/Track 32
 Size 203.41 GB (218,405,797,888 bytes)
 Total Cylinders 104,144
 Total Sectors 426,573,824
 Total Tracks 13,330,432
 Tracks/Cylinder 128
 Partition Disk #16, Partition #0
 Partition Size 203.41 GB (218,407,878,656 bytes)
 Partition Starting Offset 16,384 bytes

Description Disk drive

Manufacturer (Standard disk drives)
 Model IBM ServeRAID SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 1
 SCSI Bus 0
 SCSI Logical Unit 0
 SCSI Port 7
 SCSI Target ID 2
 Sectors/Track 32
 Size 33.90 GB (36,396,072,960 bytes)
 Total Cylinders 17,355
 Total Sectors 71,086,080
 Total Tracks 2,221,440
 Tracks/Cylinder 128
 Partition Disk #17, Partition #0
 Partition Size 33.90 GB (36,400,250,880 bytes)
 Partition Starting Offset 16,384 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM ServeRAID SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 1
 SCSI Bus 0
 SCSI Logical Unit 0
 SCSI Port 7
 SCSI Target ID 3
 Sectors/Track 32
 Size 203.41 GB (218,405,797,888 bytes)
 Total Cylinders 104,144
 Total Sectors 426,573,824
 Total Tracks 13,330,432
 Tracks/Cylinder 128
 Partition Disk #18, Partition #0
 Partition Size 203.41 GB (218,407,878,656 bytes)
 Partition Starting Offset 16,384 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM ServeRAID SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 1
 SCSI Bus 0
 SCSI Logical Unit 0
 SCSI Port 7
 SCSI Target ID 4
 Sectors/Track 32
 Size 203.41 GB (218,405,797,888 bytes)
 Total Cylinders 104,144
 Total Sectors 426,573,824
 Total Tracks 13,330,432
 Tracks/Cylinder 128
 Partition Disk #19, Partition #0
 Partition Size 203.41 GB (218,407,878,656 bytes)
 Partition Starting Offset 16,384 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM ServeRAID SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 1
 SCSI Bus 0
 SCSI Logical Unit 0
 SCSI Port 7
 SCSI Target ID 5
 Sectors/Track 32

Size 33.90 GB (36,396,072,960 bytes)
 Total Cylinders 17,355
 Total Sectors 71,086,080
 Total Tracks 2,221,440
 Tracks/Cylinder 128
 Partition Disk #20, Partition #0
 Partition Size 33.90 GB (36,400,250,880 bytes)
 Partition Starting Offset 16,384 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM-ESXS ST336753LC FN SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 1
 SCSI Bus 0
 SCSI Logical Unit 0
 SCSI Port 3
 SCSI Target ID 0
 Sectors/Track 63
 Size 33.90 GB (36,396,864,000 bytes)
 Total Cylinders 4,425
 Total Sectors 71,087,625
 Total Tracks 1,128,375
 Tracks/Cylinder 255
 Partition Disk #0, Partition #0
 Partition Size 33.89 GB (36,388,606,464 bytes)
 Partition Starting Offset 32,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM-ESXS ST336753LC FN SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 1
 SCSI Bus 0
 SCSI Logical Unit 0
 SCSI Port 3
 SCSI Target ID 1
 Sectors/Track 63
 Size 33.90 GB (36,396,864,000 bytes)
 Total Cylinders 4,425
 Total Sectors 71,087,625
 Total Tracks 1,128,375
 Tracks/Cylinder 255
 Partition Disk #1, Partition #0
 Partition Size 33.90 GB (36,396,831,744 bytes)
 Partition Starting Offset 32,256 bytes

[SCSI]

Item	Value
Name	LSI Logic 1020/1030 Ultra320 SCSI Adapter
Manufacturer	LSI Logic
Status	OK
PNP Device ID	PCI\VEN_1000&DEV_0030&SUBSYS_10001000&REV_08\3&13C0B0C5&0&10
I/O Port	0x00002000-0x00003FFF
Memory Address	0xEFFB0000-0xEFFBFFFF
Memory Address	0xEFFA0000-0xEFFAFFFF
IRQ Channel	IRQ 18
Driver	c:\windows\system32\drivers\symmpi.sys (1.09.06.00 built by: WinDDK, 33.50 KB (34,304 bytes), 5/23/2003 8:15 AM)
Name	LSI Logic 1020/1030 Ultra320 SCSI Adapter
Manufacturer	LSI Logic
Status	OK
PNP Device ID	PCI\VEN_1000&DEV_0030&SUBSYS_10001000&REV_08\3&13C0B0C5&0&11
I/O Port	0x00002100-0x000021FF
Memory Address	0xEFF90000-0xEFF9FFFF
Memory Address	0xEFF80000-0xEFF8FFFF
IRQ Channel	IRQ 19

Driver c:\windows\system32\drivers\symmpi.sys (1.09.06.00 built by: WinDDK, 33.50 KB (34,304 bytes), 5/23/2003 8:15 AM)
 Name IBM ServeRAID 6M Controller
 Manufacturer IBM Corporation
 Status OK
 PNP Device ID
 PCI\VEN_9005&DEV_0250&SUBSYS_02791014&REV_02\4&14BEBEF2&0&4040
 Memory Address 0xEFC00000-0xEFC00FFF
 IRQ Channel IRQ 26
 Driver c:\windows\system32\drivers\nfrd6m.sys (6.06.08 built by: WinDDK, 29.50 KB (30,208 bytes), 5/5/2004 10:17 AM)
 Name IBM ServeRAID 6M Controller
 Manufacturer IBM Corporation
 Status OK
 PNP Device ID
 PCI\VEN_9005&DEV_0250&SUBSYS_02791014&REV_02\4&1AA00164&0&4040
 Memory Address 0xEDC00000-0xEDC00FFF
 IRQ Channel IRQ 32
 Driver c:\windows\system32\drivers\nfrd6m.sys (6.06.08 built by: WinDDK, 29.50 KB (30,208 bytes), 5/5/2004 10:17 AM)
 Name IBM ServeRAID 6M Controller
 Manufacturer IBM Corporation
 Status OK
 PNP Device ID
 PCI\VEN_9005&DEV_0250&SUBSYS_02791014&REV_02\4&16056E07&0&4040
 Memory Address 0xE7C00000-0xE7C00FFF
 IRQ Channel IRQ 68
 Driver c:\windows\system32\drivers\nfrd6m.sys (6.06.08 built by: WinDDK, 29.50 KB (30,208 bytes), 5/5/2004 10:17 AM)
 Name IBM ServeRAID 6M Controller
 Manufacturer IBM Corporation
 Status OK
 PNP Device ID
 PCI\VEN_9005&DEV_0250&SUBSYS_02791014&REV_02\4&3827E28B&0&4040
 Memory Address 0xE9C00000-0xE9C00FFF
 IRQ Channel IRQ 53
 Driver c:\windows\system32\drivers\nfrd6m.sys (6.06.08 built by: WinDDK, 29.50 KB (30,208 bytes), 5/5/2004 10:17 AM)
 Name IBM ServeRAID 6M Controller
 Manufacturer IBM Corporation
 Status OK
 PNP Device ID
 PCI\VEN_9005&DEV_0250&SUBSYS_02791014&REV_02\4&15C019C6&0&4040
 Memory Address 0xEBC00000-0xEBC00FFF
 IRQ Channel IRQ 36
 Driver c:\windows\system32\drivers\nfrd6m.sys (6.06.08 built by: WinDDK, 29.50 KB (30,208 bytes), 5/5/2004 10:17 AM)

[IDE]

Item	Value
Name	Standard Dual Channel PCI IDE Controller
Manufacturer	(Standard IDE ATA/ATAPI controllers)
Status	OK
PNP Device ID	
PCI\VEN_1166&DEV_0213&SUBSYS_02121014&REV_A0\3&267A616A&0&79	
I/O Port	0x00000700-0x0000070F
Driver	c:\windows\system32\drivers\pciide.sys (5.2.3790.0 (srv03_rtm.030324-2048), 5.50 KB (5,632 bytes), 3/25/2003 7:00 AM)
Name	Primary IDE Channel
Manufacturer	(Standard IDE ATA/ATAPI controllers)
Status	OK
PNP Device ID	PCI\IDE\IDECHANNEL\4&1D22FEAC&0&0
I/O Port	0x000001F0-0x000001F7
I/O Port	0x000003F6-0x000003F6
IRQ Channel	IRQ 14
Driver	c:\windows\system32\drivers\atapi.sys (5.2.3790.0 (srv03_rtm.030324-2048), 89.00 KB (91,136 bytes), 3/25/2003 7:00 AM)
Name	Secondary IDE Channel

Manufacturer (Standard IDE ATA/ATAPI controllers)
 Status OK
 PNP Device ID PCI\IDE\IDECHANNEL\4&1D22FEAC&0&1
 I/O Port 0x00000170-0x00000177
 I/O Port 0x00000376-0x00000376
 Driver c:\windows\system32\drivers\atapi.sys (5.2.3790.0 (srv03_rtm.030324-2048), 89.00 KB (91,136 bytes), 3/25/2003 7:00 AM)

[Printing]

Name	Driver	Port Name	Server Name
------	--------	-----------	-------------

[Problem Devices]

Device	PNP Device ID	Error Code
--------	---------------	------------

[USB]

Device	PNP Device ID
Standard OpenHCD USB Host Controller	PCI\VEN_1166&DEV_0221&SUBSYS_02B41014&REV_05\3&267A616A&0&7A
USB Root Hub	USB\ROOT_HUB\4&1AAA84E&0
USB Composite Device	USB\VID_04B3&PID_4001\IBMRAPT0R
USB Human Interface Device	USB\VID_04B3&PID_4001&MI_00\6&2061A23&0&0000
HID Keyboard Device	HID\VID_04B3&PID_4001&MI_00\7&29EB61BE&0&0000
USB Human Interface Device	USB\VID_04B3&PID_4001&MI_01\6&2061A23&0&0001
HID-compliant mouse	HID\VID_04B3&PID_4001&MI_01\7&180B3F5&0&0000
USB Human Interface Device	USB\VID_04B3&PID_4001&MI_02\6&2061A23&0&0002
HID-compliant device	HID\VID_04B3&PID_4001&MI_02\7&14B0D033&0&0000

[Software Environment]

[System Drivers]

Name	Description	File	Type	Started	Start Mode	State
Status	Error Control	Accept Pause		Accept Stop		
abiosdsk	Abiosdsk	Not Available	Kernel Driver		No	
Disabled	Stopped	OK	Ignore	No	No	
acpi	Microsoft ACPI Driver	c:\windows\system32\drivers\acpi.sys				
Kernel Driver	Yes	Boot	Running	OK	Normal	No
Yes						
acpiec	ACPIEC	c:\windows\system32\drivers\acpiec.sys				Kernel Driver
No	Disabled	Stopped	OK	Normal	No	No
adpu160m	adpu160m	Not Available	Kernel Driver		No	
Disabled	Stopped	OK	Normal	No	No	
adpu320	adpu320	Not Available	Kernel Driver		No	
Disabled	Stopped	OK	Normal	No	No	
afcnt	afcnt	Not Available	Kernel Driver		No	
Disabled	Stopped	OK	Normal	No	No	
afd	AFD Networking Support Environment					
c:\windows\system32\drivers\afd.sys			Kernel Driver		Yes	Auto
Running	OK	Normal	No	Yes		
aha154x	Aha154x	Not Available	Kernel Driver		No	
Disabled	Stopped	OK	Normal	No	No	
aic78u2	aic78u2	Not Available	Kernel Driver		No	
Disabled	Stopped	OK	Normal	No	No	
aic78xx	aic78xx	Not Available	Kernel Driver		No	
Disabled	Stopped	OK	Normal	No	No	
aliide	Aliide	Not Available	Kernel Driver		No	
Disabled	Stopped	OK	Normal	No	No	
asynccmac	RAS Asynchronous Media Driver					
c:\windows\system32\drivers\asynccmac.sys			Kernel Driver		No	Manual
Stopped	OK	Normal	No	No		
atapi	Standard IDE/ESDI Hard Disk Controller					
c:\windows\system32\drivers\atapi.sys			Kernel Driver		Yes	Boot
Running	OK	Normal	No	Yes		
atdisk	Atdisk	Not Available	Kernel Driver		No	
Disabled	Stopped	OK	Ignore	No	No	
ati2mtag	ati2mtag	c:\windows\system32\drivers\ati2mtag.sys				Kernel Driver
Yes	Manual	Running	OK	Ignore	No	Yes

atmarpc	ATM ARP Client Protocol	c:\windows\system32\drivers\atmarpc.sys		http	HTTP	c:\windows\system32\drivers\http.sys		Kernel Driver	
Kernel Driver	No	Manual	Stopped	No	Manual	Stopped	OK	Normal	No
No			OK	No			No	No	No
audstub	Audio Stub Driver	c:\windows\system32\drivers\audstub.sys		i2omgmt	i2omgmt	Not Available		Kernel Driver	No
Driver	Yes	Manual	Running	Stopped	OK	Normal		No	System
b57w2k	Broadcom NetXtreme Gigabit Ethernet			i2omp	i2omp	Not Available		Kernel Driver	No
c:\windows\system32\drivers\b57xp32.sys	Kernel Driver			Disabled	Stopped	OK		Normal	No
Running	OK	Normal	No	i8042prt	i8042 Keyboard and PS/2 Mouse Port Driver				No
beep	Beep	c:\windows\system32\drivers\beep.sys		c:\windows\system32\drivers\i8042prt.sys	Kernel Driver		Yes		
Yes	System	Running	OK	Running	OK	Normal	No	Yes	System
cbidf2k	cbidf2k	c:\windows\system32\drivers\cbidf2k.sys		ibmhp	IBMHPA	c:\windows\system32\drivers\ibmhp.sys		Kernel Driver	
No	Disabled	Stopped	OK	Yes	Manual	Running	OK	Normal	No
cd20xrnt	cd20xrnt	Not Available		iirsp	iirsp	Not Available		Kernel Driver	No
Disabled	Stopped	OK	Normal	Disabled	Stopped	OK		Normal	No
cdfs	Cdfs	c:\windows\system32\drivers\cdfs.sys		imapi	CD-Burning Filter Driver		c:\windows\system32\drivers\imapi.sys		
Yes	Disabled	Running	OK	Kernel Driver	No	System	Stopped	OK	Normal
cdrom	CD-ROM Driver	c:\windows\system32\drivers\cdrom.sys		No					No
Driver	Yes	System	Running	intelide	IntelIde	Not Available		Kernel Driver	No
changer	Changer	Not Available		Disabled	Stopped	OK		Normal	No
Stopped	OK	Ignore	No	ipfilterdriver	IP Traffic Filter Driver				
clusdisk	Cluster Disk Driver	c:\windows\system32\drivers\clusdisk.sys		c:\windows\system32\drivers\ipfltdrv.sys	Kernel Driver		No		
Driver	No	Disabled	Stopped	Stopped	OK	Normal	No	No	Manual
cmdide	CmdIde	Not Available		ipinip	IP in IP Tunnel Driver		c:\windows\system32\drivers\ipinip.sys		
Disabled	Stopped	OK	Normal	Driver	No	Manual	Stopped	OK	Normal
cpqarray	Cpqarray	Not Available		ipnat	IP Network Address Translator		c:\windows\system32\drivers\ipnat.sys		
Disabled	Stopped	OK	Normal	Kernel Driver	No	Manual	Stopped	OK	Normal
cpqarray2	cpqarray2	Not Available		No					No
Disabled	Stopped	OK	Normal	ipsec	IPSEC driver		c:\windows\system32\drivers\ipsec.sys		
cpqcissm	cpqcissm	Not Available		Driver	Yes	System	Running	OK	Normal
Disabled	Stopped	OK	Normal	ipsraidn	ipsraidn	Not Available		Kernel Driver	No
cpqfcalm	cpqfcalm	Not Available		Disabled	Stopped	OK		Normal	No
Disabled	Stopped	OK	Normal	isapnp	PnP ISA/EISA Bus Driver		c:\windows\system32\drivers\isapnp.sys		
credisk	CRC Disk Filter Driver	c:\windows\system32\drivers\credisk.sys		Kernel Driver	Yes	Boot	Running	OK	Critical
Kernel Driver	Yes	Boot	Running	Yes					No
Yes			OK	kbdclass	Keyboard Class Driver				
dac960nt	dac960nt	Not Available		c:\windows\system32\drivers\kbdclass.sys	Kernel Driver		Yes		
Disabled	Stopped	OK	Normal	Running	OK	Normal	No	Yes	System
dellcerc	dellcerc	Not Available		kbdhid	Keyboard HID Driver		c:\windows\system32\drivers\kbdhid.sys		
Disabled	Stopped	OK	Normal	Driver	Yes	System	Running	OK	Ignore
dfsdriver	DfsDriver	c:\windows\system32\drivers\dfs.sys		ksecdd	KSecDD	c:\windows\system32\drivers\ksecdd.sys		Kernel Driver	No
Yes	Boot	Running	OK	Yes	Boot	Running	OK	Normal	Yes
disk	Disk Driver	c:\windows\system32\drivers\disk.sys		lp6nds35	lp6nds35	Not Available		Kernel Driver	No
Driver	Yes	Boot	Running	Disabled	Stopped	OK		Normal	No
dmboot	dmboot	c:\windows\system32\drivers\dmboot.sys		mnmd	mnmd	c:\windows\system32\drivers\mnmd.sys		Kernel Driver	
Yes	Disabled	Running	OK	Yes	System	Running	OK	Ignore	No
dmio	Logical Disk Manager Driver	c:\windows\system32\drivers\dmio.sys		modem	Modem	c:\windows\system32\drivers\modem.sys		Kernel Driver	
Kernel Driver	Yes	Boot	Running	No	Manual	Stopped	OK	Ignore	No
Yes			OK	mouclass	Mouse Class Driver		c:\windows\system32\drivers\mouclass.sys		
dmload	dmload	c:\windows\system32\drivers\dmload.sys		Driver	Yes	System	Running	OK	Normal
Yes	Boot	Running	OK	mouhid	Mouse HID Driver		c:\windows\system32\drivers\mouhid.sys		
dpti2o	dpti2o	Not Available		Driver	Yes	Manual	Running	OK	Ignore
Disabled	Stopped	OK	Normal	mountmgr	Mount Point Manager		c:\windows\system32\drivers\mountmgr.sys		
fastfat	Fastfat	c:\windows\system32\drivers\fastfat.sys		Driver	Yes	Boot	Running	OK	Normal
No	Disabled	Stopped	OK	mraid35x	mraid35x	Not Available		Kernel Driver	No
fdc	Floppy Disk Controller Driver	c:\windows\system32\drivers\fdc.sys		Disabled	Stopped	OK		Normal	No
Kernel Driver	Yes	Manual	Running	mrxdav	WebDav Client Redirector		c:\windows\system32\drivers\mrxdav.sys		
Yes			OK	File System Driver	No	Manual	Stopped	OK	Normal
fips	Fips	c:\windows\system32\drivers\fips.sys		No					No
Yes	System	Running	OK	mrxsmb	MRXSMB		c:\windows\system32\drivers\mrxsmb.sys		
flpydisk	Floppy Disk Driver	c:\windows\system32\drivers\flpydisk.sys		Yes	System	Running	OK	Normal	No
Driver	Yes	Manual	Running	msfs	Msfs	c:\windows\system32\drivers\msfs.sys		File System Driver	
ftdisk	Volume Manager Driver	c:\windows\system32\drivers\ftdisk.sys		Yes	System	Running	OK	Normal	No
Kernel Driver	Yes	Boot	Running	mup	Mup	c:\windows\system32\drivers\mup.sys		File System Driver	
Yes			OK	Yes	Boot	Running	OK	Normal	No
gpc	Generic Packet Classifier	c:\windows\system32\drivers\msgpc.sys		ndis	NDIS System Driver		c:\windows\system32\drivers\ndis.sys		
Kernel Driver	Yes	Manual	Running	Driver	Yes	Boot	Running	OK	Normal
Yes			OK	ndistapi	Remote Access NDIS TAPI Driver				
hidusb	Microsoft HID Class Driver	c:\windows\system32\drivers\hidusb.sys		c:\windows\system32\drivers\ndistapi.sys	Kernel Driver		Yes		
Kernel Driver	Yes	Manual	Running	Running	OK	Normal	No	Yes	Manual
Yes			OK	ndisui	NDIS Usermode I/O Protocol		c:\windows\system32\drivers\ndisui.sys		
hpn	hpn	Not Available		Kernel Driver	No	Manual	Stopped	OK	Normal
Disabled	Stopped	OK	Normal	No					No
hpt3xx	hpt3xx	Not Available		ndiswan	Remote Access NDIS WAN Driver				
Disabled	Stopped	OK	Normal	c:\windows\system32\drivers\ndiswan.sys	Kernel Driver		Yes		
			No	Running	OK	Normal	No	Yes	Manual

ndproxy	NDIS Proxy	c:\windows\system32\drivers\ndproxy.sys	Kernel				
Driver	Yes	Manual	Running	OK	Normal	No	Yes
netbios	NetBIOS Interface	c:\windows\system32\drivers\netbios.sys	File				
System Driver	Yes	System	Running	OK	Normal	No	No
Yes							
netbt	NetBios over Tcpip	c:\windows\system32\drivers\netbt.sys	Kernel				
Driver	Yes	System	Running	OK	Normal	No	Yes
nfrd6m	IBM ServeRAID 6M Device Driver						
c:\windows\system32\drivers\nfrd6m.sys			Kernel Driver		Yes		Boot
Running	OK	Normal	No	Yes			
nfrd6mpf	IBM ServeRAID 6M Performance Driver						
c:\windows\system32\drivers\nfrd6mpf.sys			Kernel Driver		Yes		Boot
Running	OK	Normal	No	Yes			
nfrd960	nfrd960	Not Available	Kernel Driver		No		
Disabled	Stopped	OK	Normal	No	No		
npfs	Npfs	c:\windows\system32\drivers\npfs.sys	File System Driver				
Yes	System	Running	OK	Normal	No	Yes	
ntfs	Ntfs	c:\windows\system32\drivers\ntfs.sys	File System Driver				
Yes	Disabled	Running	OK	Normal	No	Yes	
null	Null	c:\windows\system32\drivers\null.sys	Kernel Driver				
Yes	System	Running	OK	Normal	No	Yes	
parport	Parport	c:\windows\system32\drivers\parport.sys	Kernel Driver				
No	Manual	Stopped	OK	Ignore	No	No	
partmgr	Partition Manager	c:\windows\system32\drivers\partmgr.sys	Kernel				
Driver	Yes	Boot	Running	OK	Normal	No	Yes
pci	PCI Bus Driver	c:\windows\system32\drivers\pci.sys	Kernel				
Driver	Yes	Boot	Running	OK	Critical	No	Yes
pciide	PCIIde	c:\windows\system32\drivers\pciide.sys	Kernel Driver				
Yes	Boot	Running	OK	Normal	No	Yes	
pcmcia	Pcmcia	c:\windows\system32\drivers\pcmcia.sys	Kernel Driver				
No	Disabled	Stopped	OK	Normal	No	No	
pdcomp	PDCOMP	Not Available	Kernel Driver		No		Manual
Stopped	OK	Ignore	No	No			
pdframe	PDFRAME		Not Available		Kernel Driver		No
Manual	Stopped	OK	Ignore	No	No		
pdrel	PDRELI	Not Available	Kernel Driver		No		Manual
Stopped	OK	Ignore	No	No			
pdframe	PDRFRAME		Not Available		Kernel Driver		No
Manual	Stopped	OK	Ignore	No	No		
perc2	perc2	Not Available	Kernel Driver		No		
Disabled	Stopped	OK	Normal	No	No		
perc2hib	perc2hib	Not Available	Kernel Driver		No		
Disabled	Stopped	OK	Normal	No	No		
pnpmem	Microsoft Memory Module Driver						
c:\windows\system32\drivers\pnpmem.sys			Kernel Driver		Yes		Manual
Running	OK	Normal	No	Yes			
pptpminiport	WAN Miniport (PPTP)						
c:\windows\system32\drivers\rasppptp.sys			Kernel Driver		Yes		Manual
Running	OK	Normal	No	Yes			
processor	Processor Driver	c:\windows\system32\drivers\processr.sys	Kernel				
Driver	Yes	Manual	Running	OK	Normal	No	Yes
ptilink	Direct Parallel Link Driver	c:\windows\system32\drivers\ptilink.sys					
Kernel Driver	Yes	Manual	Running	OK	Normal	No	
Yes							
ql1080	ql1080	Not Available	Kernel Driver		No		
Disabled	Stopped	OK	Normal	No	No		
ql10wnt	Ql10wnt	Not Available	Kernel Driver		No		
Disabled	Stopped	OK	Normal	No	No		
ql12160	ql12160	Not Available	Kernel Driver		No		
Disabled	Stopped	OK	Normal	No	No		
ql1240	ql1240	Not Available	Kernel Driver		No		
Disabled	Stopped	OK	Normal	No	No		
ql1280	ql1280	Not Available	Kernel Driver		No		
Disabled	Stopped	OK	Normal	No	No		
ql2100	ql2100	Not Available	Kernel Driver		No		
Disabled	Stopped	OK	Normal	No	No		
ql2200	ql2200	Not Available	Kernel Driver		No		
Disabled	Stopped	OK	Normal	No	No		
ql2300	ql2300	Not Available	Kernel Driver		No		
Disabled	Stopped	OK	Normal	No	No		
rasacd	Remote Access Auto Connection Driver						
c:\windows\system32\drivers\rasacd.sys			Kernel Driver		Yes		System
Running	OK	Normal	No	Yes			
rasl2tp	WAN Miniport (L2TP)						
Kernel Driver	Yes	Manual	Running	OK	Normal	No	
Yes							
rasppoe	Remote Access PPPOE Driver						
c:\windows\system32\drivers\rasppoe.sys			Kernel Driver		Yes		Manual
Running	OK	Normal	No	Yes			
raspti	Direct Parallel	c:\windows\system32\drivers\raspti.sys					Kernel
Driver	Yes	Manual	Running	OK	Normal	No	Yes
rdbss	Rdbss	c:\windows\system32\drivers\rdbss.sys	File System Driver				
Yes	System	Running	OK	Normal	No	Yes	
rdpcdd	RDPCDD	c:\windows\system32\drivers\rdpcdd.sys	Kernel Driver				
Yes	System	Running	OK	Ignore	No	Yes	
rdpdr	Terminal Server Device Redirector Driver						
c:\windows\system32\drivers\rdpdr.sys			Kernel Driver		Yes		Manual
Running	OK	Normal	No	Yes			
rdpwd	RDPWD	c:\windows\system32\drivers\rdpwd.sys	Kernel Driver				
Yes	Manual	Running	OK	Ignore	No	Yes	
redbook	Digital CD Audio Playback Filter Driver						
c:\windows\system32\drivers\redbook.sys			Kernel Driver		Yes		System
Running	OK	Normal	No	Yes			
secdrv	Secdrv	c:\windows\system32\drivers\secdrv.sys	Kernel Driver				
No	Manual	Stopped	OK	Normal	No	No	
serenum	Serenum Filter Driver	c:\windows\system32\drivers\serenum.sys	Kernel				
Driver	Yes	Manual	Running	OK	Normal	No	Yes
serial	Serial port driver	c:\windows\system32\drivers\serial.sys	Kernel				
Driver	Yes	System	Running	OK	Ignore	No	Yes
sfloppy	Sfloppy	c:\windows\system32\drivers\sfloppy.sys	Kernel Driver				
No	System	Stopped	OK	Ignore	No	No	
simbad	Simbad	Not Available	Kernel Driver		No		
Disabled	Stopped	OK	Normal	No	No		
sparrow	Sparrow	Not Available	Kernel Driver		No		
Disabled	Stopped	OK	Normal	No	No		
srv	Srv	c:\windows\system32\drivers\srv.sys	File System Driver				
Yes	Manual	Running	OK	Normal	No	Yes	
swenum	Software Bus Driver	c:\windows\system32\drivers\swenum.sys	Kernel				
Driver	Yes	Manual	Running	OK	Normal	No	Yes
symc810	symc810	Not Available	Kernel Driver		No		
Disabled	Stopped	OK	Normal	No	No		
symc8xx	symc8xx	Not Available	Kernel Driver		No		
Disabled	Stopped	OK	Normal	No	No		
symmpi	symmpi	c:\windows\system32\drivers\symmpi.sys	Kernel Driver				
Yes	Boot	Running	OK	Normal	No	Yes	
sym_hi	sym_hi	Not Available	Kernel Driver		No		
Disabled	Stopped	OK	Normal	No	No		
sym_u3	sym_u3	Not Available	Kernel Driver		No		
Disabled	Stopped	OK	Normal	No	No		
tcpip	TCP/IP Protocol Driver	c:\windows\system32\drivers\tcpip.sys					
Kernel Driver	Yes	System	Running	OK	Normal	No	
Yes							
tdpipe	TDPIPE	c:\windows\system32\drivers\tdpipe.sys	Kernel Driver				
No	Manual	Stopped	OK	Ignore	No	No	
tdtcp	TDTCP	c:\windows\system32\drivers\tdtcp.sys	Kernel Driver				
Yes	Manual	Running	OK	Ignore	No	Yes	
termdd	Terminal Device Driver	c:\windows\system32\drivers\termdd.sys					
Kernel Driver	Yes	System	Running	OK	Normal	No	
Yes							
toside	TosIde	Not Available	Kernel Driver		No		
Disabled	Stopped	OK	Normal	No	No		
udfs	Udfs	c:\windows\system32\drivers\udfs.sys	File System Driver				
No	Disabled	Stopped	OK	Normal	No	No	
ultra	ultra	Not Available	Kernel Driver		No		
Disabled	Stopped	OK	Normal	No	No		
update	Microcode Update Driver	c:\windows\system32\drivers\update.sys					
Kernel Driver	Yes	Manual	Running	OK	Normal	No	
Yes							
usbccgp	Microsoft USB Generic Parent Driver						
c:\windows\system32\drivers\usbccgp.sys			Kernel Driver		Yes		Manual
Running	OK	Normal	No	Yes			
usbhub	USB2 Enabled Hub	c:\windows\system32\drivers\usbhub.sys	Kernel				
Driver	Yes	Manual	Running	OK	Normal	No	Yes
usbohci	Microsoft USB Open Host Controller Miniport Driver						
c:\windows\system32\drivers\usbohci.sys			Kernel Driver		Yes		Manual
Running	OK	Normal	No	Yes			

vgasave	VGA Display Controller.	c:\windows\system32\drivers\vga.sys	
Kernel Driver	Yes	System	Running OK Ignore No
Yes			
viaide	Vialde	Not Available	Kernel Driver No
Disabled	Stopped	OK	Normal No No
volsnap	Storage volumes	c:\windows\system32\drivers\volsnap.sys	Kernel
Driver	Yes	Boot	Running OK Normal No Yes
wanarp	Remote Access IP ARP Driver	c:\windows\system32\drivers\wanarp.sys	
Kernel Driver	Yes	Manual	Running OK Normal No
Yes			
wdica	WDICA	Not Available	Kernel Driver No Manual
Stopped	OK	Ignore	No No
wlbs	Network Load Balancing	c:\windows\system32\drivers\wlbs.sys	
Kernel Driver	No	Manual	Stopped OK Normal No
No			

[Signed Drivers]

Device Name	Signed	Device Class	Driver Version	Driver
Date	Manufacturer	INF Name	Driver Name	Device ID
Not Available	Not Available	Not Available	Not Available	Not Available
Not Available	Not Available	Not Available	Not Available	Not Available
HTREE\ROOT\0				
ACPI Multiprocessor PC	No	COMPUTER	5.2.3790.0	
10/1/2002 (Standard computers)	hal.inf	Not Available		
ROOT\ACPI_HAL\0000				
Microsoft ACPI-Compliant System	No	SYSTEM	5.2.3790.0	
10/1/2002	Microsoft	acpi.inf	Not Available	ACPI_HAL\PNP0C08\0
Processor	No	PROCESSOR	5.2.3790.0	10/1/2002 (Standard
processor types)	cpu.inf	Not Available		
ACPI\GENUINEINTEL - _X86_FAMILY_15_MODEL_2_0				
Processor	No	PROCESSOR	5.2.3790.0	10/1/2002 (Standard
processor types)	cpu.inf	Not Available		
ACPI\GENUINEINTEL - _X86_FAMILY_15_MODEL_2_1				
Processor	No	PROCESSOR	5.2.3790.0	10/1/2002 (Standard
processor types)	cpu.inf	Not Available		
ACPI\GENUINEINTEL - _X86_FAMILY_15_MODEL_2_2				
Processor	No	PROCESSOR	5.2.3790.0	10/1/2002 (Standard
processor types)	cpu.inf	Not Available		
ACPI\GENUINEINTEL - _X86_FAMILY_15_MODEL_2_3				
Processor	No	PROCESSOR	5.2.3790.0	10/1/2002 (Standard
processor types)	cpu.inf	Not Available		
ACPI\GENUINEINTEL - _X86_FAMILY_15_MODEL_2_4				
Processor	No	PROCESSOR	5.2.3790.0	10/1/2002 (Standard
processor types)	cpu.inf	Not Available		
ACPI\GENUINEINTEL - _X86_FAMILY_15_MODEL_2_5				
Processor	No	PROCESSOR	5.2.3790.0	10/1/2002 (Standard
processor types)	cpu.inf	Not Available		
ACPI\GENUINEINTEL - _X86_FAMILY_15_MODEL_2_6				
Processor	No	PROCESSOR	5.2.3790.0	10/1/2002 (Standard
processor types)	cpu.inf	Not Available		
ACPI\GENUINEINTEL - _X86_FAMILY_15_MODEL_2_7				
PCI bus	No	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)
machine.inf	Not Available	ACPI\PNP0A03\0		
PCI standard host CPU bridge	No	SYSTEM	5.2.3790.0	10/1/2002
(Standard system devices)	machine.inf	Not Available		
PCI\VEN_1014&DEV_0302&SUBSYS_00000000&REV_05\3&267A616A&0&00				
RADEON 7000 RADEON VE Family (Microsoft Corporation)	No			
DISPLAY 6.13.10.6153	8/8/2002	ATI Technologies Inc.		
atiixpag.inf	Not Available			
PCI\VEN_1002&DEV_5159&SUBSYS_029A1014&REV_00\3&267A616A&0&40				
Plug and Play Monitor	No	MONITOR	5.1.2001.0	
6/6/2001 (Standard monitor types)	monitor.inf	Not Available		
DISPLAY\IBM0000\4&3AF738E&0&10000082&00&08				
Plug and Play Monitor	No	MONITOR	5.1.2001.0	
6/6/2001 (Standard monitor types)	monitor.inf	Not Available		
DISPLAY\AVO0402\4&3AF738E&0&10000080&00&08				
ServerWorks Champion CSB6 - South Bridge 6	No	SYSTEM		
5.2.3790.0 10/1/2002	ServerWorks (RCC)	machine.inf	Not Available	
PCI\VEN_1166&DEV_0203&SUBSYS_00000000&REV_A0\3&267A616A&0&78				
Standard 101/102-Key or Microsoft Natural PS/2 Keyboard	No			
KEYBOARD	5.2.3790.0	10/1/2002 (Standard keyboards)	keyboard.inf	
Not Available	ACPI\PNP0303\4&39D9D85A&0			

PS/2 Compatible Mouse	No	MOUSE	5.2.3790.0	10/1/2002
Microsoft msmouse.inf	Not Available			
ACPI\PNP0F13\4&39D9D85A&0				
Standard floppy disk controller	No	FDC	5.2.3790.0	10/1/2002
(Standard floppy disk controllers)	fdc.inf	Not Available		
ACPI\PNP0700\4&39D9D85A&0				
Floppy disk drive	No	FLOPPYDISK	5.2.3790.0	10/1/2002
(Standard floppy disk drives)	flydisk.inf	Not Available		
FDC\GENERIC_FLOPPY_DRIVE\5&2BED2454&0&0				
Communications Port	No	PORTS	5.2.3790.0	10/1/2002 (Standard port
types)	msports.inf	Not Available		
Advanced programmable interrupt controller	No	SYSTEM	5.2.3790.0	
10/1/2002 (Standard system devices)	machine.inf	Not Available		
ACPI\PNP0003\4&39D9D85A&0				
Direct memory access controller	No	SYSTEM	5.2.3790.0	10/1/2002
(Standard system devices)	machine.inf	Not Available		
ACPI\PNP0200\4&39D9D85A&0				
System timer	No	SYSTEM	5.2.3790.0	10/1/2002 (Standard system
devices)	machine.inf	Not Available		
ACPI\PNP0100\4&39D9D85A&0				
System CMOS/real time clock	No	SYSTEM	5.2.3790.0	10/1/2002
(Standard system devices)	machine.inf	Not Available		
ACPI\PNP0B00\4&39D9D85A&0				
System speaker	No	SYSTEM	5.2.3790.0	10/1/2002 (Standard system
devices)	machine.inf	Not Available		
ACPI\PNP0800\4&39D9D85A&0				
Numeric data processor	No	SYSTEM	5.2.3790.0	10/1/2002
(Standard system devices)	machine.inf	Not Available		
ACPI\PNP0C04\4&39D9D85A&0				
Motherboard resources	No	SYSTEM	5.2.3790.0	10/1/2002
(Standard system devices)	machine.inf	Not Available		
ACPI\PNP0C02\4				
Standard Dual Channel PCI IDE Controller	No	HDC	5.2.3790.0	
10/1/2002 (Standard IDE ATA/ATAPI controllers)	mshdc.inf	Not Available		
PCI\VEN_1166&DEV_0213&SUBSYS_02121014&REV_A0\3&267A616A&0&79				
Primary IDE Channel	No	HDC	5.2.3790.0	10/1/2002 (Standard IDE
ATA/ATAPI controllers)	mshdc.inf	Not Available		
PCI\IDE\IDECHANNEL\4&1D22FEAC&0&0				
CD-ROM Drive	No	CDROM	5.2.3790.0	10/1/2002 (Standard
CD-ROM drives)	cdrom.inf	Not Available		
IDE\CDROM\TEAC_CD-224E			2.9B	\5&B1305
48&0&0.0.0				
Secondary IDE Channel	No	HDC	5.2.3790.0	10/1/2002
(Standard IDE ATA/ATAPI controllers)	mshdc.inf	Not Available		
PCI\IDE\IDECHANNEL\4&1D22FEAC&0&1				
Standard OpenHCD USB Host Controller	No	USB	5.2.3790.0	
10/1/2002 (Standard USB Host Controller)	usbport.inf	Not Available		
PCI\VEN_1166&DEV_0221&SUBSYS_02B41014&REV_05\3&267A616A&0&7A				
USB Root Hub	No	USB	5.2.3790.0	10/1/2002 (Standard USB
Host Controller)	usbport.inf	Not Available		
USB\ROOT_HUB\4&1AAA84E&0				
USB Composite Device	No	USB	5.2.3790.0	10/1/2002
(Standard USB Host Controller)	usb.inf	Not Available		
USB\VID_04B3&PID_4001\IBMRAPTOR				
USB Human Interface Device	No	HIDCLASS	5.2.3790.0	
10/1/2002 (Standard system devices)	input.inf	Not Available		
USB\VID_04B3&PID_4001&MI_00\6&2061A23&0&0000				
HID Keyboard Device	No	KEYBOARD	5.2.3790.0	
10/1/2002 (Standard keyboards)	keyboard.inf	Not Available		
HID\VID_04B3&PID_4001&MI_00\7&29EB61BE&0&0000				
USB Human Interface Device	No	HIDCLASS	5.2.3790.0	
10/1/2002 (Standard system devices)	input.inf	Not Available		
USB\VID_04B3&PID_4001&MI_01\6&2061A23&0&0001				
HID-compliant mouse	No	MOUSE	5.2.3790.0	10/1/2002
Microsoft msmouse.inf	Not Available			
HID\VID_04B3&PID_4001&MI_01\7&180B3F5&0&0000				
USB Human Interface Device	No	HIDCLASS	5.2.3790.0	
10/1/2002 (Standard system devices)	input.inf	Not Available		
USB\VID_04B3&PID_4001&MI_02\6&2061A23&0&0002				
HID-compliant device	No	HIDCLASS	5.2.3790.0	10/1/2002
(Standard system devices)	input.inf	Not Available		
HID\VID_04B3&PID_4001&MI_02\7&14B0D033&0&0000				

Serverworks Champion CSB6 - SouthBridge 6 LPC No SYSTEM 5.2.3790.0 10/1/2002 ServerWorks (RCC) machine.inf Not Available
PCI\VEN_1166&DEV_0227&SUBSYS_00000000&REV_00\3&267A616A&0&0&7B
ISAPNP Read Data Port No SYSTEM 5.2.3790.0 10/1/2002
(Standard system devices) machine.inf Not Available
ISAPNP\READDATAPORT\0
PCI bus No SYSTEM 5.2.3790.0 10/1/2002 (Standard system devices)
machine.inf Not Available ACPI\PNP0A03\1
PCI standard host CPU bridge No SYSTEM 5.2.3790.0 10/1/2002
(Standard system devices) machine.inf Not Available
PCI\VEN_1014&DEV_0302&SUBSYS_00000000&REV_05\3&13C0B0C5&0&00
Broadcom NetXtreme Gigabit Ethernet No NET 7.33.0.0
10/16/2003 Broadcom oem5.inf Not Available
PCI\VEN_14E4&DEV_1648&SUBSYS_02B41014&REV_02\3&13C0B0C5&0&08
Broadcom NetXtreme Gigabit Ethernet No NET 7.33.0.0
10/16/2003 Broadcom oem5.inf Not Available
PCI\VEN_14E4&DEV_1648&SUBSYS_02B41014&REV_02\3&13C0B0C5&0&09
LSI Logic 1020/1030 Ultra320 SCSI Adapter No SCSIADAPTER
1.9.6.0 5/23/2003 LSI Logic oem3.inf Not Available
PCI\VEN_1000&DEV_0030&SUBSYS_10001000&REV_08\3&13C0B0C5&0&10
LSI Logic 1020/1030 Ultra320 SCSI Adapter No SCSIADAPTER
1.9.6.0 5/23/2003 LSI Logic oem3.inf Not Available
PCI\VEN_1000&DEV_0030&SUBSYS_10001000&REV_08\3&13C0B0C5&0&11
Disk drive No DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk
drives) disk.inf Not Available
SCSI\DISK&VEN_IBM-ESXS&PROD_ST336753LC____FN&REV_B85B\4&1986
76A3&0&000
Disk drive No DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk
drives) disk.inf Not Available
SCSI\DISK&VEN_IBM-ESXS&PROD_ST336753LC____FN&REV_B85B\4&1986
76A3&0&010
SCSI Processor Device No SYSTEM 5.2.3790.0 10/1/2002 IBM
scsidev.inf Not Available
SCSI\PROCESSOR&VEN_IBM&PROD_32P0032A_S320_1&REV_1\4&198676A
3&0&080
PCI standard PCI-to-PCI bridge No SYSTEM 5.2.3790.0 10/1/2002
(Standard system devices) machine.inf Not Available
PCI\VEN_1014&DEV_01A7&SUBSYS_00000000&REV_02\3&13C0B0C5&0&40
IBM ServeRAID 6M Controller No SCSIADAPTER 6.6.8.0
1/14/2004 IBM Corporation oem2.inf Not Available
PCI\VEN_9005&DEV_0250&SUBSYS_02791014&REV_02\4&14BEBF2&0&40
40
Disk drive No DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk
drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.06\5&376D3368&0&000
Disk drive No DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk
drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.06\5&376D3368&0&010
Disk drive No DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk
drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.06\5&376D3368&0&020
Disk drive No DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk
drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.06\5&376D3368&0&030
Disk drive No DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk
drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.06\5&376D3368&0&040
Disk drive No DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk
drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.06\5&376D3368&0&050
SCSI Processor Device No SYSTEM 5.2.3790.0 10/1/2002 IBM
scsidev.inf Not Available
SCSI\PROCESSOR&VEN_IBM&PROD_EXP400____S320&REV_D110\5&376D33
68&0&1F0
SCSI Processor Device No SYSTEM 5.2.3790.0 10/1/2002 IBM
scsidev.inf Not Available
SCSI\PROCESSOR&VEN_IBM&PROD_EXP400____S320&REV_D110\5&376D33
68&0&2F0
IBM Dummy Device No SYSTEM 5.2.3790.0 10/1/2002 IBM
scsidev.inf Not Available
SCSI\BRIDGE&VEN_IBM&PROD_DUMMY_DEVICE&REV_6.06\5&376D3368
&0&300
PCI bus No SYSTEM 5.2.3790.0 10/1/2002 (Standard system devices)
machine.inf Not Available ACPI\PNP0A03\2

PCI standard host CPU bridge No SYSTEM 5.2.3790.0 10/1/2002
(Standard system devices) machine.inf Not Available
PCI\VEN_1014&DEV_0302&SUBSYS_00000000&REV_05\3&1070020&0&00
PCI standard PCI-to-PCI bridge No SYSTEM 5.2.3790.0 10/1/2002
(Standard system devices) machine.inf Not Available
PCI\VEN_1014&DEV_01A7&SUBSYS_00000000&REV_02\3&1070020&0&40
IBM ServeRAID 6M Controller No SCSIADAPTER 6.6.8.0
1/14/2004 IBM Corporation oem2.inf Not Available
PCI\VEN_9005&DEV_0250&SUBSYS_02791014&REV_02\4&1AA00164&0&404
0
Disk drive No DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk
drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.06\5&34B934D7&0&000
Disk drive No DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk
drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.06\5&34B934D7&0&010
Disk drive No DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk
drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.06\5&34B934D7&0&020
Disk drive No DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk
drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.06\5&34B934D7&0&030
Disk drive No DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk
drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.06\5&34B934D7&0&040
Disk drive No DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk
drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.06\5&34B934D7&0&050
SCSI Processor Device No SYSTEM 5.2.3790.0 10/1/2002 IBM
scsidev.inf Not Available
SCSI\PROCESSOR&VEN_IBM&PROD_EXP400____S320&REV_D110\5&34B934
D7&0&1F0
SCSI Processor Device No SYSTEM 5.2.3790.0 10/1/2002 IBM
scsidev.inf Not Available
SCSI\PROCESSOR&VEN_IBM&PROD_EXP400____S320&REV_D110\5&34B934
D7&0&2F0
IBM Dummy Device No SYSTEM 5.2.3790.0 10/1/2002 IBM
scsidev.inf Not Available
SCSI\BRIDGE&VEN_IBM&PROD_DUMMY_DEVICE&REV_6.06\5&34B934D7
&0&300
PCI bus No SYSTEM 5.2.3790.0 10/1/2002 (Standard system devices)
machine.inf Not Available ACPI\PNP0A03\3
PCI standard host CPU bridge No SYSTEM 5.2.3790.0 10/1/2002
(Standard system devices) machine.inf Not Available
PCI\VEN_1014&DEV_0302&SUBSYS_00000000&REV_05\3&29E81982&0&00
PCI standard PCI-to-PCI bridge No SYSTEM 5.2.3790.0 10/1/2002
(Standard system devices) machine.inf Not Available
PCI\VEN_1014&DEV_01A7&SUBSYS_00000000&REV_02\3&29E81982&0&40
IBM ServeRAID 6M Controller No SCSIADAPTER 6.6.8.0
1/14/2004 IBM Corporation oem2.inf Not Available
PCI\VEN_9005&DEV_0250&SUBSYS_02791014&REV_02\4&16056E07&0&404
0
Disk drive No DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk
drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.06\5&2C62206E&0&000
Disk drive No DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk
drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.06\5&2C62206E&0&010
Disk drive No DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk
drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.06\5&2C62206E&0&020
Disk drive No DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk
drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.06\5&2C62206E&0&030
Disk drive No DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk
drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.06\5&2C62206E&0&040
Disk drive No DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk
drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.06\5&2C62206E&0&050
SCSI Processor Device No SYSTEM 5.2.3790.0 10/1/2002 IBM
scsidev.inf Not Available
SCSI\PROCESSOR&VEN_IBM&PROD_EXP400____S320&REV_D110\5&2C6220
6E&0&1F0

SCSI Processor Device No SYSTEM 5.2.3790.0 10/1/2002 IBM
 scsdev.inf Not Available
 SCSI\PROCESSOR&VEN_IBM&PROD_EXP400___S320&REV_D110\5&2C6220
 6E&0&2F0
 IBM Dummy Device No SYSTEM 5.2.3790.0 10/1/2002 IBM
 scsdev.inf Not Available
 SCSI\BRIDGE&VEN_IBM&PROD_DUMMY_DEVICE&REV_6.06\5&2C62206E
 &0&300
 PCI bus No SYSTEM 5.2.3790.0 10/1/2002 (Standard system devices)
 machine.inf Not Available ACPI\PNP0A03\4
 PCI standard host CPU bridge No SYSTEM 5.2.3790.0 10/1/2002
 (Standard system devices) machine.inf Not Available
 PCI\VEN_1014&DEV_0302&SUBSYS_00000000&REV_05\3&172E68DD&0&00
 PCI standard PCI-to-PCI bridge No SYSTEM 5.2.3790.0 10/1/2002
 (Standard system devices) machine.inf Not Available
 PCI\VEN_1014&DEV_01A7&SUBSYS_00000000&REV_02\3&172E68DD&0&40
 IBM ServeRAID 6M Controllor No SCSIADAPTER 6.6.8.0
 1/14/2004 IBM Corporation oem2.inf Not Available
 PCI\VEN_9005&DEV_0250&SUBSYS_02791014&REV_02\4&3827E28B&0&404
 0
 Disk drive No DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk
 drives) disk.inf Not Available
 SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.06\5&3932506E&0&000
 Disk drive No DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk
 drives) disk.inf Not Available
 SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.06\5&3932506E&0&010
 Disk drive No DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk
 drives) disk.inf Not Available
 SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.06\5&3932506E&0&020
 Disk drive No DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk
 drives) disk.inf Not Available
 SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.06\5&3932506E&0&030
 Disk drive No DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk
 drives) disk.inf Not Available
 SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.06\5&3932506E&0&040
 Disk drive No DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk
 drives) disk.inf Not Available
 SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.06\5&3932506E&0&050
 SCSI Processor Device No SYSTEM 5.2.3790.0 10/1/2002 IBM
 scsdev.inf Not Available
 SCSI\PROCESSOR&VEN_IBM&PROD_EXP400___S320&REV_D110\5&3932506
 E&0&1F0
 SCSI Processor Device No SYSTEM 5.2.3790.0 10/1/2002 IBM
 scsdev.inf Not Available
 SCSI\PROCESSOR&VEN_IBM&PROD_EXP400___S320&REV_D110\5&3932506
 E&0&2F0
 IBM Dummy Device No SYSTEM 5.2.3790.0 10/1/2002 IBM
 scsdev.inf Not Available
 SCSI\BRIDGE&VEN_IBM&PROD_DUMMY_DEVICE&REV_6.06\5&3932506E
 &0&300
 PCI bus No SYSTEM 5.2.3790.0 10/1/2002 (Standard system devices)
 machine.inf Not Available ACPI\PNP0A03\5
 PCI standard host CPU bridge No SYSTEM 5.2.3790.0 10/1/2002
 (Standard system devices) machine.inf Not Available
 PCI\VEN_1014&DEV_0302&SUBSYS_00000000&REV_05\3&474B838&0&00
 PCI standard PCI-to-PCI bridge No SYSTEM 5.2.3790.0 10/1/2002
 (Standard system devices) machine.inf Not Available
 PCI\VEN_1014&DEV_01A7&SUBSYS_00000000&REV_02\3&474B838&0&40
 IBM ServeRAID 6M Controllor No SCSIADAPTER 6.6.8.0
 1/14/2004 IBM Corporation oem2.inf Not Available
 PCI\VEN_9005&DEV_0250&SUBSYS_02791014&REV_02\4&15C019C6&0&404
 0
 Disk drive No DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk
 drives) disk.inf Not Available
 SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.06\5&1BBD53DD&0&000
 SCSI Processor Device No SYSTEM 5.2.3790.0 10/1/2002 IBM
 scsdev.inf Not Available
 SCSI\PROCESSOR&VEN_IBM&PROD_EXP300___S160&REV_D014\5&1BBD5
 3DD&0&1F0
 SCSI Processor Device No SYSTEM 5.2.3790.0 10/1/2002 IBM
 scsdev.inf Not Available
 SCSI\PROCESSOR&VEN_IBM&PROD_EXP300___S160&REV_D014\5&1BBD5
 3DD&0&2F0

IBM Dummy Device No SYSTEM 5.2.3790.0 10/1/2002 IBM
 scsdev.inf Not Available
 SCSI\BRIDGE&VEN_IBM&PROD_DUMMY_DEVICE&REV_6.06\5&1BBD53D
 D&0&300
 Motherboard resources No SYSTEM 5.2.3790.0 10/1/2002
 (Standard system devices) machine.inf Not Available
 ACPI\PNP0C02\0
 System board No SYSTEM 5.2.3790.0 10/1/2002 (Standard system
 devices) machine.inf Not Available ACPI\PNP0C01\1
 Motherboard resources No SYSTEM 5.2.3790.0 10/1/2002
 (Standard system devices) machine.inf Not Available
 ACPI\PNP0C02\5
 IBM Active PCI Device No SYSTEM 5.1.1.1 2/7/2003 IBM
 Corporation oem4.inf Not Available
 ACPI\IBM37D4\2&DABA3FF&0
 Memory Module No MEMORY5.2.3790.0 10/1/2002 Microsoft
 memory.inf Not Available ACPI\PNP0C80\80
 Memory Module No MEMORY5.2.3790.0 10/1/2002 Microsoft
 memory.inf Not Available ACPI\PNP0C80\81
 Memory Module No MEMORY5.2.3790.0 10/1/2002 Microsoft
 memory.inf Not Available ACPI\PNP0C80\82
 Memory Module No MEMORY5.2.3790.0 10/1/2002 Microsoft
 memory.inf Not Available ACPI\PNP0C80\83
 Memory Module No MEMORY5.2.3790.0 10/1/2002 Microsoft
 memory.inf Not Available ACPI\PNP0C80\84
 ACPI Fixed Feature Button No SYSTEM 5.2.3790.0 10/1/2002
 (Standard system devices) machine.inf Not Available
 ACPI\FIXEDBUTTON\2&DABA3FF&0
 Logical Disk ManagerNo SYSTEM 5.2.3790.0 10/1/2002 (Standard system
 devices) machine.inf Not Available ROOT\DMIO\0000
 Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&3735C57B&0&LDM#{E210CFFF-1FC1-4BEE-AEF8-F87
 4E4894F8A}
 Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&3735C57B&0&LDM#{08FF3127-F908-460B-AA6E-F569
 03B62C1F}
 Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&3735C57B&0&LDM#{363D5F85-EA5D-467C-9E95-9488
 A9276DF7}
 Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&3735C57B&0&LDM#{2CBE30AE-AD96-463F-A629-345
 AD643AB55}
 Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&3735C57B&0&LDM#{B39146D6-0455-4BCD-B874-5BD
 CD7FA02DA}
 Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&3735C57B&0&LDM#{44DAB6D9-B0EB-46A0-B00E-48
 0B8F6C4745}
 Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&3735C57B&0&LDM#{D5F128A7-64D9-4CCD-984A-526
 696167300}
 Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&3735C57B&0&LDM#{9A5B7BCD-F288-4A1D-B455-5B
 48194D5F6F}
 Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&3735C57B&0&LDM#{CD1E44E9-A170-444A-ACD4-DB
 64D41A583B}
 Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&3735C57B&0&LDM#{7915C416-393D-4863-A509-A3D
 AF902C15F}
 Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&3735C57B&0&LDM#{959B23C6-4BF2-4BDA-8AE4-4D
 259D4E0CAD}

Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{2B97D77B-8D62-4DF9-ACB8-326
C491A85B5}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{D90BFCFB-3A0B-45CB-8FBC-72
2B1544512E}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{7804C330-EF07-4D97-A5DF-EDC
CF4CFFB64}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{6D6F8550-48AC-4A08-8CBC-F30
6280544A9}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{79A60607-6DDE-4F77-9091-0673
83D6BD96}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{F8D140DB-AF23-40E9-A00A-59B
C07FD4D09}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{30A96961-3633-4214-B7CB-0CFB
46C1A3D9}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{F80C6007-670F-4C5A-888C-C72
AFA7EFFCE}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{E418AD3E-9CD8-434A-B05E-48E
AF64B8CC1}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{EF812FE6-2EF0-4436-9CC5-AEC
588E616E5}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{24A5D1EC-517B-4496-BDBC-314
64FBA2746}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{F244B0FD-4B66-4298-9D6F-1A8
8C72436B5}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{498F6E69-A92D-4DB5-9240-2017
2636E942}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{10D5E5E8-1130-46CA-A44C-341
E03B11CAD}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{E7EB69C6-CAF4-4645-A6AE-970
8D7CD0315}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{E4F6DA80-218C-462E-97A0-3AB
99E0D16BF}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{18007467-2449-4F21-890B-A6AA
F17EB19F}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{C8F0912F-6822-4B23-961C-0AD6
FA651D5A}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available

STORAGE\VOLUME\I&3735C57B&0&LDM#{878340F0-E08A-4C53-9AB2-7470
8A6A7FA3}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{3FE7F8D0-82C9-4B96-AF96-0FA
7C59CC2BE}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{FF598177-AF6D-4094-8C9C-D81
8319F5556}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{88D32A1C-C0A8-4296-85AB-004
510A58550}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{A48C2643-5BA6-4756-8DC1-5C4
D909AE833}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{6FEF15E6-D972-4102-A530-9927
BA6212CD}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{C2727958-5B68-42F9-A701-7532
E247CD2F}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{556A4076-A83A-4479-9A82-991F
9DA5AE6F}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{2C64A9E0-4F45-4E2C-B22B-F69
9AE634235}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{9A9B3916-EF32-434F-911E-C8FF
0316F2D5}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{26E3020C-4E0F-44EE-9202-F8BF
41B12602}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{85C6F726-68BA-4345-B983-7563
092B0ACF}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{FBA96BD5-8FDA-4ACC-AB1A-B
4D624964F8B}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{AC9CBF9A-F464-4691-8C96-326
5B5DE7DDB}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{DA2A3508-F92D-4C80-8BF0-13C
196DDE53E}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{8A6ADDFC-5728-4F3D-AF55-E3
3E04D223D1}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{DEA6E8D0-07EE-4982-8876-7D4
7DCACEEF9}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{C3380ED4-8416-40B7-813A-FBA
227D6D3F9}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{9D0AADF8-9FC6-4583-8CA8-E25
050CAF50C}

Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{4EA1E346-879D-4306-A652-14F3
BFE3EDAA}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{FB155953-8EAA-46D4-B3C7-BF
D4105E9BF4}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{121D8EA7-4888-4B25-9195-70E2
4CB5E6E3}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{43C42212-86C4-4485-96FF-AE92
2E8ADB65}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{4ECAE8DF-A4DE-433A-BECD-D
7B0A35DD98E}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{C5FAF247-054F-41A4-900C-FE7
A7A0000CE}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{EA0F6AD3-1E5B-442F-9D4A-03
E04C585EBA}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{A7B3A9C1-534E-43EF-8BE8-55D
AF60AC777}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{AC806270-AE7C-4DE8-9C86-F54
50ADE3160}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{6B2ECAA0-29CA-4ABD-8F92-B9
C8B5A440EB}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{794BD770-A97C-486E-9787-2A93
5FC00413}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{51539B6D-FD60-488B-B29F-3179
A817A4A6}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{E9E66AE4-5F38-4652-BE91-234C
D5B2519D}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{6FA3FAC0-B7E8-4FFC-AA46-BE
3F232C07CE}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{72B875AF-55F9-4166-BDFF-6CA
A78DDC226}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{1AA12EE0-66DE-440B-B130-943
C463F2843}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{63CE1EA4-6488-473E-90AD-D17
8EA33136F}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{E5C5044D-3DBD-4E5A-B52D-D2
CEBA08C4F0}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available

STORAGE\VOLUME\I&3735C57B&0&LDM#{85EB3B9D-04ED-493E-8CA6-57E
6EA83D7B1}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{AD597225-FADB-44BD-AC55-89
F84C8F085D}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{E9BACBF1-8DBE-4CDF-A95E-48
33565442A3}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{165A1BD9-1239-4047-8B19-85CF
D6083BB8}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{4B17EB8E-66F6-4D25-994B-1B4
13FE2C6DC}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{AEF2AEB0-EF16-46CC-91F2-8A2
B859FDA3F}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{14FE5947-BF27-4D39-AA1F-76A
C679BB3DB}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{207A9FFC-FF93-4E2E-A972-CDF
62BA337C7}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{0A10169D-4D37-46ED-A5CD-E6
63E6AA316F}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{49A24145-820F-40E7-B0D2-B73C
B60143D0}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{CD6A2FBE-1E67-4600-B5A5-56D
2951AAEDC}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{252D80B6-70B1-4B39-82DB-628
CBF63CBD7}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{934DD7F5-7FE5-4E6E-A97F-1BC
3EE6CA4B0}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{851A6C68-33AD-494E-A287-D94
005697D5F}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{4EB28BC2-750D-4FB0-BB12-EA
7863EDE998}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{96B5D908-95A3-4CAF-A6E9-9F5
5BBA7E40A}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{3E34FE7D-5B67-4E6E-A53D-4A7
E337EB62D}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{FEE333FC-8D79-4A81-9B35-0402
280188A1}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\I&3735C57B&0&LDM#{4DB00376-AB75-4C80-B8DC-B1
5E68F712C6}

Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{F884A177-BC4F-439B-9323-E953
C1DED65A}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{CF9B644A-E1C7-492D-8432-1E1
3527939E9}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{75EC7B5F-81F5-4870-8491-F835
CD788AB7}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{F94EE2AB-8F9D-46C0-B88D-4F9
4753226BC}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{4CA744BC-2CB4-4A52-9F5B-C0
6567D76E3F}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{D01B8BF2-3FF3-4B2C-91C6-31C
9C9EA1DED}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{81B772A3-09A8-4927-A65E-B1D
243B31B82}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{8BDE809D-1064-4466-9E41-077B
93A489D6}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{3EDBBE35-B3DA-499D-A4A1-20
DD5EB211E8}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{4DDA2092-ECC0-4126-B0AB-3F
0DD2E025BB}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{4FA396C3-E805-42D2-9043-EFF1
F414CCDB}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{D105D4CA-37CC-4F5B-B4C0-1F
7271A1BE2F}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{2EFA7295-8DDF-4CA5-B151-31D
8D6C495F5}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{1EE9E1F9-3B9B-4A1D-8D34-311
E18812F5A}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{0D60BA01-DA9A-41FA-A123-93
0C5648F821}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{82AC4D59-7DA9-4778-B946-4B7
AA65E768E}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{B6B3F732-19D9-4E81-BF8A-D45
158B878D1}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{E7BD0AA8-0797-41A7-B064-376
8BCE80050}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available

STORAGE\VOLUME\1&3735C57B&0&LDM#{BD42C2EE-B629-461A-82D4-478
3EDF8F263}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{906451E3-A5FB-46F7-B675-9C68
F698F3CD}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{AE5824B9-2D08-4421-B011-6E70
2F4203B8}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{AD82DC9E-2911-4582-A053-DC7
CAFF11C4F}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{D2A90E57-8ECD-424F-962E-37C
A416AE68F}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{A2E7402-9AB2-4DB4-9027-1B8
0B2D115D3}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{F589672A-201E-4F22-AE6A-4E5
DC952E278}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{31089042-EA12-4C1D-9665-6816
F7B53B2F}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{61C4D2BF-BBDD-442A-920F-80
D7E6F7B312}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{420A1F82-DC36-43EF-99D7-D94
A61E5861B}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{517202BC-72BF-42FA-92E8-F97
B1E0FD977}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{B3F7A36A-6876-4DBF-8190-774
F4839250C}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{57310A65-BD96-4245-BCBC-9EF
5E2B49109}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{C4481F1B-1CB8-4DE2-AF8E-B12
B5F2EAE8F}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{C542C55C-1ECF-441C-8270-32A
E2B1A584D}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{200F4D1E-1810-4AA5-927D-0EF
8AD5AC82D}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{782A9146-E9E4-4DCA-BF6A-7B
A3943B81A7}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{1D8E1FB8-A5F9-4F5F-B3C9-BC0
32CABDCE9}
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.infNot Available
STORAGE\VOLUME\1&3735C57B&0&LDM#{EE4AC8AE-F6D3-41BE-9DD3-8C
E4C4A08068}

63

Video Codecs	No	MEDIA	5.2.3790.0 10/1/2002 (Standard system devices)	wave.inf	Not Available	ROOT\MEDIA\MS_MMVID
WAN Miniport (L2TP)	No	NET	5.2.3790.0 10/1/2002	Microsoft netrasa.inf	Not Available	ROOT\MS_L2TPMINIPORT\0000
WAN Miniport (IP)	No	NET	5.2.3790.0 10/1/2002	Microsoft netrasa.inf	Not Available	ROOT\MS_NDISWANIP\0000
WAN Miniport (PPPOE)	No	NET	5.2.3790.0 10/1/2002	Microsoft netrasa.inf	Not Available	ROOT\MS_PPPOEMINIPORT\0000
WAN Miniport (PPTP)	No	NET	5.2.3790.0 10/1/2002	Microsoft netrasa.inf	Not Available	ROOT\MS_PPTPMINIPORT\0000
Direct Parallel	No	NET	5.2.3790.0 10/1/2002	Microsoft netrasa.inf	Not Available	ROOT\MS_PTMINIPORT\0000
Terminal Server Device Redirector	No	SYSTEM	5.2.3790.0 10/1/2002 (Standard system devices)	machine.inf	Not Available	ROOT\RDPDR\0000
Terminal Server Keyboard Driver	No	SYSTEM	5.2.3790.0 10/1/2002 (Standard system devices)	machine.inf	Not Available	ROOT\RDP_KBD\0000
Terminal Server Mouse Driver	No	SYSTEM	5.2.3790.0 10/1/2002 (Standard system devices)	machine.inf	Not Available	ROOT\RDP_MOU\0000
Plug and Play Software Device Enumerator	No	SYSTEM	5.2.3790.0 10/1/2002 (Standard system devices)	machine.inf	Not Available	ROOT\SYSTEM\0000
Microcode Update Device	No	SYSTEM	5.2.3790.0 10/1/2002 (Standard system devices)	machine.inf	Not Available	ROOT\SYSTEM\0001

[Environment Variables]

Variable	Value	User Name
----------	-------	-----------

CLASSPATH
.;D:\SQLLIB\java\db2java.zip;D:\SQLLIB\java\db2jcc.jar;D:\SQLLIB\java\sqlj.zip;D:\SQLLIB\java\db2jcc_license_cisuz.jar;D:\SQLLIB\java\db2jcc_license_cu.jar;D:\SQLLIB\bin;D:\SQLLIB\java\common.jar <SYSTEM>

ClusterLog	C:\WINDOWS\Cluster\cluster.log	<SYSTEM>
------------	--------------------------------	----------

ComSpec	%SystemRoot%\system32\cmd.exe	<SYSTEM>
---------	-------------------------------	----------

DB2INSTANCE	tpch	<SYSTEM>
-------------	------	----------

DB2TEMPDIR	D:\SQLLIB\	<SYSTEM>
------------	------------	----------

INCLUDE D:\SQLLIB\INCLUDE;D:\SQLLIB\LIB;C:\Program Files\Microsoft Visual Studio .NET 2003\Vc7\include;C:\Program Files\Microsoft Visual Studio .NET 2003\Vc7\PlatformSDK\Include <SYSTEM>

LIB D:\SQLLIB\LIB;C:\Program Files\Microsoft Visual Studio .NET 2003\Vc7\lib;C:\Program Files\Microsoft Visual Studio .NET 2003\Vc7\PlatformSDK\Lib <SYSTEM>

NUMBER_OF_PROCESSORS	8	<SYSTEM>
----------------------	---	----------

OS	Windows_NT	<SYSTEM>
----	------------	----------

Path
%SystemRoot%\system32;%SystemRoot%;%SystemRoot%\System32\Wbem;D:\SQLLIB\BIN;D:\SQLLIB\FUNCTION;D:\SQLLIB\SAMPLES\REPL;C:\Program

Files\Microsoft Visual Studio .NET 2003\Vc7\bin;C:\Program Files\Microsoft Visual Studio .NET 2003\Common7\IDE;D:\tools;D:\nttools <SYSTEM>

PATHEXT .COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH <SYSTEM>

PROCESSOR_ARCHITECTURE	x86	<SYSTEM>
------------------------	-----	----------

PROCESSOR_IDENTIFIER	x86 Family 15 Model 2 Stepping 6, GenuineIntel	<SYSTEM>
----------------------	--	----------

PROCESSOR_LEVEL	15	<SYSTEM>
-----------------	----	----------

PROCESSOR_REVISION	0206	<SYSTEM>
--------------------	------	----------

RAH_DO_ALL_LOGICAL_NODES	TRUE	<SYSTEM>
--------------------------	------	----------

RAHSLEEPTIME	999999	<SYSTEM>
--------------	--------	----------

TEMP	%SystemRoot%\TEMP	<SYSTEM>
------	-------------------	----------

TMP	%SystemRoot%\TEMP	<SYSTEM>
-----	-------------------	----------

VS71COMNTOOLS C:\Program Files\Microsoft Visual Studio .NET 2003\Common7\Tools\ <SYSTEM>

windir	%SystemRoot%	<SYSTEM>
--------	--------------	----------

TEMP	%USERPROFILE%\Local Settings\Temp	NT AUTHORITY\SYSTEM
------	-----------------------------------	---------------------

TMP	%USERPROFILE%\Local Settings\Temp	NT AUTHORITY\SYSTEM
-----	-----------------------------------	---------------------

TEMP	%USERPROFILE%\Local Settings\Temp	NT AUTHORITY\LOCAL SERVICE
------	-----------------------------------	----------------------------

TMP	%USERPROFILE%\Local Settings\Temp	NT AUTHORITY\LOCAL SERVICE
-----	-----------------------------------	----------------------------

TEMP	%USERPROFILE%\Local Settings\Temp	NT AUTHORITY\NETWORK SERVICE
------	-----------------------------------	------------------------------

TMP	%USERPROFILE%\Local Settings\Temp	NT AUTHORITY\NETWORK SERVICE
-----	-----------------------------------	------------------------------

TEMP	%USERPROFILE%\Local Settings\Temp	RELENTLESS1\tpch
------	-----------------------------------	------------------

TMP	%USERPROFILE%\Local Settings\Temp	RELENTLESS1\tpch
-----	-----------------------------------	------------------

[Print Jobs]

Document Size	Owner	Notify	Status	Time Submitted	Start
Time	Until	Time Elapsed	Pages Printed	Job ID	Priority
Parameters	Driver	Print Processor	Host	Print Queue	Data Type
					Name

[Network Connections]

Local Name	Remote Name	Type	Status	User Name
------------	-------------	------	--------	-----------

[Running Tasks]

Name	Path	Process ID	Priority	Min Working Set	Max Working Set
Start Time	Version	Size	File Date		

system idle process	Not Available	0	0	Not Available	
Not Available	Not Available	Not Available	Not Available	Not Available	
system	Not Available	4	8	0	1413120 Not Available
Available	Not Available	Not Available	Not Available	Not Available	
smss.exe	Not Available	380	11	204800	1413120
5/12/2004 7:24 AM	Not Available	Not Available	Not Available	Not Available	
csrss.exe	Not Available	580	13	Not Available	Not Available
Available	5/12/2004 7:24 AM	Not Available	Not Available	Not Available	Not Available
winlogon.exe	c:\windows\system32\winlogon.exe	604	13		
204800	1413120	5/12/2004 7:24 AM	5.2.3790.0 (srv03_rtm.030324-2048)		
536.50 KB (549,376 bytes)	3/25/2003 7:00 AM				
services.exe	c:\windows\system32\services.exe	648	9		
204800	1413120	5/12/2004 7:24 AM	5.2.3790.0 (srv03_rtm.030324-2048)		
102.00 KB (104,448 bytes)	3/25/2003 7:00 AM				
lsass.exe	c:\windows\system32\lsass.exe	684	9	204800	
1413120	5/12/2004 7:24 AM	5.2.3790.0 (srv03_rtm.030324-2048)		13.00	
KB (13,312 bytes)	3/25/2003 7:00 AM				
svchost.exe	c:\windows\system32\svchost.exe	828	8		
204800	1413120	5/12/2004 7:24 AM	5.2.3790.0 (srv03_rtm.030324-2048)		
13.00 KB (13,312 bytes)	3/25/2003 7:00 AM				
svchost.exe	c:\windows\system32\svchost.exe	884	8		
204800	1413120	5/12/2004 7:24 AM	5.2.3790.0 (srv03_rtm.030324-2048)		
13.00 KB (13,312 bytes)	3/25/2003 7:00 AM				
svchost.exe	Not Available	1012	8	Not Available	
Not Available	5/12/2004 7:24 AM	Not Available	Not Available	Not Available	
svchost.exe	Not Available	1040	8	Not Available	
Not Available	5/12/2004 7:24 AM	Not Available	Not Available	Not Available	
svchost.exe	c:\windows\system32\svchost.exe	1072	8		
204800	1413120	5/12/2004 7:24 AM	5.2.3790.0 (srv03_rtm.030324-2048)		
13.00 KB (13,312 bytes)	3/25/2003 7:00 AM				
msdtc.exe	Not Available	1168	8	Not Available	Not Available
Available	5/12/2004 7:24 AM	Not Available	Not Available	Not Available	Not Available
db2licd.exe	d:\sql\lib\bin\db2licd.exe	1388	8	204800	
1413120	5/12/2004 7:24 AM	8.1.5.449	84.06 KB (86,082 bytes)		
5/4/2004 3:53 PM					
db2sec.exe	d:\sql\lib\bin\db2sec.exe	1404	8	204800	
1413120	5/12/2004 7:24 AM	8.1.5.449	24.06 KB (24,641 bytes)		
5/4/2004 4:01 PM					
db2rcmd.exe	Not Available	1416	8	Not Available	
Not Available	5/12/2004 7:24 AM	Not Available	Not Available	Not Available	
ibmhpasv.exe	c:\windows\system32\ibmhpasv.exe	1448	8		
204800	1413120	5/12/2004 7:24 AM	5.1.1.1	14.50 KB (14,848 bytes)	
2/7/2003 9:55 PM					
mdm.exe	c:\program files\common files\microsoft shared\vs7debug\mdm.exe	1472	8	204800	
1413120	5/12/2004 7:24 AM	7.10.3077	328.00 KB (335,872 bytes)		
3/19/2003 4:55 AM					
db2jds.exe	d:\sql\lib\bin\db2jds.exe	1688	8	204800	
1413120	5/12/2004 7:24 AM	8.1.5.449	184.07 KB (188,488 bytes)		
5/4/2004 3:53 PM					

dfssvc.exe	c:\windows\system32\dfssvc.exe	1704	8	204800	
1413120	5/12/2004 7:24 AM	5.2.3790.0 (srv03_rtm.030324-2048)		130.50	
KB (133,632 bytes)	3/25/2003 7:00 AM				
wmiprvse.exe	Not Available	1836	8	Not Available	
Not Available	5/12/2004 7:25 AM	Not Available	Not Available	Not Available	
csrss.exe	Not Available	2360	13	Not Available	Not Available
Available	5/12/2004 10:23 AM	Not Available	Not Available	Not Available	Not Available
winlogon.exe	c:\windows\system32\winlogon.exe	2384	13		
204800	1413120	5/12/2004 10:23 AM	5.2.3790.0 (srv03_rtm.030324-2048)		
536.50 KB (549,376 bytes)	3/25/2003 7:00 AM				
rdpclip.exe	c:\windows\system32\rdpclip.exe	2552	8	204800	
1413120	5/12/2004 10:23 AM	5.2.3790.0 (srv03_rtm.030324-2048)		53.00	
KB (54,272 bytes)	4/1/2004 2:30 PM				
explorer.exe	c:\windows\explorer.exe	2644	8	204800	
1413120	5/12/2004 10:23 AM	6.00.3790.0 (srv03_rtm.030324-2048)			
1,008.50 KB (1,032,704 bytes)	3/25/2003 7:00 AM				
cmd.exe	c:\windows\system32\cmd.exe	2720	8	204800	
1413120	5/12/2004 10:23 AM	5.2.3790.0 (srv03_rtm.030324-2048)		374.00	
KB (382,976 bytes)	3/25/2003 7:00 AM				
cmd.exe	c:\windows\system32\cmd.exe	2840	8	204800	
1413120	5/12/2004 10:27 AM	5.2.3790.0 (srv03_rtm.030324-2048)		374.00	
KB (382,976 bytes)	3/25/2003 7:00 AM				
explorer.exe	c:\windows\explorer.exe	3056	8	204800	
1413120	5/12/2004 10:54 AM	6.00.3790.0 (srv03_rtm.030324-2048)			
1,008.50 KB (1,032,704 bytes)	3/25/2003 7:00 AM				
helpctr.exe	c:\windows\pchealth\helpctr\binaries\helpctr.exe	3404	8		
204800	1413120	5/12/2004 11:27 AM	5.2.3790.0 (srv03_rtm.030324-2048)		
764.00 KB (782,336 bytes)	4/1/2004 2:34 PM				
wmiprvse.exe	Not Available	3496	8	Not Available	
Not Available	5/12/2004 11:27 AM	Not Available	Not Available	Not Available	
helpsvc.exe	c:\windows\pchealth\helpctr\binaries\helpsvc.exe	3504	8		
204800	1413120	5/12/2004 11:27 AM	5.2.3790.0 (srv03_rtm.030324-2048)		
720.00 KB (737,280 bytes)	4/1/2004 2:34 PM				
svchost.exe	c:\windows\system32\svchost.exe	3752	8		
204800	1413120	5/12/2004 11:31 AM	5.2.3790.0 (srv03_rtm.030324-2048)		
13.00 KB (13,312 bytes)	3/25/2003 7:00 AM				
[Loaded Modules]					
Name	Version	Size	File Date	Manufacturer	Path
winlogon	5.2.3790.0 (srv03_rtm.030324-2048)	536.50 KB (549,376 bytes)			
3/25/2003 7:00 AM	Microsoft Corporation				
c:\windows\system32\winlogon.exe					
ntdll	5.2.3790.0 (srv03_rtm.030324-2048)	722.50 KB (739,840 bytes)			
3/25/2003 7:00 AM	Microsoft Corporation				c:\windows\system32\ntdll.dll
kernel32	5.2.3790.0 (srv03_rtm.030324-2048)	965.00 KB (988,160 bytes)			
3/25/2003 7:00 AM	Microsoft Corporation				
c:\windows\system32\kernel32.dll					

msvcrt	7.0.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\msvcrt.dll	319.50 KB (327,168 bytes)	setupapi	5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\setupapi.dll	1,014.50 KB (1,038,848 bytes)
advapi32	5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\advapi32.dll	559.50 KB (572,928 bytes)	msgina	5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\msgina.dll	1.14 MB (1,191,936 bytes)
rpcrt4	5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rpcrt4.dll	643.50 KB (658,944 bytes)	shsvcs	6.00.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\shsvcs.dll	121.50 KB (124,416 bytes)
user32	5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\user32.dll	562.00 KB (575,488 bytes)	shlwapi	6.00.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\shlwapi.dll	281.00 KB (287,744 bytes)
gdi32	5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\gdi32.dll	263.00 KB (269,312 bytes)	sfc	5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\sfc.dll	4.50 KB (4,608 bytes)
userenv	5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\userenv.dll	732.50 KB (750,080 bytes)	sfc_os	5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\sfc_os.dll	133.00 KB (136,192 bytes)
nddeapi	5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\nddeapi.dll	16.00 KB (16,384 bytes)	wintrust	5.131.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wintrust.dll	161.50 KB (165,376 bytes)
crypt32	5.131.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\crypt32.dll	598.00 KB (612,352 bytes)	ole32	5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ole32.dll	1.13 MB (1,187,328 bytes)
msasn1	5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\msasn1.dll	58.00 KB (59,392 bytes)	imagehlp	5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\imagehlp.dll	142.50 KB (145,920 bytes)
secur32	5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\secur32.dll	63.00 KB (64,512 bytes)	comctl32	6.0 (srv03_rtm.030324-2048) 4/1/2004 8:59 AM Microsoft Corporation c:\windows\winsxs\x86_microsoft.windows.common-controls_6595b64144ccf1df_6.0.100.0_x-ww_8417450b\comctl32.dll	907.00 KB (928,768 bytes)
winsta	5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\winsta.dll	51.00 KB (52,224 bytes)	winscard	5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\winscard.dll	98.50 KB (100,864 bytes)
netapi32	5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\netapi32.dll	317.00 KB (324,608 bytes)	wsapi32	5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wsapi32.dll	17.50 KB (17,920 bytes)
profmap	5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\profmap.dll	22.00 KB (22,528 bytes)	sxs	5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\sxs.dll	733.00 KB (750,592 bytes)
regapi	5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\regapi.dll	48.50 KB (49,664 bytes)	winmm	5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\winmm.dll	166.00 KB (169,984 bytes)
ws2_32	5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ws2_32.dll	87.50 KB (89,600 bytes)	shell32	6.00.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\shell32.dll	7.79 MB (8,166,400 bytes)
ws2help	5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ws2help.dll	19.50 KB (19,968 bytes)	wldap32	5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wldap32.dll	158.00 KB (161,792 bytes)
psapi	5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\psapi.dll	21.50 KB (22,016 bytes)	rsaenh	5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rsaenh.dll	176.83 KB (181,072 bytes)
version	5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\version.dll	17.00 KB (17,408 bytes)	csdll	5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\csdll.dll	99.00 KB (101,376 bytes)
			wlnotify	5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wlnotify.dll	87.50 KB (89,600 bytes)

winspool 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\winspool.drv	140.00 KB (143,360 bytes)	netui1 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\netui1.dll	184.00 KB (188,416 bytes)
mpr 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\mpr.dll	56.00 KB (57,344 bytes)	davclnt 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\davclnt.dll	23.50 KB (24,064 bytes)
comctl32 5.82 (srv03_rtm.030324-2048) 4/1/2004 8:59 AM Microsoft Corporation c:\windows\winsxs\x86_microsoft.windows.common-controls_6595b64144ccf1df_5.8.2.0.0_x-ww_8a69ba05\comctl32.dll	561.00 KB (574,464 bytes)	mprui 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\mprui.dll	49.00 KB (50,176 bytes)
uxtheme 6.00.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\uxtheme.dll	196.00 KB (200,704 bytes)	netui2 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\netui2.dll	309.50 KB (316,928 bytes)
clbcatq 2001.12.4720.0 (srv03_rtm.030324-2048) 4/1/2004 2:30 PM Microsoft Corporation c:\windows\system32\clbcatq.dll	481.00 KB (492,544 bytes)	comdlg32 6.00.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\comdlg32.dll	261.00 KB (267,264 bytes)
oleaut32 5.2.3790.0 486.00 KB (497,664 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\oleaut32.dll	3/25/2003 7:00 AM	netmsg 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\netmsg.dll	178.00 KB (182,272 bytes)
comres 2001.12.4720.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\comres.dll	778.00 KB (796,672 bytes)	es 2001.12.4720.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\es.dll	221.50 KB (226,816 bytes)
wbemprox 5.2.3790.0 (srv03_rtm.030324-2048) 4/1/2004 2:30 PM Microsoft Corporation c:\windows\system32\wbem\wbemprox.dll	17.50 KB (17,920 bytes)	ntmarta 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntmarta.dll	114.00 KB (116,736 bytes)
wbemcomn 5.2.3790.0 (srv03_rtm.030324-2048) (216,576 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wbem\wbemcomn.dll	211.50 KB	services 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\services.exe	102.00 KB (104,448 bytes)
wbemsvc 5.2.3790.0 (srv03_rtm.030324-2048) 4/1/2004 2:30 PM Microsoft Corporation c:\windows\system32\wbem\wbemsvc.dll	42.50 KB (43,520 bytes)	scesrv 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\scesrv.dll	316.50 KB (324,096 bytes)
fastprox 5.2.3790.0 (srv03_rtm.030324-2048) 4/1/2004 2:30 PM Microsoft Corporation c:\windows\system32\wbem\fastprox.dll	443.00 KB (453,632 bytes)	authz 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\authz.dll	67.00 KB (68,608 bytes)
msvcp60 6.05.2144.0 388.00 KB (397,312 bytes) AM Microsoft Corporation c:\windows\system32\msvcp60.dll	3/25/2003 7:00	umpnprmgr 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\umpnprmgr.dll	121.50 KB (124,416 bytes)
ntdsapi 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntdsapi.dll	76.00 KB (77,824 bytes)	ncobjapi 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ncobjapi.dll	34.50 KB (35,328 bytes)
dnsapi 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dnsapi.dll	147.50 KB (151,040 bytes)	eventlog 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\eventlog.dll	60.50 KB (61,952 bytes)
samlib 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\samlib.dll	49.00 KB (50,176 bytes)	lsass 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\lsass.exe	13.00 KB (13,312 bytes)
cscui 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\cscui.dll	305.00 KB (312,320 bytes)	lsasrv 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\lsasrv.dll	780.50 KB (799,232 bytes)
drprov 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\drprov.dll	12.50 KB (12,800 bytes)	samsrv 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\samsrv.dll	452.00 KB (462,848 bytes)
ntlanman 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntlanman.dll	41.00 KB (41,984 bytes)	cryptdll 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\cryptdll.dll	34.00 KB (34,816 bytes)
netui0 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\netui0.dll	75.50 KB (77,312 bytes)	msprivs 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\msprivs.dll	46.50 KB (47,616 bytes)

kerberos 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\kerberos.dll	332.50 KB (340,480 bytes)	pstorsvc 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\pstorsvc.dll	24.00 KB (24,576 bytes)
msv1_0 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\msv1_0.dll	127.00 KB (130,048 bytes)	psbase 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\psbase.dll	81.00 KB (82,944 bytes)
netlogon 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\netlogon.dll	409.00 KB (418,816 bytes)	dssenh 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dssenh.dll	131.33 KB (134,480 bytes)
w32time 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\w32time.dll	216.00 KB (221,184 bytes)	wlbcctrl 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wlbcctrl.dll	78.00 KB (79,872 bytes)
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll	82.50 KB (84,480 bytes)	svchost 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\svchost.exe	13.00 KB (13,312 bytes)
schannel 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll	149.50 KB (153,088 bytes)	rpss 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rpss.dll	276.50 KB (283,136 bytes)
wdigest 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wdigest.dll	61.00 KB (62,464 bytes)	termsrv 5.2.3790.0 (srv03_rtm.030324-2048) 4/1/2004 2:31 PM Microsoft Corporation c:\windows\system32\termsrv.dll	216.50 KB (221,696 bytes)
rassfm 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rassfm.dll	20.50 KB (20,992 bytes)	icaapi 5.2.3790.0 (srv03_rtm.030324-2048) 4/1/2004 2:30 PM Microsoft Corporation c:\windows\system32\icaapi.dll	10.50 KB (10,752 bytes)
kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\kdcsvc.dll	221.00 KB (226,304 bytes)	mstlsapi 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\mstlsapi.dll	104.50 KB (107,008 bytes)
ntdsa 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntdsa.dll	1.45 MB (1,520,640 bytes)	activeds 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\activeds.dll	189.00 KB (193,536 bytes)
ntdsatq 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntdsatq.dll	32.00 KB (32,768 bytes)	adslsdp 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\adslsdp.dll	142.50 KB (145,920 bytes)
mswsock 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\mswsock.dll	254.00 KB (260,096 bytes)	credui 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\credui.dll	159.00 KB (162,816 bytes)
esent 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\esent.dll	1.01 MB (1,056,256 bytes)	atl 3.05.2283 83.00 KB (84,992 bytes) Microsoft Corporation c:\windows\system32\atl.dll	3/25/2003 7:00 AM
scecli 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\scecli.dll	179.50 KB (183,808 bytes)	rdpwsx 5.2.3790.0 (srv03_rtm.030324-2048) 4/1/2004 2:30 PM Microsoft Corporation c:\windows\system32\rdpwsx.dll	80.13 KB (82,056 bytes)
wshtcpip 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wshtcpip.dll	18.00 KB (18,432 bytes)	schedsvc 5.2.3790.0 (srv03_rtm.030324-2048) 4/1/2004 2:34 PM Microsoft Corporation c:\windows\system32\schedsvc.dll	176.00 KB (180,224 bytes)
ipsecsvc 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ipsecsvc.dll	162.50 KB (166,400 bytes)	msidle 6.00.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\msidle.dll	5.50 KB (5,632 bytes)
oakley 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\oakley.dll	325.50 KB (333,312 bytes)	wkssvc 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wkssvc.dll	125.00 KB (128,000 bytes)
winipsec 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\winipsec.dll	34.50 KB (35,328 bytes)	wiarpc 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wiarpc.dll	30.00 KB (30,720 bytes)
		dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll	24.00 KB (24,576 bytes)

srvsvc 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\srvsvc.dll	89.00 KB (91,136 bytes)	tapi32 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\tapi32.dll	175.00 KB (179,200 bytes)
seclogon 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\seclogon.dll	16.50 KB (16,896 bytes)	wcserv 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:15 AM Microsoft Corporation c:\windows\system32\wcserv.dll	272.50 KB (279,040 bytes)
sens 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation	35.50 KB (36,352 bytes) c:\windows\system32\sens.dll	wmi 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation	6.50 KB (6,656 bytes) c:\windows\system32\wmi.dll
trkws 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\trkws.dll	85.00 KB (87,040 bytes)	dhcpcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dhcpcsvc.dll	101.50 KB (103,936 bytes)
wmisvc 5.2.3790.0 (srv03_rtm.030324-2048) 4/1/2004 2:30 PM Microsoft Corporation c:\windows\system32\wbem\wmisvc.dll	131.00 KB (134,144 bytes)	wcsapi 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:15 AM Microsoft Corporation c:\windows\system32\wcsapi.dll	24.50 KB (25,088 bytes)
vssapi 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\vssapi.dll	528.00 KB (540,672 bytes)	netshell 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\netshell.dll	1.67 MB (1,747,456 bytes)
comsvcs 2001.12.4720.0 (srv03_rtm.030324-2048) 4/1/2004 2:30 PM Microsoft Corporation c:\windows\system32\comsvcs.dll	1.14 MB (1,199,616 bytes)	clusapi 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\clusapi.dll	56.00 KB (57,344 bytes)
wbemcore 5.2.3790.0 (srv03_rtm.030324-2048) 4/1/2004 2:30 PM Microsoft Corporation c:\windows\system32\wbem\wbemcore.dll	457.00 KB (467,968 bytes)	netcfgx 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\netcfgx.dll	726.00 KB (743,424 bytes)
esscli 5.2.3790.0 (srv03_rtm.030324-2048) 4/1/2004 2:30 PM Microsoft Corporation c:\windows\system32\wbem\esscli.dll	235.50 KB (241,152 bytes)	hnetcfg 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\hnetcfg.dll	243.50 KB (249,344 bytes)
wmiutils 5.2.3790.0 (srv03_rtm.030324-2048) 4/1/2004 2:30 PM Microsoft Corporation c:\windows\system32\wbem\wmiutils.dll	90.50 KB (92,672 bytes)	wininet 6.00.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wininet.dll	609.00 KB (623,616 bytes)
repdrvfs 5.2.3790.0 (srv03_rtm.030324-2048) 4/1/2004 2:30 PM Microsoft Corporation c:\windows\system32\wbem\repdrvfs.dll	165.00 KB (168,960 bytes)	rasmans 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rasmans.dll	163.50 KB (167,424 bytes)
wmiprvsd 5.2.3790.0 (srv03_rtm.030324-2048) 4/1/2004 2:30 PM Microsoft Corporation c:\windows\system32\wbem\wmiprvsd.dll	405.50 KB (415,232 bytes)	rasdlg 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rasdlg.dll	642.00 KB (657,408 bytes)
wbemess 5.2.3790.0 (srv03_rtm.030324-2048) 4/1/2004 2:30 PM Microsoft Corporation c:\windows\system32\wbem\wbemess.dll	256.50 KB (262,656 bytes)	rasadhlp 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rasadhlp.dll	6.50 KB (6,656 bytes)
ncprov 5.2.3790.0 (srv03_rtm.030324-2048) 4/1/2004 2:30 PM Microsoft Corporation c:\windows\system32\wbem\ncprov.dll	43.00 KB (44,032 bytes)	pchsvc 5.2.3790.0 (srv03_rtm.030324-2048) 4/1/2004 2:34 PM Microsoft Corporation c:\windows\pchealth\helpctr\binaries\pchsvc.dll	31.50 KB (32,256 bytes)
netman 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\netman.dll	209.00 KB (214,016 bytes)	wbemcons 5.2.3790.0 (srv03_rtm.030324-2048) 4/1/2004 2:30 PM Microsoft Corporation c:\windows\system32\wbem\wbemcons.dll	69.00 KB (70,656 bytes)
mprapi 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\mprapi.dll	81.00 KB (82,944 bytes)	rastapi 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rastapi.dll	57.00 KB (58,368 bytes)
rtutils 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rtutils.dll	32.00 KB (32,768 bytes)	rasppp 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rasppp.dll	195.00 KB (199,680 bytes)
rasapi32 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rasapi32.dll	227.50 KB (232,960 bytes)	ntlsapi 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntlsapi.dll	8.00 KB (8,192 bytes)
rasman 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rasman.dll	56.50 KB (57,856 bytes)		

raschap	5.2.3790.0 (srv03_rtm.030324-2048)	106.00 KB (108,544 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\raschap.dll	msdbg2	7.10.3077	188.00 KB (192,512 bytes)	3/19/2003 4:58 AM	Microsoft Corporation	c:\program files\common files\microsoft shared\vs7debug\msdbg2.dll
rastls	5.2.3790.0 (srv03_rtm.030324-2048)	155.00 KB (158,720 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\rastls.dll	db2jds	8.1.5.449	184.07 KB (188,488 bytes)	5/4/2004 3:53 PM	International Business Machines Corporation	d:\sqllib\bin\db2jds.exe
cryptui	5.131.3790.0 (srv03_rtm.030324-2048)	473.50 KB (484,864 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\cryptui.dll	winnr	5.2.3790.0 (srv03_rtm.030324-2048)	15.00 KB (15,360 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\winnr.dll
ipbootp	5.2.3790.0 (srv03_rtm.030324-2048)	34.50 KB (35,328 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\ipbootp.dll	dfssvc	5.2.3790.0 (srv03_rtm.030324-2048)	130.50 KB (133,632 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\dfssvc.exe
db2licd	8.1.5.449	84.06 KB (86,082 bytes)	5/4/2004 3:53 PM	International Business Machines Corporation	d:\sqllib\bin\db2licd.exe	resutils	5.2.3790.0 (srv03_rtm.030324-2048)	59.00 KB (60,416 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\resutils.dll
db2sys	8.1.5.449	2.38 MB (2,490,433 bytes)	5/4/2004 4:04 PM	International Business Machines Corporation	d:\sqllib\bin\db2sys.dll	mfc42u	6.05.3014.0	960.00 KB (983,040 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\mfc42u.dll
db2wint	8.1.5.449	48.06 KB (49,218 bytes)	5/4/2004 4:09 PM	International Business Machines Corporation	d:\sqllib\bin\db2wint.dll	wsock32	5.2.3790.0 (srv03_rtm.030324-2048)	22.00 KB (22,528 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\wsock32.dll
db2sysp	8.1.5.449	88.06 KB (90,178 bytes)	5/4/2004 4:04 PM	International Business Machines Corporation	d:\sqllib\bin\db2sysp.dll	rdpsnd	5.2.3790.0 (srv03_rtm.030324-2048)	18.00 KB (18,432 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\rdpsnd.dll
db2app	8.1.5.449	2.49 MB (2,613,313 bytes)	5/4/2004 3:30 PM	International Business Machines Corporation	d:\sqllib\bin\db2app.dll	scredir	5.2.3790.0 (srv03_rtm.030324-2048)	27.00 KB (27,648 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\scredir.dll
db2util	8.1.5.449	1.18 MB (1,237,058 bytes)	5/4/2004 4:07 PM	International Business Machines Corporation	d:\sqllib\bin\db2util.dll	msacm32	5.2.3790.0 (srv03_rtm.030324-2048)	21.00 KB (21,504 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\msacm32.drv
db2abind	8.1.5.449	244.07 KB (249,923 bytes)	5/4/2004 3:29 PM	International Business Machines Corporation	d:\sqllib\bin\db2abind.dll	msacm32	5.2.3790.0 (srv03_rtm.030324-2048)	67.50 KB (69,120 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\msacm32.dll
db2g11n	8.1.5.449	364.06 KB (372,802 bytes)	5/4/2004 3:50 PM	International Business Machines Corporation	d:\sqllib\bin\db2g11n.dll	imaadp32	5.2.3790.0 (srv03_rtm.030324-2048)	15.50 KB (15,872 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\imaadp32.acm
db2osse	8.1.5.449	312.07 KB (319,563 bytes)	5/4/2004 3:57 PM	International Business Machines Corporation	d:\sqllib\bin\db2osse.dll	msadp32	5.2.3790.0 (srv03_rtm.030324-2048)	14.50 KB (14,848 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\msadp32.acm
db2locale	8.1.5.449	48.07 KB (49,220 bytes)	5/4/2004 3:54 PM	International Business Machines Corporation	d:\sqllib\bin\db2locale.dll	msg711	5.2.3790.0 (srv03_rtm.030324-2048)	10.00 KB (10,240 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\msg711.acm
db2cli	8.1.5.449	2.68 MB (2,805,825 bytes)	5/4/2004 3:34 PM	International Business Machines Corporation	d:\sqllib\bin\db2cli.dll	msgsm32	5.2.3790.0 (srv03_rtm.030324-2048)	20.50 KB (20,992 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\msgsm32.acm
db2install	8.1.5.449	28.07 KB (28,741 bytes)	5/4/2004 3:52 PM	International Business Machines Corporation	d:\sqllib\bin\db2install.dll	tssoft32	1.01	9.50 KB (9,728 bytes)	3/25/2003 7:00 AM	DSP GROUP, INC.	c:\windows\system32\tssoft32.acm
db2trcapi	8.1.5.449	36.08 KB (36,941 bytes)	5/4/2004 4:06 PM	International Business Machines Corporation	d:\sqllib\bin\db2trcapi.dll	tsd32	1.03	16.50 KB (16,896 bytes)	3/25/2003 7:00 AM	DSP GROUP, INC.	c:\windows\system32\tsd32.dll
db2sec	8.1.5.449	24.06 KB (24,641 bytes)	5/4/2004 4:01 PM	International Business Machines Corporation	d:\sqllib\bin\db2sec.exe	msg723	4.4.4000	116.00 KB (118,784 bytes)	4/1/2004 2:34 PM	Microsoft Corporation	c:\windows\system32\msg723.acm
ibmhpasv	5.1.1.1	14.50 KB (14,848 bytes)	2/7/2003 9:55 PM	IBM Corporation	c:\windows\system32\ibmhpasv.exe	msaud32	8.00.00.4487	288.00 KB (294,912 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\msaud32.acm
mdm	7.10.3077	328.00 KB (335,872 bytes)	3/19/2003 4:55 AM	Microsoft Corporation	c:\program files\common files\microsoft shared\vs7debug\mdm.exe	sl_anet	3.02	84.00 KB (86,016 bytes)	3/25/2003 7:00 AM	Sipro Lab Telecom Inc.	c:\windows\system32\sl_anet.acm
csm	7.10.3077	56.00 KB (57,344 bytes)	3/19/2003 4:56 AM	Microsoft Corporation	c:\program files\common files\microsoft shared\vs7debug\csm.dll						
msvcr71	7.10.3052.4	340.00 KB (348,160 bytes)	2/21/2003 7:42 AM	Microsoft Corporation	c:\windows\system32\msvcr71.dll						

l3codeca	1, 9, 0, 0305	284.00 KB (290,816 bytes)	3/25/2003 7:00 AM
Fraunhofer Institut Integrierte Schaltungen IIS			
c:\windows\system32\l3codeca.acm			
rdpclip	5.2.3790.0 (srv03_rtm.030324-2048)	53.00 KB (54,272 bytes)	4/1/2004 2:30 PM
Microsoft Corporation			
c:\windows\system32\rdpclip.exe			
explorer	6.00.3790.0 (srv03_rtm.030324-2048)	1,008.50 KB (1,032,704 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\explorer.exe			
browseui	6.00.3790.0 (srv03_rtm.030324-2048)	1.01 MB (1,057,280 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\browseui.dll			
shdocvw	6.00.3790.0 (srv03_rtm.030324-2048)	1.33 MB (1,393,664 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\shdocvw.dll			
apphelp	5.2.3790.0 (srv03_rtm.030324-2048)	122.00 KB (124,928 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\apphelp.dll			
themeui	6.00.3790.0 (srv03_rtm.030324-2048)	360.50 KB (369,152 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\themeui.dll			
msimg32	5.2.3790.0 (srv03_rtm.030324-2048)	4.50 KB (4,608 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\msimg32.dll			
actxprxy	6.00.3790.0 (srv03_rtm.030324-2048)	95.00 KB (97,280 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\actxprxy.dll			
linkinfo	5.2.3790.0 (srv03_rtm.030324-2048)	16.50 KB (16,896 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\linkinfo.dll			
ntshrui	6.00.3790.0 (srv03_rtm.030324-2048)	136.00 KB (139,264 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\ntshrui.dll			
urlmon	6.00.3790.0 (srv03_rtm.030324-2048)	501.50 KB (513,536 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\urlmon.dll			
webcheck	6.00.3790.0 (srv03_rtm.030324-2048)	261.50 KB (267,776 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\webcheck.dll			
stobject	5.2.3790.0 (srv03_rtm.030324-2048)	117.50 KB (120,320 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\stobject.dll			
batmeter	6.00.3790.0 (srv03_rtm.030324-2048)	28.50 KB (29,184 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\batmeter.dll			
powrprof	6.00.3790.0 (srv03_rtm.030324-2048)	14.50 KB (14,848 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\powrprof.dll			
printui	5.2.3790.0 (srv03_rtm.030324-2048)	536.50 KB (549,376 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\printui.dll			
cfgmgr32	5.2.3790.0 (srv03_rtm.030324-2048)	17.50 KB (17,920 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\cfgmgr32.dll			
cmd	5.2.3790.0 (srv03_rtm.030324-2048)	374.00 KB (382,976 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\cmd.exe			

browseui	6.00.3790.0 (srv03_rtm.030324-2048)	62.00 KB (63,488 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\browseui.dll			
shdoclc	6.00.3790.0 (srv03_rtm.030324-2048)	588.50 KB (602,624 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\shdoclc.dll			
netplwiz	5.2.3790.0 (srv03_rtm.030324-2048)	843.00 KB (863,232 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\netplwiz.dll			
helpctr	5.2.3790.0 (srv03_rtm.030324-2048)	764.00 KB (782,336 bytes)	4/1/2004 2:34 PM
Microsoft Corporation			
c:\windows\pchealth\helpctr\binaries\helpctr.exe			
hcappres	5.2.3790.0 (srv03_rtm.030324-2048)	6.50 KB (6,656 bytes)	4/1/2004 2:34 PM
Microsoft Corporation			
c:\windows\pchealth\helpctr\binaries\hcappres.dll			
itss	5.2.3790.0 (srv03_rtm.030324-2048)	119.50 KB (122,368 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\itss.dll			
msxml3	8.40.9419.0	1.28 MB (1,337,344 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\msxml3.dll			
pchshell	5.2.3790.0 (srv03_rtm.030324-2048)	100.50 KB (102,912 bytes)	4/1/2004 2:34 PM
Microsoft Corporation			
c:\windows\pchealth\helpctr\binaries\pchshell.dll			
mlang	6.00.3790.0 (srv03_rtm.030324-2048)	570.00 KB (583,680 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\mlang.dll			
mshtml	6.00.3790.0 (srv03_rtm.030324-2048)	2.78 MB (2,916,352 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\mshtml.dll			
msimtf	5.2.3790.0 (srv03_rtm.030324-2048)	149.00 KB (152,576 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\msimtf.dll			
msctf	5.2.3790.0 (srv03_rtm.030324-2048)	287.00 KB (293,888 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\msctf.dll			
jscript	5.6.0.8515	436.00 KB (446,464 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\jscript.dll			
msls31	3.10.349.0	147.00 KB (150,528 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\msls31.dll			
imm32	5.2.3790.0 (srv03_rtm.030324-2048)	105.50 KB (108,032 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\imm32.dll			
mshtml	6.00.3790.0 (srv03_rtm.030324-2048)	443.50 KB (454,144 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\mshtml.dll			
vbscript	5.6.0.8515	404.00 KB (413,696 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\vbscript.dll			
mfc42	6.05.3014.0	960.00 KB (983,040 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\mfc42.dll			
msinfo	5.2.3790.0 (srv03_rtm.030324-2048)	358.50 KB (367,104 bytes)	4/1/2004 2:34 PM
Microsoft Corporation			
c:\windows\pchealth\helpctr\binaries\msinfo.dll			
riched32	5.2.3790.0 (srv03_rtm.030324-2048)	3.50 KB (3,584 bytes)	3/25/2003 7:00 AM
Microsoft Corporation			
c:\windows\system32\riched32.dll			

riched20	5.31.23.1218	406.00 KB (415,744 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\riched20.dll
mydocs	6.00.3790.0 (srv03_rtm.030324-2048)	88.00 KB (90,112 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\mydocs.dll
helpsvc	5.2.3790.0 (srv03_rtm.030324-2048)	720.00 KB (737,280 bytes)	4/1/2004 2:34 PM	Microsoft Corporation	c:\windows\pchealth\helpctr\binaries\helpsvc.exe
tapisrv	5.2.3790.0 (srv03_rtm.030324-2048)	238.50 KB (244,224 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\tapisrv.dll
unimdm	5.2.3790.0 (srv03_rtm.030324-2048)	190.50 KB (195,072 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\unimdm.tsp
uniplat	5.2.3790.0 (srv03_rtm.030324-2048)	15.00 KB (15,360 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\uniplat.dll
kmddsp	5.2.3790.0 (srv03_rtm.030324-2048)	34.00 KB (34,816 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\kmddsp.tsp
ndptsp	5.2.3790.0 (srv03_rtm.030324-2048)	54.50 KB (55,808 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\ndptsp.tsp
ipconf	5.2.3790.0 (srv03_rtm.030324-2048)	16.50 KB (16,896 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\ipconf.tsp
h323	5.2.3790.0 (srv03_rtm.030324-2048)	250.00 KB (256,000 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\h323.tsp
hidphone	5.2.3790.0 (srv03_rtm.030324-2048)	28.00 KB (28,672 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\hidphone.tsp
hid	5.2.3790.0 (srv03_rtm.030324-2048)	17.50 KB (17,920 bytes)	3/24/2003 8:48 PM	Microsoft Corporation	c:\windows\system32\hid.dll

[Services]

Display Name	Name	State	Start Mode	Service Type	Path
Error Control	Start NameTag ID				
Alerter	Alerter	Stopped	Disabled	Share Process	
	c:\windows\system32\svchost.exe -k localservice		Normal	NT	
	AUTHORITY\LocalService	0			
Application Layer Gateway Service		ALG	Stopped	Manual	Own
Process	c:\windows\system32\alg.exe	Normal	NT		
	AUTHORITY\LocalService	0			
Application Management	AppMgmt	Stopped	Manual	Share Process	
	c:\windows\system32\svchost.exe -k netsvcs	Normal	LocalSystem	0	
Windows Audio	AudioSrv	Stopped	Disabled	Share Process	
	c:\windows\system32\svchost.exe -k netsvcs	Normal	LocalSystem	0	
Background Intelligent Transfer Service	BITS	Stopped	Manual	Share	
Process	c:\windows\system32\svchost.exe -k netsvcs	Normal	LocalSystem		
0					
Computer Browser	Browser	Stopped	Manual	Share Process	
	c:\windows\system32\svchost.exe -k netsvcs	Normal	LocalSystem	0	

Indexing Service	CiSvc	Stopped	Disabled	Share Process	
	c:\windows\system32\cisvc.exe	Normal	LocalSystem	0	
ClipBook	ClipSrv	Stopped	Disabled	Own Process	
	c:\windows\system32\clipsrv.exe	Normal	LocalSystem	0	
COM+ System Application	COMSysApp	Stopped	Manual	Own	
Process	c:\windows\system32\dlhst.exe				
	/processid:{02d4b3f1-fd88-11d1-960d-00805fc79235}	Normal	LocalSystem	0	
Cryptographic Services	CryptSvc	Stopped	Manual	Share Process	
	c:\windows\system32\svchost.exe -k netsvcs	Normal	LocalSystem	0	
DB2DAS - DB2DAS00	DB2DAS00	Stopped	Auto	Own	
Process	"d:\sql\lib\bin\db2dasrm.exe"	Normal	.\tpch	0	
DB2EXT - QUAL	DB2EXT - QUAL	Stopped	Manual	Own Process	
	d:\sql\lib\bin\ctelock.exe	Normal	.\tpch	0	
DB2 Governor	DB2GOVERNOR	Stopped	Manual	Own Process	
	"d:\sql\lib\bin\db2govds.exe"	Normal	.\tpch	0	
DB2 JDBC Applet Server	DB2JDS	Running	Auto	Own Process	
	"d:\sql\lib\bin\db2jds.exe"	Normal	LocalSystem	0	
DB2 License Server	DB2LICD	Running	Auto	Own Process	
	d:\sql\lib\bin\db2licd.exe	Ignore	LocalSystem	0	
DB2 Security Server	DB2NTSECSERVER	Running	Auto	Own Process	
	"d:\sql\lib\bin\db2sec.exe"	Normal	LocalSystem	0	
DB2 Remote Command Server	DB2REMOTECMD	Running	Auto	Own	
Process	"d:\sql\lib\bin\db2rcmd.exe"	Ignore	.\tpch	0	
Distributed File System	Dfs	Running	Auto	Own Process	
	c:\windows\system32\dfssvc.exe	Normal	LocalSystem	0	
DHCP Client	Dhcp	Stopped	Manual	Share Process	
	c:\windows\system32\svchost.exe -k networkservice	Normal	NT		
AUTHORITY\NetworkService	0				
Logical Disk Manager	Administrative Service		dmadmin	Stopped	Manual
Share Process	c:\windows\system32\dmadmin.exe /com		Normal		
LocalSystem	0				
Logical Disk Manager	dmserver	Running	Auto	Share Process	
	c:\windows\system32\svchost.exe -k netsvcs	Normal	LocalSystem	0	
DNS Client	Dnscache	Running	Auto	Share Process	
	c:\windows\system32\svchost.exe -k networkservice	Normal	NT		
AUTHORITY\NetworkService	0				
Error Reporting Service	ERSvc	Stopped	Manual	Share Process	
	c:\windows\system32\svchost.exe -k winerr	Ignore	LocalSystem	0	
Event Log	Eventlog	Running	Auto	Share Process	
	c:\windows\system32\services.exe	Normal	LocalSystem	0	
COM+ Event System	EventSystem	Running	Manual	Share Process	
	c:\windows\system32\svchost.exe -k netsvcs	Normal	LocalSystem	0	
Help and Support	helpsvc	Running	Manual	Share Process	
	c:\windows\system32\svchost.exe -k netsvcs	Normal	LocalSystem	0	
Human Interface Device Access	HidServ	Stopped	Disabled	Share Process	
	c:\windows\system32\svchost.exe -k netsvcs	Normal	LocalSystem	0	
HTTP SSLHTTPFilter		Stopped	Manual	Share Process	
	c:\windows\system32\lsass.exe	Normal	LocalSystem	0	
IBM Active PCI Alert Service	IBMHPAS	Running	Auto	Own Process	
	c:\windows\system32\ibmhpasv.exe	Normal	LocalSystem	0	
IMAPI CD-Burning COM Service	ImapiService	Stopped			
Disabled	Own Process		c:\windows\system32\imapi.exe	Normal	
LocalSystem	0				
InterSite Messaging	IsmServ	Stopped	Disabled	Own Process	
	c:\windows\system32\ismserv.exe		Normal	LocalSystem	0
Kerberos Key Distribution Center	kdc	Stopped	Disabled	Share	
Process	c:\windows\system32\lsass.exe	Normal	LocalSystem	0	
Server	lanmanserver	Running	Auto	Share Process	
	c:\windows\system32\svchost.exe -k netsvcs	Normal	LocalSystem	0	
Workstation	lanmanworkstation	Running	Auto	Share Process	
	c:\windows\system32\svchost.exe -k netsvcs	Normal	LocalSystem	0	
License Logging	LicenseService	Stopped	Disabled	Own Process	
	c:\windows\system32\llsrv.exe	Normal	NT AUTHORITY\NetworkService	0	
TCP/IP NetBIOS Helper	LmHosts	Running	Auto	Share Process	
	c:\windows\system32\svchost.exe -k localservice		Normal	NT	
AUTHORITY\LocalService	0				
Machine Debug Manager	MDM	Running	Auto	Own Process	
	"c:\program files\common files\microsoft shared\vs7debug\mdm.exe"		Normal		
LocalSystem	0				
Messenger	Messenger	Stopped	Disabled	Share Process	
	c:\windows\system32\svchost.exe -k netsvcs	Normal	LocalSystem	0	

NetMeeting Remote Desktop Sharing	mnmsrvc	Stopped	Disabled	Own
Process c:\windows\system32\mnmsrvc.exe		Normal	LocalSystem	
Distributed Transaction Coordinator	MSDTC	Running	Auto	Own
Process c:\windows\system32\msdtc.exe	Normal	NT		
AUTHORITY\NetworkService	0			
Windows Installer	MSIServer	Stopped	Manual	Share Process
c:\windows\system32\msiexec.exe /v	Normal		LocalSystem	0
Network DDE	NetDDE	Stopped	Disabled	Share Process
c:\windows\system32\netdde.exe	Normal		LocalSystem	0
Network DDE DSDM	NetDDEdsdm	Stopped	Disabled	Share Process
c:\windows\system32\netdde.exe	Normal		LocalSystem	0
Net Logon	Netlogon	Stopped	Manual	Share Process
c:\windows\system32\lsass.exe	Normal		LocalSystem	0
Network Connections	Netman	Running	Manual	Share Process
c:\windows\system32\svchost.exe -k netsvcs	Normal		LocalSystem	0
Network Location Awareness (NLA)	Nla	Running	Manual	Share Process
Process c:\windows\system32\svchost.exe -k netsvcs	Normal		LocalSystem	0
File Replication	NtFrs	Stopped	Manual	Own Process
c:\windows\system32\ntfrs.exe	Ignore		LocalSystem	0
NT LM Security Support Provider	NtLmSsp	Stopped	Manual	Share Process
Process c:\windows\system32\lsass.exe	Normal		LocalSystem	0
Removable Storage	NtmsSvc	Stopped	Manual	Share Process
c:\windows\system32\svchost.exe -k netsvcs	Normal		LocalSystem	0
Plug and Play	PlugPlay	Running	Auto	Share Process
c:\windows\system32\services.exe	Normal		LocalSystem	0
IPSEC Services	PolicyAgent	Running	Auto	Share Process
c:\windows\system32\lsass.exe	Normal		LocalSystem	0
Protected Storage	ProtectedStorage	Running	Auto	Share Process
c:\windows\system32\lsass.exe	Normal		LocalSystem	0
DB2 - QUAL-0	QUAL-0	Stopped	Manual	Own Process
d:\sql\lib\bin\db2syscs.exe	Normal		.\tpch	0
DB2 - QUAL-1	QUAL-1	Stopped	Manual	Own Process
d:\sql\lib\bin\db2syscs.exe	Normal		.\tpch	0
Remote Access Auto Connection Manager	RasAuto	Stopped	Manual	Share Process
Process c:\windows\system32\svchost.exe -k netsvcs	Normal		LocalSystem	0
Remote Access Connection Manager	RasMan	Running	Manual	Share Process
Process c:\windows\system32\svchost.exe -k netsvcs	Normal		LocalSystem	0
Remote Desktop Help Session Manager	RDSessMgr		Stopped	Manual
Own Process c:\windows\system32\sessmgr.exe		Normal		
LocalSystem	0			
Routing and Remote Access	RemoteAccess	Stopped	Disabled	Share Process
Process c:\windows\system32\svchost.exe -k netsvcs	Normal		LocalSystem	0
Remote Registry	RemoteRegistry	Stopped	Manual	Share Process
c:\windows\system32\svchost.exe -k regsvc	Normal		NT	
AUTHORITY\LocalService	0			
Remote Procedure Call (RPC) Locator	RpcLocator		Stopped	Manual
Own Process c:\windows\system32\locator.exe	Normal		NT	
AUTHORITY\NetworkService	0			
Remote Procedure Call (RPC)	RpcSs	Running	Auto	Share Process
c:\windows\system32\svchost.exe -k rpcss	Normal		LocalSystem	0
Resultant Set of Policy Provider	RSOPProv	Stopped	Manual	Share Process
c:\windows\system32\rsopprov.exe	Normal		LocalSystem	0
Special Administration Console Helper	sacsvr	Stopped	Manual	Share Process
Process c:\windows\system32\svchost.exe -k netsvcs	Normal		LocalSystem	0
Security Accounts Manager	SamSs	Running	Auto	Share Process
c:\windows\system32\lsass.exe	Normal		LocalSystem	0
Smart Card	SCardSvr	Stopped	Manual	Share Process
c:\windows\system32\scardsvr.exe	Ignore		NT	
AUTHORITY\LocalService	0			
Task Scheduler	Schedule	Running	Auto	Share Process
c:\windows\system32\svchost.exe -k netsvcs	Normal		LocalSystem	0
Secondary Logon	seclogon	Running	Auto	Share Process
c:\windows\system32\svchost.exe -k netsvcs	Ignore		LocalSystem	0
System Event Notification	SENS	Running	Auto	Share Process
c:\windows\system32\svchost.exe -k netsvcs	Normal		LocalSystem	0
ServeRAID Manager Agent	ServeRAIDManagerAgent		Stopped	Manual
Own Process c:\program files\raidman\raidserv.exe		Normal		
LocalSystem	0			

Internet Connection Firewall (ICF) / Internet Connection Sharing (ICS)				
SharedAccess	Stopped	Disabled	Share Process	
c:\windows\system32\svchost.exe -k netsvcs	Normal		LocalSystem	0
Shell Hardware Detection	ShellHWDetection	Running	Auto	Share Process
Process c:\windows\system32\svchost.exe -k netsvcs	Ignore		LocalSystem	0
Print Spooler	Spooler	Stopped	Manual	Own Process
c:\windows\system32\spoolsv.exe		Normal	LocalSystem	0
Windows Image Acquisition (WIA)	stisvc	Stopped	Disabled	Share Process
Process c:\windows\system32\svchost.exe -k imgs	svcs	Normal	NT	
AUTHORITY\LocalService	0			
Microsoft Software Shadow Copy Provider	swprv	Stopped	Manual	Own Process
Process c:\windows\system32\svchost.exe -k swprv	Normal		LocalSystem	0
Performance Logs and Alerts	SysmonLog		Stopped	Manual
Process c:\windows\system32\smlogsvc.exe		Normal	NT	
Authority\NetworkService	0			
Telephony TapiSrv	Running	Manual	Share Process	
c:\windows\system32\svchost.exe -k tapisrv	Normal		LocalSystem	0
Terminal Services	TermService	Running	Manual	Share Process
c:\windows\system32\svchost.exe -k termsvcs		Normal	LocalSystem	0
Themes	Themes	Stopped	Disabled	Share Process
c:\windows\system32\svchost.exe -k netsvcs	Normal		LocalSystem	0
Telnet	TlntSvr	Stopped	Disabled	Own Process
c:\windows\system32\tlntsvr.exe	Normal		NT AUTHORITY\LocalService	0
DB2 - TPCCH-0	TPCH-0	Stopped	Manual	Own Process
d:\sql\lib\bin\db2syscs.exe	Normal		.\tpch	0
DB2 - TPCCH-1	TPCH-1	Stopped	Manual	Own Process
d:\sql\lib\bin\db2syscs.exe	Normal		.\tpch	0
DB2 - TPCCH-2	TPCH-2	Stopped	Manual	Own Process
d:\sql\lib\bin\db2syscs.exe	Normal		.\tpch	0
DB2 - TPCCH-3	TPCH-3	Stopped	Manual	Own Process
d:\sql\lib\bin\db2syscs.exe	Normal		.\tpch	0
DB2 - TPCCH-4	TPCH-4	Stopped	Manual	Own Process
d:\sql\lib\bin\db2syscs.exe	Normal		.\tpch	0
DB2 - TPCCH-5	TPCH-5	Stopped	Manual	Own Process
d:\sql\lib\bin\db2syscs.exe	Normal		.\tpch	0
DB2 - TPCCH-6	TPCH-6	Stopped	Manual	Own Process
d:\sql\lib\bin\db2syscs.exe	Normal		.\tpch	0
DB2 - TPCCH-7	TPCH-7	Stopped	Manual	Own Process
d:\sql\lib\bin\db2syscs.exe	Normal		.\tpch	0
Distributed Link Tracking Server	TrkSvr	Stopped	Disabled	Share Process
c:\windows\system32\svchost.exe -k netsvcs	Normal		LocalSystem	0
Distributed Link Tracking Client	TrkWks	Running	Auto	Share Process
c:\windows\system32\svchost.exe -k netsvcs	Normal		LocalSystem	0
Terminal Services Session Directory	Tssdis	Stopped	Disabled	Own Process
Process c:\windows\system32\tssdis.exe	Normal		LocalSystem	0
Upload Manager	uploadmgr	Stopped	Manual	Share Process
c:\windows\system32\svchost.exe -k netsvcs	Normal		LocalSystem	0
Uninterruptible Power Supply	UPS	Stopped	Manual	Own Process
c:\windows\system32\ups.exe	Normal		NT AUTHORITY\LocalService	0
Virtual Disk Service	vds	Stopped	Manual	Own Process
c:\windows\system32\vds.exe	Normal		LocalSystem	0
Volume Shadow Copy	VSS	Stopped	Manual	Own Process
c:\windows\system32\vssvc.exe	Normal		LocalSystem	0
Windows Time	W32Time	Stopped	Manual	Share Process
c:\windows\system32\svchost.exe -k netsvcs	Normal		LocalSystem	0
WebClient WebClient	Stopped	Disabled	Share Process	
c:\windows\system32\svchost.exe -k local	service	Normal	NT	
AUTHORITY\LocalService	0			
WinHTTP Web Proxy Auto-Discovery Service			WinHttpAutoProxySvc	
Stopped	Manual	Share Process	c:\windows\system32\svchost.exe -k	
localservice	Normal	NT	AUTHORITY\LocalService	0
Windows Management Instrumentation	wimgmt	Running	Auto	Share Process
Process c:\windows\system32\svchost.exe -k netsvcs	Ignore		LocalSystem	0
Portable Media Serial Number Service	WmdmPmSN	Stopped	Manual	
Share Process c:\windows\system32\svchost.exe -k netsvcs	Normal		LocalSystem	0
Windows Management Instrumentation Driver Extensions			Wmi	
Stopped	Manual	Share Process	c:\windows\system32\svchost.exe -k	
netsvcs	Normal	LocalSystem	0	

WMI Performance Adapter WmiApSrv Stopped Manual Own
Process c:\windows\system32\wbem\wmiaprv.exe Normal LocalSystem
0

Automatic Updates wuauserv Stopped Manual Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0
Wireless Configuration WZCSVC Stopped Manual Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0

[Program Groups]

Group Name	Name	User Name	Default User
Accessories	Default User:Accessories	Default User	
Accessories\Accessibility	Default User:Accessories\Accessibility	Default User	Default
Accessories\Entertainment	Default User:Accessories\Entertainment	Default User	Default
Startup	Default User:Startup	Default User	
Accessories	All Users:Accessories	All Users	
Accessories\Accessibility	All Users:Accessories\Accessibility	All Users	All
Accessories\Communications	All Users:Accessories\Communications	All Users	All
Accessories\Entertainment	All Users:Accessories\Entertainment	All Users	All
Accessories\System Tools	All Users:Accessories\System Tools	All Users	All
Administrative Tools	All Users:Administrative Tools	All Users	
IBM DB2	All Users:IBM DB2	All Users	
IBM DB2\Command Line Tools	All Users:IBM DB2\Command Line Tools	All Users	All
IBM DB2\Development Tools	All Users:IBM DB2\Development Tools	All Users	All
IBM DB2\General Administration Tools	All Users:IBM DB2\General Administration Tools	All Users	
IBM DB2\Information	All Users:IBM DB2\Information	All Users	
IBM DB2\Monitoring Tools	All Users:IBM DB2\Monitoring Tools	All Users	All
IBM DB2\Set-up Tools	All Users:IBM DB2\Set-up Tools	All Users	All
Microsoft Visual Studio .NET 2003	All Users:Microsoft Visual Studio .NET 2003	All Users	
Microsoft Visual Studio .NET 2003\Visual Studio .NET Tools	All Users:Microsoft Visual Studio .NET 2003\Visual Studio .NET Tools	All Users	All
ServeRAID Manager	All Users:ServeRAID Manager	All Users	
Startup	NT AUTHORITY\SYSTEM:Accessories	NT AUTHORITY\SYSTEM	
Accessories\Accessibility	NT AUTHORITY\SYSTEM:Accessories\Accessibility	NT AUTHORITY\SYSTEM	
Accessories\Entertainment	NT AUTHORITY\SYSTEM:Accessories\Entertainment	NT AUTHORITY\SYSTEM	
Startup	NT AUTHORITY\SYSTEM:Startup	NT AUTHORITY\SYSTEM	
Accessories	RELENTLESS1\tpch:Accessories	RELENTLESS1\tpch	
Accessories\Accessibility	RELENTLESS1\tpch:Accessories\Accessibility	RELENTLESS1\tpch	
Accessories\Entertainment	RELENTLESS1\tpch:Accessories\Entertainment	RELENTLESS1\tpch	
Administrative Tools	RELENTLESS1\tpch:Administrative Tools	RELENTLESS1\tpch	
Startup	RELENTLESS1\tpch:Startup	RELENTLESS1\tpch	

[Startup Programs]

Program	Command	User Name	Location
desktop	desktop.ini	NT AUTHORITY\SYSTEM	Startup
desktop	desktop.ini	RELENTLESS1\tpch	Startup
desktop	desktop.ini	DEFAULT	Startup
desktop	desktop.ini	All Users	Common Startup
KernelFaultCheck	%systemroot%\system32\dumprep 0 -k	All Users	
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run			

[OLE Registration]

Object	Local Server
Sound (OLE2)	sndrec32.exe

Media Clip mplay32.exe
Video Clip mplay32.exe /avi
MIDI Sequence mplay32.exe /mid
Sound Not Available
Media Clip Not Available
WordPad Document "%programfiles%\windows nt\accessories\wordpad.exe"
Windows Media Services DRM Storage object Not Available
Bitmap Image mspaint.exe
[Windows Error Reporting]
Time Type Details

[Internet Settings]

[Internet Explorer]
[Following are sub-categories of this main category]
[Summary]
Item Value
Version 6.0.3790.0
Build 63790
Application Path C:\Program Files\Internet Explorer
Language English (United States)
Active Printer Not Available
Cipher Strength 128-bit
Content Advisor Disabled
IEAK Install No

[File Versions]

File	Version	Size	Date	Path	Company
actxprxy.dll	6.0.3790.0	95 KB	3/25/2003 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
actxprxy.dll	4.70.0.1215	37 KB	10/15/1996 6:40:38 AM	D:\tools	Microsoft Corporation
advpack.dll	6.0.3790.0	94 KB	3/25/2003 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
advpack.dll	4.70.0.1144	23 KB	10/15/1996 6:40:38 AM	D:\tools	Microsoft Corporation
asctrls.ocx	6.0.3790.0	90 KB	3/25/2003 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
browsecl.dll	6.0.3790.0	62 KB	3/25/2003 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
browseui.dll	6.0.3790.0	1,033 KB	3/25/2003 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
cdview.dll	6.0.3790.0	144 KB	3/25/2003 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
comctl32.dll	5.82.3790.0	561 KB	3/25/2003 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
dxtrans.dll	6.3.3790.0	198 KB	3/25/2003 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
dxtmsft.dll	6.3.3790.0	344 KB	3/25/2003 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
iecont.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
iecont.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
iedkcs32.dll	16.0.3790.0	300 KB	3/25/2003 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
iepeers.dll	6.0.3790.0	230 KB	3/25/2003 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
iesetup.dll	6.0.3790.0	59 KB	3/25/2003 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
ieunit.inf	Not Available	20 KB	3/25/2003 8:00:00 AM	C:\WINDOWS\system32	Not Available
ieexplore.exe	6.0.3790.0	90 KB	3/25/2003 8:00:00 AM	C:\Program Files\Internet Explorer	Microsoft Corporation
imgutil.dll	5.2.3790.0	35 KB	3/25/2003 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
inetcp.cpl	6.0.3790.0	303 KB	3/25/2003 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
inetcp.dll	6.0.3790.0	109 KB	3/25/2003 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
inseng.dll	6.0.3790.0	72 KB	3/25/2003 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
mlang.dll	6.0.3790.0	570 KB	3/25/2003 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation

msencode.dll 2002.10.4.0 112 KB 3/25/2003 8:00:00 AM
 C:\WINDOWS\system32 Not Available
 mshta.exe 6.0.3790.0 26 KB 3/25/2003 8:00:00 AM
 C:\WINDOWS\system32 Microsoft Corporation
 mshtml.dll 6.0.3790.0 2,848 KB 3/25/2003 8:00:00 AM
 C:\WINDOWS\system32 Microsoft Corporation
 mshtml.tlb 6.0.3790.0 1,319 KB 3/25/2003 8:00:00 AM
 C:\WINDOWS\system32 Microsoft Corporation
 mshtml.dll 6.0.3790.0 444 KB 3/25/2003 8:00:00 AM
 C:\WINDOWS\system32 Microsoft Corporation
 mshtml.dll 6.0.3790.0 55 KB 3/25/2003 8:00:00 AM
 C:\WINDOWS\system32 Microsoft Corporation
 msident.dll 6.0.3790.0 47 KB 3/25/2003 8:00:00 AM
 C:\WINDOWS\system32 Microsoft Corporation
 msidntld.dll 6.0.3790.0 15 KB 3/25/2003 8:00:00 AM
 C:\WINDOWS\system32 Microsoft Corporation
 msieftp.dll 6.0.3790.0 230 KB 3/25/2003 8:00:00 AM
 C:\WINDOWS\system32 Microsoft Corporation
 msrating.dll 6.0.3790.0 132 KB 3/25/2003 8:00:00 AM
 C:\WINDOWS\system32 Microsoft Corporation
 mstime.dll 6.0.3790.0 491 KB 3/25/2003 8:00:00 AM
 C:\WINDOWS\system32 Microsoft Corporation
 occache.dll 6.0.3790.0 89 KB 3/25/2003 8:00:00 AM
 C:\WINDOWS\system32 Microsoft Corporation
 proctexe.ocx 6.3.3790.0 78 KB 3/25/2003 8:00:00 AM
 C:\WINDOWS\system32 Intel Corporation
 sendmail.dll 6.0.3790.0 52 KB 3/25/2003 8:00:00 AM
 C:\WINDOWS\system32 Microsoft Corporation
 shdoclc.dll 6.0.3790.0 589 KB 3/25/2003 8:00:00 AM
 C:\WINDOWS\system32 Microsoft Corporation
 shdocvw.dll 6.0.3790.0 1,361 KB 3/25/2003 8:00:00 AM
 C:\WINDOWS\system32 Microsoft Corporation
 shfolder.dll 6.0.3790.0 23 KB 3/25/2003 8:00:00 AM
 C:\WINDOWS\system32 Microsoft Corporation
 shlwapi.dll 6.0.3790.0 281 KB 3/25/2003 8:00:00 AM
 C:\WINDOWS\system32 Microsoft Corporation
 tdc.ocx 1.3.0.3130 58 KB 3/25/2003 8:00:00 AM
 C:\WINDOWS\system32 Microsoft Corporation
 url.dll 6.0.3790.0 36 KB 3/25/2003 8:00:00 AM
 C:\WINDOWS\system32 Microsoft Corporation
 urlmon.dll 6.0.3790.0 502 KB 3/25/2003 8:00:00 AM
 C:\WINDOWS\system32 Microsoft Corporation
 webcheck.dll 6.0.3790.0 262 KB 3/25/2003 8:00:00 AM
 C:\WINDOWS\system32 Microsoft Corporation
 wininet.dll 6.0.3790.0 609 KB 3/25/2003 8:00:00 AM
 C:\WINDOWS\system32 Microsoft Corporation

[Connectivity]

Item	Value
Connection Preference	Never dial
LAN Settings	
AutoConfigProxy	Not Available
AutoProxyDetectMode	Disabled
AutoConfigURL	
Proxy	Disabled
ProxyServer	
ProxyOverride	

[Cache]

[Following are sub-categories of this main category]

[Summary]

Item	Value
Page Refresh Type	Automatic
Temporary Internet Files Folder	C:\Documents and Settings\NetworkService\Local Settings\Temporary Internet Files
Total Disk Space	Not Available
Available Disk Space	Not Available
Maximum Cache Size	Not Available
Available Cache Size	Not Available

[List of Objects]

Program File	Status	CodeBase
No cached object information available		

[Content]

[Following are sub-categories of this main category]

[Summary]

Item	Value
Content Advisor	Disabled
[Personal Certificates]	
Issued To	Issued By Validity Signature Algorithm
No personal certificate information available	
[Other People Certificates]	
Issued To	Issued By Validity Signature Algorithm
No other people certificate information available	
[Publishers]	
Name	
No publisher information available	
[Security]	
Zone	Security Level
My Computer	Custom
Local intranet	Medium-low
Trusted sites	Medium
Internet	High
Restricted sites	High

Appendix B: Database Build Scripts

affinity_8m1n_4P

```
rem Set affinities for LN0 -> LN7 for 8 processor system with hyperthreading.
db2set db2processors=0 -i tpch 0
db2set db2processors=1 -i tpch 1
db2set db2processors=2 -i tpch 2
db2set db2processors=3 -i tpch 3
db2set db2processors=4 -i tpch 4
db2set db2processors=5 -i tpch 5
db2set db2processors=6 -i tpch 6
db2set db2processors=7 -i tpch 7
```

backuptestdb

```
rd o:\backup /s /q
rd p:\backup /s /q
md o:\backup
md p:\backup

time < xxx
db2start
set db2node=4
db2 backup database tpcd to p:\backup WITH 16 BUFFERS PARALLELISM 8
without prompting
time < xxx
rem End of backup node 4
db2stop

db2start
set db2node=5
db2 backup database tpcd to p:\backup WITH 16 BUFFERS PARALLELISM 8
without prompting
time < xxx
rem End of backup node 5
db2stop
time < xxx

db2start
set db2node=6
db2 backup database tpcd to p:\backup WITH 16 BUFFERS PARALLELISM 8
without prompting
time < xxx
rem End of backup node 6
db2stop
time < xxx

db2start
set db2node=7
db2 backup database tpcd to p:\backup WITH 16 BUFFERS PARALLELISM 8
without prompting
time < xxx
rem End of backup node 7
db2stop
time < xxx

db2start
set db2node=0
db2 backup database tpcd to o:\backup WITH 16 BUFFERS PARALLELISM 8
without prompting
time < xxx
rem End of backup node 0
db2stop
time < xxx

db2start
set db2node=1
db2 backup database tpcd to o:\backup WITH 16 BUFFERS PARALLELISM 8
without prompting
```

```
time < xxx
rem End of backup node 1
db2stop
time < xxx

db2start
set db2node=2
db2 backup database tpcd to o:\backup WITH 16 BUFFERS PARALLELISM 8
without prompting
time < xxx
rem End of backup node 2
db2stop
time < xxx

db2start
set db2node=3
db2 backup database tpcd to o:\backup WITH 16 BUFFERS PARALLELISM 8
without prompting
time < xxx
rem End of backup node 3
time < xxx
```

buildtpcd

```
#!/usr/bin/perl
# usage buildtpcd [QUAL]
# ASSUMPTIONS: all ddl files have commits in them!
($myName = $0) =~ s@.*\/@@; $usage=""
Usage: buildtpcd [QUAL]
      where QUAL is the optional parameter saying to build the qualification
      database (sf = 1 - 1GB)\n";

$qual="";
if (@ARGV == 1){
    $qual = $ARGV[0];
}

# get TPC-D specific environment variables
require "getvars";
require "macro.pl";
require "tpcdmacro.pl";
require "version";

# Make output unbuffered.
select(STDOUT);
$_ = 1 ;
#-----#
# verify that necessary environment variables for building the database #
# are present. Default those that aren't necessary #
#-----#

# variables that must be specified for script to run
@reqVars = ("TPCD_PLATFORM",
            "TPCD_PRODUCT",
            "TPCD_VERSION",
            "TPCD_DBNAME",
            "TPCD_MODE",
            "TPCD_SF",
            "TPCD_DDLPATH",
            "TPCD_AUDIT",
            "TPCD_AUDIT_DIR",
            "TPCD_BUILD_STAGE");

# variables default to 'NULL' if unspecified
@defNullVars = ("TPCD_LOAD_SCRIPT",
                "TPCD_LOAD_SCRIPT_QUAL",
                "TPCD_INPUT",
                "TPCD_QUAL_INPUT",
                "TPCD_DBGEN",
                "TPCD_LOGPRIMARY",
                "TPCD_LOGSECOND",
                "TPCD_LOGFILSIZ",
```

```

"TPCD_LOG_DIR",
"TPCD_MACHINE",
"TPCD_AGENTPRI",
"TPCD_STAGING_TABLE_DDL",
"TPCD_PRELOAD_STAGING_TABLE_SCRIPT",
"TPCD_DELETE_STAGING_TABLE_SQL",
"TPCD_RUNSTATSHORT",
"TPCD_ADD_RI",
"TPCD_AST",
"TPCD_DBM_CONFIG",
"TPCD_EXPLAIN_DDL",
"TPCD_NODEGROUP_DEF",
"TPCD_BUFFERPOOL_DEF",
"TPCD_LOAD_DB2SET_SCRIPT",
"TPCD_DB2SET_SCRIPT",
"TPCD_LOG_DIR_SETUP_SCRIPT",
"TPCD_LOAD_CONFIGFILE",
"TPCD_LOAD_DBM_CONFIGFILE",
"TPCD_TEMP");

&setVar(@reqVars, "ERROR");
&setVar(@defNullVars, "NULL");

if ( $qual eq "QUAL" ){
    @reqQualVars = ("TPCD_QUAL_DBNAME",
                    "TPCD_QUAL_DDL",
                    "TPCD_QUAL_TBSP_DDL",
                    "TPCD_QUALCONFIGFILE",
                    "TPCD_DBM_QUALCONFIG",
                    "TPCD_LOAD_QUALCONFIGFILE",
                    "TPCD_LOAD_DBM_QUALCONFIGFILE");

    &setVar(@reqQualVars, "ERROR");

    if ( ( $ENV{"TPCD_QUAL_INPUT"} ) eq "NULL" ){
        if ( ( ( $ENV{"TPCD_DBGEN"} ) eq "NULL" ) ||
              ( ( $ENV{"TPCD_TEMP"} ) eq "NULL" ) ){
            die "TPCD_DBGEN and TPCD_TEMP must be set if flatfiles are not
provided.\n";
        }
    }
}

$platform=$ENV{"TPCD_PLATFORM"};

if (length($ENV{"TPCD_DBPATH"}) <= 0){
    # if no db pathname specified, build the db in the home directory
    if ( $platform eq "aix" ||
        $platform eq "sun" ||
        $platform eq "ptx" ||
        $platform eq "hp" ||
        $platform eq "linux" ){
        $ENV{"TPCD_DBPATH"} = $ENV{"HOME"};
    }
    elsif ( $platform eq "nt" ){
        $ENV{"TPCD_DBPATH"} = $ENV{"HOMEDRIVE"};
    }
    else{
        die "platform '$platform' not supported yet!\n";
    }
}

if ( ( $ENV{"TPCD_INPUT"} ) eq "NULL" ){
    if ( ( ( $ENV{"TPCD_DBGEN"} ) eq "NULL" ) ||
          ( ( $ENV{"TPCD_TEMP"} ) eq "NULL" ) ){
        die "TPCD_DBGEN and TPCD_TEMP must be set if flatfiles are not
provided.\n";
    }
}

#-----#
# ddl script files found under custom directory      #
#-----#

```

```

if (length($ENV{"TPCD_DDL"}) <= 0){
    $ENV{"TPCD_DDL"} = "dss.ddl";
}

if (length($ENV{"TPCD_TBSP_DDL"}) <= 0){
    $ENV{"TPCD_TBSP_DDL"} = "dss.tbsp.ddl";
}

if (length($ENV{"TPCD_INDEXDDL"}) <= 0){
    $ENV{"TPCD_INDEXDDL"} = "dss.index";
}

if (length($ENV{"TPCD_RUNSTATS"}) <= 0){
    $ENV{"TPCD_RUNSTATS"} = "dss.runstats";
}

if (length($ENV{"TPCD_CONFIGFILE"}) <= 0){
    $ENV{"TPCD_CONFIGFILE"} = "dss.dbconfig";
}

#-----#
# other settings                                     #
#-----#

if (length($ENV{"TPCD_BACKUP_DIR"}) <= 0){
    $ENV{"TPCD_BACKUP_DIR"} = "${delim}dev${delim}null";
}

if (length($ENV{"TPCD_COPY_DIR"}) <= 0){
    $ENV{"TPCD_COPY_DIR"} = "${delim}dev${delim}null";
}

if (length($ENV{"TPCD_TEMP"}) <= 1){
    $ENV{"TPCD_TEMP"} = "/u/$instance/sql/lib/tmp";
}

if (length($ENV{"TPCD_PHYS_NODE"}) <= 0){
    $ENV{"TPCD_NODEGROUP_DEF"}="NULL"
}

if (length($ENV{"TPCD_GENERATE_SEED_FILE"}) <= 0){
    $ENV{"TPCD_GENERATE_SEED_FILE"} = "no";
}

if (length($ENV{"TPCD_SORTBUF"}) <= 0){
    $ENV{"TPCD_SORTBUF"} = 4096;
}

if (length($ENV{"TPCD_LOAD_PARALLELISM"}) <= 0){
    $ENV{"TPCD_LOAD_PARALLELISM"} = 0;
}

if (length($ENV{"TPCD_LOADSTATS"}) <= 0){
    $ENV{"TPCD_LOADSTATS"} = "no";
}

if (length($ENV{"TPCD_FASTPARSE"}) <= 0){
    $ENV{"TPCD_FASTPARSE"} = "no";
}

if (length($ENV{"TPCD_LOG"}) <= 0){
    $ENV{"TPCD_LOG"} = "no";
}

if (length($ENV{"TPCD_SMPDEGREE"}) <= 0 ){
    $ENV{"TPCD_SMPDEGREE"} = 1;
}

if (length($ENV{"TPCD_ACTIVATE"}) <= 0){
    $ENV{"TPCD_ACTIVATE"} = "no";
}

if (length($ENV{"TPCD_APPEND_ON"}) <= 0){
    $ENV{"TPCD_APPEND_ON"}="yes"
}

if (length($ENV{"TPCD_GENERATE_SEED_FILE"}) <= 0){
    $ENV{"TPCD_GENERATE_SEED_FILE"}="no";
}

#setup global variables
$tpcdVersion=    $ENV{"TPCD_VERSION"};
$buildStage=    $ENV{"TPCD_BUILD_STAGE"};
$mode=          $ENV{"TPCD_MODE"};
$delim =        $ENV{"TPCD_PATH_DELIM"};
$sep =          $ENV{"COMMAND_SEP"};
$dddlpath=      $ENV{"TPCD_DDL_PATH"};
$extraindex=    $ENV{"TPCD_EXTRAINDEX"};

```

```

$earlyindex=      $ENV{"TPCD_EARLYINDEX"};
$loadstats=      $ENV{"TPCD_LOADSTATS"};
$addRI=          $ENV{"TPCD_ADD_RI"};
$astFile=        $ENV{"TPCD_AST"};
$genSeed=        $ENV{"TPCD_GENERATE_SEED_FILE"};
$log=            $ENV{"TPCD_LOG"};
$activate=       $ENV{"TPCD_ACTIVATE"};
$realAudit=      $ENV{"TPCD_AUDIT"};
$auditDir=       $ENV{"TPCD_AUDIT_DIR"};
$loadsetScript=  $ENV{"TPCD_LOAD_DB2SET_SCRIPT"};
$user=           $ENV{"USER"};
$logDirScript=   $ENV{"TPCD_LOG_DIR_SETUP_SCRIPT"};
$logprimary=     $ENV{"TPCD_LOGPRIMARY"};
$logsecond=      $ENV{"TPCD_LOGSECOND"};
$logfilsize=     $ENV{"TPCD_LOGFILSIZE"};
$dbpath=         $ENV{"TPCD_DBPATH"};
$explainDDL=     $ENV{"TPCD_EXPLAIN_DDL"};
$platform=       $ENV{"TPCD_PLATFORM"};
$buffpooldef=    $ENV{"TPCD_BUFFERPOOL_DEF"};
$stagingTbl=     $ENV{"TPCD_STAGING_TABLE_DDL"};
$preloadSampleUF=
$ENV{"TPCD_PRELOAD_STAGING_TABLE_SCRIPT"};
$deleteSampleUF= $ENV{"TPCD_DELETE_STAGING_TABLE_SQL"};
$machine=        $ENV{"TPCD_MACHINE"};
$runstatShort=   $ENV{"TPCD_RUNSTATSHORT"};
$runstats=       $ENV{"TPCD_RUNSTATS"};
$smpdegree=      $ENV{"TPCD_SMPDEGREE"};
$agentpri=       $ENV{"TPCD_AGENTPRI"};
$setScript=      $ENV{"TPCD_DB2SET_SCRIPT"};
$backupdir=      $ENV{"TPCD_BACKUP_DIR"};
$nodegroupdef=   $ENV{"TPCD_NODEGROUP_DEF"};
$dbgen=          $ENV{"TPCD_DBGEN"};
$appendOn=       $ENV{"TPCD_APPEND_ON"};
$indexddl=       $ENV{"TPCD_INDEXDDL"};

if($qual eq "QUAL"){
    $logDir=      $ENV{"TPCD_LOG_QUAL_DIR"};
    $dbname=      $ENV{"TPCD_QUAL_DBNAME"};
    $input=       $ENV{"TPCD_QUAL_INPUT"};
    $sf=          $ENV{"TPCD_QUAL_SF"};
    $loadconfigfile=$ENV{"TPCD_LOAD_QUALCONFIGFILE"};
    $loadDBMconfig=$ENV{"TPCD_LOAD_DBM_QUALCONFIGFILE"};
    $loadscript=  $ENV{"TPCD_LOAD_SCRIPT_QUAL"};
    $configfile=  $ENV{"TPCD_QUALCONFIGFILE"};
    $dbmconfig=   $ENV{"TPCD_DBM_QUALCONFIG"};
    $ddl=         $ENV{"TPCD_QUAL_DDL"};
    $tbpsddl=     $ENV{"TPCD_QUAL_TBSP_DDL"};
}else{
    $logDir=      $ENV{"TPCD_LOG_DIR"};
    $dbname=      $ENV{"TPCD_DBNAME"};
    $input=       $ENV{"TPCD_INPUT"};
    $sf=          $ENV{"TPCD_SF"};
    $loadconfigfile=$ENV{"TPCD_LOAD_CONFIGFILE"};
    $loadDBMconfig=$ENV{"TPCD_LOAD_DBM_CONFIGFILE"};
    $loadscript=  $ENV{"TPCD_LOAD_SCRIPT"};
    $configfile=  $ENV{"TPCD_CONFIGFILE"};
    $dbmconfig=   $ENV{"TPCD_DBM_CONFIG"};
    $ddl=         $ENV{"TPCD_DDL"};
    $tbpsddl=     $ENV{"TPCD_TBSP_DDL"};
}

if(( $mode eq "uni" ) || ( $mode eq "smp" )){
    $all_in="once";
    $all_pn="once";
    $once="once";
}else{
    $all_in="all_in";
    $all_pn="all_pn";
    $once="once";
}

#-----#

```

```

# echo parameter settings to acknowledge what is being built      #
# and set db2set options for database load                          #
#-----#

&printSummary;

print "\nSleeping for 15 seconds to give you a chance to reconsider...\n";
sleep 15;

system("db2start");

if ( $platform eq "nt" ){
    if (( $mode eq "uni" ) || ( $mode eq "smp" )){
        #spaces required for NT
        $rc=&doddb_noconn("db2set DB2OPTIONS=\n -t -v +c\n",db2set
        DB2NTNOCACHE=ON",$all_in);
    }
    else{
        $rc=&doddb_noconn("db2set DB2OPTIONS=\\n -t -v +c\\n",db2set
        DB2NTNOCACHE=ON",$all_in);
    }
}
else{
    if (( $mode eq "uni" ) || ( $mode eq "smp" )){
        $rc=&doddb_noconn("db2set DB2OPTIONS=\n -t -v +c\n",$all_in);
    }
    else{
        $rc=&doddb_noconn("db2set DB2OPTIONS=\\n -t -v +c\\n",$all_in);
    }
}
if ( $rc != 0 ){
    die "failure setting db2 environment variable : rc = $rc\n";
}

#-----#
# set the db2 env vars for loading, from the TPCD_LOAD_DB2SET_SCRIPT
script #
#-----#

if ( $loadsetScript ne "NULL" )
{
    if ( $platform eq "nt" ){
        if (( $mode eq "uni" ) || ( $mode eq "smp" )){
            $rc=system("$ {ddlpath} $ {delim} $loadsetScript");
        }
        else{
            print " db2_all \n call $ {ddlpath} $ {delim} $loadsetScript\n ";
            $rc=system(" db2_all \n call $ {ddlpath} $ {delim} $loadsetScript\n ");
        }
    }
    else{
        $rc=system("$ {ddlpath} $ {delim} $loadsetScript");
    }
}
($rc == 0) || die "failure loading db2set parms from $loadsetScript\n";

#!&stopStart || die;
!&stopStart;
#-----#
# Begin complete build: TPCD_BUILDSTAGE = ALL                      #
#-----#

if($buildStage eq "ALL") {
    #create the database
    $rc = &createDb;
    ($rc == 0) || die "ERROR: create database failed. rc = $rc\n ";
    &setLog;
};

$rc = &setLoadConfig;

#-----#
# Begin build from CreateTablespace or early Indexes              #
#-----#

```

```

if( $buildStage eq "ALL" ||
    $buildStage eq "CRTTBSP" ||
    ($buildStage eq "INDEX" && $earlyindex eq "yes")){
    !&createNodegroups || print "ERROR: create nodegroups failed.\n";
    !&createBufferPools || print "ERROR: create bufferpools failed.\n";
    &outtime("*** Start of audited Load Time - starting to create tables");
    !&createTablespaces || print "WARNING: create tablespaces error.\n";
    !&createExplainTbls || print "ERROR: create EXPLAIN tables failed.\n";
    !&createTables || print "ERROR: create tables failed.\n";

    mkdir("${delim}tmp${delim}${instance}",0777);

    # if earlyindex requested, create indexes
    if( $earlyindex eq "yes" ){
        !&createIndexes("early") || die "ERROR: create early indexes failed.\n";
    }
    # start the dbgen and load.....call the specific mode for loading (uni,smp,mln)
    !&loadData || die "ERROR: failure during load data.\n";

    # remove the update.pair.num file so when setupDir runs, it doesn't
    # hang waiting for an answer on nt
    &rm("${auditDir}${delim}${dbname}.${user}.update.pair.num");
    # verify that the audit directory exists
    $filename="auditDir";
    if(-e $filename){
        # set up the auditDir/${dbname}.${user}.update.pair.num file
        # to start at update pair 1
        $filename="auditDir${delim}${dbname}.${user}.update.pair.num";
    }else{
        mkdir ("auditDir", 0775) || die "cannot mkdir auditDir";
    }
    print "setting update pair num to 1\n";
    system("echo 1 > $filename");
};
#-----#
# Begin build from Index or Load                                #
#-----#

if( $buildStage eq "ALL" ||
    $buildStage eq "CRTTBSP" ||
    $buildStage eq "LOAD" ||
    $buildStage eq "INDEX"){

    # if indexes haven't been created, do so now
    if( $earlyindex ne "yes" ){
        !&createIndexes("normal") || die "ERROR: create indexes failed.\n";
    }
    if( $extraindex ne "no" ){
        !&createIndexes("extra") || die "ERROR: create extra indexes failed.\n";
    }
}

}; # end create/load/index phase of the build

#-----#
# Begin build from runstats                                    #
#-----#

if( $buildStage eq "ALL" ||
    $buildStage eq "CRTTBSP" ||
    $buildStage eq "LOAD" ||
    $buildStage eq "INDEX" ||
    $buildStage eq "RUNSTATS"){
    # if statistics not gathered on the load, run runstats (we have to run the
    # stats at the same time as the index creation whether it be both during load,
    # or after load)
    # We need to run the runstats as well if we have specified an extra index file
    # for "after load" indexes
    if(( $loadstats eq "no" ) || ( $earlyindex eq "no" ) || ( $extraindex ne "no" )){
        &doRunStats;
    }
};
#-----#

```

```

# End build phase: all/load/index/runstats                                #
#-----#
# Add RI/AST, set run configuration                                    #
#-----#

if ( $addRI ne "NULL" ){
    &outtime("*** Adding RI constraints started");
    &dodb2file($dbname,"$ddlpath${delim}$addRI",&once);
    &outtime("*** Adding RI constraints completed");
}

#add the AST if it has been requested
if ( $astFile ne "NULL" ){
    &outtime("*** Adding AST started");
    &dodb2file($dbname,"$ddlpath${delim}$astFile",&once);
    &outtime("*** Adding AST completed");
}

# check tbsp info
&dodb_conn($dbname,"db2 list tablespaces show detail",&once);

# set the configuration
&outtime("*** Set Configuration started");
&outtime("*** Setting degree of parallelism");

&setConfiguration;
# if logging is enabled, we must take a backup of the database
!&stopStart;
print "Logging Enabled   $log \n";
if ( $log eq "YES" ){
    &outtime("*** Create Backups");
    print "Creating Backups\n";
    &createBackup;
    &outtime("*** Backups done");
}

# stop and restart the database to get config parms recognized
!&stopStart;
#!&stopStart || die;

&outtime("*** Set Configuration completed");
&outtime("*** End of audited Load Time");

#create generated seeds
if ( $genSeed ne "no" ){
    $rc = system("perl createmseedme.pl 1000");
    ($rc != 0) || warn "createmseedme failed\n";
}

#-----#
# Call buildtpcdbatch to compile tpcdbatch                                #
#-----#
# - if we are in real audit mode then we have to do a number of things  #
#   set up the audit directory structure and the run directory structure #
#   so that once we have completed the buildtpcd, we are ready to run.  #
#   first remove any old "update pair number" file so we won't be prompted #
#   doing setupDir.                                                         #
# - before we stop the database for the final time                        #
#   if we are in the real audit mode then run dbtables and dbcheck before #
#   we print out the notice that we're ready to run performance tests    #
#   if we are building the qualification database then we'll bind to both #
#   the dbname database and the qualification database                    #
#-----#

$rc = system("perl buildtpcdbatch $qual");
($rc == 0) || die "buildtpcdbatch failed rc=$rc\n";

if ( $realAudit eq "yes" ){
    &rm("${auditDir}${delim}tools${delim}tpcd.runsetup");
    system("perl setupRun");
    if ( $qual eq "QUAL" ){
        $verifyType="q";
    }
}

```

```

else{
    $verifyType="t";
}
system("perl tablespdb $verifyType");

&dodb2file($dbname,"$auditDir${delim}tools${delim}first10rows.sql",$once);
}

#-----#
# Create Catalog info                                     #
#-----#
$rc = system("perl catinfo.pl b");
($rc == 0) || warn "catinfo failed!!! rc = $rc\n";

$rc=system("db2stop");
#DJD ($rc == 0) || die "failure during db2stop rc = $rc\n";

&outtime("*** Ready to run the performance tests once the dbm has restarted");

if ( $RealAudit ne "yes" ){
    # if we are not in a real audit, then we can restart the database manager
    # if we are in a real audit, then we don't want to do this until the
    # power test starts
    $rc=system("db2start");
    #($rc == 0) || die "failure during db2start rc = $rc\n";
    if ( $activate eq "yes" ){
        &dodb_noconn("db2 activate database $dbname",$once);
    }
}

&outtime("*** Finished creating the database");
#-----#
# finished creating the database                             #
#-----#

#-----#
# Function: setLog                                           #
#-----#
sub setLog{
    # update the log information first
    # set up the log directory before we do any index creation
    my $rc;
    my $setLogs;
    my $setLogString;

    if($logDirScript ne "NULL"){
        system ("perl $ddlpath${delim}$logDirScript");
    }
    elsif ( $logDir ne "NULL" ){
        &dodb_noconn("db2 update database configuration for $dbname using
newlogpath $logDir",$all_in);
    }
    $setLogs=0;
    $setLogString="";
    if ( $logprimary ne "NULL" ){
        $setLogString="db2 update db cfg for $dbname using logprimary
$logprimary";
        $setLogs=1;
    }
    if ( $logsecond ne "NULL" ){
        if ( $setLogs != 0 ){
            $setLogString.=" $sep ";
        }
        $setLogString.="db2 update db cfg for $dbname using logsecond
$logsecond";
        $setLogs=1;
    }
    if ( $logfilsiz ne "NULL" ){
        if ( $setLogs != 0 ){
            $setLogString.=" $sep ";
        }
    }
}

```

```

    $setLogString.="db2 update db cfg for $dbname using logfilsiz
$logfilsiz";
    $setLogs=1;
}
if ( $setLogs != 0 ){
    $setLogString.=" $sep ";
}
$setLogString.="db2 update db cfg for $dbname using logbufsiz 128";
$rc = &dodb_noconn("$setLogString",$all_in);
}

#-----#
# Function: createDb                                         #
#-----#
sub createDb{
    &outtime("*** Starting to create the database");
    # setup required variables
    my $rc;
    $rc = &dodb_noconn("db2 \create database $dbname on $dbpath collate
using identity with 'TPC-D $sf GB'",$once);
    ($rc == 0) || return($rc);
    # reset the db and dbm configuration before we start
    &dodb_noconn("db2 reset database configuration for $dbname",$all_in);
    &dodb_conn($dbname,"db2 alter bufferpool ibmdefaultbp size -1 $sep \
db2 grant connect on database to public $sep \
db2 grant dbadm on database to $dbname $sep \
db2 commit",$once);
    &dodb_noconn("db2 reset database manager configuration",$once);
}

#-----#
# Function: createNodegroups                                 #
#-----#
sub createNodegroups{
    &outtime("*** Creating the nodegroups.");
    my $rc;
    if ( $nodegroupdef ne "NULL"){
        $rc = &dodb2file($dbname,"$ddlpath${delim}$nodegroupdef",$once);
    }
}

#-----#
# Function: createExplainTbls                                #
#-----#
sub createExplainTbls{
    &outtime("*** Creating the EXPLAIN tables.");
    my $rc;
    my $explnPathFile;
    my $home;
    my $sqlpath;

    if ( $explainDDL ne "NULL" ){
        $explnPathFile="$explainDDL";
    }
    else{
        if ( $platform eq "ptx" ){
            $home=$ENV{"HOME"};
            $sqlpath="$home${delim}sqllib";
        }
        if ( $platform ne "nt" ){
            $home=$ENV{"HOME"};
            $sqlpath="$home${delim}sqllib";
        }
        else{
            $sqlpath=$ENV{"DB2PATH"};
        }
        $explnPathFile="$sqlpath${delim}misc${delim}EXPLAIN.DDL";
    }
    $rc = &dodb_conn($dbname,
"db2 -tvf $explnPathFile $sep \
db2 alter table explain_instance locksize table append on $sep \
db2 alter table explain_statement locksize table append on $sep \
db2 alter table explain_argument locksize table append on $sep \

```



```

db2 alter table explain_object locksize table append on $sep \
db2 alter table explain_operator locksize table append on $sep \
db2 alter table explain_predicate locksize table append on $sep \
db2 alter table explain_stream locksize table append on",
$once);
}
#-----#
# Function: createBufferPools                                #
#-----#
sub createBufferPools{
    my $src;
    &outtime("*** Creating the bufferpools");
    if ( $buffpooldef ne "NULL" ){
        #run the create bufferpool ddl
        $src = &dodb2file($dbname,"$ddlpath${delim}$buffpooldef",$once);
    }
}
#-----#
# Function: createTablespaces                                #
#-----#
sub createTablespaces{
    &outtime("*** Ready to start creating the tablespaces");
    # setup required variables
    my $src;
    $src = &dodb2file($dbname,"$ddlpath${delim}$tblspddl",$once);
    ($src == 0) || return $src;
    # create/populate the staging tables
    if ( $stagingTbl ne "NULL" ){
        # staging tables must be created for both test and qualification database
        # but they do not need to be populated for the qualification database
        $src = &dodb2file($dbname,"$ddlpath${delim}$stagingTbl",$once);
        ($src == 0) || return $src;
        if ( $qual ne "QUAL" ){
            if ( $preloadSampleUF ne "NULL" ){
                # preload the sample UF data for statistics gathering
                $src = system ("perl $ddlpath${delim}$preloadSampleUF");
                #($src == 0) || return $src;
            }
            if ( $deleteSampleUF ne "NULL" ){
                # delete the sample rows now that stats have been gathered
                $src =
&dodb2file($dbname,"$ddlpath${delim}$deleteSampleUF",$once);
                #($src == 0) || return $src;
            }
        }
    }
}
#-----#
# Function: createTables                                    #
#-----#
sub createTables{
    my $src;
    $src = &dodb2file($dbname,"$ddlpath${delim}$ddl",$once);
    ($src == 0) || return $src;
    # update the locksize on the non-updated tables to be table level locking
    # update the tables for appendmode
    if ($appendOn eq "yes"){
        $src = &dodb_conn($dbname,
            "db2 alter table tpcd.nation locksize table $sep \
            db2 alter table tpcd.region locksize table $sep \
            db2 alter table tpcd.customer locksize table $sep \
            db2 alter table tpcd.supplier locksize table $sep \
            db2 alter table tpcd.part locksize table $sep \
            db2 alter table tpcd.partsupp locksize table $sep \
            db2 alter table tpcd.lineitem append on $sep \
            db2 alter table tpcd.orders append on",
            $once);
    }
    else{
        $src = &dodb_conn($dbname,
            "db2 alter table tpcd.nation locksize table $sep \
            db2 alter table tpcd.region locksize table $sep \

```

```

            db2 alter table tpcd.customer locksize table $sep \
            db2 alter table tpcd.supplier locksize table $sep \
            db2 alter table tpcd.part locksize table $sep \
            db2 alter table tpcd.partsupp locksize table $sep \
            db2 alter table tpcd.lineitem pctfree 0 $sep \
            db2 alter table tpcd.orders pctfree 0",
            $once);
        }
    }
#-----#
# Function: createIndexes                                    #
#-----#
sub createIndexes{
    # setup required variables
    local @args = @_;
    my $indexType = @args[0];
    my $src;
    &outtime("*** Starting to create $indexType indexes");
    if ( $indexType eq "extra" ){
        $src = &dodb2file($dbname,"$ddlpath${delim}$extraindex",$once);
    }
    elsif ( $indexType eq "early" || $indexType eq "normal" ){
        $src = &dodb2file($dbname,"$ddlpath${delim}$indexddl",$once);
    }
    &outtime("*** Create $indexType index completed");
    # $src = &dodb_conn($dbname,
        "db2 alter table tpcd.orders add primary key (o_orderkey) $sep \
        db2 alter table tpcd.lineitem add primary key (l_orderkey,
        l_linenumbr)",
        $once);
    return $src;
}
#-----#
# Function: setLoadConfig                                    #
#-----#
sub setLoadConfig{
    &outtime("*** Setting LOAD configuration.");
    my $src;
    my $buffpage;
    my $sortheap;
    my $sheapthres;
    my $util_heap_sz;
    my $ioservers;
    my $ioclnrs= 1;
    my $schngpgs= 60;

    if ($loadconfigfile eq "NULL"){
        if ( $machine eq "small" ){
            $buffpage = 5000;
            $sortheap = 3000;
            $sheapthres = 8000;
            $util_heap_sz = 5000;
            $ioservers = 6;
        }
        elsif ( $machine eq "medium" ){
            $buffpage = 10000;
            $sortheap = 8000;
            $sheapthres = 20000;
            $util_heap_sz = 10000;
            $ioservers = 10;
        }
        elsif ( $machine eq "big" ){
            $buffpage = 30000;
            $sortheap = 20000;
            $sheapthres = 50000;
            $util_heap_sz = 30000;
            $ioservers = 20;
        }
    }
    else {

```

```

        die "Neither a LOAD config filename nor a valid machine size has
        been specified!\n";
    }
    $rc = &dodb_noconn("db2 update db cfg for $dbname using buffpage
$buffpage $sep \
db2 update db cfg for $dbname using sortheap $sortheap $sep \
db2 update db cfg for $dbname using num_iocleaners $ioclnrs $sep \
db2 update db cfg for $dbname using num_ioservers $ioservers $sep \
db2 update db cfg for $dbname using util_heap_sz $util_heap_sz $sep \
db2 update db cfg for $dbname using chngpgs_thresh
$chngpgs", $all_in);
    }
    else{
        print "$ddlpath${delim}$loadconfigfile\n";
        $rc = &dodb2file_noconn("$ddlpath${delim}$loadconfigfile", $all_in);
        print "Rc= $rc\n";
        print "Calling Setlogs.\n";
        if($qual eq "QUAL"){
            $rc = system("$ddlpath${delim}$setlogs_qual $dbname");
        }
        else {
            $rc = system("$ddlpath${delim}$setlogs $dbname");
        }
        print "Set Logs Rc= $rc\n";
    }
    ($rc == 0) || return $rc;
    if($loadDBMconfig ne "NULL"){
        $rc =
&dodb2file_noconn("$ddlpath${delim}$loadDBMconfig", $once);
    }
    else{
        $rc = &dodb_noconn("db2 update dbm cfg using sheapthres
$sheapthres", $once);
    }
    ($rc == 0) || return $rc;
    &dodb_noconn("db2 terminate", $once);
    $rc = &stopStart;
    return $rc;
}
#-----#
# Function: loadData                                     #
#-----#
sub loadData{
    # start the dbgen and load.....call the specific mode for loading (uni,smp,mln)
    my $rc;
    if (( $mode eq "uni" ) || ( $mode eq "smp" )){
        &outtime("*** Starting the load");
        # call the appropriate dbgen/load for uni/smp
        if ( $loadscript eq "NULL"){
            $rc = system("perl genloaduni $qual");
            ($rc == 0) || print "ERROR: genloaduni failed rc = $rc\n";
        }
        else{
            $rc = &dodb2file_noconn("$ddlpath${delim}$loadscript", $once);
            ($rc == 0) || print "ERROR: load script: $loadscript failed. rc =
$rc\n";
        }
    }
    elsif (( $mode eq "mln" ) || ( $mode eq "mpp" )){
        &outtime("*** Starting the load");
        # call the appropriate dbgen/split(sort)/load for mln/mpp
        if ( $loadscript eq "NULL"){
            $rc = system("perl genloadmpp $qual");
            ($rc == 0) || print "ERROR: genloadmpp failed. rc = $rc\n";
        }
        else{
            system("$ddlpath${delim}$loadscript");
            # $rc = &dodb2file_noconn("$ddlpath${delim}$loadscript $sf");
            # ($rc == 0) || print "ERROR: load script $loadscript failed. rc =
$rc\n";
        }
    }
}

```

```

    else{
        print "TPCD_MODE not set to one of uni, smp, mln or mpp\n";
        $rc = 1;
    }
    ($rc == 0) || &outtime("*** Load complete");
    return $rc;
}
#-----#
# Function: doRunStats                                     #
#-----#
sub doRunStats{
    # if loadstats not gathered, then index stats not gathered either.
    &outtime("*** Runstats started");
    if ( $runstatShort ne "NULL" ){
        # we've specified a second runstats file...This runstats file should do
        # runstats for all table except lineitem. The lineitem runstats command
        # should be left in the main runstats file.
        if ( $platform eq "aix" || $platform eq "sun" || $platform eq "ptx" ){
            print "runstats from $ddlpath${delim}$runstatShort running now\n";
            $rc = system("db2 -tvf \"$ddlpath${delim}$runstatShort\" >
\"$auditDir${delim}$tools${delim}$runstatShort.out\" & ");
            print "rc from runstatshort=$rc\n";
        }
        elsif ( $platform eq "nt" ){
            system("start db2 -tvf $ddlpath${delim}$runstatShort");
        }
        else
        {
            print "Don't know how to start in background on $platform platform\n";
            print "therefore running runstats serially\n";
            &dodb2file($dbname, "$ddlpath${delim}$runstatShort", $once);
        }
    }
    # run the full runstats, or the remainder of what wasn't put into the short
    # runstats file. You should be sure that this runstats will take longer
    # than the short runstats that is running in the background, otherwise
    # setting the config will happen before this is done.
    &dodb2file($dbname, "$ddlpath${delim}$runstats", $once);
    &outtime("*** Runstats completed");
}
#-----#
# Function: setConfiguration                               #
#-----#
sub setConfiguration{
    my $ret = 0;
    &dodb_noconn("db2 update database configuration for $dbname using
dft_degree $smpdegree", $all_in);
    &dodb_noconn("db2 update database manager configuration using
max_querydegree $smpdegree", $once);
    &dodb2file_noconn("$ddlpath${delim}$configfile", $all_in);
    &dodb2file_noconn("$ddlpath${delim}$dbmconfig", $once);

    if ( $agentpri ne "NULL" ){
        &dodb_noconn("db2 update dbm cfg using AGENTPRI
$agentpri", $once);
    }
    # set the db2 environment variables for running the benchmark
    if ( $setScript ne "NULL" ){
        if ( $platform eq "aix" || $platform eq "sun" || $platform eq "ptx" ){
            $ret=system("$ddlpath${delim}$setScript");
        }
        elsif ( $platform eq "nt" ){
            if (( $mode eq "uni" ) || ( $mode eq "smp" )){
                $ret = system("perl $ddlpath${delim}$setScript");
            }
            else{
                $ret=system(" db2_all \" call $ddlpath${delim}$setScript \" ");
            }
        }
        # ($ret == 0) || die "failure setting runtime db2set parms from $setScript
\n";
    }
}

```

```

&outtime("*** Setting up 4K-Temp");
system("CALL ${ddlpath}${delim}temp_4k");
&outtime("*** Config Completed.");
}

#-----#
# Function: createBackup                                     #
#-----#

sub createBackup{
    my $rc;
    &dodb_noconn("db2 update database configuration for $dbname using
LOGRETAIN yes", $all_in);
    print "\n NOTE: DO NOT RESET THE DATABASE CONFIGURATION
or you will lose logretain\n";
    # force a connection to the database on all nodes to ensure LOGRETAIN is
    # set in effect.
    # An error message will print to screen if the logretain is set properly
    # i.e. SQL116N A connection to or activation of database <database name>
    # cannot be made.
    # This is expected and the lack of this error message should be seen as an
    # error in the database build.
    &dodb_conn($dbname,"db2 \" select count(*) from tpcd.region\"", $all_in);
    system("db2 connect reset");
    system("db2 terminate");

    if ( $qual eq "QUAL" ){
        &outtime("*** Starting the backup");
        $rc=system("CALL BKUPQUAL2.BAT");
        ($rc == 0) || &outtime("*** Finished the backup");

        #if ( ( $mode eq "mln" ) || ( $mode eq "mpp" ) ){
        #    # must back up catalog node first...assume node 00
        #    #DJD $rc=system("db2_all \\\}<<+000< db2 \"backup database
$dbname to $backupdir without prompting\" \");

        #($rc == 0) || print "ERROR: backup of catalog node failed rc =
$rc\n";

        # back up remaining nodes
        # $rc=system("db2_all \\\}<<+000< db2 backup database $dbname to
$backupdir without prompting\" \");
        #($rc == 0) || print "ERROR: backup of remaining nodes failed rc =
$rc\n";
        #}
        #else{
        #    $rc = &dodb_noconn("db2 backup database $dbname to
$backupdir", $once);
        #}
        #($rc == 0) || &outtime("*** Finished the backup");
    }
    else{
        &outtime("*** Starting BACKUP");
        # This is the test database. Clause 3.1.4 states that "the test sponsor is
        # not required to make or have backup copies of the test database;
however
        # all other mechanisms that guarantee durability of the qualification
        # database must be enabled in the same way for the test database".
        # According to this clause we do need to keep the backup of the
database.
        #DJD $rc = &dodb_noconn("db2dart $dbname /CHST /WHAT DBBP
OFF", $all_in);
        print "Start taking backups, as RAID 0 is being used\n";
        ($rc == 0) || &outtime("*** Finished the backup");
        &outtime("*** Starting the test db DB backups");
        $rc=system("CALL BKUPTESTDB.BAT");
        &outtime("*** Finished taking backups of test DB ");

    }
    return $rc;
}

#-----#
# Function: printSummary                                     #
#-----#

```

```

sub printSummary{
    if ( $buildStage ne "ALL" ){
        print " ***** STARTING the build process at the $buildStage Stage
*****\n";
    }
    print "Building a TPC-D Version $tpcdVersion $sf GB database on $dbpath
with: \n";
    print "   Mode = $mode \n";
    print "   Tablespace ddl in $ddlpath${delim}$tbspddl \n";
    if ( $nodelgroupdef ne "NULL" ){
        print "   Nodelgroup ddl in $ddlpath${delim}$nodelgroupdef \n";
    }
    if ( $buffpooldef ne "NULL" ){
        print "   Bufferpool ddl in $ddlpath${delim}$buffpooldef \n";
    }
    print "   Table ddl in $ddlpath${delim}$ddl \n";
    print "   Index ddl in $ddlpath${delim}$indexddl \n";
    if ( $extraindex ne "no" ){
        print "   Indices to create after the load $ddlpath${delim}$extraindex \n";
    }
    if ( $loadscript eq "NULL" ){
        if ( $input eq "NULL" ){
            print "   Data generated by DBGEN in $dbgen\n";
        }
        else{
            print "   Data loaded from flat files in $input\n";
        }
    }
    if ( $earlyindex eq "yes" ){
        print "   Indexes created before loading\n";
    }
    else{
        print "   Indexes created after loading\n";
    }
    if ( $addri ne "NULL" ){
        print "   RI being used from $ddlpath${delim}$addri \n";
    }
    if ( $astfile ne "NULL" ){
        print "   AST being used from $ddlpath${delim}$astfile \n";
    }
    if ( $loadstats eq "yes" ){
        if ( $earlyindex eq "yes" ){
            print "   Statistics for tables and indexes gathered during load\n";
        }
        else{
            if ( $runstatShort eq "NULL" ){
                print "   Statistics for tables and indexes gathered after load using
$ddlpath${delim}$runstats \n";
            }
            else{
                print "   Statistics for tables and indexes gathered after load using
$ddlpath${delim}$runstats and $ddlpath${delim}$runstatShort \n";
            }
        }
    }
    else{
        if ( $runstatShort eq "NULL" ){
            print "   Statistics for tables and indexes gathered after load using
$ddlpath${delim}$runstats \n";
        }
        else{
            print "   Statistics for tables and indexes gathered after load using
$ddlpath${delim}$runstats and $ddlpath${delim}$runstatShort \n";
        }
    }
    if ( $loadconfigfile ne "NULL" ){
        print "   Database Configuration parameters for LOAD taken from
$ddlpath${delim}$loadconfigfile \n";
    }
    if ( $loadDBMconfig ne "NULL" ){
        print "   Database manager Configuration parameters for LOAD taken from
$ddlpath${delim}$loadDBMconfig \n";
    }
    if ( $configfile ne "NULL" ){

```

```

    print " Database Configuration parameters taken from
    $ddlpath${delim}$sconfigfile\n";
}
else{
    print " Database Configuration paramters taken from
    $ddlpath${delim}dss.dbconfig${sfReal}GB\n";
    $sconfigfile="dss.dbconfig${sfReal}GB";
}
if ( $dbmconfig ne "NULL" ){
    print " Database Manager Configuration parameters taken from
    $ddlpath${delim}$dbmconfig\n";
}
else{
    print " Database Manager Configuration paramters taken from
    $ddlpath${delim}dss.dbmconfig${sfReal}GB\n";
    $sconfigfile="dss.dbmconfig${sfReal}GB";
}
#print " Copy image for load command created in $copydir\n";
if ( $log eq "yes" ){
    print " Backup files placed in $backupdir\n";
}
else{
    print " No backup will be taken.\n";
}
print " Log retain set to $log\n";
if ( $logDir eq "NULL" ){
    print " Log files remain in database path\n";
}
else{
    print " Log file path set to $logDir\n";
}
if ( $logprimary eq "NULL" ){
    print " Log Primary left at default\n";
}
else{
    print " Log Primary set to $logprimary\n";
}
if ( $logsecond eq "NULL" ){
    print " Log Second left at default\n";
}
else{
    print " Log second set to $logsecond\n";
}
if ( $logfilsiz eq "NULL" ){
    print " Logfilsiz left at default\n";
}
else{
    print " Logfilsiz set to $logfilsiz\n";
}
if ( ($loadconfigfile eq "") || ($loadconfigfile eq "NULL") ){
    print " Machine size set to $machine so the following configuration\n";
    print " parameters are used for load, create index and runstats: \n";
    print " BUFFPAGE = $buffpage \n";
    print " SORTHEAP = $sorheap \n";
    print " SHEAPTHRES = $sheapthres\n";
    print " NUM_IOSERVERS = $ioservers\n";
    print " NUM_IOCLEANERS = $ioclnrs\n";
    print " CHNGPGS_THRESH = $chnpggs\n";
    print " UTIL_HEAP_SZ = $util_heap_sz\n";
    print " Degree of parallelism (dft_degree and max_querydegree) set to
    $smpdegree\n";
    print " Parameters for load are: temp file = $ldtemp\n";
    print " sort buf = $sortbuf\n";
    print " ld parallelism = $load_parallelism\n";
    if ( $fparse eq "yes" ){
        print " FASTPARSE used on load\n";
    }
}
if ( $loadscript ne "NULL" ){
    print " Load commands in $ddlpath${delim}$loadscript\n";
}
print " Degree of parallelism (dft_degree and max_querydegree) set to
    $smpdegree\n";
if ( $agentpri ne "NULL" ){

```

```

    print " AGENTPRI set to $agentpri\n";
}
if ( $activate eq "yes" ){
    print " Database will be activated when build is complete\n";
}
if ( $explainDDL ne "NULL" ){
    print " EXPLAIN DDL being used from
    $ddlpath${delim}$explainDDL\n";
}
else{
    print " EXPLAIN DDL being used from default sqllib directory\n";
}
}
1;

```

clrlogs.bat

```

rd e:\logs /s /q
rd h:\logs /s /q
rd m:\logs /s /q
rd n:\logs /s /q
rd i:\logs /s /q
rd j:\logs /s /q
rd g:\logs /s /q
rd k:\logs /s /q
md e:\logs
md h:\logs
md m:\logs
md n:\logs
md i:\logs
md j:\logs
md g:\logs
md k:\logs

```

create_bufferpools

```

-- Create 16K page bufferpool
alter bufferpool ibmdefaultbp size 250;
create bufferpool BP16K size -1 pagesize 16k;

```

create_indexes

```

values(current timestamp);
CREATE UNIQUE INDEX "TPCD"."R_RK" ON "TPCD"."REGION"
("R_REGIONKEY" ASC)
PCTFREE 0 ;
commit work;

values(current timestamp);
CREATE UNIQUE INDEX "TPCD"."N_NK" ON "TPCD"."NATION"
("N_NATIONKEY" ASC)
PCTFREE 0 ;
commit work;

values(current timestamp);
CREATE INDEX "TPCD"."N_RK" ON "TPCD"."NATION"
("N_REGIONKEY" ASC)
PCTFREE 0 ;
commit work;

values(current timestamp);
CREATE UNIQUE INDEX "TPCD"."S_SK" ON "TPCD"."SUPPLIER"
("S_SUPPKEY" ASC)
PCTFREE 0 ;
commit work;

values(current timestamp);
CREATE INDEX "TPCD"."S_NK" ON "TPCD"."SUPPLIER"
("S_NATIONKEY" ASC)
PCTFREE 0 ;
commit work;

```

```

values(current timestamp);

values(current timestamp);
CREATE UNIQUE INDEX "TPCD"."P_PK" ON "TPCD"."PART"
    ("P_PARTKEY" ASC)
    PCTFREE 0 ;
commit work;
values(current timestamp);

values(current timestamp);
CREATE INDEX "TPCD"."PS_SK" ON "TPCD"."PARTSUPP"
    ("PS_SUPPKEY" ASC)
    PCTFREE 0 ;
commit work;

values(current timestamp);
CREATE INDEX "TPCD"."PS_PK" ON "TPCD"."PARTSUPP"
    ("PS_PARTKEY" ASC)
    PCTFREE 0 ;
commit work;

values(current timestamp);
CREATE UNIQUE INDEX "TPCD"."PS_PKS" ON "TPCD"."PARTSUPP"
    ("PS_PARTKEY" ASC,
    "PS_SUPPKEY" ASC)
    PCTFREE 0 ;
commit work;

values(current timestamp);
CREATE UNIQUE INDEX "TPCD"."PS_SKPK" ON "TPCD"."PARTSUPP"
    ("PS_SUPPKEY" ASC,
    "PS_PARTKEY" ASC)
    PCTFREE 0 ;
commit work;

values(current timestamp);
CREATE UNIQUE INDEX "TPCD"."C_CK" ON "TPCD"."CUSTOMER"
    ("C_CUSTKEY" ASC)
    PCTFREE 0 ;
commit work;

values(current timestamp);
CREATE INDEX "TPCD"."C_NK" ON "TPCD"."CUSTOMER"
    ("C_NATIONKEY" ASC)
    PCTFREE 0 ;
commit work;

values(current timestamp);
CREATE UNIQUE INDEX "TPCD"."O_OK" ON "TPCD"."ORDERS"
    ("O_ORDERKEY" ASC)
    PCTFREE 3 ;
commit work;

values(current timestamp);
create unique index tpcd.l_ok_ln on tpcd.lineitem (L_orderkey, l_linenum)
pctfree 3;
commit work;
values(current timestamp);

-- values(current timestamp);
-- create unique index tpcd.l_ok on tpcd.lineitem (L_orderkey) pctfree 3;
-- commit work;
-- values(current timestamp);

alter table tpcd.orders add primary key(o_orderkey);
alter table tpcd.lineitem add primary key(l_orderkey,l_linenum);
commit work;

```

create_nodegroups

```

-- Create nodegroups.
create nodegroup catalog_node on node (0);
create nodegroup all_nodes;

```

create_tables

```

CREATE TABLE TPCD.NATION ( N_NATIONKEY INTEGER NOT NULL,
    N_NAME CHAR(25) NOT NULL,
    N_REGIONKEY INTEGER NOT NULL,
    N_COMMENT VARCHAR(152) NOT NULL WITH
DEFAULT)
    IN SMALL_TABLES;

CREATE TABLE TPCD.REGION ( R_REGIONKEY INTEGER NOT NULL,
    R_NAME CHAR(25) NOT NULL,
    R_COMMENT VARCHAR(152) NOT NULL WITH
DEFAULT)
    IN SMALL_TABLES;

CREATE TABLE TPCD.PART ( P_PARTKEY INTEGER NOT NULL,
    P_NAME VARCHAR(55) NOT NULL,
    P_MFGR CHAR(25) NOT NULL,
    P_BRAND CHAR(10) NOT NULL,
    P_TYPE VARCHAR(25) NOT NULL,
    P_SIZE INTEGER NOT NULL,
    P_CONTAINER CHAR(10) NOT NULL,
    P_RETAILPRICE FLOAT NOT NULL,
    P_COMMENT VARCHAR(23) NOT NULL WITH
DEFAULT)
    IN OTHER_STUFF
    INDEX IN OTHER_INDEX
    PARTITIONING KEY(P_PARTKEY) USING HASHING;

CREATE TABLE TPCD.SUPPLIER ( S_SUPPKEY INTEGER NOT NULL,
    S_NAME CHAR(25) NOT NULL,
    S_ADDRESS VARCHAR(40) NOT NULL,
    S_NATIONKEY INTEGER NOT NULL,
    S_PHONE CHAR(15) NOT NULL,
    S_ACCTBAL FLOAT NOT NULL,
    S_COMMENT VARCHAR(101) NOT NULL WITH
DEFAULT)
    IN OTHER_STUFF
    INDEX IN OTHER_INDEX
    PARTITIONING KEY(S_SUPPKEY) USING HASHING
    ORGANIZE BY (
    ("S_NATIONKEY" ) )
    ;

CREATE TABLE TPCD.PARTSUPP ( PS_PARTKEY INTEGER NOT
NULL,
    PS_SUPPKEY INTEGER NOT NULL,
    PS_AVAILQTY INTEGER NOT NULL,
    PS_SUPPLYCOST FLOAT NOT NULL,
    PS_COMMENT VARCHAR(199) NOT NULL WITH
DEFAULT)
    IN OTHER_STUFF
    INDEX IN OTHER_INDEX
    PARTITIONING KEY(PS_PARTKEY) USING HASHING;

CREATE TABLE TPCD.CUSTOMER ( C_CUSTKEY INTEGER NOT
NULL,
    C_NAME CHAR(25) NOT NULL,
    C_ADDRESS VARCHAR(40) NOT NULL,
    C_NATIONKEY INTEGER NOT NULL,
    C_PHONE CHAR(15) NOT NULL,
    C_ACCTBAL FLOAT NOT NULL,
    C_MKTSEGMENT CHAR(10) NOT NULL,
    C_COMMENT VARCHAR(117) NOT NULL WITH
DEFAULT)
    IN OTHER_STUFF
    INDEX IN OTHER_INDEX

```

```

PARTITIONING KEY(C_CUSTKEY) USING HASHING
ORGANIZE BY (
("C_NATIONKEY" ));

CREATE TABLE TPCD.ORDERS ( O_ORDERKEY    INTEGER NOT
NULL,
        O_CUSTKEY    INTEGER NOT NULL,
        O_ORDERSTATUS CHAR(1) NOT NULL,
        O_TOTALPRICE  FLOAT NOT NULL,
        O_ORDERDATE   DATE NOT NULL,
        O_ORDERPRIORITY CHAR(15) NOT NULL,
        O_CLERK       CHAR(15) NOT NULL,
        O_SHIPPRIORITY INTEGER NOT NULL,
        O_COMMENT     VARCHAR(79) NOT NULL WITH
DEFAULT
--          PRIMARY KEY (O_ORDERKEY)
)
IN OTHER_STUFF
INDEX IN OTHER_INDEX
PARTITIONING KEY(O_ORDERKEY) USING HASHING
ORGANIZE BY (( O_ORDERDATE ));

CREATE TABLE TPCD.LINEITEM ( L_ORDERKEY  INTEGER NOT
NULL,
        L_PARTKEY  INTEGER NOT NULL,
        L_SUPPKEY  INTEGER NOT NULL,
        L_LINENUMBER INTEGER NOT NULL,
        L_QUANTITY  FLOAT NOT NULL,
        L_EXTENDEDPRICE FLOAT NOT NULL,
        L_DISCOUNT  FLOAT NOT NULL,
        L_TAX        FLOAT NOT NULL,
        L_RETURNFLAG CHAR(1) NOT NULL,
        L_LINestatus CHAR(1) NOT NULL,
        L_SHIPDATE   DATE NOT NULL,
        L_COMMITDATE DATE NOT NULL,
        L_RECEIPTDATE DATE NOT NULL,
        L_SHIPINSTRUCT CHAR(25) NOT NULL,
        L_SHIPMODE    CHAR(10) NOT NULL,
        L_COMMENT     VARCHAR(44) NOT NULL WITH
DEFAULT
--          PRIMARY KEY (L_ORDERKEY, L_LINENUMBER)
)
IN LINEITEM_TABLE
INDEX IN LINEITEM_INDEXES
PARTITIONING KEY(L_ORDERKEY) USING HASHING
ORGANIZE BY ( (L_SHIPDATE));

```

COMMIT WORK;

create_tablespace

```

create regular tablespace small_tables
in nodegroup catalog_node
managed by system
using ('d:\small_tables')
on node (0)
overhead 1.60
transferrate 0.22;

```

```

create regular tablespace LINEITEM_TABLE
pagesize 16K
managed by database
using ( device 'd:\MP\li_data_D15' 17800M,
        device 'd:\MP\li_data_D21' 17800M) on node(0)
using ( device 'd:\MP\li_data_D16' 17800M,
        device 'd:\MP\li_data_D22' 17800M) on node(1)
using ( device 'd:\MP\li_data_D18' 17800M,
        device 'd:\MP\li_data_D24' 17800M) on node(2)
using ( device 'd:\MP\li_data_D19' 17800M,
        device 'd:\MP\li_data_D25' 17800M) on node(3)
using ( device 'd:\MP\li_data_D2' 17800M,
        device 'd:\MP\li_data_D9' 17800M) on node(4)
using ( device 'd:\MP\li_data_D3' 17800M,

```

```

        device 'd:\MP\li_data_D10' 17800M) on node(5)
using ( device 'd:\MP\li_data_D5' 17800M,
        device 'd:\MP\li_data_D12' 17800M) on node(6)
using ( device 'd:\MP\li_data_D6' 17800M,
        device 'd:\MP\li_data_D13' 17800M) on node(7)
bufferpool BP16K
extentsize 24
prefetchsize 48
overhead 3.0
transferrate 0.40;

```

```

create regular tablespace LINEITEM_INDEXES
pagesize 16K
managed by database
using ( device 'd:\MP\LI_INDEX_D15' 2900M,
        device 'd:\MP\LI_INDEX_D21' 2900M) on node(0)
using ( device 'd:\MP\LI_INDEX_D16' 2900M,
        device 'd:\MP\LI_INDEX_D22' 2900M) on node(1)
using ( device 'd:\MP\LI_INDEX_D18' 2900M,
        device 'd:\MP\LI_INDEX_D24' 2900M) on node(2)
using ( device 'd:\MP\LI_INDEX_D19' 2900M,
        device 'd:\MP\LI_INDEX_D25' 2900M) on node(3)
using ( device 'd:\MP\LI_INDEX_D2' 2900M,
        device 'd:\MP\LI_INDEX_D9' 2900M) on node(4)
using ( device 'd:\MP\LI_INDEX_D3' 2900M,
        device 'd:\MP\LI_INDEX_D10' 2900M) on node(5)
using ( device 'd:\MP\LI_INDEX_D5' 2900M,
        device 'd:\MP\LI_INDEX_D12' 2900M) on node(6)
using ( device 'd:\MP\LI_INDEX_D6' 2900M,
        device 'd:\MP\LI_INDEX_D13' 2900M) on node(7)
bufferpool BP16K
extentsize 24
prefetchsize 48
overhead 3.0
transferrate 0.40;

```

```

create regular tablespace OTHER_STUFF
pagesize 16K
managed by database
using ( device 'd:\MP\OTHER_TABLES_D15' 7900M,
        device 'd:\MP\OTHER_TABLES_D21' 7900M) on node(0)
using ( device 'd:\MP\OTHER_TABLES_D16' 7900M,
        device 'd:\MP\OTHER_TABLES_D22' 7900M) on node(1)
using ( device 'd:\MP\OTHER_TABLES_D18' 7900M,
        device 'd:\MP\OTHER_TABLES_D24' 7900M) on node(2)
using ( device 'd:\MP\OTHER_TABLES_D19' 7900M,
        device 'd:\MP\OTHER_TABLES_D25' 7900M) on node(3)
using ( device 'd:\MP\OTHER_TABLES_D2' 7900M,
        device 'd:\MP\OTHER_TABLES_D9' 7900M) on node(4)
using ( device 'd:\MP\OTHER_TABLES_D3' 7900M,
        device 'd:\MP\OTHER_TABLES_D10' 7900M) on node(5)
using ( device 'd:\MP\OTHER_TABLES_D5' 7900M,
        device 'd:\MP\OTHER_TABLES_D12' 7900M) on node(6)
using ( device 'd:\MP\OTHER_TABLES_D6' 7900M,
        device 'd:\MP\OTHER_TABLES_D13' 7900M) on node(7)
bufferpool BP16K
extentsize 24
prefetchsize 48
overhead 3.0
transferrate 0.40;

```

```

create regular tablespace OTHER_INDEX
pagesize 16K
managed by database
using ( device 'd:\MP\OTHER_INDEX_D15' 1990M,
        device 'd:\MP\OTHER_INDEX_D21' 1990M) on node(0)
using ( device 'd:\MP\OTHER_INDEX_D16' 1990M,
        device 'd:\MP\OTHER_INDEX_D22' 1990M) on node(1)
using ( device 'd:\MP\OTHER_INDEX_D18' 1990M,
        device 'd:\MP\OTHER_INDEX_D24' 1990M) on node(2)
using ( device 'd:\MP\OTHER_INDEX_D19' 1990M,
        device 'd:\MP\OTHER_INDEX_D25' 1990M) on node(3)
using ( device 'd:\MP\OTHER_INDEX_D2' 1990M,

```

```

        device 'd:\MP\OTHER_INDEX_D9' 1990M) on node(4)
using ( device 'd:\MP\OTHER_INDEX_D3' 1990M,
        device 'd:\MP\OTHER_INDEX_D10' 1990M) on node(5)
using ( device 'd:\MP\OTHER_INDEX_D5' 1990M,
        device 'd:\MP\OTHER_INDEX_D12' 1990M) on node(6)
using ( device 'd:\MP\OTHER_INDEX_D6' 1990M,
        device 'd:\MP\OTHER_INDEX_D13' 1990M) on node(7)
bufferpool BP16K
extentsize 24
prefetchsize 48
overhead 3.0
transferrate 0.40;

```

```

create temporary tablespace TEMP_TABLES
pagesize 16K
managed by database
using ( device 'd:\MP\temp_D15' 34900M,
        device 'd:\MP\temp_D21' 34900M) on node(0)
using ( device 'd:\MP\temp_D16' 34900M,
        device 'd:\MP\temp_D22' 34900M) on node(1)
using ( device 'd:\MP\temp_D18' 34900M,
        device 'd:\MP\temp_D24' 34900M) on node(2)
using ( device 'd:\MP\temp_D19' 34900M,
        device 'd:\MP\temp_D25' 34900M) on node(3)
using ( device 'd:\MP\temp_D2' 34900M,
        device 'd:\MP\temp_D9' 34900M) on node(4)
using ( device 'd:\MP\temp_D3' 34900M,
        device 'd:\MP\temp_D10' 34900M) on node(5)
using ( device 'd:\MP\temp_D5' 34900M,
        device 'd:\MP\temp_D12' 34900M) on node(6)
using ( device 'd:\MP\temp_D6' 34900M,
        device 'd:\MP\temp_D13' 34900M) on node(7)
bufferpool BP16K
extentsize 24
prefetchsize 48
overhead 3.0
transferrate 0.40;

```

```

create temporary tablespace TEMP2_TABLES
pagesize 4K
managed by database
using ( device 'd:\MP\temp2_D15' 34900M,
        device 'd:\MP\temp2_D21' 34900M) on node(0)
using ( device 'd:\MP\temp2_D16' 34900M,
        device 'd:\MP\temp2_D22' 34900M) on node(1)
using ( device 'd:\MP\temp2_D18' 34900M,
        device 'd:\MP\temp2_D24' 34900M) on node(2)
using ( device 'd:\MP\temp2_D19' 34900M,
        device 'd:\MP\temp2_D25' 34900M) on node(3)
using ( device 'd:\MP\temp2_D2' 34900M,
        device 'd:\MP\temp2_D9' 34900M) on node(4)
using ( device 'd:\MP\temp2_D3' 34900M,
        device 'd:\MP\temp2_D10' 34900M) on node(5)
using ( device 'd:\MP\temp2_D5' 34900M,
        device 'd:\MP\temp2_D12' 34900M) on node(6)
using ( device 'd:\MP\temp2_D6' 34900M,
        device 'd:\MP\temp2_D13' 34900M) on node(7)
bufferpool IBMDEFAULTBP
extentsize 96
prefetchsize 192
overhead 3.0
transferrate 0.40;

```

commit work;

createUFtbls

connect to tpcd;

```

drop table TPCDTEMP.ORDERS_NEW;
drop table TPCDTEMP.ORDERS_DEL;
drop table TPCDTEMP.LINEITEM_NEW;

```

commit;

```

CREATE TABLE TPCDTEMP.ORDERS_NEW ( APP_ID INTEGER NOT
NULL,

```

```

        O_ORDERKEY INTEGER NOT NULL,
        O_CUSTKEY INTEGER NOT NULL,
        O_ORDERSTATUS CHAR(1) NOT NULL,
        O_TOTALPRICE FLOAT NOT NULL,
        O_ORDERDATE DATE NOT NULL,
        O_ORDERPRIORITY CHAR(15) NOT NULL,
        O_CLERK CHAR(15) NOT NULL,
        O_SHIPPRIORITY INTEGER NOT NULL,
        O_COMMENT VARCHAR(79) NOT NULL WITH

```

```

DEFAULT)
IN OTHER_STUFF
INDEX IN OTHER_INDEX
PARTITIONING KEY(O_ORDERKEY) USING HASHING;

```

```

CREATE INDEX "TPCDTEMP"."I_ORDERS_NEW" ON
"TPCDTEMP"."ORDERS_NEW"
("APP_ID" ASC,
"O_ORDERKEY" ASC,
"O_CUSTKEY" ASC,
"O_ORDERSTATUS" ASC,
"O_TOTALPRICE" ASC,
"O_ORDERDATE" ASC,
"O_ORDERPRIORITY" ASC,
"O_CLERK" ASC,
"O_SHIPPRIORITY" ASC,
"O_COMMENT" ASC) PCTFREE 0;

```

```

CREATE TABLE TPCDTEMP.ORDERS_DEL ( APP_ID INTEGER NOT
NULL,

```

```

        O_ORDERKEY INTEGER NOT NULL)
IN OTHER_STUFF
INDEX IN OTHER_INDEX
PARTITIONING KEY(O_ORDERKEY) USING HASHING;

```

```

CREATE UNIQUE INDEX "TPCDTEMP"."I_ORDERS_DEL" ON
"TPCDTEMP"."ORDERS_DEL"
("APP_ID" ASC,
"O_ORDERKEY" ASC) PCTFREE 0;

```

```

CREATE TABLE TPCDTEMP.LINEITEM_NEW ( APP_ID INTEGER NOT
NULL,

```

```

        L_ORDERKEY INTEGER NOT NULL,
        L_PARTKEY INTEGER NOT NULL,
        L_SUPPKEY INTEGER NOT NULL,
        L_LINENUMBER INTEGER NOT NULL,
        L_QUANTITY FLOAT NOT NULL,
        L_EXTENDEDPRICE FLOAT NOT NULL,
        L_DISCOUNT FLOAT NOT NULL,
        L_TAX FLOAT NOT NULL,
        L_RETURNFLAG CHAR(1) NOT NULL,
        L_LINestatus CHAR(1) NOT NULL,
        L_SHIPDATE DATE NOT NULL,
        L_COMMITDATE DATE NOT NULL,
        L_RECEIPTDATE DATE NOT NULL,
        L_SHIPINSTRUCT CHAR(25) NOT NULL,
        L_SHIPMODE CHAR(10) NOT NULL,
        L_COMMENT VARCHAR(44) NOT NULL WITH

```

```

DEFAULT)
IN LINEITEM_TABLE
INDEX IN LINEITEM_INDEXES
PARTITIONING KEY(L_ORDERKEY);

```

```

CREATE INDEX "TPCDTEMP"."I_LINEITEM_NEW" ON
"TPCDTEMP"."LINEITEM_NEW"
("APP_ID" ASC) PCTFREE 0;

```

COMMIT WORK;

```
alter table tpcdtemp.orders_new locksize table;
alter table tpcdtemp.orders_del locksize table;
alter table tpcdtemp.lineitem_new locksize table;
```

```
COMMIT WORK;
```

```
connect reset;
```

dss.runstats

```
values (current timestamp, 'TS*** runstats nation START like ');
RUNSTATS ON TABLE TPCD.NATION WITH DISTRIBUTION on all
columns
and columns (
  n_name like statistics,
  n_comment like statistics )
AND INDEXES ALL;
commit;
values (current timestamp, 'TS*** runstats done nation ');
RUNSTATS ON TABLE TPCD.REGION WITH DISTRIBUTION on all
columns
and columns (
  r_name like statistics,
  r_comment like statistics )
AND INDEXES ALL;
commit;
RUNSTATS ON TABLE TPCD.SUPPLIER WITH DISTRIBUTION on all
columns
and columns (
  s_name like statistics,
  s_address like statistics,
  s_phone like statistics,
  s_comment like statistics )
AND INDEXES ALL;
commit;
values (current timestamp, 'TS*** runstats done part ');
RUNSTATS ON TABLE TPCD.PART WITH DISTRIBUTION on all
columns
and columns (
  p_name like statistics,
  p_mfgr like statistics,
  p_brand like statistics,
  p_type like statistics,
  p_container like statistics,
  p_comment like statistics )
AND INDEXES ALL;
commit;
values (current timestamp, 'TS*** runstats done partsupp ');
RUNSTATS ON TABLE TPCD.PARTSUPP WITH DISTRIBUTION on all
columns
and columns (
  ps_comment like statistics )
AND INDEXES ALL;
commit;
values (current timestamp, 'TS*** runstats done customer ');
RUNSTATS ON TABLE TPCD.CUSTOMER WITH DISTRIBUTION on all
columns
and columns (
  c_name like statistics,
  c_address like statistics,
  c_phone like statistics,
  c_mktsegment like statistics,
  c_comment like statistics )
AND INDEXES ALL;
commit;
values (current timestamp, 'TS*** runstats done orders ');
RUNSTATS ON TABLE TPCD.ORDERS WITH DISTRIBUTION on all
columns
and columns (
  o_orderstatus like statistics,
  o_orderpriority like statistics,
  o_clerk like statistics,
```

```
  o_comment like statistics)
AND INDEXES ALL;
commit;
values (current timestamp, 'TS*** runstats done lineitem ');
RUNSTATS ON TABLE TPCD.LINEITEM WITH DISTRIBUTION on all
columns
and columns (
  l_returnflag like statistics,
  l_linestatus like statistics,
  l_shipinstruct like statistics,
  l_shipmode like statistics,
  l_comment like statistics)
AND INDEXES ALL;
COMMIT WORK;
values (current timestamp, 'TS*** runstats END like');
```

load.db2set_8mIn.bat

```
db2set DB2_ANTIJOIN=Y
db2set DB2BPVARS=d:\tpch\ddl\scattered_read
db2set DB2OPTIONS="-t -v +c"
db2set DB2NTNOCACHE=ON
db2set DB2_PARALLEL_IO=*
db2set db2memmaxfree=1000000000
db2set DB2_EXTENDED_OPTIMIZATION=YES
```

load.dbcfg_8mIn

```
update database configuration for tpcd using
NUM_FREQVALUES 0
NUM_QUANTILES 300
buffpage 10000
catalogcache_sz 386
chngpgs_thresh 50
dbheap 6654
locklist 16384
logbufsz 2048
logfilsiz 16384
logprimary 20
logsecond 2
newlogpath e:\logs
maxappls 40
maxlocks 20
mincommit 1
num_iocleaners 4
num_ioservers 4
pckcachesz 320
softmax 1800
sortheap 50000
stat_heap_sz 16000
stmtheap 4096
util_heap_sz 128000
applheapsz 768
app_ctl_heap_sz 1024
dft_queryopt 7
dft_degree 1;
```

```
get database configuration for tpcd;
```

```
--connect reset;
```

load_dbmcfg_8mIn

```
update database manager configuration using
cpuspeed 7.951129e-7
comm_bandwidth 7
sheapthres 430000
agent_stack_sz 16
aslheapsz 15
rqrioblk 32767
intra_parallel NO
max_querydegree -1
```



```
maxagents 200
num_poolagents 4
num_initagents 4
diaglevel 3
svccname db2_db2_end;
```

get database manager configuration;

--connect reset;

load_all.sql

connect to tpcd;

```
values(current timestamp, 'TS*** Load Supplier Started ');
load from supplier.tbl.1,
    supplier.tbl.2,
    supplier.tbl.3,
    supplier.tbl.4,
    supplier.tbl.5,
    supplier.tbl.6,
    supplier.tbl.7,
    supplier.tbl.8 of del
MODIFIED BY COLDEL|
FASTPARSE ANYORDER MESSAGES d:\tmp0\supplier.msg
REPLACE INTO TPCD.SUPPLIER NONRECOVERABLE
partitioned db config mode load_only output_dbpartnums (0,1,2,3,4,5,6,7)
part_file_location w:\300GB_8mIn_flatfiles;
commit work;
```

```
values(current timestamp, 'TS*** Load done partsupp ');
load from orders.tbl.1,
    orders.tbl.2,
    orders.tbl.3,
    orders.tbl.4,
    orders.tbl.5,
    orders.tbl.6,
    orders.tbl.7,
    orders.tbl.8,
    orders.tbl.9 of del
MODIFIED BY COLDEL| FASTPARSE ANYORDER MESSAGES
d:\tmp0\orders.msg
REPLACE INTO TPCD.orders NONRECOVERABLE
partitioned db config mode load_only output_dbpartnums (0,1,2,3,4,5,6,7)
part_file_location w:\300GB_8mIn_flatfiles;
commit work;
```

```
values(current timestamp, 'TS*** Load done orders ');
load from lineitem.tbl.1,
    lineitem.tbl.2,
    lineitem.tbl.3,
    lineitem.tbl.4,
    lineitem.tbl.5,
    lineitem.tbl.6,
    lineitem.tbl.7,
    lineitem.tbl.8,
    lineitem.tbl.9,
    lineitem.tbl.10,
    lineitem.tbl.11,
    lineitem.tbl.12,
    lineitem.tbl.13,
    lineitem.tbl.14,
    lineitem.tbl.15,
    lineitem.tbl.16 of del
MODIFIED BY COLDEL| FASTPARSE ANYORDER MESSAGES
d:\tmp0\lineitem.msg
REPLACE INTO TPCD.lineitem NONRECOVERABLE
partitioned db config mode load_only output_dbpartnums (0,1,2,3,4,5,6,7)
part_file_location w:\300GB_8mIn_flatfiles;
commit work;
```

```
values(current timestamp, 'TS*** Load done supplier ');
load from customer.tbl.1,
```

```
customer.tbl.2,
customer.tbl.3,
customer.tbl.4,
customer.tbl.5,
customer.tbl.6,
customer.tbl.7,
customer.tbl.8 of del
MODIFIED BY COLDEL|
FASTPARSE ANYORDER MESSAGES d:\tmp0\customer.msg
REPLACE INTO TPCD.customer NONRECOVERABLE
partitioned db config mode load_only output_dbpartnums (0,1,2,3,4,5,6,7)
part_file_location w:\300GB_8mIn_flatfiles;
commit work;
```

```
values(current timestamp, 'TS*** Load done customer ');
load from part.tbl.1,
    part.tbl.2,
    part.tbl.3,
    part.tbl.4,
    part.tbl.5,
    part.tbl.6,
    part.tbl.7,
    part.tbl.8 of del
MODIFIED BY COLDEL| FASTPARSE ANYORDER MESSAGES
d:\tmp0\part.msg
REPLACE INTO TPCD.part NONRECOVERABLE
partitioned db config mode load_only output_dbpartnums (0,1,2,3,4,5,6,7)
part_file_location w:\300GB_8mIn_flatfiles;
commit work;
```

```
values(current timestamp, 'TS*** Load done part ');
load from partsupp.tbl.1,
    partsupp.tbl.2,
    partsupp.tbl.3,
    partsupp.tbl.4,
    partsupp.tbl.5,
    partsupp.tbl.6,
    partsupp.tbl.7,
    partsupp.tbl.8 of del
MODIFIED BY COLDEL| FASTPARSE ANYORDER MESSAGES
d:\tmp0\partsupp.msg
REPLACE INTO TPCD.partsupp NONRECOVERABLE
partitioned db config mode load_only output_dbpartnums (0,1,2,3,4,5,6,7)
part_file_location w:\300GB_8mIn_flatfiles;
commit work;
values(current timestamp, 'TS*** Load done lineitem ');
LOAD FROM w:\300GB_8mIn_flatfiles\region.tbl OF DEL
MODIFIED BY COLDEL| FASTPARSE ANYORDER MESSAGES
d:\tmp\region.msg
REPLACE INTO TPCD.REGION STATISTICS NO NONRECOVERABLE;
commit work;
```

```
values(current timestamp, 'TS*** Load done region ');
LOAD FROM w:\300GB_8mIn_flatfiles\nation.tbl OF DEL
MODIFIED BY COLDEL| FASTPARSE ANYORDER MESSAGES
d:\tmp\nation.msg
REPLACE INTO TPCD.NATION STATISTICS NO NONRECOVERABLE;
values(current timestamp, 'TS*** Load done nation ');
commit work;
connect reset;
```

loadcfg.sql

```
NUM_FREQVALUES 0
NUM_QUANTILES 300
buffpage 20000
catalogcache_sz 386
chnpgs_thresh 10
dbheap 10000
locklist 6000
logbufsz 4096
logfilsiz 16384
logprimary 10
```

```
logsecond 2
maxappls 8
maxlocks 20
mincommit 1
num_iocleaners 4
num_ioservers 4
pckcachesz 320
softmax 4000
sortheap 20000
stat_heap_sz 16000
stmthep 4096
util_heap_sz 32000
applheapsz 768
app_ctl_heap_sz 1024
dft_degree 2;
get database configuration for tpcd;
```

```
--connect reset;
```

load_8mIn.bat

```
cd \tpch\ddl
db2 connect to tpcd
db2 -tvf load_all.sql
db2 connect reset
cd \tpch\tools
```

run.db2set_8mIn.bat

```
db2set DB2_ANTIJOIN=Y
db2set DB2BPVARS=d:\tpch\ddl\scattered_read
db2set DB2OPTIONS="-t -v +c"
db2set DB2NTNOCACHE=ON
db2set DB2_PARALLEL_IO=*
db2set db2memmaxfree=1000000000
db2set DB2_EXTENDED_OPTIMIZATION=YES
```

run.dbcfg_8mIn

```
update database configuration for tpcd using
buffpage 50000
catalogcache_sz 386
dbheap 20000
locklist 16384
maxlocks 5
maxappls 40
mincommit 1
num_iocleaners 4
num_ioservers 4
DLCHKTIME 5000
pckcachesz 640
softmax 600
sortheap 6400
stat_heap_sz 10000
stmthep 10000
util_heap_sz 40000
applheapsz 16000
logbufsz 2048
app_ctl_heap_sz 2048
dft_degree 1
dft_queryopt 7
maxfilop 1024
chnpggs_thresh 15;
```

```
get database configuration for tpcd;
```

```
--connect reset;
```

run.dbmcfg_8mIn

```
update database manager configuration using
cpuspeed 7.951129e-7
comm_bandwidth 7
min_priv_mem 32
priv_mem_thresh 20000
sheapthres 250000
agent_stack_sz 16
aslheapsz 15
rqrioblk 32767
intra_parallel no
max_querydegree -1
maxagents 3600
num_poolagents 64
num_initagents 4
fcm_num_buffers 30000
numdb 1
svcname DB2_TPCD_END
diaglevel 0;
get database manager configuration;
```

runstats_UF.bat

```
db2start
db2 connect to tpcd
db2 runstats on table tpcdtemp.lineitem_new with distribution and detailed
indexes all
db2 runstats on table tpcdtemp.orders_new with distribution and detailed indexes
all
db2 runstats on table tpcdtemp.orders_del with distribution and detailed indexes
all
db2 commit
db2 connect reset
```

scattered_read

```
# 1/2 -> 1/3 of number of prefetchers
NUMPREFETCHQUEUES=2
PREFETCHQUEUESIZE=200
NUMHATESTACKS=4,*
```

```
# turn on scatter read for these types
NT_SCATTER_SMS=1
NT_SCATTER_DMSFILE=1
NT_SCATTER_DMSDEVICE=1
```

```
#diaglevels at which it writes configuration
LOG_CFG=4
LOG_PERF_HINTS=5
```

setlogs.bat

```
set db2node=0
db2start
db2 update db cfg for %1 using newlogpath i:\logs
set db2node=1
db2 update db cfg for %1 using newlogpath j:\logs
set db2node=2
db2 update db cfg for %1 using newlogpath k:\logs
set db2node=3
db2 update db cfg for %1 using newlogpath l:\logs
set db2node=4
db2 update db cfg for %1 using newlogpath e:\logs
set db2node=5
db2 update db cfg for %1 using newlogpath f:\logs
set db2node=6
db2 update db cfg for %1 using newlogpath g:\logs
set db2node=7
db2 update db cfg for %1 using newlogpath h:\logs
db2stop
set db2node=0
db2start
```

temp_4K.bat

```
db2start
db2 connect to tpcd
db2 alter bufferpool IBMDEFAULTBP size 100000
db2 connect reset
db2stop
db2start
db2 connect to tpcd
db2 drop tablespace userspace1
db2 commit
db2 drop tablespace tempspace1
db2 commit
db2 connect reset
db2stop
db2start
```

tpcd.setup

```
# NOTE: ALL variable defitions must have a comment at the end.
TPCD_PLATFORM=nt          # aix, nt, sun ....
TPCD_VERSION=2            # 1 or 2 (Version of tpcd). Default 1
TPCD_DBNAME=TPCD          # name to create database under
TPCD_WORKLOAD=H           # TPC version (R for TPCR, H for TPCB)
TPCD_AUDIT_DIR=d:\tpch    # top level directory of tar file for
                          # all the tpcd scripts
TPCD_PRODUCT=v5           # v5 or pe
                          # Use pe if you really are using pe v1.2!
                          # but I won't guarantee that it will work!
TPCD_MODE=mln             # uni/smp/mln/mpp
TPCD_PHYS_NODE=1          # number of physical nodes
TPCD_LN_PER_PN=8          # number of logical nodes per physical node
TPCD_SF=300               # size of the database (1=1GB,...) to
                          # get test size databases use:
                          # 0.012 = 12MB
                          # 0.1 = 100MB
TPCD_BUILD_STAGE=ALL      # where to start the build - currently the
                          # following is possible:
                          # ALL - do everything (create,load,
                          # index,stats,config) (Default)
                          # CRTTBSP - start after create db and
                          # config setting. Start righ at
                          # create tbps
                          # LOAD - start from the load of the tables
                          # INDEX - start from the index creation
                          # (NOTE if earlyindex is specified,
                          # then this will do the create index
                          # followed by the load...)
                          # RUNSTATS - start from the runstats
                          # (NOTE Do not use this option if
                          # distribution stats are gathered
                          # as part of the load, this will
                          # start after the load and indices
                          # have been created.
                          # CONFIG - start from the setting up of
                          # the benchmark runs config setup
                          #
TPCD_DBPATH=D:             # path for database (defaults to home)
TPCD_DDLPATH=d:\tpch\ddl  # path for all ddl files and customized
                          # scripts (load script), config files,etc
TPCD_BUFFERPOOL_DEF=create_bufferpools # name of file with bufferpool
definitions
                          # and sizes
TPCD_NODEGROUP_DEF=create_nodegroups # name of file in ddlpath with
nodegroup
                          # definitions
TPCD_EXPLAIN_DDL=NULL     # file with DDL for explains statments
                          # if this is NULL then uses the default
                          # and puts it in USERSPACE1 across all
                          # nodes...nt 1TB found it was faster if
                          # just in a single node nodegroup
TPCD_TBSP_DDL=create_tablespaces # ddl file for tablespaces
```

```
TPCD_DDL=create_tables    # ddl file for tables
TPCD_QUAL_TBSP_DDL=NULL   # ddl file for tablespaces for qual
TPCD_QUAL_DDL=NULL        # ddl file for qualification database
                          # tablespaces and tables should be identical
                          # to regular ddl except container names
TPCD_INDEXDDL=create_indexes # ddl file for indexes
TPCD_EXTRAINDEX=no        # no = no extra indexes
                          # filename = If you want to create some
                          # indices before
                          # the load, and some indices after, then
                          # use this additional file to specify the
TPCD_ADD_RI=NULL          # file name that contains any RI
                          # constraints to add after index creation
                          # set to NULL (default) if unused
                          # indices to create after the load.
TPCD_AST=NULL             # file name that contains complete AST
                          # definition including connection to
                          # the database, summary table creation,
                          # population, indexing and runstats.
TPCD_RUNSTATS=dss.runstats # ddl file for runstats. If you have
                          # created indices before the load (ie
                          # TPCD_EARLYINDEX=yes), have specified to
                          # gather stats on the load command (either
                          # through your own load script or by using
                          # TPCD_LOADSTATS=yes, AND you have
                          # specified a file for TPCD_EXTRAINDEX
                          # then this runstats file should include
                          # the runstats commands specifically for
                          # the extra indices.
TPCD_RUNSTATSHORT=NULL    # NOTE!! THIS IS BUGGY....I
can't get it to
                          # work on UNI successfully
                          # ddl file for short runstats that are
                          # run in the background while the
                          # TPCD_RUNSTATS are run in the foreground
                          # of the build. If this is used, then
                          # TPCD_RUNSTATS should have the runstats
                          # command for lineitem and
                          # TPCD_RUNSTATSHORT should have runstats
                          # commands for all other tables.
TPCD_DBGEN=\tpch\appendix.v2\dbgen # path name to data generation code
                          # Parameters used to specify source of
                          # data for load scripts
TPCD_INPUT=300GB_8mln_flatfiles # NULL - use dbgen generated data
OR
                          # path name - to the pre-generated
                          # flat files
                          # /gwl/dss/12MB - path for pregenerated 12MB
                          # /gwl/dss/100MB - path for pregen'd 100MB
                          #
TPCD_QUAL_INPUT=100GB_QUAL_FLATFILES # NULL - use dbgen
generated data OR
                          # path name - to the pre-generated
                          # flat files
TPCD_TAILOR_DIR=D:\tailor # path name for the directory used to
                          # generate split specific config files
                          # only used for partitioned environment
TPCD_EARLYINDEX=no        # create indexes before the load
                          #
                          # LOAD specific parameters follow:
TPCD_LOAD_DB2SET_SCRIPT=load.db2set_8mln.bat # Script that contains
the db2set commands
                          # for the load process Use NULL if not
                          # specified
TPCD_LOAD_CONFIGFILE=load.dbcfg_8mln # config file with specific
database config
                          # parms for the load/index/runstats part
                          # of the build.
                          # set to NULL if use defaults
TPCD_LOAD_DBM_CONFIGFILE=load.dbmcfg_8mln # config file with
specific
```

```

# database manager config parts for the
# load/index/runstats part of the build.
# set to NULL if use defaults
TPCD_LOAD_QUALCONFIGFILE=load.dbcfg_8mln # config file with specific
database config

# parms for the load/index/runstats part
# of the build for qualification db.
# set to NULL if use defaults
TPCD_LOAD_DBM_QUALCONFIGFILE=load.dbmcfg_8mln # config file with
specific

# database manager config parts for the
# load/index/runstats part of the build.
# set to NULL if use defaults
TPCD_LOADSTATS=NO # gather statistics during load
# ignored if EARLYINDEX is not set
# due to runstats limitation
TPCD_TEMP=d:\tmp # path for LOAD temp files
# defaults to /u/<instance>/sql/lib/tmp
# used in load script only
TPCD_SORTBUF=4096 # sortbuf size for LOAD
# used in load script only
TPCD_LOAD_PARALLELISM=0 # degree of parallelism to use on
load

# 0 = use the "intelligent default" that
# the load will chose at run time
# used in load script only
TPCD_COPY_DIR=NULL # directory where copy image is created
# on load command CURRENTLY UNUSED
# used in load script only
TPCD_FASTPARSE=yes # use fastparse on load
# used in load script only

# Backup and logfile specific parameters follow:
TPCD_BACKUP_DIR=\\backupdir # directory where backup files are
placed
TPCD_LOGPRIMARY=NULL # NULL/value = how many primary
log files

# to configure. If NULL is specified then
# the default is not changed.
TPCD_LOGFILSIZ=NULL # NULL/value = how 4KB pages to use
for

# logfilsiz db cfg parameter. If NULL is
# specified then the default is not changed
TPCD_LOGSECOND=NULL # NULL/value = how many secondary
log files

# to configure. If NULL is specified then
# the default is not changed.
TPCD_LOG_DIR=d:\logs # directory where log files stored..
# NULL leaves them in the dbpath
TPCD_LOG_QUAL_DIR=NULL # directory where qual log files
stored

# NULL leaves them in the dbpath
TPCD_LOG=YES # yes/no - whether to turn LOG_RETAIN on
# i.e. are backups needed to be taken

# CONFIG specific parameters
TPCD_DB2SET_SCRIPT=run.db2set_8mln.bat # Script that contains the db2set
commands

# for the benchmark run. Use NULL if not
# specified
TPCD_CONFIGFILE=run.dbcfg_8mln # name of configuration file in ddl
path

# that will be used for the benchmark run
TPCD_DBM_CONFIG=run.dbmcfg_8mln # name of config file for database
manager

# cfg parms
TPCD_QUALCONFIGFILE=run.dbcfg_8mln # name of database cfg file in
ddl path

# for qualification database
TPCD_DBM_QUALCONFIG=run.dbmcfg_8mln # name of config file for
database

# manager cfg parms

```

```

TPCD_MACHINE=NULL # set to NULL if using load config file
# big/medium/small size of machine used to
# determine buffpage, sortheap, sheapthres
# and ioservers parms for load, create
# index and runstats
# NOTE that this parameter is ignored if
# a TPCD_LOAD_CONFIGFILE

TPCD_SMPDEGREE=1 # 1...# of degrees of parallelism to run
# with
TPCD_AGENTPRI=NULL # set agentpri to this value (default
# is SYSTEM)
TPCD_ACTIVATE=yes # activate the database upon build
# completion
# run specific parameters
TPCD_AUDIT=yes # no/yes
# no - don't set up qualification db stuff
# yes - set up qualification db queries
# - build the update function tables
# and data before we get into the
# timing of the creation of the
# tables and the load.
TPCD_TMP_DIR=d:\tmp # place to put temp working files

TPCD_SHARED_TEMP_FULL_PATHNAME=d:\sql/lib/tmp # just added
TPCD_QUERY_TEMPLATE_DIR=standard.V2 # subdirectory in
AUDIT_DIR/queries

# to use as the source of the query
# templates. Currently there are
# v2 ones and pe ones. You can make
# your own directory following the same
# form as in the v2 directory using
# any variant you wish
TPCD_QUAL_DBNAME=TPCD # name of qualification database
TPCD_NUMSTREAM=6 # number of streams for the throughput
test
TPCD_FLATFILES=O:\300GB_8mln_UF_flatfiles # where to generate flat files
# for update functions
TPCD_STAGING_TABLE_DDL=createUFtables # script that contains the ddl
for creating

# the staging tables if they are used for
# the update functions
TPCD_PRELOAD_STAGING_TABLE_SCRIPT=LoadSampleUFData # File
containing

# the sql for preloading
# and gathering stats on sample UF data
# Note that the data used is sample data
# and is not data from any of the applied
# update pairs
TPCD_DELETE_STAGING_TABLE_SQL=DELETEDUMMYUFDATA.SQL #
file that contains the sql for deleting
# the preloaded data from the staging
# tables
TPCD_UPDATE_IMPORT=false # true = use import for the staging
tables

# for UNI/SMP mode only (code change in
# tpcdbatch) (if not uni mode then must
# change load_update)
# false = use load for staging tables
# The default is false if not set.
# NOTE that this parm is only for UNI/SMP
# it is not for multi node invocation

TPCD_SPLIT_UPDATES=256 # number of chunks to split the update
# function into.
TPCD_CONCURRENT_INSERTS=32 # number of insert chunks that are
run

# concurrently. TPCD_SPLIT_UPDATES
# should be evenly divisible by this number
TPCD_CONCURRENT_INSERTS_LOAD=16 # number of insert chunks
that are loaded

# concurrently. TPCD_SPLIT_UPDATES should

```

```

# be evenly divisible by this number.
# this controls the load portion of the
# insert routine for partitioned databases
TPCD_SPLIT_DELETES=256      # number of portions to split the delete
                             # function into.
                             # this variable is only valid in UNI/SMP
                             # mode.
TPCD_CONCURRENT_DELETES=32  # number of DELETE chunks
that are run
                             # concurrently. TPCD_SPLIT_UPDATES
                             # should be evenly divisible by this number
TPCD_GEN_UPDATEPAIRS=40     # number of pairs of update function
data
                             # to generate
                             # if 0 the update data generation and
                             # setup will not be done. use this if
                             # you don't want to run the update
                             # functions (Update functions not
                             # fully tested in new env't yet)
TPCD_GENERATE_SEED_FILE=yes  # yes/no These are the seed files
for
                             # generating the query substitution values
                             # yes - generate a seed file base on
                             #   year/month/day (for audited runs)
                             # no - use qgen's default seeds
TPCD_RUN_ON_MULTIPLE_NODES=NO # pe V1.2 only - will we be
running each
                             # query stream of throughput starting at
                             # different nodes or from same node
TPCD_STATS_INTERVAL=3       # timing interval for vmstats/iostats
TPCD_STATS_THRU_INT=300     # timing interval for vmstats/iostats
for
                             # throughput run
TPCD_GATHER_STATS=off       # on/off - only implement for AIX yet
                             # on = gather statistics around power
                             #   test run (vmstat,iostat,netstat)
                             # off = no stats gathered during power run

TPCD_UFTEMP=UFTEMP         # base name of tablespace(s) where the
                             # staging tables for the update functions
                             # are created
                             # this name will be used as the
                             # basename for the tablespaces...eg
                             # UFTEMP1 UFTEMP2 ....
TPCD_HAVECOMPILER=yes      # rebuild tpcdbatch executable
                             # yes/no
TPCD_SLEEP=5               # ?
TPCD_INLISTMAX=default      # max num of keys to delete at a time
                             # for UF2, use "default" for default.
TPCD_LOAD_SCRIPT=load_8mln.bat # script to run for loading tables
                             # in TPCD_DDLPATH directory under mln/mpp
                             # leave as NULL if using default genloaduni
TPCD_LOAD_SCRIPT_QUAL=NULL  # script to run for loading tables
in
                             # TPCD_DDLPATH directory under mln/mpp
                             # for QUAL db
TPCD_ROOTPRIV=no           # do you have root privileges to be able
                             # get values of things like schedtune
                             # and vmtune (currently on AIX only)
                             # acid test specific information
TPCD_DB2LOG=d:\sqlib\db2    # directory where the db2diag.log can
                             # be found for the durability tests
TPCD_APPEND_ON=no          # set to no if the cluster indexes are used

```

Qualification Database

2mln_qual_load.sql

```

LOAD FROM w:\1GB_qual_flatfiles\region.tbl OF DEL
MODIFIED BY COLDEL| FASTPARSE
MESSAGES d:\tmpqual\region.msg

```

```

REPLACE INTO TPCD.REGION
STATISTICS NO
NONRECOVERABLE;
commit work;
values(current timestamp);

LOAD FROM w:\1GB_qual_flatfiles\nation.tbl OF DEL
MODIFIED BY COLDEL| FASTPARSE
MESSAGES d:\tmpqual\nation.msg
REPLACE INTO TPCD.NATION
STATISTICS NO
NONRECOVERABLE;
commit work;
values(current timestamp);

load from
    orders.1,
    orders.2
  of DEL MODIFIED BY COLDEL| FASTPARSE
MESSAGES d:\tmpqual\orders.msg
REPLACE INTO TPCD.orders STATISTICS no NONRECOVERABLE
partitioned db config mode load_only
output_dbpartnums (0,1)
part_file_location w:\1GB_qual_flatfiles;
commit work;

values(current timestamp);
load from
    supplier.1,
    supplier.2
  of DEL MODIFIED BY COLDEL| FASTPARSE
MESSAGES d:\tmpqual\supplier.msg
REPLACE INTO TPCD.SUPPLIER STATISTICS no
NONRECOVERABLE
partitioned db config mode load_only
output_dbpartnums (0,1)
part_file_location w:\1GB_qual_flatfiles;
commit work;

values(current timestamp);
load from
    customer.1,
    customer.2
  of DEL MODIFIED BY COLDEL| FASTPARSE
MESSAGES d:\tmpqual\customer.msg
REPLACE INTO TPCD.customer STATISTICS no NONRECOVERABLE
partitioned db config mode load_only
output_dbpartnums (0,1)
part_file_location w:\1GB_qual_flatfiles;
commit work;

values(current timestamp);
load from
    part.1,
    part.2
  of DEL MODIFIED BY COLDEL| FASTPARSE
MESSAGES d:\tmpqual\part.msg
REPLACE INTO TPCD.part STATISTICS no NONRECOVERABLE
partitioned db config mode load_only
output_dbpartnums (0,1)
part_file_location w:\1GB_qual_flatfiles;
commit work;

values(current timestamp);
load from
    partsupp.1,
    partsupp.2
  of DEL MODIFIED BY COLDEL| FASTPARSE
MESSAGES d:\tmpqual\partsupp.msg
REPLACE INTO TPCD.partsupp STATISTICS no NONRECOVERABLE
partitioned db config mode load_only
output_dbpartnums (0,1)
part_file_location w:\1GB_qual_flatfiles;
commit work;

```

```

values(current timestamp);
load from
    lineitem.1,
    lineitem.2
of DEL MODIFIED BY COLDEL| FASTPARSE
MESSAGES d:\tmpqual\lineitem.msg
REPLACE INTO TPCD.lineitem STATISTICS no NONRECOVERABLE
partitioned db config mode load_only
output_dbpartnums (0,1)
part_file_location w:\1GB_qual_flatfiles;
commit work;

```

```
values(current timestamp);
```

affinity_2mIn.bat

```

rem Set affinities for LN0 -> LN7 for 8 processor system with hyperthreading.
db2set db2processors=0,1,2,3,8,9,10,11 -i QUAL 0
db2set db2processors=4,5,6,7,12,13,14,15 -i QUAL 1

```

bkupqual2.bat

```

time < xxx
rem Start of backup1 node 0
db2start
set db2node=0
rd o:\backupqual /s /q
md o:\backupqual
db2 backup database tpcd to o:\backupqual WITH 16 BUFFERS
PARALLELISM 8 without prompting
rem End of backup1 node 0
set db2node=1
rem Start of backup1 node 1
db2 backup database tpcd to o:\backupqual WITH 16 BUFFERS
PARALLELISM 8 without prompting
time < xxx
rem End of backup1 node 1
time < xxx

```

clrqlogs.bat

```

i:
rd quallogs /s /q
md quallogs
j:
rd quallogs /s /q
md quallogs
d:

```

create_mIn_instance_qual.bat

```

REM set db2instance=%1 in all db2cmd windows...
db2icrt QUAL /s ese /u:tpch,tpch1tpch
rem edit services file
pause
set db2instance=qual
db2nrcrt /n:1 /u:tpch,tpch1tpch /i:QUAL /p:1 /o:relentless1
db2nlist
db2ilist

```

create_tablespaces_2mIn_qual

```

create regular tablespace small_tables
in nodegroup catalog_node
managed by system
using ('d:\small_tables_qual')
on node (0)
overhead 3.0
transferrate 0.40;

```

```

create regular tablespace LINEITEM_TABLE
pagesize 16K
managed by database
using ( device 'd:\QMP\li_data_D15' 990M,
        device 'd:\QMP\li_data_D21' 990M) on node(0)
using ( device 'd:\QMP\li_data_D16' 990M,
        device 'd:\QMP\li_data_D22' 990M) on node(1)
bufferpool BP16K
extentsize 24
prefetchsize 48
overhead 3.0
transferrate 0.40;

```

```

create regular tablespace LINEITEM_INDEXES
pagesize 16K
managed by database
using ( device 'd:\QMP\LI_INDEX_D15' 200M,
        device 'd:\QMP\LI_INDEX_D21' 200M) on node(0)
using ( device 'd:\QMP\LI_INDEX_D16' 200M,
        device 'd:\QMP\LI_INDEX_D22' 200M) on node(1)
bufferpool BP16K
extentsize 24
prefetchsize 48
overhead 3.0
transferrate 0.40;

```

```

create regular tablespace OTHER_STUFF
pagesize 16K
managed by database
using ( device 'd:\QMP\OTHER_TABLES_D15' 780M,
        device 'd:\QMP\OTHER_TABLES_D21' 780M) on node(0)
using ( device 'd:\QMP\OTHER_TABLES_D16' 780M,
        device 'd:\QMP\OTHER_TABLES_D22' 780M) on node(1)
bufferpool BP16K
extentsize 24
prefetchsize 48
overhead 3.0
transferrate 0.40;

```

```

create regular tablespace OTHER_INDEX
pagesize 16K
managed by database
using ( device 'd:\QMP\OTHER_INDEX_D15' 200M,
        device 'd:\QMP\OTHER_INDEX_D21' 200M) on node(0)
using ( device 'd:\QMP\OTHER_INDEX_D16' 200M,
        device 'd:\QMP\OTHER_INDEX_D22' 200M) on node(1)
bufferpool BP16K
extentsize 24
prefetchsize 48
overhead 3.0
transferrate 0.40;

```

```

create temporary tablespace TEMP_TABLES
pagesize 16K
managed by database
using ( device 'd:\QMP\temp_D15' 1500M,
        device 'd:\QMP\temp_D21' 1500M) on node(0)
using ( device 'd:\QMP\temp_D16' 1500M,
        device 'd:\QMP\temp_D22' 1500M) on node(1)
bufferpool BP16K
extentsize 24
prefetchsize 48
overhead 3.0
transferrate 0.40;

```

```

create temporary tablespace TEMP2_TABLES
pagesize 4K
managed by database
using ( device 'd:\QMP\temp2_D15' 1500M,
        device 'd:\QMP\temp2_D21' 1500M) on node(0)
using ( device 'd:\QMP\temp2_D16' 1500M,
        device 'd:\QMP\temp2_D22' 1500M) on node(1)
bufferpool IBMDEFAULTBP

```

```

extentsize 96
prefetchsize 192
overhead 3.0
transferrate 0.40;

```

commit work;

go_qual.bat

```

db2stop
rah killdb2

```

set db2instance=QUAL

REM Cleanup and rebuild the qualification DB. (30 Mins)

```

call d:\tpch_qual\ddl\clrqlogs.bat
db2start
db2 drop db tpcd
db2stop
cd d:\tpch_qual\tools
perl setupdir
perl buildtpcd QUAL 2>&1 |tee buildqual.log
perl buildtpcdbatch
db2stop
db2start

```

REM Database is build start the run of the qualification queries.

```

perl runqual
db2stop
db2start
cd \tpch_qual\auditexe\acid

```

REM compile and run the acid tests.

```

db2start
perl setup.pl
db2stop
db2start
db2 connect to tpcd
call nt_compile TPCD
db2 connect reset
db2stop
perl runaci.pl 2>&1 |tee runaci.log

```

load_2mln_qual.bat

```

cd \tpch_qual\ddl
db2 connect to tpcd
db2 -tvf 2mln_qual_load.sql
db2 connect reset
cd \tpch_qual\tools

```

load_dbcfg_qual

update database configuration for tpcd using

```

NUM_FREQVALUES 0
NUM_QUANTILES 300
buffpage 10000
catalogcache_sz 386
chnngpgs_thresh 50
dbheap 6654
locklist 16384
logbufsz 2048
logfilsiz 16384
logprimary 20
logsecond 2
newlogpath i:\quallogs
maxappls 40
maxlocks 20
mincommit 1
num_iocleaners 4

```

```

num_ioservers 4
pckcachesz 320
softmax 1800
sortheap 50000
stat_heap_sz 16000
stmtheap 4096
util_heap_sz 128000
applheapsz 768
app_ctl_heap_sz 1024
dft_queryopt 7
dft_degree 1;

```

get database configuration for tpcd;

--connect reset;

load_qual.bat

```

cd \tpch_qual\ddl
db2 connect to tpcd
db2 -tvf 2mln_qual_load.sql
db2 connect reset
cd \tpch_qual\tools

```

qual_untag.bat

```

D:\sqllib\bin\db2untag -f d:\QMP\li_data_D8
D:\sqllib\bin\db2untag -f d:\QMP\li_data_D9
D:\sqllib\bin\db2untag -f d:\QMP\li_data_D11
D:\sqllib\bin\db2untag -f d:\QMP\li_data_D12
D:\sqllib\bin\db2untag -f d:\QMP\LI_INDEX_D8
D:\sqllib\bin\db2untag -f d:\QMP\LI_INDEX_D9
D:\sqllib\bin\db2untag -f d:\QMP\LI_INDEX_D11
D:\sqllib\bin\db2untag -f d:\QMP\LI_INDEX_D12
D:\sqllib\bin\db2untag -f d:\QMP\OTHER_TABLES_D8
D:\sqllib\bin\db2untag -f d:\QMP\OTHER_TABLES_D9
D:\sqllib\bin\db2untag -f d:\QMP\OTHER_TABLES_D11
D:\sqllib\bin\db2untag -f d:\QMP\OTHER_TABLES_D12
D:\sqllib\bin\db2untag -f d:\QMP\OTHER_INDEX_D8
D:\sqllib\bin\db2untag -f d:\QMP\OTHER_INDEX_D9
D:\sqllib\bin\db2untag -f d:\QMP\OTHER_INDEX_D11
D:\sqllib\bin\db2untag -f d:\QMP\OTHER_INDEX_D12
D:\sqllib\bin\db2untag -f d:\QMP\temp_D8
D:\sqllib\bin\db2untag -f d:\QMP\temp_D9
D:\sqllib\bin\db2untag -f d:\QMP\temp_D11
D:\sqllib\bin\db2untag -f d:\QMP\temp_D12

```

run.db2set.qual.bat

```

db2set DB2_ANTIJOIN=Y
db2set DB2BPVARS=d:\tpch\ddl\scattered_read
db2set DB2OPTIONS="-t -v +c"
db2set DB2NTNOCACHE=ON
db2set DB2_PARALLEL_IO=*
db2set db2memmaxfree=1000000000
db2set DB2_EXTENDED_OPTIMIZATION=YES

```

run.dbcfg_qual

update database configuration for tpcd using

```

buffpage 10000
catalogcache_sz 386
dbheap 20000
locklist 16384
maxlocks 5
maxappls 40
mincommit 1
num_iocleaners 4
num_ioservers 2
DLCHKTIME 5000
pckcachesz 640
softmax 600

```

```

sortheap 6400
stat_heap_sz 10000
stmtheap 10000
util_heap_sz 40000
applheapsz 16000
logbufsz 2048
app_ctl_heap_sz 2048
dft_degree 1
dft_queryopt 7
maxfilop 1024
chngpgs_thresh 15;

```

get database configuration for tpcd;

--connect reset;

run.dbmcfg_qual

update database configuration for tpcd using

```

buffpage 10000
catalogcache_sz 386
dbheap 20000
locklist 16384
maxlocks 5
maxappls 40
mincommit 1
num_iocleaners 4
num_ioservers 2
DLCHKTIME 5000
pckcachesz 640
softmax 600
sortheap 6400
stat_heap_sz 10000
stmtheap 10000
util_heap_sz 40000
applheapsz 16000
logbufsz 2048
app_ctl_heap_sz 2048
dft_degree 1
dft_queryopt 7
maxfilop 1024
chngpgs_thresh 15;

```

get database configuration for tpcd;

--connect reset;

setlogs_qual.bat

```

set db2node=0
db2start
db2 update db cfg for tpcd using newlogpath i:\quallogs
db2stop
set db2node=1
db2start
db2 update db cfg for tpcd using newlogpath j:\quallogs
db2stop
set db2node=0
db2start

```

temp_4K.bat

```

db2start
db2 connect to tpcd
db2 alter bufferpool IBMDEFAULTBP size 20000
db2 connect reset
db2stop
db2start
db2 connect to tpcd
db2 drop tablespace userspace1
db2 commit
db2 drop tablespace tempspace1

```

```

db2 commit
db2 connect reset
db2stop
db2start

```

tpcd.setup

NOTE: ALL variable definitions must have a comment at the end.

```

TPCD_PLATFORM=nt # aix, nt, sun ....
TPCD_VERSION=2 # 1 or 2 (Version of tpcd). Default 1
TPCD_DBNAME=TPCD # name to create database under
TPCD_WORKLOAD=H # TPC version (R for TPCR, H for Tpch)
TPCD_AUDIT_DIR=d:\tpch_qual # top level directory of tar file for
# all the tpcd scripts
TPCD_PRODUCT=v5 # v5 or pe
# Use pe if you really are using pe v1.2!
# but I won't guarantee that it will work!
TPCD_MODE=mln # uni/smp/mln/mpp
TPCD_PHYS_NODE=1 # number of physical nodes
TPCD_LN_PER_PN=2 # number of logical nodes per physical node
TPCD_SF=1 # size of the database (1=1GB,...) to
# get test size databases use:
# 0.012 = 12MB
# 0.1 = 100MB
TPCD_BUILD_STAGE=ALL # where to start the build - currently the
# following is possible:
# ALL - do everything (create,load,
# index,stats,config) (Default)
# CRTTBSP - start after create db and
# config setting. Start righ at
# create tbsp
# LOAD - start from the load of the tables
# INDEX - start from the index creation
# (NOTE if earlyindex is specified,
# then this will do the create index
# followed by the load...)
# RUNSTATS - start from the runstats
# (NOTE Do not use this option if
# distribution stats are gathered
# as part of the load, this will
# start after the load and indices
# have been created.
# CONFIG - start from the setting up of
# the benchmark runs config setup
#
TPCD_DBPATH=D: # path for database (defaults to home)
TPCD_DDLPATH=d:\tpch_qual\ddl # path for all ddl files and customized
# scripts (load script), config files,etc
TPCD_BUFFERPOOL_DEF=create_bufferpools # name of file with bufferpool
definitions
# and sizes
TPCD_NODEGROUP_DEF=create_nodegroups # name of file in ddlpath with
nodegroup
# definitions
TPCD_EXPLAIN_DDL=NULL # file with DDL for explains statments
# if this is NULL then uses the default
# and puts it in USERSPACE1 across all
# nodes...nt 1TB found it was faster if
# just in a single node nodegroup
TPCD_TBSP_DDL=create_tablespace_2mln_qual # ddl file for tablespaces
TPCD_DDL=create_tables # ddl file for tables
TPCD_QUAL_TBSP_DDL=create_tablespace_2mln_qual # ddl file for
tablespaces for
TPCD_QUAL_DDL=create_tables # ddl file for qualification database
# tablespaces and tables should be identical
# to regular ddl except container names
TPCD_INDEXDDL=create_indexes # ddl file for indexes
TPCD_EXTRAINDEX=no # no = no extra indexes
# filename = If you want to create some
# indices before
# the load, and some indices after, then
# use this additional file to specify the

```



```

TPCD_ADD_RI=NULL          # file name that contains any RI
                          # constraints to add after index creation
                          # set to NULL (default) if unused
                          # indices to create after the load.

TPCD_AST=NULL            # file name that contains complete AST
                          # definition including connection to
                          # the database, summary table creation,
                          # population, indexing and runstats.

TPCD_RUNSTATS=dss.runstats # ddl file for runstats. If you have
                          # created indices before the load (ie
                          # TPCD_EARLYINDEX=yes), have specified to
                          # gather stats on the load command (either
                          # through your own load script or by using
                          # TPCD_LOADSTATS=yes, AND you have
                          # specified a file for TPCD_EXTRAINDEX
                          # then this runstats file should include
                          # the runstats commands specifically for
                          # the extra indices.

TPCD_RUNSTATSHORT=NULL   # NOTE!! THIS IS BUGGY....I
can't get it to

                          # work on UNI successfully
                          # ddl file for short runstats that are
                          # run in the background while the
                          # TPCD_RUNSTATS are run in the foreground
                          # of the build. If this is used, then
                          # TPCD_RUNSTATS should have the runstats
                          # command for lineitem and
                          # TPCD_RUNSTATSHORT should have runstats
                          # commands for all other tables.

TPCD_DBGEN=\tpch_qual\appendix.v2\dbgen # path name to data generation
code

                          # Parameters used to specify source of
                          # data for load scripts

TPCD_INPUT=100GB_8mnl_flatfiles # NULL - use dbgen generated data
OR

                          # path name - to the pre-generated
                          # flat files
                          # /gw1/dss/12MB - path for pregenerated 12MB
                          # /gw1/dss/100MB - path for pregen'd 100MB
                          #

TPCD_QUAL_INPUT=\1GB_QUAL_FLATFILES # NULL - use dbgen
generated data OR

                          # path name - to the pre-generated
                          # flat files

TPCD_TAILOR_DIR=D:\tailor # path name for the directory used to
                          # generate split specific config files
                          # only used for partitioned environment

TPCD_EARLYINDEX=no       # create indexes before the load

                          # LOAD specific parameters follow:
TPCD_LOAD_DB2SET_SCRIPT=load.db2set_8mnl.bat # Script that contains
the db2set commands

                          # for the load process Use NULL if not
                          # specified

TPCD_LOAD_CONFIGFILE=load.dbcfg_8mnl # config file with specific
database config

                          # parms for the load/index/runstats part
                          # of the build.
                          # set to NULL if use defaults

TPCD_LOAD_DBM_CONFIGFILE=load.dbmcfg_8mnl # config file with
specific

                          # database manager config parts for the
                          # load/index/runstats part of the build.
                          # set to NULL if use defaults

TPCD_LOAD_QUALCONFIGFILE=load_dbcfg_qual # config file with specific
database config

                          # parms for the load/index/runstats part
                          # of the build for qualification db.
                          # set to NULL if use defaults

TPCD_LOAD_DBM_QUALCONFIGFILE=load.dbmcfg_8mnl # config file with
specific

```

```

                          # database manager config parts for the
                          # load/index/runstats part of the build.
                          # set to NULL if use defaults

TPCD_LOADSTATS=NO        # gather statistics during load
                          # ignored if EARLYINDEX is not set
                          # due to runstats limitation

TPCD_TEMP=d:\tmp         # path for LOAD temp files
                          # defaults to /u/<instance>/sql/lib/tmp
                          # used in load script only

TPCD_SORTBUF=4096        # sortbuf size for LOAD
                          # used in load script only

TPCD_LOAD_PARALLELISM=0  # degree of parallelism to use on
load

                          # 0 = use the "intelligent default" that
                          # the load will chose at run time
                          # used in load script only

TPCD_COPY_DIR=NULL       # directory where copy image is created
                          # on load command CURRENTLY UNUSED
                          # used in load script only

TPCD_FASTPARSE=yes       # use fastparse on load
                          # used in load script only

                          # Backup and logfile specific parameters follow:
TPCD_BACKUP_DIR=o:\backupqual # directory where backup files are
placed

TPCD_LOGPRIMARY=NULL     # NULL/value = how many primary
log files

                          # to configure. If NULL is specified then
                          # the default is not changed.

TPCD_LOGFILSIZ=NULL      # NULL/value = how 4KB pages to use
for

                          # logfilsiz db cfg parameter. If NULL is
                          # specified then the default is not changed

TPCD_LOGSECOND=NULL      # NULL/value = how many secondary
log files

                          # to configure. If NULL is specified then
                          # the default is not changed.

TPCD_LOG_DIR=d:\logs     # directory where log files stored..
                          # NULL leaves them in the dbpath

TPCD_LOG_QUAL_DIR=NULL   # directory where qual log files
stored

                          # NULL leaves them in the dbpath

TPCD_LOG=YES             # yes/no - whether to turn LOG_RETAIN on
                          # i.e. are backups needed to be taken

                          # CONFIG specific parameters

TPCD_DB2SET_SCRIPT=run.db2set.qual # Script that contains the db2set
commands

                          # for the benchmark run. Use NULL if not
                          # specified

TPCD_CONFIGFILE=run.dbcfg_8mnl # name of configuration file in ddl
path

                          # that will be used for the benchmark run

TPCD_DBM_CONFIG=run.dbmcfg_8mnl # name of config file for database
manager

                          # cfg parms

TPCD_QUALCONFIGFILE=run.dbcfg_qual # name of database cfg file in ddl
path

                          # for qualification database

TPCD_DBM_QUALCONFIG=run.dbmcfg_qual # name of config file for
database

                          # manager cfg parms

TPCD_MACHINE=NULL        # set to NULL if using load config file
                          # big/medium/small size of machine used to
                          # determine buffpage, sortheap, sheapthres
                          # and ioservers parms for load, create
                          # index and runstats
                          # NOTE that this parameter is ignored if
                          # a TPCD_LOAD_CONFIGFILE

TPCD_SMPDEGREE=1         # 1...# of degrees of parallelism to run
                          # with

```

<pre> TPCD_AGENTPRI=NULL # set agentpri to this value (default # is SYSTEM) TPCD_ACTIVATE=yes # activate the database upon build # completion # run specific parameters TPCD_AUDIT=yes # no/yes # no - don't set up qualification db stuff # yes - set up qualification db queries # - build the update function tables # and data before we get into the # timing of the creation of the # tables and the load. TPCD_TMP_DIR=d:\tmp # place to put temp working files TPCD_SHARED_TEMP_FULL_PATHNAME=c:\sqllib\tmp # just added TPCD_QUERY_TEMPLATE_DIR=standard.V2 # subdirectory in AUDIT_DIR/queries # to use as the source of the query # templates. Currently there are # v2 ones and pe ones. You can make # your own directory following the same # form as in the v2 directory using # any variant you wish TPCD_QUAL_DBNAME=TPCD # name of qualification database TPCD_NUMSTREAM=6 # number of streams for the throughput test TPCD_FLATFILES=1GB_qual_flatfiles # where to generate flat files # for update functions TPCD_STAGING_TABLE_DDL=NULL # script that contains the ddl for creating # the staging tables if they are used for # the update functions. Set to NULL for Qual DB. TPCD_PRELOAD_STAGING_TABLE_SCRIPT=LoadSampleUFData # File containing # the sql for preloading # and gathering stats on sample UF data # Note that the data used is sample data # and is not data from any of the applied # update pairs TPCD_DELETE_STAGING_TABLE_SQL=DELETEDUMMYUFDATA.SQL # file that contains the sql for deleting # the preloaded data from the staging # tables TPCD_UPDATE_IMPORT=false # true = use import for the staging tables # for UNI/SMP mode only (code change in # tpcdbatch) (if not uni mode then must # change load_update) # false = use load for staging tables # The default is false if not set. # NOTE that this parm is only for UNI/SMP # it is not for multi node invocation TPCD_SPLIT_UPDATES=32 # number of chunks to split the update # function into. TPCD_CONCURRENT_INSERTS=16 # number of insert chunks that are run # concurrently. TPCD_SPLIT_UPDATES # should be evenly divisible by this number TPCD_CONCURRENT_INSERTS_LOAD=4 # number of insert chunks that are loaded # concurrently. TPCD_SPLIT_UPDATES should # be evenly divisible by this number. # this controls the load portion of the # insert routine for partitioned databases TPCD_SPLIT_DELETES=32 # number of portions to split the delete # function into. # this variable is only valid in UNI/SMP # mode. TPCD_CONCURRENT_DELETES=16 # number of DELETE chunks that are run # concurrently. TPCD_SPLIT_UPDATES </pre>	<pre> # should be evenly divisible by this number TPCD_GEN_UPDATEPAIRS=30 # number of pairs of update function data # to generate # if 0 the update data generation and # setup will not be done. use this if # you don't want to run the update # functions (Update functions not # fully tested in new env't yet) TPCD_GENERATE_SEED_FILE=yes # yes/no These are the seed files for # generating the query substitution values # yes - generate a seed file base on # year/month/day (for audited runs) # no - use qgen's default seeds TPCD_RUN_ON_MULTIPLE_NODES=NO # pe V1.2 only - will we be running each # query stream of throughput starting at # different nodes or from same node TPCD_STATS_INTERVAL=5 # timing interval for vmstats/iostats TPCD_STATS_THRU_INT=300 # timing interval for vmstats/iostats for # throughput run TPCD_GATHER_STATS=off # on/off - only implement for AIX yet # on = gather statistics around power # test run (vmstat,iostat,netstat) # off = no stats gathered during power run TPCD_UFTEMP=UFTEMP # base name of tablespace(s) where the # staging tables for the update functions # are created # this name will be used as the # basename for the tablespaces...eg # UFTEMP1 UFTEMP2 TPCD_HAVECOMPILER=yes # rebuild tpcdbatch executable # yes/no TPCD_SLEEP=5 # ? TPCD_INLISTMAX=default # max num of keys to delete at a time # for UF2, use "default" for default. TPCD_LOAD_SCRIPT=load_8mln.bat # script to run for loading tables # in TPCD_DDLPATH directory under mln/mpp # leave as NULL if using default genloaduni TPCD_LOAD_SCRIPT_QUAL=LOAD_QUAL.BAT # script to run for loading tables in # TPCD_DDLPATH directory under mln/mpp # for QUAL db TPCD_ROOTPRIV=no # do you have root privileges to be able # get values of things like schedtune # and vmtune (currently on AIX only) # acid test specific information TPCD_DB2LOG=d:\sqlib\qual # directory where the db2diag.log can # be found for the durability tests TPCD_APPEND_ON=no # set to no if the cluster indexes are used </pre>
--	--

Appendix C: Qualification Query Output

Qualification Queries

Query 1

Start timestamp 05/11/04 14:33:07.078

-- Query 01 - Var_0 Rev_01 - Pricing Summary Report Query

Tag: Q1 Stream: -1 Sequence number: 17

select l_returnflag, l_linestatus, sum(l_quantity) as sum_qty, sum(l_extendedprice) as sum_base_price, sum(l_extendedprice * (1 - l_discount)) as sum_disc_price, sum(l_extendedprice * (1 - l_discount) * (1 + l_tax)) as sum_charge, avg(l_quantity) as avg_qty, avg(l_extendedprice) as avg_price, avg(l_discount) as avg_disc, count(*) as count_order from tpcd.lineitem where l_shipdate <= date ('1998-12-01') - 90 day group by l_returnflag, l_linestatus order by l_returnflag, l_linestatus				
L_RETURNFLAG	L_LINESTATUS	SUM_QTY	SUM_BASE_PRICE	
SUM_DISC_PRICE	SUM_CHARGE	AVG_QTY		
AVG_PRICE	AVG_DISC	COUNT_ORDER		

A	F	37734107.000	56586554400.731	
53758257134.871		55909065222.828	25.522	38273.130
0.050	1478493			
N	F	991417.000	1487504710.380	1413082168.054
1469649223.194		25.516	38284.468	0.050
38854				
N	O	74476040.000	111701729697.739	
106118230307.604		110367043872.496	25.502	38249.118
0.050	2920374			
R	F	37719753.000	56568041380.898	
53741292684.603		55889619119.832	25.506	38250.855
0.050	1478870			

Number of rows retrieved is: 4

Stop timestamp 05/11/04 14:33:11.812
Query Time = 4.7 secs

Query 2

Start timestamp 05/11/04 14:28:20.953

-- Query 02 - Var_0 Rev_02 - Minimum Cost Supplier Query

Tag: Q2 Stream: -1 Sequence number: 2

select s_acctbal, s_name, n_name, p_partkey, p_mfgr, s_address, s_phone, s_comment from tpcd.part, tpcd.supplier, tpcd.partsupp, tpcd.nation, tpcd.region where p_partkey = ps_partkey and s_suppkey = ps_suppkey and p_size = 15 and p_type like '%BRASS' and s_nationkey = n_nationkey and n_regionkey = r_regionkey and r_name = 'EUROPE' and ps_supplycost = (select min(ps_supplycost) from tpcd.partsupp, tpcd.supplier, tpcd.nation, tpcd.region where p_partkey = ps_partkey and s_suppkey = ps_suppkey and s_nationkey = n_nationkey and n_regionkey = r_regionkey and r_name = 'EUROPE') order by s_acctbal desc, n_name, s_name, p_partkey fetch first 100 rows only				
S_ACCTBAL	S_NAME	N_NAME		
P_PARTKEY	P_MFGR	S_ADDRESS		
S_PHONE	S_COMMENT			

9938.530	Supplier#000005359	UNITED KINGDOM		
185358	Manufacturer#4	QKuHYh,vZGiwu2FWt6JdLDx04		
33-429-790-6131		blithely silent pinto beans are furiously. slyly final deposits		
acros				
9937.840	Supplier#000005969	ROMANIA	108438	
Manufacturer#1		ANDENSOSmk,miq23Xfb5RWt6dvUcvt6Qa		
29-520-692-3537		carefully slow deposits use furiously. slyly ironic platelets		
above the ironic				
9936.220	Supplier#000005250	UNITED KINGDOM		
249	Manufacturer#4	B3rqp0xbSEim4Mpy2RH J		

33-320-228-2957 blithely special packages are. stealthily express deposits across the closely final instructi

9923.770 Supplier#000002324 GERMANY 29821
Manufacturer#4 y3OD9UywSTOK 17-779-299-1839
quickly express packages breach quiet pinto beans. requ

9871.220 Supplier#000006373 GERMANY 43868
Manufacturer#5 J8fcXWsTqM 17-813-485-8637
never silent deposits integrate furiously blit

9870.780 Supplier#000001286 GERMANY 81285
Manufacturer#2 YKA,E2fjiVd7eUrzp2Ef8j1QxGo2DFnosaTEH
17-516-924-4574 final theodolites cajole slyly special,

9870.780 Supplier#000001286 GERMANY 181285
Manufacturer#4 YKA,E2fjiVd7eUrzp2Ef8j1QxGo2DFnosaTEH
17-516-924-4574 final theodolites cajole slyly special,

9852.520 Supplier#000008973 RUSSIA 18972
Manufacturer#2 t5L67YdBYYH6o,Vz24jpDyQ9
32-188-594-7038 quickly regular instructions wake-- carefully unusual braids into the expres
9847.830 Supplier#000008097 RUSSIA 130557
Manufacturer#2 xMe97bpE69NzdWLoX 32-375-640-3593
slyly regular dependencies sleep slyly furiously express dep

9847.570 Supplier#000006345 FRANCE 86344
Manufacturer#1 VSt3rzK3qG698u6ld8HhOByvrTcSTSvQIDQDag
16-886-766-7945 silent pinto beans should have to snooze carefully along the final reques

... 80 lines Deleted

7937.930 Supplier#000009012 ROMANIA
83995 Manufacturer#2 iUiTziH,Ek3i4lwSgunXMgrcTzwdb
29-250-925-9690 blithely bold ideas haggle quickly final, regular request

7914.450 Supplier#000001013 RUSSIA 125988
Manufacturer#2 riRcntps4KEDtYScjpMIWeYF6mNnR
32-194-698-3365 final, ironic theodolites alongside of the ironic

7912.910 Supplier#000004211 GERMANY 159180
Manufacturer#5 2wQRVovHrm3,v03IKzfTd,IPYsFXQFFOG
17-266-947-7315 final requests integrate slyly above the silent, even

7912.910 Supplier#000004211 GERMANY 184210
Manufacturer#4 2wQRVovHrm3,v03IKzfTd,IPYsFXQFFOG
17-266-947-7315 final requests integrate slyly above the silent, even

7894.560 Supplier#000007981 GERMANY 85472
Manufacturer#4 NSJ96vMROAbeXP 17-963-404-3760
regular, even theodolites integrate carefully. bold, special theodolites are slyly fluffily iron

7887.080 Supplier#000009792 GERMANY 164759
Manufacturer#3 Y28ITVeYriT3kiGdV2K8fSZ V2UqT5H1Otz
17-988-938-4296 pending, ironic packages sleep among the carefully ironic accounts. quickly final accounts

7871.500 Supplier#000007206 RUSSIA 104695
Manufacturer#1 3w fNCnrVmvJJE95sgWZzvW
32-432-452-7731 furiously dogged pinto beans cajole. bold, express notornis until the slyly pending

7852.450 Supplier#000005864 RUSSIA 8363
Manufacturer#4 WCNfBPZeSXh3h,c 32-454-883-3821
blithely regular deposits

7850.660 Supplier#000001518 UNITED KINGDOM
86501 Manufacturer#1 ONda3YJiHKJOC
33-730-383-3892 furiously final accounts wake carefully idle requests. even dolphins wake acc

7843.520 Supplier#000006683 FRANCE 11680
Manufacturer#4 220JGkiv01Y00oCFwUGfviIbhzCdy
16-464-517-8943 carefully bold accounts doub

Number of rows retrieved is: 100

Stop timestamp 05/11/04 14:28:21.718
Query Time = 0.8 secs

Query 3

Start timestamp 05/11/04 14:32:09.156

-- Query 03 - Var_0 Rev_01 - Shipping Priority Query

Tag: Q3 Stream: -1 Sequence number: 11

```
select
l_orderkey,
sum(l_extendedprice * (1 - l_discount)) as revenue,
o_orderdate,
o_shippriority
from
tpcd.customer,
tpcd.orders,
tpcd.lineitem
where
c_mktsegment = 'BUILDING'
and c_custkey = o_custkey
and l_orderkey = o_orderkey
and o_orderdate < date ('1995-03-15')
and l_shipdate > date ('1995-03-15')
group by
l_orderkey,
o_orderdate,
o_shippriority
order by
revenue desc,
o_orderdate
fetch first 10 rows only
```

L_ORDERKEY	REVENUE	O_ORDERDATE	O_SHIPPRIORITY
2456423	406181.011	1995-03-05	0
3459808	405838.699	1995-03-04	0
492164	390324.061	1995-02-19	0
1188320	384537.936	1995-03-09	0
2435712	378673.056	1995-02-26	0
4878020	378376.795	1995-03-12	0
5521732	375153.922	1995-03-13	0
2628192	373133.309	1995-02-22	0
993600	371407.460	1995-03-05	0
2300070	367371.145	1995-03-13	0

Number of rows retrieved is: 10

Stop timestamp 05/11/04 14:32:26.593
Query Time = 17.4 secs

Query 4

Start timestamp 05/11/04 14:33:00.703

-- Query 04 - Var_0 Rev_01 - Order Priority Checking Query

Tag: Q4 Stream: -1 Sequence number: 14

```
select
o_orderpriority,
count(*) as order_count
from
tpcd.orders
where
o_orderdate >= date ('1993-07-01')
and o_orderdate < date ('1993-07-01') + 3 month
and exists (
select
*
from
tpcd.lineitem
where
l_orderkey = o_orderkey
and l_commitdate < l_receiptdate
)
group by
o_orderpriority
order by
o_orderpriority
```

O_ORDERPRIORITY ORDER_COUNT

1-URGENT 10594
2-HIGH 10476
3-MEDIUM 10410
4-NOT SPECIFIED 10556
5-LOW 10487

Number of rows retrieved is: 5

Stop timestamp 05/11/04 14:33:05.843
Query Time = 5.1 secs

Query 5

Start timestamp 05/11/04 14:33:22.078

-- Query 05 - Var_0 Rev_02 Local Supplier Volume Query

Tag: Q5 Stream: -1 Sequence number: 20

```
select
n_name,
sum(l_extendedprice * (1 - l_discount)) as revenue
from
tpcd.customer,
tpcd.orders,
tpcd.lineitem,
tpcd.supplier,
tpcd.nation,
tpcd.region
where
c_custkey = o_custkey
and o_orderkey = l_orderkey
and l_suppkey = s_suppkey
and c_nationkey = s_nationkey
and s_nationkey = n_nationkey
```

```
and n_regionkey = r_regionkey
and r_name = 'ASIA'
and o_orderdate >= date ('1994-01-01')
and o_orderdate < date ('1994-01-01') + 1 year
group by
n_name
order by
revenue desc
```

N_NAME	REVENUE
INDONESIA	55502041.170
VIETNAM	55295086.997
CHINA	53724494.257
INDIA	52035512.000
JAPAN	45410175.695

Number of rows retrieved is: 5

Stop timestamp 05/11/04 14:33:29.890
Query Time = 7.8 secs

Query 6

Start timestamp 05/11/04 14:29:03.968

-- Query 06 - Var_0 Rev_01 - Forecasting Revenue Change Query

Tag: Q6 Stream: -1 Sequence number: 5

```
select
sum(l_extendedprice * l_discount) as revenue
from
tpcd.lineitem
where
l_shipdate >= date ('1994-01-01')
and l_shipdate < date ('1994-01-01') + 1 year
and l_discount between .06 - 0.01 and .06 + 0.01
and l_quantity < 24
```

REVENUE

123141078.228

Number of rows retrieved is: 1

Stop timestamp 05/11/04 14:29:06.484
Query Time = 2.5 secs

Query 7

Start timestamp 05/11/04 14:33:29.890

-- Query 07 - Var_0 Rev_01 - Volume Shipping Query

Tag: Q7 Stream: -1 Sequence number: 21

```
select
supp_nation,
cust_nation,
l_year,
```

```
sum(volume) as revenue
from
(
select
n1.n_name as supp_nation,
n2.n_name as cust_nation,
year(l_shipdate) as l_year,
l_extendedprice * (1 - l_discount) as volume
from
tpcd.supplier,
tpcd.lineitem,
tpcd.orders,
tpcd.customer,
tpcd.nation n1,
tpcd.nation n2
where
s_suppkey = l_suppkey
and o_orderkey = l_orderkey
and c_custkey = o_custkey
and s_nationkey = n1.n_nationkey
and c_nationkey = n2.n_nationkey
and (
(n1.n_name = 'FRANCE' and n2.n_name = 'GERMANY')
or (n1.n_name = 'GERMANY' and n2.n_name = 'FRANCE')
)
and l_shipdate between date('1995-01-01') and date('1996-12-31')
) as shipping
group by
supp_nation,
cust_nation,
l_year
order by
supp_nation,
cust_nation,
l_year
```

SUPP_NATION	CUST_NATION	L_YEAR	REVENUE

FRANCE	GERMANY	1995	54639732.734
FRANCE	GERMANY	1996	54633083.308
GERMANY	FRANCE	1995	52531746.670
GERMANY	FRANCE	1996	52520549.022

Number of rows retrieved is: 4

Stop timestamp 05/11/04 14:34:05.531
Query Time = 35.6 secs

Query 8

Start timestamp 05/11/04 14:31:12.390

-- Query 08 - Var_0 Rev_01 - National Market Share Query

Tag: Q8 Stream: -1 Sequence number: 8

```
select
o_year,
sum(case
when nation = 'BRAZIL' then volume
else 0
end) / sum(volume) as mkt_share
from
(
```

```
select
year(o_orderdate) as o_year,
l_extendedprice * (1 - l_discount) as volume,
n2.n_name as nation
from
tpcd.part,
tpcd.supplier,
tpcd.lineitem,
tpcd.orders,
tpcd.customer,
tpcd.nation n1,
tpcd.nation n2,
tpcd.region
where
p_partkey = l_partkey
and s_suppkey = l_suppkey
and l_orderkey = o_orderkey
and o_custkey = c_custkey
and c_nationkey = n1.n_nationkey
and n1.n_regionkey = r_regionkey
and r_name = 'AMERICA'
and s_nationkey = n2.n_nationkey
and o_orderdate between date('1995-01-01') and date('1996-12-31')
and p_type = 'ECONOMY ANODIZED STEEL'
) as all_nations
group by
o_year
order by
o_year
```

O_YEAR	MKT_SHARE

1995	0.034
1996	0.041

Number of rows retrieved is: 2

Stop timestamp 05/11/04 14:31:22.437
Query Time = 10.0 secs

Query 9

Start timestamp 05/11/04 14:28:21.718

-- Query 09 - Var_0 Rev_01 - Product Type Profit Measure Query

Tag: Q9 Stream: -1 Sequence number: 3

```
select
nation,
o_year,
sum(amount) as sum_profit
from
(
select
n_name as nation,
year(o_orderdate) as o_year,
l_extendedprice * (1 - l_discount) - ps_supplycost * l_quantity as amount
from
tpcd.part,
tpcd.supplier,
tpcd.lineitem,
tpcd.partsupp,
tpcd.orders,
tpcd.nation

```

```

where
s_suppkey = l_suppkey
and ps_suppkey = l_suppkey
and ps_partkey = l_partkey
and p_partkey = l_partkey
and o_orderkey = l_orderkey
and s_nationkey = n_nationkey
and p_name like '%green%'
) as profit
group by
nation,
o_year
order by
nation,
o_year desc

```

NATION	O_YEAR	SUM_PROFIT
ALGERIA	1998	31342867.234
ALGERIA	1997	57138193.023
ALGERIA	1996	56140140.133
ALGERIA	1995	53051469.653
ALGERIA	1994	53867582.129
ALGERIA	1993	54942718.132
ALGERIA	1992	54628034.713
ARGENTINA	1998	30211185.708
ARGENTINA	1997	50805741.752
ARGENTINA	1996	51923746.575
ARGENTINA	1995	49298625.767
ARGENTINA	1994	50835610.109
ARGENTINA	1993	51646079.178
ARGENTINA	1992	50410314.995
BRAZIL	1998	27217924.383
BRAZIL	1997	48378669.199
BRAZIL	1996	50482870.357
BRAZIL	1995	47623383.635
BRAZIL	1994	47840165.726
BRAZIL	1993	49054694.035
BRAZIL	1992	48667639.084

... 153 lines deleted

UNITED KINGDOM	1997	49381810.899
UNITED KINGDOM	1996	51386853.960
UNITED KINGDOM	1995	51509586.789
UNITED KINGDOM	1994	48086499.711
UNITED KINGDOM	1993	49166827.223
UNITED KINGDOM	1992	49349122.082
UNITED STATES	1998	25126238.946
UNITED STATES	1997	50077306.419
UNITED STATES	1996	48048649.470
UNITED STATES	1995	48809032.423
UNITED STATES	1994	49296747.183
UNITED STATES	1993	48029946.801
UNITED STATES	1992	48671944.498
VIETNAM	1998	30442736.059
VIETNAM	1997	50309179.794
VIETNAM	1996	50488161.410
VIETNAM	1995	49658284.613
VIETNAM	1994	50596057.261
VIETNAM	1993	50953919.152
VIETNAM	1992	49613838.315

Number of rows retrieved is: 175

Stop timestamp 05/11/04 14:29:00.546
Query Time = 38.8 secs

Query 10

Start timestamp 05/11/04 14:33:11.812

-- Query 10 - Var_0 Rev_01 - Returned Item Reporting Query

Tag: Q10 Stream: -1 Sequence number: 18

```

select
c_custkey,
c_name,
sum(l_extendedprice * (1 - l_discount)) as revenue,
c_acctbal,
n_name,
c_address,
c_phone,
c_comment
from
tpcd.customer,
tpcd.orders,
tpcd.lineitem,
tpcd.nation
where
c_custkey = o_custkey
and l_orderkey = o_orderkey
and o_orderdate >= date ('1993-10-01')
and o_orderdate < date ('1993-10-01') + 3 month
and l_returnflag = 'R'
and c_nationkey = n_nationkey
group by
c_custkey,
c_name,
c_acctbal,
c_phone,
n_name,
c_address,
c_comment
order by
revenue desc
fetch first 20 rows only

```

C_CUSTKEY	C_NAME	REVENUE	C_ACCTBAL
N_NAME	C_ADDRESS		C_PHONE
C_COMMENT			

57040	Customer#000057040	734235.246	632.870
JAPAN	Eioyzjf4pp	22-895-641-3466	requests
sleep blithely about the furiously i			
143347	Customer#000143347	721002.695	2557.470
EGYPT	1aReFYv,Kw4	14-742-935-3718	fluffily
bold excuses haggle finally after the u			
60838	Customer#000060838	679127.308	2454.770
BRAZIL	64EaJ5vMAHWJIBOxJklpNc2RJiWE		
12-913-494-9813	furiously even pinto beans integrate under the ruthless foxes;		
ironic, even dolphins across the slyl			
101998	Customer#000101998	637029.567	3790.890
UNITED KINGDOM	01c9CILnNtfOQYmZj		
33-593-865-6378	accounts doze blithely! enticing, final deposits sleep blithely		
special accounts. slyly express accounts pla			
125341	Customer#000125341	633508.086	4983.510
GERMANY	S29ODD6bceU8QSuuEJznkNaK		

17-582-695-5962 quickly express requests wake quickly blithely

25501 Customer#000025501 620269.785 7725.040
ETHIOPIA W556MXuoiaYCCZamJI,Rn0B4ACUGdkQ8DZ
15-874-808-6793 quickly special requests sleep evenly among the special
deposits. special deposi

115831 Customer#000115831 596423.867 5098.100
FRANCE rFeBbEEYk dl ne7zV5fDrmiq1oK09wV7pxqCgIc
16-715-386-3788 carefully bold excuses sleep alongside of the thinly idle

84223 Customer#000084223 594998.024 528.650
UNITED KINGDOM nAVZCs6BaWap rrM27N 2qBnzc5WBauxbA
33-442-824-8191 pending, final ideas haggle final requests. unusual, regular
asymptotes affix according to the even foxes.

54289 Customer#000054289 585603.392 5583.020
IRAN vXCxoCsU0Bad5JQI ,oobkZ 20-834-292-4707
express requests sublate blithely regular requests. regular, even ideas solve.

39922 Customer#000039922 584878.113 7321.110
GERMANY Zgy4s50l2GKN4pLDPBU8m342gIw6R
17-147-757-8036 even pinto beans haggle. slyly bold accounts inte

6226 Customer#00006226 576783.761 2230.090
UNITED KINGDOM 8gPu8,NPGkfyQQ0hcIYUGPIBWc,ybP5g,
33-657-701-3391 quickly final requests against the regular instructions wake
blithely final instructions. pa

922 Customer#00000922 576767.533 3869.250
GERMANY Az9RFaut7NkPnc5zSD2PwHgVwr4jRzq
17-945-916-9648 boldly final requests cajole blith

147946 Customer#000147946 576455.132 2030.130
ALGERIA iANyZHjghyy7Ajah0pTrYyhJ
10-886-956-3143 furiously even accounts are blithely above the furiousl

115640 Customer#000115640 569341.193 6436.100
ARGENTINA Vtgfia9qI 7EpHgecU1X 11-411-543-4901
final instructions are slyly according to the

73606 Customer#000073606 568656.858 1785.670
JAPAN xuR0Tro5yChdFOCrjkd2ol 22-437-653-6966
furiously bold orbits about the furiously busy requests wake across the furiously
quiet theodolites. d

110246 Customer#000110246 566842.981 7763.350
VIETNAM 7KzflgX MDOq7sOkI 31-943-426-9837
dolphins sleep blithely among the slyly final

142549 Customer#000142549 563537.237 5085.990
INDONESIA ChqEoK43OysjdHbtKCp6dKqjNyyvi9
19-955-562-2398 regular, unusual dependencies boost slyly; ironic attainments
nag fluffily into the unusual packages?

146149 Customer#000146149 557254.987 1791.550
ROMANIA s87fvzFQpU 29-744-164-6487 silent,
unusual requests detect quickly slyly regul

52528 Customer#000052528 556397.351 551.790
ARGENTINA NFztyTOR10UOJ 11-208-192-3205
unusual requests detect. slyly dogged theodolites use slyly. deposit

23431 Customer#000023431 554269.536 3381.860
ROMANIA HgiV0phqhaIa9aydNolIb 29-915-458-2654
instructions nag quickly. furiously bold accounts cajol

Number of rows retrieved is: 20

Stop timestamp 05/11/04 14:33:17.656

Query Time = 5.8 secs

Query 11

Start timestamp 05/11/04 14:33:05.843

-- Query 11 - Var_0 Rev_01 - Important Stock Identification Query

Tag: Q11 Stream: -1 Sequence number: 15

```
select
ps_partkey,
sum(ps_supplycost * ps_availqty) as value
from
tpcd.partsupp,
tpcd.supplier,
tpcd.nation
where
ps_suppkey = s_suppkey
and s_nationkey = n_nationkey
and n_name = 'GERMANY'
group by
ps_partkey having
sum(ps_supplycost * ps_availqty) > (
select
sum(ps_supplycost * ps_availqty) * 0.0001000000
from
tpcd.partsupp,
tpcd.supplier,
tpcd.nation
where
ps_suppkey = s_suppkey
and s_nationkey = n_nationkey
and n_name = 'GERMANY'
)
order by
value desc
```

PS_PARTKEY VALUE

129760	17538456.860
166726	16503353.920
191287	16474801.970
161758	16101755.540
34452	15983844.720
139035	15907078.340
9403	15451755.620
154358	15212937.880
38823	15064802.860
85606	15053957.150
33354	14408297.400
154747	14407580.680
82865	14235489.780
76094	14094247.040
222	13937777.740
121271	13908336.000
55221	13716120.470
22819	13666434.280
76281	13646853.680
85298	13581154.930

... 1033 lines Deleted

77207	7897752.720
96712	7897575.270
10157	7897046.250
171154	7896814.500
79373	7896186.000

113808	7893353.880
27901	7892952.000
128820	7892882.720
25891	7890511.200
122819	7888881.020
154731	7888301.330
101674	7879324.600
51968	7879102.210
72073	7877736.110
5182	7874521.730

Number of rows retrieved is: 1048

Stop timestamp 05/11/04 14:33:06.328
Query Time = 0.5 secs

Query 12

Start timestamp 05/11/04 14:34:05.531

-- Query 12 - Var_0 Rev_02 - Shipping Modes and Order Priority Query

Tag: Q12 Stream: -1 Sequence number: 22

```
select
l_shipmode,
sum(case
when o_orderpriority = '1-URGENT'
or o_orderpriority = '2-HIGH'
then 1
else 0
end) as high_line_count,
sum(case
when o_orderpriority <> '1-URGENT'
and o_orderpriority <> '2-HIGH'
then 1
else 0
end) as low_line_count
from
tpcd.orders,
tpcd.lineitem
where
o_orderkey = l_orderkey
and l_shipmode in ('MAIL', 'SHIP')
and l_commitdate < l_receiptdate
and l_shipdate < l_commitdate
and l_receiptdate >= date ('1994-01-01')
and l_receiptdate < date ('1994-01-01') + 1 year
group by
l_shipmode
order by
l_shipmode
```

L_SHIPMODE HIGH_LINE_COUNT LOW_LINE_COUNT

MAIL	6202	9324
SHIP	6200	9262

Number of rows retrieved is: 2

Stop timestamp 05/11/04 14:34:33.234
Query Time = 27.7 secs

Query 13

Start timestamp 05/11/04 14:32:05.953

-- Query 13 - Var_0 Rev_01 - Customer Distribution Query

Tag: Q13 Stream: -1 Sequence number: 10

```
select
c_count,
count(*) as custdist
from
(
select
c_custkey,
count(o_orderkey)
from
tpcd.customer left outer join tpcd.orders on
c_custkey = o_custkey
and o_comment not like '%special%requests%'
group by
c_custkey
) as c_orders (c_custkey, c_count)
group by
c_count
order by
custdist desc,
c_count desc
```

C_COUNT CUSTDIST

0	50004
9	6641
10	6566
11	6058
8	5949
12	5553
13	4989
19	4748
7	4707
18	4625
15	4552
17	4530
14	4484
20	4461
16	4323
21	4217
22	3730
6	3334
23	3129
24	2622
25	2079
5	1972
26	1593
27	1185
4	1033
28	869
29	559
3	398
30	373
31	235
2	144
32	128
33	71
34	48
35	33

1	23
36	17
37	7
40	4
38	4
39	2
41	1

Number of rows retrieved is: 42

Stop timestamp 05/11/04 14:32:09.156
Query Time = 3.2 secs

Query 14

Start timestamp 05/11/04 14:28:19.265

--#SET ROWS_OUT -1 ROWS_FETCH -1

-- Query 14 - Var_0 Rev_01 - Promotion Effect Query

Tag: Q14 Stream: -1 Sequence number: 1

```
select
100.00 * sum(case
when p_type like 'PROMO%'
then l_extendedprice * (1 - l_discount)
else 0
end) / sum(l_extendedprice * (1 - l_discount)) as promo_revenue
from
tpcd.lineitem,
tpcd.part
where
l_partkey = p_partkey
and l_shipdate >= date ('1995-09-01')
and l_shipdate < date ('1995-09-01') + 1 month
```

PROMO_REVENUE

16.381

Number of rows retrieved is: 1

Stop timestamp 05/11/04 14:28:20.953
Query Time = 1.7 secs

Query 15a

Start timestamp 05/11/04 14:33:06.328

-- Query 15 - Var_a Rev_01 - Top Supplier Query

Tag: Q15a Stream: -1 Sequence number: 16

```
with revenue (supplier_no, total_revenue) as (
select
l_suppkey,
sum(l_extendedprice * (1-l_discount))
from
tpcd.lineitem
where
l_shipdate >= date ('1996-01-01')
```

```
and l_shipdate < date ('1996-01-01') + 3 month
group by
l_suppkey
)
select
s_suppkey,
s_name,
s_address,
s_phone,
total_revenue
from
tpcd.supplier,
revenue
where
s_suppkey = supplier_no
and total_revenue = (
select
max(total_revenue)
from
revenue
)
order by
s_suppkey
```

S_SUPPKEY	S_NAME	S_ADDRESS
S_PHONE	TOTAL_REVENUE	

8449	Supplier#000008449	Wp34zim9qYFbVctdW
20-469-856-8873	1772627.209	

Number of rows retrieved is: 1

Stop timestamp 05/11/04 14:33:07.078
Query Time = 0.8 secs

Query 16

Start timestamp 05/11/04 14:32:59.718

-- Query 16 - Var_0 Rev_01 - Parts/Supplier Relationship Query

Tag: Q16 Stream: -1 Sequence number: 13

```
select
p_brand,
p_type,
p_size,
count(distinct ps_suppkey) as supplier_cnt
from
tpcd.partsupp,
tpcd.part
where
p_partkey = ps_partkey
and p_brand <> 'Brand#45'
and p_type not like 'MEDIUM POLISHED%'
and p_size in (49, 14, 23, 45, 19, 3, 36, 9)
and ps_suppkey not in (
select
s_suppkey
from
tpcd.supplier
where
s_comment like '%Customer%Complaints%'
)
```

group by
p_brand,
p_type,
p_size
order by
supplier_cnt desc,
p_brand,
p_type,
p_size

P_BRAND	P_TYPE	P_SIZE	SUPPLIER_CNT
Brand#41	MEDIUM BRUSHED TIN	3	28
Brand#54	STANDARD BRUSHED COPPER	14	27
Brand#11	STANDARD BRUSHED TIN	23	24
Brand#11	STANDARD BURNISHED BRASS	36	24
Brand#15	MEDIUM ANODIZED NICKEL	3	24
Brand#15	SMALL ANODIZED BRASS	45	24
Brand#15	SMALL BURNISHED NICKEL	19	24
Brand#21	MEDIUM ANODIZED COPPER	3	24
Brand#22	SMALL BRUSHED NICKEL	3	24
Brand#22	SMALL BURNISHED BRASS	19	24

... 18294 lines Deleted

Brand#25	LARGE PLATED STEEL	19	3
Brand#32	STANDARD ANODIZED COPPER	23	3
Brand#33	SMALL ANODIZED BRASS	9	3
Brand#35	MEDIUM ANODIZED TIN	19	3
Brand#51	SMALL PLATED BRASS	23	3
Brand#52	MEDIUM BRUSHED BRASS	45	3
Brand#53	MEDIUM BRUSHED TIN	45	3
Brand#54	ECONOMY POLISHED BRASS	9	3
Brand#55	PROMO PLATED BRASS	19	3
Brand#55	STANDARD PLATED TIN	49	3

Number of rows retrieved is: 18314

Stop timestamp 05/11/04 14:33:00.703
Query Time = 1.0 secs

Query 17

Start timestamp 05/11/04 14:29:06.484

-- Query 17 - Var_0 Rev_01 - Small-Quantity-Order Revenue Query

Tag: Q17 Stream: -1 Sequence number: 6

```
select
sum(l_extendedprice) / 7.0 as avg_yearly
from
tpcd.lineitem,
tpcd.part
where
p_partkey = l_partkey
and p_brand = 'Brand#23'
and p_container = 'MED BOX'
and l_quantity < (
select
0.2 * avg(l_quantity)
from
tpcd.lineitem
where
l_partkey = p_partkey
```

)

AVG_YEARLY

348406.054

Number of rows retrieved is: 1

Stop timestamp 05/11/04 14:29:11.046
Query Time = 4.6 secs

Query 18

Start timestamp 05/11/04 14:29:11.046

-- Query 18 - Var_0 Rev_01 - Large Volume Customer Query

Tag: Q18 Stream: -1 Sequence number: 7

```
select
c_name,
c_custkey,
o_orderkey,
o_orderdate,
o_totalprice,
sum(l_quantity)
from
tpcd.customer,
tpcd.orders,
tpcd.lineitem
where
o_orderkey in (
select
l_orderkey
from
tpcd.lineitem
group by
l_orderkey having
sum(l_quantity) > 300
)
and c_custkey = o_custkey
and o_orderkey = l_orderkey
group by
c_name,
c_custkey,
o_orderkey,
o_orderdate,
o_totalprice
order by
o_totalprice desc,
o_orderdate
fetch first 100 rows only
```

C_NAME	C_CUSTKEY	O_ORDERKEY	O_ORDERDATE
O_TOTALPRICE	6		
Customer#000128120	128120	4722021	1994-04-07
544089.090	323.000		
Customer#000144617	144617	3043270	1997-02-12
530604.440	317.000		
Customer#000013940	13940	2232932	1997-04-13
522720.610	304.000		
Customer#000066790	66790	2199712	1996-09-30
515531.820	327.000		

Customer#000046435	46435	4745607	1997-07-03	
508047.990	309.000			
Customer#000015272	15272	3883783	1993-07-28	
500241.330	302.000			
Customer#000146608	146608	3342468	1994-06-12	
499794.580	303.000			
Customer#000096103	96103	5984582	1992-03-16	
494398.790	312.000			
Customer#000024341	24341	1474818	1992-11-15	
491348.260	302.000			
Customer#000137446	137446	5489475	1997-05-23	
487763.250	311.000			
... 37 lines deleted				
Customer#000112987	112987	4439686	1996-09-17	
418161.490	305.000			
Customer#000012599	12599	4259524	1998-02-12	
415200.610	304.000			
Customer#000105410	105410	4478371	1996-03-05	
412754.510	302.000			
Customer#000149842	149842	5156581	1994-05-30	
411329.350	302.000			
Customer#000010129	10129	5849444	1994-03-21	
409129.850	309.000			
Customer#000069904	69904	1742403	1996-10-19	
408513.000	305.000			
Customer#000017746	17746	6882	1997-04-09	408446.930
303.000				
Customer#000013072	13072	1481925	1998-03-15	
399195.470	301.000			
Customer#000082441	82441	857959	1994-02-07	
382579.740	305.000			
Customer#000088703	88703	2995076	1994-01-30	
363812.120	302.000			

Number of rows retrieved is: 57

Stop timestamp 05/11/04 14:31:12.390
Query Time = 121.3 secs

Query 19

Start timestamp 05/11/04 14:33:17.656

-- Query 19 - Var_0 Rev_01 - Discounted Revenue Query

Tag: Q19 Stream: -1 Sequence number: 19

```

select
sum(l_extendedprice* (1 - l_discount)) as revenue
from
tpcd.lineitem,
tpcd.part
where
(
p_partkey = l_partkey
and p_brand = 'Brand#12'
and p_container in ('SM CASE', 'SM BOX', 'SM PACK', 'SM PKG')
and l_quantity >= 1 and l_quantity <= 1 + 10
and p_size between 1 and 5
and l_shipmode in ('AIR', 'AIR REG')
and l_shipinstruct = 'DELIVER IN PERSON'
)
or

```

```

(
p_partkey = l_partkey
and p_brand = 'Brand#23'
and p_container in ('MED BAG', 'MED BOX', 'MED PKG', 'MED PACK')
and l_quantity >= 10 and l_quantity <= 10 + 10
and p_size between 1 and 10
and l_shipmode in ('AIR', 'AIR REG')
and l_shipinstruct = 'DELIVER IN PERSON'
)
or
(
p_partkey = l_partkey
and p_brand = 'Brand#34'
and p_container in ('LG CASE', 'LG BOX', 'LG PACK', 'LG PKG')
and l_quantity >= 20 and l_quantity <= 20 + 10
and p_size between 1 and 15
and l_shipmode in ('AIR', 'AIR REG')
and l_shipinstruct = 'DELIVER IN PERSON'
)

```

REVENUE

3083843.058

Number of rows retrieved is: 1

Stop timestamp 05/11/04 14:33:22.078
Query Time = 4.4 secs

Query 20

Start timestamp 05/11/04 14:29:00.546

-- Query 20 - Var_0 Rev_01 - Potential Part Promotion Query

Tag: Q20 Stream: -1 Sequence number: 4

```

select
s_name,
s_address
from
tpcd.supplier,
tpcd.nation
where
s_suppkey in (
select
ps_suppkey
from
tpcd.partsupp
where
ps_partkey in (
select
p_partkey
from
tpcd.part
where
p_name like 'forest%'
)
and ps_availqty > (
select
0.5 * sum(l_quantity)
from
tpcd.lineitem
where
l_partkey = ps_partkey
)
)

```

```

and l_suppkey = ps_suppkey
and l_shipdate >= date ('1994-01-01')
and l_shipdate < date ('1994-01-01') + 1 year
)
)
and s_nationkey = n_nationkey
and n_name = 'CANADA'
order by
s_name

```

S_NAME	S_ADDRESS
Supplier#000000020	iybAE,RmTymrZVYaFZva2SH,j
Supplier#000000091	YV45D7TkfdQanOOZ7q9QxkyGUapU1oOWU6q3
Supplier#000000197	YC2Acon6kjY3zj3Fbxs2k4Vdf7X0cd2F
Supplier#000000226	83qOdU2EYRdPQAQhEtn GRZEd
Supplier#000000285	Br7e1nntlyxrw6ImgpJ7YdhFDjuBf
Supplier#000000378	FfbhyCxWvcPro8ltp9
Supplier#000000402	i9Sw4DoyMhzhKXCH9By,AYSgmD
Supplier#000000530	0qwCMwobKY OcmLyfRXlagA8ukENJv,
Supplier#000000688	D fw5ocppmZpYBBIP1718hCihLDZ5KhKX
Supplier#000000710	fl19YPvOyb QoYwjKC,oPycpGfieBAcwKJo

... 184 Lines deleted ...

Supplier#000009753	wLhVEcRmd7PkJF4FBnGK7Z
Supplier#000009796	z,y4Idmr15DOvPUqYG
Supplier#000009799	4wNjXGa4OKWl
Supplier#000009811	E3iuyq7UnZxU7oPZle2Gu6
Supplier#000009812	APFRMy3lCbGFga53n5t9DxzFPQPgnjrGt32
Supplier#000009862	rJzweWeN58
Supplier#000009868	ROjGgx5gvtkmnUUoeyy7v
Supplier#000009869	ucLqxzrpBTRMewGSM29t0rNTM30g1Tu3Xgg3mKag
Supplier#000009899	7XdPAHrzt,UQFZE
Supplier#000009974	7wJ,J5DKcxSU4Kp1cQLpbcAvB5AsvKT

Number of rows retrieved is: 204

Stop timestamp 05/11/04 14:29:03.968
Query Time = 3.4 secs

Query 21

Start timestamp 05/11/04 14:31:22.437

-- Query 21 - Var_0 Rev_01 - Suppliers Who Kept Orders Waiting Query

Tag: Q21 Stream: -1 Sequence number: 9

```

select
s_name,
count(*) as numwait
from
tpcd.supplier,
tpcd.lineitem l1,
tpcd.orders,
tpcd.nation
where
s_suppkey = l1.l_suppkey
and o_orderkey = l1.l_orderkey
and o_orderstatus = 'F'
and l1.l_receiptdate > l1.l_commitdate

```

```

and exists (
select
*
from
tpcd.lineitem l2
where
l2.l_orderkey = l1.l_orderkey
and l2.l_suppkey <> l1.l_suppkey
)
and not exists (
select
*
from
tpcd.lineitem l3
where
l3.l_orderkey = l1.l_orderkey
and l3.l_suppkey <> l1.l_suppkey
and l3.l_receiptdate > l3.l_commitdate
)
and s_nationkey = n_nationkey
and n_name = 'SAUDI ARABIA'
group by
s_name
order by
numwait desc,
s_name
fetch first 100 rows only

```

S_NAME	NUMWAIT
Supplier#000002829	20
Supplier#000005808	18
Supplier#000000262	17
Supplier#000000496	17
Supplier#000002160	17
Supplier#000002301	17
Supplier#000002540	17
Supplier#000003063	17
Supplier#000005178	17
Supplier#000008331	17

... 80 lines deleted

Supplier#000000673	12
Supplier#000000762	12
Supplier#000000811	12
Supplier#000000821	12
Supplier#000001337	12
Supplier#000001916	12
Supplier#000001925	12
Supplier#000002039	12
Supplier#000002357	12
Supplier#000002483	12

Number of rows retrieved is: 100

Stop timestamp 05/11/04 14:32:05.953
Query Time = 43.5 secs

Query 22

Start timestamp 05/11/04 14:32:26.593

-- Query 22 - Var_0 Rev_01 - Global Sales Opportunity Query

Tag: Q22 Stream: -1 Sequence number: 12

```
select
cntrycode,
count(*) as numcust,
sum(c_acctbal) as totacctbal
from
(
select
substr(c_phone, 1, 2) as cntrycode,
c_acctbal
from
tpcd.customer
where
substr(c_phone, 1, 2) in
('13', '31', '23', '29', '30', '18', '17')
and c_acctbal > (
select
avg(c_acctbal)
from
tpcd.customer
where
c_acctbal > 0.00
and substr(c_phone, 1, 2) in
('13', '31', '23', '29', '30', '18', '17')
)
and not exists (
select
*
from
tpcd.orders
where
o_custkey = c_custkey
)
) as custsale
group by
cntrycode
order by
cntrycode
```

CNTRYCODE NUMCUST TOTACCTBAL

13	888	6737713.990
17	861	6460573.720
18	964	7236687.400
23	892	6701457.950
29	948	7158866.630
30	909	6808436.130
31	922	6806670.180

Number of rows retrieved is: 7

Stop timestamp 05/11/04 14:32:59.718
Query Time = 33.1 secs

First 10 Rows of the Database

```
D:\tpch\tools>db2 connect to TPCD & db2 -f d:\tpch\tools\first10rows.sql &
db2 connect reset & db2 terminate
connect to TPCD
```

Database Connection Information

Database server = DB2/NT 8.1.5
SQL authorization ID = TPCH
Local database alias = TPCD

```
SELECT * FROM TPCD.REGION FETCH FIRST 10 ROWS ONLY
```

R_REGIONKEY	R_NAME	R_COMMENT
0	AFRICA	special Tiresias about the furiously even dolphins are furi
1	AMERICA	even, ironic theodolites according to the bold platelets wa
2	ASIA	silent, bold requests sleep slyly across the quickly sly dependencies. furiously silent instructions alongside
3	EUROPE	special, bold deposits haggle foxes. platelet
4	MIDDLE EAST	furiously unusual packages use carefully above the unusual, exp
5 record(s) selected.		

```
SELECT * FROM TPCD.NATION FETCH FIRST 10 ROWS ONLY
```

N_NATIONKEY	N_NAME	N_REGIONKEY	N_COMMENT
0	ALGERIA	0	final accounts wake quickly. special reques
5	ETHIOPIA	0	fluffily ruthless requests integrate fluffily. pending ideas wake blithely acco
14	KENYA	0	ironic requests boost. quickly pending pinto beans cajole slyly slyly even deposits. ironic packages
15	MOROCCO	0	ideas according to the fluffily final pinto beans sleep furiously
16	MOZAMBIQUE	0	ironic courts wake fluffily even, bold deposi
1	ARGENTINA	1	idly final instructions cajole stealthily. regular instructions wake carefully blithely express accounts. fluffi
2	BRAZIL	1	always pending pinto beans sleep sil
3	CANADA	1	foxes among the bold requests
17	PERU	1	final, final accounts sleep slyly across the requests.
24	UNITED STATES	1	blithely regular deposits serve furiously blithely regular warthogs! slyly fi
10 record(s) selected.			

```
SELECT * FROM TPCD.PART FETCH FIRST 10 ROWS ONLY
```

P_PARTKEY	P_NAME	P_SIZE	P_MFGR
P_BRAND	P_TYPE	P_CONTAINER	P_RETAILPRICE
			P_COMMENT

```

-----
11 chocolate turquoise sandy snow misty      Manufacturer#2
Brand#25 STANDARD BURNISHED NICKEL          43 WRAP BOX
+9.110100000000000E+002 furiousl

37 turquoise ivory orange sandy maroon        Manufacturer#4
Brand#45 LARGE POLISHED TIN                  48 JUMBO BOX
+9.370300000000000E+002 blithely regular

43 medium khaki chocolate rosy blush          Manufacturer#4
Brand#44 PROMO POLISHED STEEL                5 WRAP CASE
+9.430400000000000E+002 carefully iro

47 sky firebrick red linen dim                Manufacturer#4
Brand#45 LARGE BURNISHED BRASS               14 JUMBO PACK
+9.470400000000000E+002 bold, unusual a

50 yellow cornflower royal blush almond       Manufacturer#3
Brand#33 LARGE ANODIZED TIN                  25 WRAP PKG
+9.500500000000000E+002 regular dinos ar

55 antique cream pale tomato rose            Manufacturer#2
Brand#23 ECONOMY BRUSHED COPPER              9 MED BAG
+9.550500000000000E+002 furiously

68 bisque frosted pale puff sandy            Manufacturer#1
Brand#11 PROMO ANODIZED STEEL               10 WRAP BOX
+9.680600000000000E+002 carefully

76 bisque light yellow puff salmon           Manufacturer#3
Brand#34 MEDIUM BRUSHED COPPER              9 SM PKG
+9.760700000000000E+002 final

87 pale khaki sandy antique black            Manufacturer#4
Brand#41 LARGE PLATED STEEL                  41 WRAP PACK
+9.870800000000000E+002 slyly even instruction

89 ghost khaki lawn pale dim                 Manufacturer#5
Brand#53 STANDARD BURNISHED STEEL           7 MED JAR
+9.890800000000000E+002 quickly ironi

10 record(s) selected.

SELECT * FROM TPCD.SUPPLIER FETCH FIRST 10 ROWS ONLY

S_SUPPKEY S_NAME      S_ADDRESS
S_NATIONKEY S_PHONE   S_ACCTBAL      S_COMMENT
-----
--

9 Supplier#000000009 1KhUgZegwM3ua7dsYmekYBsK
10 20-403-398-8662 +5.302370000000000E+003 slyly regular decoys mold
slyly ironic dugouts. requests are carefully-- carefully

398 Supplier#000000398 WTV,rOOS1KZFKlhLsUpAH
10 20-536-426-8920 +9.354750000000000E+003 blithely pending instructions
above

504 Supplier#000000504 P8k2mjRiRUFcJfxw7KrEdRpNNQPDxiI
10 20-322-544-5770 +9.050120000000000E+003 quickly regular courts
doze furiously slyly regular packages. blithely unusual platelets cajole

```

```

1233 Supplier#000001233 V26hJzvCaekpg9WP8DL7f,AXiL
10 20-440-730-2308 +6.600160000000000E+003 slyly regular requests sleep
slyly pending platelets. final, unusual inst

1264 Supplier#000001264 GCfirAZS3kqchx5T7fZlH5qEDCN82iaZu
10 20-989-349-5053 +5.728440000000000E+003 quickly regular
dolphins wake car

1441 Supplier#000001441
uWzJDzjltr4ZX5PJLbe7g8zNtho8C5Awq9Chuu 10 20-682-239-5942
+5.851670000000000E+003 requests thrash slyly ironic theod

1523 Supplier#000001523 vqDQGhm58v0 10
20-315-869-6648 +9.915480000000000E+003 final, final theodolites haggle after

1652 Supplier#000001652 RSKj9qiJ0xZBwW4 10
20-895-421-3569 +3.312980000000000E+003 blithely express ideas thrash
furiously.

1953 Supplier#000001953 hpv9Dsezjk8Zw7qSlnzGhMF6kkdJ9zl
10 20-371-309-5922 +6.825470000000000E+003 carefully express accounts
use slyly regular a

1981 Supplier#000001981 Q9YktbRu7FxiQiaT1U
10 20-705-553-6657 +7.086420000000000E+003 furiously unusual dependen

10 record(s) selected.

SELECT * FROM TPCD.PARTSUPP FETCH FIRST 10 ROWS ONLY

PS_PARTKEY PS_SUPPKEY PS_AVAILQTY PS_SUPPLYCOST
PS_COMMENT
-----
-----
-

11 12 4540 +7.098700000000000E+002 final packages mold after
the carefully unusual requests. quickly fi

11 750012 4729 +8.949000000000000E+002 regular packages sleep
carefully fluffily ironic ac

11 1500012 3708 +8.187400000000000E+002 slyly pending
theodolites wake quickly unusual, express accounts. fluffily regular requests
cajole furiously quickly even dugouts. slyly bold platelets

11 2250012 3213 +4.719800000000000E+002 ideas nag regular
instructions. regular, thin pinto beans unwind furiously ironic accounts. quickly
express platele

37 38 7171 +8.249600000000000E+002 pending packages wake
quietly. special, even accounts haggle unusual

37 750038 5542 +1.265900000000000E+002 ironic, ironic packages
are slyly. furiously regular excuses cajole blithely quickly ironic dependencies.
regular packages use carefully furiously pending depths

37 1500038 7113 +1.572000000000000E+001 regular, final requests
nag across the regular, even requests. carefully regular accounts cajole carefully
carefully ruthless dep

```

37 2250038 1449 +7.4564000000000E+002 regular, silent foxes
nag. carefully ironic deposits are fluffily. carefully regular pinto beans around the
bravely even instructions cajole carefully furiously regul

43 44 3211 +8.0578000000000E+002 final, express
dependencies sleep according to the express requests. bold, regular accounts
detect outside the slyly

43 750044 6770 +4.9319000000000E+002 furiously special pinto
beans cajole. ironic decoys across the

10 record(s) selected.

SELECT * FROM TPCD.CUSTOMER FETCH FIRST 10 ROWS ONLY

C_CUSTKEY	C_NAME	C_ADDRESS
C_NATIONKEY	C_PHONE	C_ACCTBAL
C_COMMENT		C_MKTSEGMENT

15 Customer#000000015
YtWggXoOLdwdo7b0y,BZaGUQMLJMX1Y,EC,6Dn 23
33-687-542-7601 +2.7885200000000E+003 HOUSEHOLD fluffily bold
dolphins detect quickly about the special instruction

31 Customer#000000031 LUACbO0viaAv6eXOAebryDB xjVst
23 33-197-837-7094 +5.2368900000000E+003 HOUSEHOLD carefully
even theodolites doubt final, final deposits. regular, ironic tithes sleep

441 Customer#000000441 gjYpcBx6MP8GvDa6 23
33-438-355-3491 +9.4518400000000E+003 HOUSEHOLD slyly unusual
accounts cajole sometimes. express accounts haggle furiously alongside of the
silently final ins

771 Customer#000000771 J9UMiYJznHVHZDuCG,ErV0iiMStETS
23 33-761-371-4753 +7.4617400000000E+003 AUTOMOBILE
furiously regular platelets within the furiously even ideas haggle c

828 Customer#000000828 0PkG ELwBIgU4AsNcjDc5Q,9Gj
23 33-108-680-4317 +7.0596800000000E+003 BUILDING braids nag!
express theodolites haggle after the express packages. fluffily regular

985 Customer#000000985 0uAMe1ICB,wtS4STD4eLL
23 33-408-194-5161 +2.7012100000000E+003 HOUSEHOLD bold, ironic
accounts sleep slyly. carefully special deposits haggle regular requests. slowly
even dep

1059 Customer#000001059 OHwYMiDjmgelQXhLINW,8LIwIEr
23 33-683-418-9460 +1.5475000000000E+003 HOUSEHOLD
pending accounts sleep carefully after the furiously even pinto beans. packages
wake slyly. re

1117 Customer#000001117 80NfzBRWj5tUUArdnsFE7Eg
23 33-461-439-5684 +2.8290700000000E+003 FURNITURE furiously
regular foxes cajole furiously ironic requests. carefully permae

1648 Customer#000001648 e3oTXQ7OOTzwcRFXr
23 33-718-723-5373 +2.3891400000000E+003 AUTOMOBILE regular
requests are. blithely special

1727 Customer#000001727 g7iJ1HSEmxElsd4lWLV
XcE64HA2JHYqPEJBA 23 33-952-217-5029

+3.6280300000000E+003 AUTOMOBILE express, even requests boost.
blithely special accounts thrash slyly alo

10 record(s) selected.

SELECT * FROM TPCD.ORDERS FETCH FIRST 10 ROWS ONLY

O_ORDERKEY	O_CUSTKEY	O_ORDERSTATUS	O_TOTALPRICE
O_ORDERDATE	O_ORDERPRIORITY	O_CLERK	O_SHIPRIORITY
O_COMMENT			

126753	40951322 F	+2.24458810000000E+005	01/01/1992
3-MEDIUM	Clerk#000171722	0 pinto beans sleep quickly	
176768	33854756 F	+3.12039110000000E+005	01/01/1992
1-URGENT	Clerk#000241823	0 carefully unusual accounts use	
		slyly. ironic accounts integrat	
300416	32144095 F	+9.4546840000000E+004	01/01/1992
4-NOT SPECIFIED	Clerk#000270154	0 fluffily silent ideas must affi	
343335	9645688 F	+2.8118208000000E+005	01/01/1992
2-HIGH	Clerk#000149299	0 bold deposits are carefully sent	
433446	37907813 F	+1.03566760000000E+005	01/01/1992
2-HIGH	Clerk#000031969	0 furiously unusual courts wake	
		carefully furiously	
434722	2812378 F	+1.35128570000000E+005	01/01/1992
4-NOT SPECIFIED	Clerk#000082529	0 courts near the carefully ironic accounts	
		are sl	
497314	26507002 F	+6.5310880000000E+004	01/01/1992
3-MEDIUM	Clerk#000285089	0 even, final dependencies wake	
		unusual sauternes. careful	
515077	35649802 F	+2.82384590000000E+005	01/01/1992
1-URGENT	Clerk#000020437	0 deposits are furiousl	
639553	32997580 F	+2.31567590000000E+005	01/01/1992
5-LOW	Clerk#000057504	0 bold theodolites wake furiously inside	
		the regular theo	
706371	22199488 F	+9.4253900000000E+003	01/01/1992
1-URGENT	Clerk#000184294	0 slyly regular packages wake.	

10 record(s) selected.

SELECT * FROM TPCD.LINEITEM FETCH FIRST 10 ROWS ONLY

L_ORDERKEY	L_PARTKEY	L_SUPPKEY	L_LINENUMBER
L_QUANTITY	L_EXTENDEDPRICE	L_DISCOUNT	
L_TAX	L_RETURNFLAG	L_LINESTATUS	L_SHIPDATE
L_COMMITDATE	L_RECEIPTDATE	L_SHIPINSTRUCT	
L_SHIPMODE	L_COMMENT		

1054181	4865078	1865079	1	+4.5000000000000E+001
+4.6927350000000E+004	+3.0000000000000E+002			


```

+8.000000000000000E-002 R      F      01/02/1992 02/05/1992 01/15/1992
NONE      MAIL      even instructions kindle furio

5018977 24611600 611601      1 +2.000000000000000E+001
+3.020740000000000E+004 +0.000000000000000E+000
+0.000000000000000E+000 A      F      01/02/1992 03/19/1992
01/15/1992 NONE      SHIP      quickly ironic excu

14168833 46737982 988028      3 +4.900000000000000E+001
+9.886485000000000E+004 +9.000000000000000E-002
+5.000000000000000E-002 R      F      01/02/1992 02/01/1992 01/28/1992
TAKE BACK RETURN      SHIP      furiously bold courts are careful

15413986 53978084 2978085      4 +1.400000000000000E+001
+1.623146000000000E+004 +5.000000000000000E-002
+5.000000000000000E-002 A      F      01/02/1992 01/31/1992 01/04/1992
COLLECT COD      TRUCK      final, bold dolphins

18436929 39951321 2451348      1 +5.000000000000000E+000
+6.851650000000000E+003 +0.000000000000000E+000
+7.000000000000000E-002 A      F      01/02/1992 02/27/1992 01/26/1992
TAKE BACK RETURN      SHIP      even, regular platelets na

19547653 37197566 1947579      1 +4.800000000000000E+001
+7.976208000000000E+004 +0.000000000000000E+000
+7.000000000000000E-002 R      F      01/02/1992 02/08/1992 01/26/1992
NONE      MAIL      regular accounts haggle. h

19918976 6334057 2584064      6 +3.100000000000000E+001
+3.381294000000000E+004 +5.000000000000000E-002
+6.000000000000000E-002 R      F      01/02/1992 02/23/1992 01/03/1992
DELIVER IN PERSON      REG AIR      carefully fluffy requests

22142214 56458378 1708433      7 +1.900000000000000E+001
+2.533745000000000E+004 +0.000000000000000E+000
+7.000000000000000E-002 R      F      01/02/1992 03/17/1992 01/03/1992
NONE      AIR      carefully express requests

22518080 50882189 1382222      6 +3.900000000000000E+001
+4.557696000000000E+004 +5.000000000000000E-002
+7.000000000000000E-002 R      F      01/02/1992 03/06/1992 01/30/1992
COLLECT COD      SHIP      carefully special ideas

26601603 39300513 2550553      4 +2.000000000000000E+000
+3.023100000000000E+003 +0.000000000000000E+000
+2.000000000000000E-002 R      F      01/02/1992 03/28/1992 01/08/1992
COLLECT COD      RAIL      special platelets wake about the slyly fin

10 record(s) selected.

```

connect reset

DB20000I The SQL command completed successfully.

Query Substitution Parameters

```

"Power stream Seed = 514201105"
-- TPC TPC-H Parameter Substitution (Version 1.3.0)
-- using 514201105 as a seed to the RNG
Q1 DELTA 60
Q2 SIZE 31
TYPE NICKEL
REGION AFRICA
Q3 SEGMENT HOUSEHOLD
DATE 1995-03-06
Q4 DATE 1993-12-01
Q5 REGION ASIA
DATE 1996-01-01

```

```

Q6 DATE 1996-01-01
DISCOUNT 0.06
QUANTITY 25
Q7 NATION1 JORDAN
NATION2 JAPAN
Q8 NATION JAPAN
REGION ASIA
TYPE PROMO PLATED COPPER
Q9 COLOR thistle
Q10 DATE 1993-02-01
Q11 NATION IRAQ
FRACTION 0.0000003333
Q12 SHIPMODE1 AIR
SHIPMODE2 TRUCK
DATE 1997-01-01
Q13 WORD1 pending
WORD2 deposits
Q14 DATE 1997-02-01
Q15 DATE 1996-08-01
Q16 BRAND Brand#55
TYPE SMALL POLISHED
SIZE1 28
SIZE2 11
SIZE3 7
SIZE4 26
SIZE5 37
SIZE6 4
SIZE7 35
SIZE8 25
Q17 BRAND Brand#52
CONTAINER LG DRUM
Q18 QUANTITY 313
Q19 BRAND1 Brand#22
BRAND2 Brand#33
BRAND3 Brand#41
QUANTITY1 7
QUANTITY2 14
QUANTITY3 30
Q20 COLOUR honeydew
DATE 1995-01-01
NATION KENYA
Q21 NATION KENYA
Q22 I1 16
I2 23
I3 12
I4 27
I5 25
I6 19
I7 15

```

```

"Throughput Stream = 1 Seed = 514201106"
-- TPC TPC-H Parameter Substitution (Version 1.3.0)
-- using 514201106 as a seed to the RNG

```

```

Q1 DELTA 68
Q2 SIZE 19
TYPE TIN
REGION EUROPE
Q3 SEGMENT AUTOMOBILE
DATE 1995-03-23
Q4 DATE 1996-07-01
Q5 REGION EUROPE
DATE 1996-01-01
Q6 DATE 1996-01-01
DISCOUNT 0.03
QUANTITY 24
Q7 NATION1 ETHIOPIA
NATION2 EGYPT
Q8 NATION EGYPT

```

REGION MIDDLE EAST
TYPE PROMO ANODIZED COPPER

Q9 COLOR slate
Q10 DATE 1993-11-01
Q11 NATION UNITED STATES
FRACTION 0.0000003333
Q12 SHIPMODE1 REG AIR
SHIPMODE2 TRUCK
DATE 1997-01-01
Q13 WORD1 pending
WORD2 deposits
Q14 DATE 1997-06-01
Q15 DATE 1994-05-01
Q16 BRAND Brand#35

TYPE ECONOMY ANODIZED
SIZE1 23
SIZE2 41
SIZE3 14
SIZE4 9
SIZE5 33
SIZE6 17
SIZE7 11
SIZE8 28

Q17 BRAND Brand#54
CONTAINER MED BAG

Q18 QUANTITY 314

Q19 BRAND1 Brand#35
BRAND2 Brand#21
BRAND3 Brand#45
QUANTITY1 3
QUANTITY2 15
QUANTITY3 27

Q20 COLOUR salmon

DATE 1994-01-01
NATION EGYPT

Q21 NATION FRANCE

Q22 I1 14

I2 26
I3 10
I4 15
I5 22
I6 13
I7 19

"Throughput Stream = 2 Seed = 514201107"

-- TPC TPC-H Parameter Substitution (Version 1.3.0)

-- using 514201107 as a seed to the RNG

Q1 DELTA 76

Q2 SIZE 7

TYPE COPPER
REGION AMERICA

Q3 SEGMENT FURNITURE

DATE 1995-03-08

Q4 DATE 1994-04-01

Q5 REGION MIDDLE EAST
DATE 1996-01-01

Q6 DATE 1996-01-01

DISCOUNT 0.09

QUANTITY 25

Q7 NATION1 RUSSIA

NATION2 VIETNAM

Q8 NATION VIETNAM

REGION ASIA

TYPE ECONOMY POLISHED COPPER

Q9 COLOR saddle

Q10 DATE 1994-08-01

Q11 NATION JAPAN

FRACTION 0.0000003333

Q12 SHIPMODE1 FOB
SHIPMODE2 TRUCK

DATE 1997-01-01

Q13 WORD1 unusual

WORD2 deposits

Q14 DATE 1997-09-01

Q15 DATE 1996-11-01

Q16 BRAND Brand#25

TYPE STANDARD BURNISHED

SIZE1 34

SIZE2 12

SIZE3 48

SIZE4 20

SIZE5 30

SIZE6 41

SIZE7 4

SIZE8 32

Q17 BRAND Brand#51

CONTAINER MED PKG

Q18 QUANTITY 312

Q19 BRAND1 Brand#32

BRAND2 Brand#54

BRAND3 Brand#35

QUANTITY1 8

QUANTITY2 16

QUANTITY3 23

Q20 COLOUR cyan

DATE 1997-01-01

NATION CHINA

Q21 NATION UNITED KINGDOM

Q22 I1 28

I2 16

I3 30

I4 10

I5 25

I6 14

I7 34

"Throughput Stream = 3 Seed = 514201108"

-- TPC TPC-H Parameter Substitution (Version 1.3.0)

-- using 514201108 as a seed to the RNG

Q1 DELTA 84

Q2 SIZE 45

TYPE BRASS

REGION EUROPE

Q3 SEGMENT AUTOMOBILE

DATE 1995-03-25

Q4 DATE 1996-11-01

Q5 REGION AMERICA

DATE 1996-01-01

Q6 DATE 1996-01-01

DISCOUNT 0.06

QUANTITY 25

Q7 NATION1 KENYA

NATION2 JORDAN

Q8 NATION JORDAN

REGION MIDDLE EAST

TYPE ECONOMY BURNISHED COPPER

Q9 COLOR puff

Q10 DATE 1993-05-01

Q11 NATION ALGERIA

FRACTION 0.0000003333

Q12 SHIPMODE1 MAIL

SHIPMODE2 TRUCK

DATE 1997-01-01

Q13 WORD1 unusual

WORD2 deposits

Q14 DATE 1997-12-01

Q15 DATE 1994-08-01
 Q16 BRAND Brand#55
 TYPE MEDIUM POLISHED
 SIZE1 30
 SIZE2 5
 SIZE3 32
 SIZE4 19
 SIZE5 26
 SIZE6 2
 SIZE7 14
 SIZE8 4

Q17 BRAND Brand#53
 CONTAINER MED DRUM

Q18 QUANTITY 313

Q19 BRAND1 Brand#34

BRAND2 Brand#42

BRAND3 Brand#34

QUANTITY1 3

QUANTITY2 17

QUANTITY3 30

Q20 COLOUR orange

DATE 1996-01-01

NATION INDIA

Q21 NATION MOROCCO

Q22 I1 32

I2 31

I3 26

I4 20

I5 28

I6 13

I7 33

"Throughput Stream = 4 Seed = 514201109"

-- TPC TPC-H Parameter Substitution (Version 1.3.0)

-- using 514201109 as a seed to the RNG

Q1 DELTA 92

Q2 SIZE 32

TYPE NICKEL

REGION AMERICA

Q3 SEGMENT FURNITURE

DATE 1995-03-10

Q4 DATE 1994-08-01

Q5 REGION ASIA

DATE 1997-01-01

Q6 DATE 1997-01-01

DISCOUNT 0.04

QUANTITY 24

Q7 NATION1 FRANCE

NATION2 ETHIOPIA

Q8 NATION ETHIOPIA

REGION AFRICA

TYPE ECONOMY ANODIZED TIN

Q9 COLOR papaya

Q10 DATE 1994-03-01

Q11 NATION JORDAN

FRACTION 0.0000003333

Q12 SHIPMODE1 TRUCK

SHIPMODE2 MAIL

DATE 1993-01-01

Q13 WORD1 unusual

WORD2 packages

Q14 DATE 1993-03-01

Q15 DATE 1997-03-01

Q16 BRAND Brand#35

TYPE ECONOMY BRUSHED

SIZE1 34

SIZE2 3

SIZE3 9

SIZE4 47

SIZE5 2

SIZE6 16

SIZE7 40

SIZE8 38

Q17 BRAND Brand#55

CONTAINER JUMBO BAG

Q18 QUANTITY 315

Q19 BRAND1 Brand#41

BRAND2 Brand#25

BRAND3 Brand#33

QUANTITY1 8

QUANTITY2 18

QUANTITY3 26

Q20 COLOUR bisque

DATE 1994-01-01

NATION UNITED KINGDOM

Q21 NATION INDIA

Q22 I1 33

I2 21

I3 23

I4 29

I5 30

I6 11

I7 15

"Throughput Stream = 5 Seed = 514201110"

-- TPC TPC-H Parameter Substitution (Version 1.3.0)

-- using 514201110 as a seed to the RNG

Q1 DELTA 100

Q2 SIZE 20

TYPE TIN

REGION MIDDLE EAST

Q3 SEGMENT MACHINERY

DATE 1995-03-27

Q4 DATE 1997-03-01

Q5 REGION EUROPE

DATE 1997-01-01

Q6 DATE 1997-01-01

DISCOUNT 0.09

QUANTITY 25

Q7 NATION1 UNITED KINGDOM

NATION2 RUSSIA

Q8 NATION RUSSIA

REGION EUROPE

TYPE LARGE POLISHED TIN

Q9 COLOR navajo

Q10 DATE 1994-12-01

Q11 NATION ARGENTINA

FRACTION 0.0000003333

Q12 SHIPMODE1 RAIL

SHIPMODE2 MAIL

DATE 1993-01-01

Q13 WORD1 unusual

WORD2 packages

Q14 DATE 1993-06-01

Q15 DATE 1994-12-01

Q16 BRAND Brand#25

TYPE SMALL BURNISHED

SIZE1 7

SIZE2 39

SIZE3 13

SIZE4 27

SIZE5 40

SIZE6 16

SIZE7 17

SIZE8 36

Q17 BRAND Brand#52

CONTAINER JUMBO PKG
 Q18 QUANTITY 312
 Q19 BRAND1 Brand#43
 BRAND2 Brand#13
 BRAND3 Brand#22
 QUANTITY1 4
 QUANTITY2 19
 QUANTITY3 22
 Q20 COLOUR lemon
 DATE 1993-01-01
 NATION JORDAN
 Q21 NATION ALGERIA
 Q22 I1 20
 I2 11
 I3 13
 I4 12
 I5 17
 I6 29
 I7 19

"Throughput Stream = 6 Seed = 514201111"
 -- TPC TPC-H Parameter Substitution (Version 1.3.0)
 -- using 514201111 as a seed to the RNG

Q1 DELTA 108
 Q2 SIZE 8
 TYPE COPPER
 REGION AMERICA
 Q3 SEGMENT BUILDING
 DATE 1995-03-12
 Q4 DATE 1994-12-01
 Q5 REGION MIDDLE EAST
 DATE 1997-01-01
 Q6 DATE 1997-01-01
 DISCOUNT 0.07
 QUANTITY 25
 Q7 NATION1 MOROCCO
 NATION2 KENYA
 Q8 NATION KENYA
 REGION AFRICA
 TYPE LARGE BURNISHED TIN
 Q9 COLOR medium
 Q10 DATE 1993-09-01
 Q11 NATION KENYA
 FRACTION 0.0000003333
 Q12 SHIPMODE1 AIR
 SHIPMODE2 MAIL
 DATE 1993-01-01
 Q13 WORD1 unusual
 WORD2 packages
 Q14 DATE 1993-10-01
 Q15 DATE 1997-06-01
 Q16 BRAND Brand#55
 TYPE LARGE PLATED
 SIZE1 27
 SIZE2 45
 SIZE3 36
 SIZE4 6
 SIZE5 16
 SIZE6 10
 SIZE7 43
 SIZE8 9
 Q17 BRAND Brand#53
 CONTAINER JUMBO DRUM
 Q18 QUANTITY 314
 Q19 BRAND1 Brand#45
 BRAND2 Brand#41
 BRAND3 Brand#21
 QUANTITY1 9

QUANTITY2 10
 QUANTITY3 30
 Q20 COLOUR spring
 DATE 1996-01-01
 NATION BRAZIL
 Q21 NATION PERU
 Q22 I1 10
 I2 17
 I3 14
 I4 31
 I5 12
 I6 32
 I7 25

Appendix D: Driver Source Code

doufload_v8.bat

```
REM Takes UfType and update_pair as parameters.
set RAHSLEEPTIME=999999
d:
cd \tpch\tools
perl load_UF%1_data_v8 %2
cd \tpch\ddl
```

load_UF1_data_V8

```
: # *-Perl-*
eval `exec perl5 -S $0 ${1+"$@"}' # Horrible kludge to convert this
    if 0;                          # into a "portable" perl script

# usage perl loadUFD [update pair number]

push(@INC, split(':', $ENV{'PATH'}));

# Get TPC-D specific environment variables
require 'getvars';

# Use the macros in here so that they can handle the platform differences.
# macro.pl should be sourced from cmvc, other people wrote and maintain it.
require "macro.pl";

# Make output unbuffered.
select(STDOUT);
$| = 1;

if (length($ENV{"TPCD_AUDIT_DIR"}) <= 0)
{
    die "TPCD_AUDIT_DIR environment variable not set\n";
}
if (length($ENV{"TPCD_DBNAME"}) <= 0)
{
    die "TPCD_DBNAME environment variable not set\n";
}
if (length($ENV{"TPCD_SF"}) <= 0)
{
    die "TPCD_SF environment variable not set\n";
}
if (length($ENV{"TPCD_PLATFORM"}) <= 0)
{
    die "TPCD_PLATFORM environment variable not set\n";
}
if (length($ENV{"TPCD_PATH_DELIM"}) <= 0)
{
    die "TPCD_PATH_DELIM environment variable not set\n";
}
if (length($ENV{"TPCD_PRODUCT"}) <= 0)
{
    die "TPCD_PRODUCT environment variable not set\n";
}
if (length($ENV{"TPCD_AUDIT"}) <= 0)
{
    die "Must set TPCD_AUDIT env't var. Real audit timing sequence run if
yes\n";
}
if (length($ENV{"TPCD_PHYS_NODE"}) <= 0)
{
    die "TPCD_PHYS_NODE env't var not set\n";
}

#set up local variables
$auditDir=$ENV{"TPCD_AUDIT_DIR"};
$dbname=$ENV{"TPCD_DBNAME"};
$sf=$ENV{"TPCD_SF"};
```

```
$splatform=$ENV{"TPCD_PLATFORM"};
$delim=$ENV{"TPCD_PATH_DELIM"};
$gatherstats=$ENV{"TPCD_GATHER_STATS"};
$product=$ENV{"TPCD_PRODUCT"};
$RealAudit=$ENV{"TPCD_AUDIT"};
$inlistmax=$ENV{"TPCD_INLISTMAX"};
$pn=$ENV{"TPCD_PHYS_NODE"};
$flatfilepath=$ENV{"TPCD_FLATFILES"};
$coldel="|";
$dblquote="";
```

```
$PairNum=$ARGV[0];
```

```
system("db2 connect to tpcd\n");
print "Beginning ....Preload of Update Function Data. Orders \n";
$str="db2 \" load from ";
$str="$str$flatfilepath${delim}lineitem.tbl.new.u" ;
$str="$str$PairNum";
$str="$str of del modified by coldel| fastparse messages
\\tmp\\TPCD\\line.msg.u";
# $str="$str$PairNum replace into TPCDTEMP.LINEITEM_NEW statistics no
nonrecoverable CPU_PARALLELISM 8";
# $str="$str$PairNum replace into TPCDTEMP.LINEITEM_NEW statistics yes
nonrecoverable DATA BUFFER 16 CPU_PARALLELISM 8 ";
$str="$str$PairNum replace into TPCDTEMP.LINEITEM_NEW statistics no
nonrecoverable ";
$str="$str partitioned db config mode load_only partitioning_dbpartnums
(0,1,2,3,4,5,6,7)";
$str="$str part_file_location $flatfilepath \" " ;
```

```
print "$str \n";
$ret=system($str);
system("db2 commit\n");
if ($ret == 0)
{
    print "Preload Orders updates completed successfully.\n";
}
else
{
    print "Preload Orders updates failed. ret=$ret\n";
    exit -1;
}
```

```
print "Beginning ....Preload of Update Function Data. Lineitem \n";
$str="db2 \" load from ";
$str="$str$flatfilepath${delim}order.tbl.new.u" ;
$str="$str$PairNum";
$str="$str of del modified by coldel| fastparse messages
\\tmp\\TPCD\\orders.msg.u";
# $str="$str$PairNum replace into TPCDTEMP.ORDERS_NEW statistics no
nonrecoverable DATA BUFFER 16 CPU_PARALLELISM 8";
# $str="$str$PairNum replace into TPCDTEMP.ORDERS_NEW statistics yes
nonrecoverable DATA BUFFER 16 CPU_PARALLELISM 8 ";
$str="$str$PairNum replace into TPCDTEMP.ORDERS_NEW statistics no
nonrecoverable ";
$str="$str partitioned db config mode load_only partitioning_dbpartnums
(0,1,2,3,4,5,6,7)";
$str="$str part_file_location $flatfilepath \" " ;
```

```
print "$str \n";
$ret=system($str);
system("db2 commit\n");
if ($ret == 0)
{
    print "Preload Lineitem updates completed successfully.\n";
}
else
{
    print "Preload Lineitem updates failed. ret=$ret\n";
    exit -1;
}
```

```
# print "$str\n";
# $ret=system($str);
# system("db2 commit\n");
if ($ret == 0)
{
    print "Preload Orders updates completed successfully.\n";
}
else
{
    print "Preload Orders updates failed. ret=$ret\n";
    exit -1;
}
```

load_UF2_data_V8

```
: # -*-Perl-*-
eval `exec perl5 -S $0 ${1+"$@"}` # Horrible kludge to convert this
    if 0;                          # into a "portable" perl script

# usage perl loadUFD [update pair number] [nodenumber]

push(@INC, split(':', $ENV{'PATH'}));

# Get TPC-D specific environment variables
require 'getvars';

# Use the macros in here so that they can handle the platform differences.
# macro.pl should be sourced from cmvc, other people wrote and maintain it.
require "macro.pl";
```

```
# Make output unbuffered.
select(STDOUT);
$| = 1 ;

if (length($ENV{"TPCD_AUDIT_DIR"}) <= 0)
{
    die "TPCD_AUDIT_DIR environment variable not set\n";
}
if (length($ENV{"TPCD_DBNAME"}) <= 0)
{
    die "TPCD_DBNAME environment variable not set\n";
}
if (length($ENV{"TPCD_SF"}) <= 0)
{
    die "TPCD_SF environment variable not set\n";
}
if (length($ENV{"TPCD_PLATFORM"}) <= 0)
{
    die "TPCD_PLATFORM environment variable not set\n";
}
if (length($ENV{"TPCD_PATH_DELIM"}) <= 0)
{
    die "TPCD_PATH_DELIM environment variable not set\n";
}
if (length($ENV{"TPCD_PRODUCT"}) <= 0)
{
    die "TPCD_PRODUCT environment variable not set\n";
}
if (length($ENV{"TPCD_AUDIT"}) <= 0)
{
    die "Must set TPCD_AUDIT env't var. Real audit timing sequence run if
yes\n";
}
if (length($ENV{"TPCD_PHYS_NODE"}) <= 0)
{
    die "TPCD_PHYS_NODE env't var not set\n";
}

#set up local variables
$auditDir=$ENV{"TPCD_AUDIT_DIR"};
$dbname=$ENV{"TPCD_DBNAME"};
```

```
$sf=$ENV{"TPCD_SF"};
$platform=$ENV{"TPCD_PLATFORM"};
$delim=$ENV{"TPCD_PATH_DELIM"};
$gatherstats=$ENV{"TPCD_GATHER_STATS"};
$product=$ENV{"TPCD_PRODUCT"};
$RealAudit=$ENV{"TPCD_AUDIT"};
$inlistmax=$ENV{"TPCD_INLISTMAX"};
$pn=$ENV{"TPCD_PHYS_NODE"};
$flatfilepath=$ENV{"TPCD_FLATFILES"};
$coldel="|";
$dblquote="";
```

```
$PairNum=$ARGV[0];
```

```
print "Beginning ....Preload of Update Function Data. Deletes\n";
```

```
system("db2 connect to tpcd\n");
$str="db2 \" load from ";
$str="$str$flatfilepath$ {delim}delete.new." ;
$str="$str$PairNum";
$str="$str of del modified by coldel| fastparse messages \\tmp\\TPCD\\del.msg.u";
#$str="$str$PairNum replace into TPCDTEMP.ORDERS_DEL statistics yes
nonrecoverable DATA BUFFER 16 CPU_PARALLELISM 8";
$str="$str$PairNum replace into TPCDTEMP.ORDERS_DEL statistics no
nonrecoverable ";
$str="$str partitioned db config mode load_only output_dbpartnums
(0,1,2,3,4,5,6,7)";
$str="$str part_file_location $flatfilepath \" " ;
```

```
print "$str\n";
$ret=system($str);
system("db2 commit\n");
if ($ret == 0)
{
    print "Preload Deletes updates completed successfully.\n";
}
else
{
    print "Preload Deletes updates failed. ret=$ret\n";
    exit -1;
}
```

loadSampleUFData

```
#!/usr/bin/perl
# usage LoadSampleUFData
```

```
($myName = $0) =~ s@.*@/@; $usage="
Usage: loaddata [nodenumber] [source dir]
      nodenumber = local node number
      source dir = source dir to load data from\n";
```

```
die $usage if (@ARGV > 2);
```

```
push(@INC, split(':', $ENV{'PATH'}));
```

```
# Get TPC-D specific environment variables
require 'getvars';
```

```
# Use the macros in here so that they can handle the platform differences.
# macro.pl should be sourced from cmvc, other people wrote and maintain it.
require "macro.pl";
```

```
# Make output unbuffered. I'm not sure why we would want to do this but many
# of the fvt testcases do it.
select(STDOUT);
$| = 1 ;
```

```
open(OUTFILE, ">temp_UF_load.bat");
## print OUTFILE "cd d:\tpch\ddl\n";
print OUTFILE "call doUFload_v8 1 30\n";
```

```

    print OUTFILE "call doUfload_v8 2 30\n";
    print OUTFILE "call runstats_ufl\n";
    close(OUTFILE);
system("temp_UF_load.bat");
1;

```

runpower

```

: # -*-Perl-*-
eval 'exec perl5 -S $0 ${1+"$@"}' # Horrible kludge to convert this
    if 0;          # into a "portable" perl script

# usage  runpower [UF]
# where UF is the optional parameter that says to run the power test
# with the update functions.  By default, the update functions are not
# run

push(@INC, split(':', $ENV{'PATH'}));

# Get TPC-D specific environment variables
require 'getvars';

# Use the macros in here so that they can handle the platform differences.
# macro.pl should be sourced from cmvc, other people wrote and maintain it.
require "macro.pl";
require "tpcdmacro.pl";

# Make output unbuffered.
select(STDOUT);
$|= 1;

if (@ARGV > 0)
{
    $runUF=$ARGV[0];
}
else
{
    $runUF="no";
}

if (length($ENV{"TPCD_AUDIT_DIR"}) <= 0)
{
    die "TPCD_AUDIT_DIR environment variable not set\n";
}
if (length($ENV{"TPCD_RUN_DIR"}) <= 0)
{
    die "TPCD_RUN_DIR environment variable not set\n";
}
if (length($ENV{"TPCD_DBNAME"}) <= 0)
{
    die "TPCD_DBNAME environment variable not set\n";
}
if (length($ENV{"TPCD_RUNNUMBER"}) <= 0)
{
    die "TPCD_RUNNUMBER environment variable not set\n";
}
if (length($ENV{"TPCD_SF"}) <= 0)
{
    die "TPCD_SF environment variable not set\n";
}
if (length($ENV{"TPCD_PLATFORM"}) <= 0)
{
    die "TPCD_PLATFORM environment variable not set\n";
}
if (length($ENV{"TPCD_PATH_DELIM"}) <= 0)
{
    die "TPCD_PATH_DELIM environment variable not set\n";
}
if (length($ENV{"TPCD_PRODUCT"}) <= 0)
{
    die "TPCD_PRODUCT environment variable not set\n";
}
if (length($ENV{"TPCD_AUDIT"}) <= 0)

```

```

{
    die "Must set TPCD_AUDIT env't var.  Real audit timing sequence run if
yes\n";
}
if (length($ENV{"TPCD_PHYS_NODE"}) <= 0)
{
    die "TPCD_PHYS_NODE env't var not set\n";
}
if (length($ENV{"TPCD_LOG_DIR"}) <= 0)
{
    $ENV{"TPCD_LOG_DIR"} = "NULL";
}
if (length($ENV{"TPCD_MODE"}) <= 0)
{
    die "TPCD_MODE environment variable not set - uni/smp/mln\n";
}
if (length($ENV{"TPCD_ROOTPRIV"}) <= 0)
{
    die "TPCD_ROOTPRIV environment variable not set - yes/no\n";
}

#set up local variables
$runNum=$ENV{"TPCD_RUNNUMBER"};
$runDir=$ENV{"TPCD_RUN_DIR"};
$auditDir=$ENV{"TPCD_AUDIT_DIR"};
$dbname=$ENV{"TPCD_DBNAME"};
$sf=$ENV{"TPCD_SF"};
$platform=$ENV{"TPCD_PLATFORM"};
$delim=$ENV{"TPCD_PATH_DELIM"};
$gatherstats=$ENV{"TPCD_GATHER_STATS"};
$product=$ENV{"TPCD_PRODUCT"};
$RealAudit=$ENV{"TPCD_AUDIT"};
$sinlistmax=$ENV{"TPCD_INLISTMAX"};
$pn=$ENV{"TPCD_PHYS_NODE"};
$logDir=$ENV{"TPCD_LOG_DIR"};
$rootPriv=$ENV{"TPCD_ROOTPRIV"};
$mode=$ENV{"TPCD_MODE"};
if (( $mode eq "uni" ) || ( $mode eq "smp" ))
{
    $all_ln="once";
    $all_pn="once";
    $once="once";
}
else
{
    $all_ln="all_ln";
    $all_pn="all_pn";
    $once="once";
}

if ($sinlistmax eq "default")
{
    $sinlistmax = 400;
}

# the auditruns directory is where we have already generate the sql files for the
# updates and the power tests

# append isolation level information about tpcdbatch to the miso file
# the miso file is created here but appended to for power and throughput
#information

$misofile="$runDir${delim}miso$runNum";
if ( -e $misofile )
{
    &rm("$misofile");
}
# if we are in real audit mode then we must start the db manager now since
# there must be no activity on the database between the time the build script
# has finished and the time the power test is started
if ( $RealAudit eq "yes" )
{
    system("db2start");
}

```

```

system("db2 activate database $dbname");
sleep 10;
}

# do not activate the database
#if ( $RealAudit ne "yes" )
#{
#   system("db2 activate database $dbname");
#}

#Report current log info to the run# directory in a file called startLog.Info
#system("perl getLogInfo.pl startLog");
system("getlog.bat startlog");

open(MISO, ">$misofile") || die "Can't open $misofile: $!\n";
$curTs = `perl gettimestamp "long"`;
print MISO "Timestamp and isolation level of tpcdbatch before power run at :
$curTs\n";
close(MISO);
if ( $product eq "pe" )
{
    system("db2 \"connect to $dbname\"; db2 \"select
name,creator,valid,unique_id,isolation from sysibm.sysplan where name like
'TPCD%\"; db2 connect reset; db2 terminate >> $runDir${delim}miso$runNum
\");
}
else
{
    &verifyTPCDBatch("$misofile", "$dbname");
}

if ($platform eq "aix")
{
    # Create the sysunused file. This reports what disks are attached, and which
    # ones are being used. Its use spans both the runpower and runthroughput tests
    system("echo \"The following disks are assigned to the indicated volume
groups\" > $runDir/sysunused$runNum") && die "cannot create
$runDir/sysunused$runNum";

    system("lspsv >> $runDir/sysunused$runNum");
    system("echo \"The following volume groups are currently online\" >>
$runDir/sysunused$runNum");
    $curTs = `perl gettimestamp "long"`;
    system("echo \"$curTs\" >> $runDir/sysunused$runNum");
    system("lsvg -o >> $runDir/sysunused$runNum");
    # show the disks that are used/unused
    #system("getdisks \"Before the start of the Power Test\");

}
else
{
    # for all other platforms
    system("echo Assume that all portions of the system are used >>
$runDir${delim}sysunused$runNum");
}

&getConfig("p");
if ( $rootPriv eq "yes" )
{
    # get the o/s tuning parameters...currently AIX only and only if your
    # user has root privileges to run this
    &getOSTune("p");
}
if ($gatherstats eq "on")
{
    # gather vm io and net stats
    if ($platform eq "aix" || $platform eq "sun" || $platform eq "ptx" ||
        $platform eq "hp" || $platform eq "linux")
    {
        # gather vmstats and iostats (and net stats if in mpp mode)
        system("perl getstats p &");
    }
}
else
{
    print "Stats gather not set up for current platform $platform\n";
}

# print to screen what type of run is running and set variables to run
# the query and update streams in parallel
if ($runUF ne "UF")
{
    $semcontrol = "off";
    print "Beginning power stream.....no update functions\n";

    $streamEx = "";
    $streamExNT = "";
}
else
{
    $semcontrol = "on";
    print "Beginning power stream.....with update functions\n";
    if ( $platform eq "nt" )
    {
        $streamExNT = "start /b";
        $streamEx = "";
    }
    else
    {
        $streamExNT = "";
        $streamEx = "&";
    }
}

# bbe This new line (below) runs queries for power test

print "Starting tpcdbatch...\n";
$ret=system("$streamExNT $auditDir${delim}auditruns${delim}tpcdbatch -d
$dbname -f $runDir${delim}qtextpow.sql -r on -b on -s $sf -u p1 -m $inlistmax -n
0 -p $semcontrol $streamEx");

if ( $runUF eq "UF" )
{
    $ret2 = system("$auditDir${delim}auditruns${delim}tpcdbatch -d $dbname -f
$runDir${delim}qtextquf.sql -r on -b on -s $sf -u p2 -m $inlistmax -n 0");
}
else
{
    $ret2 = 0; # If UFs were not running, then the stream cannot fail
}

if (($ret2 == 0) && ($ret == 0))
{
    print "Power stream completed succesfully.\n";
}
else
{
    print "Power stream failed. ret=$ret\n";
}

if ($platform eq "aix")
{
    # show that the same disks are still used or unused
    # system("getdisks \"After completion of the Power Test\");

    #clean up
}
if ($gatherstats eq "on")
{
    # gather vm io and net stats
    if ($platform eq "aix" || $platform eq "sun" || $platform eq "ptx" || $platform eq
        "linux")
    {

```



```

# kill the stats that were being gathered
if ($platform eq "ptx")
{
    $src= `perl5 zap "-f" "sar";`
    $src= `perl5 zap "-f" "sadc";`
}
else
{
    $src= `perl5 zap "-f" "vmstat";`
    $src= `perl5 zap "-f" "iostat";`
}
if ( $pn > 1 )
{
    $src= `perl5 zap "-f" "netstat";`
}
$src= `perl5 zap "-f" "getstats";`
}
}

open(MISO, ">>$misofile") || die "Can't open $misofile: $!\n";
$curTs = `perl gettimestamp "long";`
print MISO "Timestamp and isolation level of tpcdbatch after power run at :
$curTs\n";
close(MISO);

if ( $product eq "pe" )
{
    system("db2 \"connect to $dbname\"; db2 \"select
name,creator,valid,unique_id,isolation from sysibm.sysplan where name like
'TPCD%'\";db2 connect reset;db2 terminate >>
$runDir${delim}miso$runNum");
}
else
{
    &verifyTPCDBatch("$misofile", "$dbname");
}
if ( $RealAudit ne "yes" )
{
    $curTs = `perl gettimestamp "short";`
    # grab the db and dbm snapshot before we deactivate
    system("db2 get snapshot for all on $dbname >
$runDir${delim}dbrun$runNum.snap.$curTs");
    system("db2 get snapshot for database manager >>
$runDir${delim}dbrun$runNum.snap.$curTs");
}

#####

# now copy the reports from the count of streams files into one final file
&cat("$runDir${delim}pstrcnt*","$runDir${delim}mpstrcnt$runNum");
#(NOTE: there is a dependency that this mpstrcnt file exist before the
# calcmetrics.pl script is called, both because it is used as input for
# calcmetrics.pl, and because the output from calcmetrics is used as
# the trigger for watchstreams to complete, and watchstreams cats its
# output at the end of the mstrcnt file.

# generate the mpinter?.metrics file in the run directory
#require 'calcmetrics.pl';
if ( $runUF eq "UF" )
{
    system("perl calcmetrics.pl UF");
}
else
{
    system("perl calcmetrics.pl");
}

# concatenate all the throughput inter files that were used to
# generate these results into the calcmetrics output file (mpinterX.metrics)
#cd $TPCD_RUN_DIR
&cat("$runDir${delim}mpqinter*","$runDir${delim}mpinter$runNum.metrics");

if ($runUF eq "UF") {
    &cat("$runDir${delim}mpufinter*","$runDir${delim}mpinter$runNum.metrics")
    ;
    ;
}

# if ($runUF eq "no") {
#   &rm("$runDir${delim}mpuf*");
# }

#####

# no longer activate/deactivate the database
# if ( $RealAudit ne "yes" )
# {
#   # deactivate the database
#   system("db2 deactivate database $dbname");
# }

# do not stop the database after the power test
# if ( $RealAudit ne "yes" )
# {
#   system("db2stop");
# }

1;

sub getConfig
{
    $testtype=$_[0];
    print "Getting database configuration.\n";
    $dbtune="$runDir${delim}m${testtype}dbtune${runNum}";
    open(DBTUNE, ">$dbtune") || die "Can't open $dbtune: $!\n";
    $timestamp=`perl gettimestamp "long";`
    print DBTUNE "Database and Database manager configuration taken at :
$timestamp";
    close(DBTUNE);
    system("db2level >> $dbtune");
    system("db2 get database configuration for $dbname >> $dbtune");
    system("db2 get database manager configuration >> $dbtune");
    system("db2set >> $dbtune");
    if (( $mode eq "mln" ) || ( $mode eq "mpp" ))
    {
        $cfgfile="$runDir${delim}dbtune${runNum}.";
        #removed by Alex due to hang
        #system("db2 _all \"||\" typeset -i ln=###; db2 get db cfg for $dbname >
$cfgfile${ln} ; db2 get dbm cfg >> $cfgfile${ln} ; db2set >> $cfgfile${ln} ; db2
terminate \"");
    }
}

sub getOSTune
{
    $testtype=$_[0];
    if ( $platform eq "aix" )
    {
        print "Getting OS and VMdatabase configuration.\n";
        $ostune="$runDir${delim}m${testtype}ostune${runNum}";
        open(OSTUNE, ">$ostune") || die "Can't open $ostune: $!\n";
        $timestamp=`perl gettimestamp "long";`
        print OSTUNE "Operating System and Virtual Memory configuration taken at
: $timestamp";
        close(OSTUNE);
        system("$ ${delim}usr${delim}samples${delim}kernel${delim}schedtune >>
$ostune");
        system("$ ${delim}usr${delim}samples${delim}kernel${delim}vmtune >>
$ostune");
    }
    else
    {
        print "OS parameters retrieval not supported for $platform \n";
    }
}

```

```

sub verifyTPCDBatch
{
    $logfile=$_[0];
    $dbname=$_[1];
    $file="verifytpcdbatch.clp";
    open(VERTBL, ">$file") || die "Can't open $file: $!\n";
    print VERTBL "connect to $dbname;\n";
    print VERTBL "select name,creator,valid,last_bind_time,isolation from
sysibm.sysplan where name like 'TPCD%'\n";
    print VERTBL "connect reset;\n";
    print VERTBL "terminate;\n";
    close(VERTBL);
    system("db2 -vtf $file >> $logfile");
}

```

runthroughput

```

: # *-Perl-*
eval `exec perl5 -S $0 ${1+"$@"}' # Horrible kludge to convert this
    if 0;          # into a "portable" perl script

# usage runthroughput [UF]
# where UF is the optional parameter that says to run the throughput test
# with the update functions. By default, the update functions are not
# run
# If UF is not supplied and a number is supplied, then that number is taken
# as the number of concurrent throughput streams to run. This is also optional

push(@INC, split(':', $ENV{'PATH'}));

# Get TPC-D specific environment variables
require 'getvars';

# Use the macros in here so that they can handle the platform differences.
# macro.pl should be sourced from cmvc, other people wrote and maintain it.
require "macro.pl";
require "tpcdmacro.pl";

$runUF="no";
if (@ARGV > 0)
{
    if ($ARGV[0] eq "UF")
    {
        $runUF=$ARGV[0];
    }
}

@reqVars = ("TPCD_AUDIT_DIR",
            "TPCD_RUN_DIR",
            "TPCD_DBNAME",
            "TPCD_RUNNUMBER",
            "TPCD_SF",
            "TPCD_PLATFORM",
            "TPCD_PATH_DELIM",
            "TPCD_PRODUCT",
            "TPCD_AUDIT",
            "TPCD_PHYS_NODE",
            "TPCD_MODE",
            "TPCD_ROOTPRIV",
            "TPCD_NUMSTREAM");
##      "TPCD_NUMSTREAM",
##      "TPCD_RUN_ON_MULTIPLE_NODES");

&setVar(@reqVars, "ERROR");

if (length($ENV{"TPCD_LOG_DIR"}) <= 0)
{
    $ENV{"TPCD_LOG_DIR"} = "NULL";
}

#set up local variables

```

```

$runNum=$ENV{"TPCD_RUNNUMBER";
$numStream=$ENV{"TPCD_NUMSTREAM";
$runDir=$ENV{"TPCD_RUN_DIR";
$auditDir=$ENV{"TPCD_AUDIT_DIR";
$dbname=$ENV{"TPCD_DBNAME";
$sf=$ENV{"TPCD_SF";
$product=$ENV{"TPCD_PRODUCT";
$platform=$ENV{"TPCD_PLATFORM";
$delim=$ENV{"TPCD_PATH_DELIM";
$realAudit=$ENV{"TPCD_AUDIT";
$inlistmax=$ENV{"TPCD_INLISTMAX";
$gatherstats=$ENV{"TPCD_GATHER_STATS";
$logDir=$ENV{"TPCD_LOG_DIR";
$rootPriv=$ENV{"TPCD_ROOTPRIV";
$mode=$ENV{"TPCD_MODE";

```

```
$path="$auditDir${delim}auditruns";
```

```
if (( $mode eq "uni" ) || ( $mode eq "smp" ))
```

```
{
    $all_in="once";
    $all_pn="once";
    $once="once";
}
```

```
else
```

```
{
    $all_in="all_in";
    $all_pn="all_pn";
    $once="once";
}
```

```
# return 1 if the given pattern(parameter $_[0]) matches any file
```

```
sub existfile {
    if ($platform eq "aix" || $platform eq "sun" || $platform eq "ptx" || $platform eq
"linux")
    {
        `ls $_[0] 2> /dev/null | wc -l` + 0 != 0;
    }
    else
    {
        `dir /b $_[0] 2> NUL | wc -l` + 0 != 0;
    }
}
```

```
if ($inlistmax eq "default")
```

```
{
    $inlistmax = 400;
}
```

```
# no longer stop and start the dbm between runs when not in realaudit mode
```

```
#if ( $RealAudit ne "yes" )
```

```
{
```

```
# # if we are not in real audit mode then we must start the db manager now
```

```
# system("db2start");
```

```
# # activate the database
```

```
# system("db2 activate database $dbname");
```

```
#}
```

```
$misofile="$runDir${delim}miso$runNum";
```

```
# append isolation level information about tpcdbatch to the miso file
```

```
open(MISO, ">>$misofile") || die "Can't open $misofile: $!\n";
```

```
$curTs = `perl gettimestamp "long";`
```

```
print MISO "Timestamp and isolation level of tpcdbatch before throughput run at
```

```
: $curTs\n";
```

```
close(MISO);
```

```
if ( $product eq "pe" )
```

```
{
    system("db2 \"connect to $dbname\"; db2 \"select
name,creator,valid,unique_id,isolation from sysibm.sysplan where name like
'TPCD%'\n\" >> $runDir${delim}miso$runNum ");
}
else
{

```

```

    &verifyTPCDBatch("$misofile","$dbname");
}

# kick off the script that will monitor for the database applications during
# the running of the throughput tests. This will quit when the mtinterX.metrics
# (where X=runnumber) file has been created.

# set variables to run streams in parallel
if ( $platform eq "nt" )
{
    $streamExNT = "start /b";
    $streamEx = "";
}
else
{
    $streamExNT = "";
    $streamEx = "&";
}
if ( $platform eq "aix" || $platform eq "sun" || $platform eq "nt" || $platform eq
"hp" || $platform eq "linux")
{
    system("$streamExNT perl watchstreams $streamEx");
}
else
{
    die "platform not supported, can't start watchstreams in background";
}

# show the disks that are used/unused
if ($platform eq "aix")
{
    system("getdisks \"Before the start of the Throughput Test\"");
}
if ($gatherstats eq "on")
{
    # gather vm io and net stats
    if ($platform eq "aix" || $platform eq "sun" || $platform eq "ptx" || $platform eq
"hp" || $platform eq "linux")
    {
        # gather vmstats and iostats (and net stats if in mpp mode)
        system("perl getstats t &");
    }
    else
    {
        print "Stats gather not set up for current platform $platform\n";
    }
}

# the auditruns directory is where we have already generated the sql files
# for the updates and the power tests

$loopStream=1;

for ( $loopStream = 1; $loopStream <= $numStream; $loopStream++)
{
    print "starting stream $loopStream\n";
    system("echo Executing stream $loopStream out of $numStream.");
    # run the queries
    if ( $platform eq "aix" || $platform eq "sun" || $platform eq "nt" || $platform eq
"ptx" ||
        $platform eq "hp" || $platform eq "linux")
    {
        system("$streamExNT $path${delim}tpcdbatch -d $dbname -f
$runDir${delim}qtextt$loopStream.sql -r on -b on -s $sf -u t1 -m $inlistmax -n
$loopStream $streamEx");
    }
    else
    {
        die "platform $platform not supported yet";
    }
}

# run the update function stream....this will wait until the queries have

```

```

# completed to kick off the updates
print "starting update stream\n";

if ($runUF eq "no") {
    $ret=system("$auditDir${delim}auditruns${delim}tpcdbatch -d $dbname -f
$runDir${delim}quft.sql -r on -b on -s $sf -u t -m $inlistmax -n $numStream");
}
else {
    $ret=system("$auditDir${delim}auditruns${delim}tpcdbatch -d $dbname -f
$runDir${delim}quft.sql -r on -b on -s $sf -u t2 -m $inlistmax -n $numStream");
}
print "update stream done\n";

&getConfig("t");
if ( $rootPriv eq "yes" )
{
    # get the o/s tuning parameters...currently AIX only and only if your
    # user has root privileges to run this
    &getOSTune("t");
}

if ($platform eq "aix")
{
    # show the disks that are used/unused
    system("getdisks \"After the completion of the Throughput Test\"");
}
if ($gatherstats eq "on")
{
    # gather vm io and net stats
    if ($platform eq "aix" || $platform eq "sun" || $platform eq "ptx" || $platform eq
"linux")
    {
        # kill the stats that were being gathered
        if ($platform eq "ptx")
        {
            $src= `perl5 zap -f "sar"`;
            $src= `perl5 zap -f "sadc"`;
        }
        else
        {
            $src= `perl5 zap -f "vmstat"`;
            $src= `perl5 zap -f "iostat"`;
        }
        if ( $pn > 1 )
        {
            $src= `perl5 zap -f "netstat"`;
        }
        $src= `perl5 zap -f "getstats"`;
    }
}

open(MISO, ">>$misofile") || die "Can't open $misofile: $!\n";
$curTs = `perl gettimestamp "long"`;
print MISO "Timestamp and isolation level of tpcdbatch after throughput run at :
$curTs\n";
close(MISO);

if ( $product eq "pe" )
{
    system("db2 \"connect to $dbname\"; db2 \"select
name,creator,valid,unique_id,isolation from sysibm.sysplan where name like
'TPCD%'\>> $runDir${delim}miso$runNum");
}
else
{
    &verifyTPCDBatch("$misofile","$dbname");
}

if ( $RealAudit ne "yes" )
{
    $curTs = `perl gettimestamp "short"`;
    # grab the db and dbm snapshot before we deactivate

```

```

system("db2 get snapshot for all on $dbname >
$runDir${delim}dbTrun$runNum.snap.$curTs");
system("db2 get snapshot for database manager >>
$runDir${delim}dbTrun$runNum.snap.$curTs");
}

# now copy the reports from the count of streams files into one final file
&cat("$runDir${delim}strcnt*","$runDir${delim}mstrcnt$runNum");
#(NOTE: there is a dependancy that this mstrcnt file exist before the
# calcmetrics.pl script is called, both because it is used as input for
# calcmetrics.pl, and because the output from calcmetrics is used as
# the trigger for watchstreams to complete, and watchstreams cats its
# output at the end of the mstrcnt file.

# generate the mtinter?.metrics file in the run directory
#require 'calcmetrics.pl';

if ( $runUF ne "no")
{
system("perl calcmetrics.pl $numStream UF");
}
else
{
system("perl calcmetrics.pl $numStream");
}

# concatenate all the throughput inter files that were used to
# generate these results into the calcmetrics output file (mtinterX.metrics)
#cd STPCD_RUN_DIR
&cat("$runDir${delim}mts*inter*","$runDir${delim}mtinter$runNum.metrics");

if ($runUF ne "no") {

&cat("$runDir${delim}mtufinter*","$runDir${delim}mtinter$runNum.metrics");
}

if (&existfile("$runDir${delim}mp*")) {
# generate the mplot stuff
system("perl gen_mplot");

# generate the mlog information file
require 'buildmlog';
}

#if ($runUF eq "no") {
# &rm("$runDir${delim}mtuf*");
#}

# deactivate the database this needs to remain at the end of run throughput so
# asynchronous writing of the log files completes.
system("db2 deactivate database $dbname");
$src=&dodb_noconn("db2 get db cfg for $dbname | grep -i log >>
$runDir${delim}endLog.Info",$all_ln);
if ( $logDir ne "NULL" )
{
$src=&dodb_noconn("$dircmd $logDir >>
$runDir${delim}endLog.Info",$all_ln);
}

#system("db2_all `}]`db2 get db cfg for tpcd | grep -i log >>
$runDir${delim}endLog.Info ; db2 terminate\ '");
#system("ls -ltr /node??vg.log/NODE00* >> $runDir${delim}endLog.Info");

#Create Catalog info
$src = system("perl catinfo.pl p");

if ( $src != 0 )
{
warn "catinfo failed!!!\n";
}

#Report current log info to the run# directory in a file called endLog.Info
#system("perl getLogInfo.pl endLog");

```

```

system("getlog.bat endLog");

# if we are in audit mode we must do a db2stop at the end of the
power/throughput run
if ( $RealAudit eq "yes" )
{
system("db2stop");
}

1;

sub getConfig
{
$testtype=$_[0];
print "Getting database configuration.\n";
$dbtunefile="$runDir${delim}m${testtype}dbtune${runNum}";
open(DBTUNE, ">$dbtunefile") || die "Can't open $dbtunefile: $!\n";
$timestamp=`perl gettimestamp "long"`;
print DBTUNE "Database and Database manager configuration taken at :
$timestamp";
close(DBTUNE);
system("db2level >> $dbtunefile");
system("db2 get database configuration for $dbname >> $dbtunefile");
system("db2 get database manager configuration >> $dbtunefile");
system("db2set >> $dbtunefile");
}

sub getOSTune
{
$testtype=$_[0];
if ( $platform eq "aix" || $platform eq "linux")
{
print "Getting OS and VMdatabase configuration.\n";
$ostunefile="$runDir${delim}m${testtype}ostune${runNum}";
open(OSTUNE, ">$ostunefile") || die "Can't open $ostunefile: $!\n";
$timestamp=`perl gettimestamp "long"`;
print OSTUNE "Operating System and Virtual Memory configuration taken at
: $timestamp";
close(OSTUNE);
system("$delim usr$delim samples$delim kernel$delim schedtune >>
$ostunefile");
system("$delim usr$delim samples$delim kernel$delim vmtune >>
$ostunefile");
}
else
{
print "OS parameters retrieval not supported for $platform \n";
}
}

sub verifyTPCDBatch
{
$logfile=$_[0];
$dbname=$_[1];
$file="verifytpcdbatch.clp";
open(VERTBL, ">$file") || die "Can't open $file: $!\n";
print VERTBL "connect to $dbname;\n";
print VERTBL "select name,creator,valid,last_bind_time,isolation from
sysibm.sysplan where name like 'TPCD%'\n";
print VERTBL "connect reset;\n";
print VERTBL "terminate;\n";
close(VERTBL);
system("db2 -vtf $file >> $logfile");
}

tpcd_cl.bat

erase tpcdUF.c
erase tpcdUF.obj
erase tpcdbatch.c
erase tpcdbatch.obj

```

```
erase tpcdbatch.map
```

```
set db2options=+c -t +p -v
db2start
db2 connect to %1
db2 prep tpcdbatch.sqc bindfile package isolation rr blocking all OPTLEVEL 1
DATETIME ISO
db2 prep tpcdUF.sqc bindfile package isolation rr blocking all OPTLEVEL 1
DEGREE 1 DATETIME ISO
db2 connect reset
db2 terminate
REM make sure LIBPATH is set to include the compiler libraries and db2
libraries
cl -c -Z7 -DSQLWINT -W3 -J tpcdbatch.C
cl -c -Z7 -DSQLWINT -W3 -J tpcdUF.C
link -debug -out:tpcdbatch.exe tpcdbatch.obj tpcdUF.obj user32.lib kernel32.lib
db2api.lib -subsystem:console
```

tpcdbatch.h

```
/*
*****
*
*   TPCDBATCH.H
*
*   Revision History:
*
*   * 27 may 99 bbe from (24 nov 98 jen) fixNTtimestamp - fixed NT timestamp to
print millisecond correctly
*   * 27 may 99 bbe from (10 dec 98 jen) SUN - added Haider's changes necessary
for SUN
*   * 17 jun 99 jen Increased version to 5.1
*   * 10 aug 99 bbe Increased version to 5.2
*   * 13 aug 99 bbe Increased version to 5.3
*   * 18 mar 02 ken Increased version to 5.7
*****
*****/

/** Necessary header files **/

/** System header files **/
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>
#include <fcntl.h> /* SUN bbe */

#include <time.h>
#include <ctype.h>
#if (defined(SQLAIX) || defined(SQLPTX) || defined(LINUX) || defined(SQLHP))
#include <unistd.h> /* SUN */
#include <sys/stat.h> /* SUN */
#endif
#if ((defined(SQLAIX) || defined(SQLPTX)) && !defined(LINUX))
#include <sys/vnode.h> /* SUN */
#endif
#ifndef SQLWINT
#include <sys/time.h> /*@d33143aha*/
#include <sys/ipc.h>
#include <sys/sem.h>
#if (!defined(SQLPTX) && !defined(LINUX)&& !defined(SQLHP))
#include <sys/mode.h>
#endif
#include <sys/timeb.h>
#include <sys/types.h>
#else
#include <windows.h>
#include <sys/timeb.h>
#endif
#include <errno.h>

/** External header files **/
```

```
#include "sqlda.h"
#include "sqlenv.h"
#include "sql.h"
#include "sqlmon.h"
#include "sqlca.h"
#include "sqlutil.h"
#include "sqlcodes.h"

/** Internal header files **/
/** #ifdef __cplusplus **/
/** #include "sqlz.h" **/
/** #include "sqlzcopy.h" **/
/** #endif **/

*****
*****/
/* Define synonyms here */
*****
#define TPCDBATCH_VERSION "5.7"

#define TPCDBATCH_NONSQL 10 /* @d23684 tlg */
#define TPCDBATCH_SELECT 20
#define TPCDBATCH_NONSELECT 30
#define TPCDBATCH_EOBLOCK 40 /* @d30369 tlg */
#define TPCDBATCH_INSERT 50
#define TPCDBATCH_DELETE 60

#define TPCDBATCH_MAX_COLS 100 /* @d30369 tlg */

#define TPCDBATCH_CHAR char

#define TPCDBATCH_PRINT_FLOAT_WIDTH 20
/* kmw - allow 15 whole digit for %#.3f format */
/* - note: use > 18, size of long identifier so that it will */
/* be larger than any column heading */
#define TPCDBATCH_PRINT_FLOAT_MAX 1e15 /* kmw */
/* #define TPCD_PREPARETIME 1 */ /* for separate prep/exec on uf jen
1106 */

#ifdef SQLWINT
#define PATH_DELIM '\\'
// #define sleep(a) Sleep((a)*1000)
#define sleep(a) Sleep(a)
#else
#define PATH_DELIM '/'
#endif

#define PARALLEL_UPDATES 1

#ifdef PARALLEL_UPDATES
#define UF1OUTSTREAMPATTERN "%s%cuf1.%02d.%d.out"
#define TPCD_NONPARTITIONED
#define UF2OUTSTREAMPATTERN "%s%cuf2.%02d.%d.out"
#else
/* kelly add same as NONPART. */
#define UF2OUTSTREAMPATTERN "%s%cuf2.%02d.%d.out"
/* kelly ... take this out ... should be same name as for non-partitioned
#define UF2OUTSTREAMPATTERN "%s%cuf2.%02d.%d.%d.out" */
/*DELjen add delchunk*/
#endif
#define BUFSIZE 1024
#endif

#define T_STAMP_FORM_1 1
#define T_STAMP_FORM_2 2
/* jen TIME_ACC start */
#define T_STAMP_FORM_3 3
#define T_STAMP_ILLEN 17
#if defined (SQLUNIX) || defined (SQLAIX) || defined (SQLHP)
#define T_STAMP_3LEN 24
```

```

#elif (defined (SQLOS2) || defined (SQLWINT) || defined (SQLWIN) ||
defined (SQLDOS))
#define T_STAMP_3LEN 21 /* WIN NT timestamp fix bbe */
#else
#error Unknown operating system
#endif
/* jen TIME_ACC start */

#define BLANKS ""
#define READMODE "r"
#define WRITEMODE "w"
#define APPENDMODE "a"
#define mem_error(xx) \
{ fprintf(stderr, "n--Out of memory when %s.\n", xx); }
/* Display out-of-memory and end */

#define TPCDBATCH_MIN(x,y) ((x) < (y) ? (x) : (y))
/** Returns the smaller of both x and y */
#define TPCDBATCH_MAX(x,y) ((x) > (y) ? (x) : (y)) /* @d22817 tlg */
/** Returns the larger of both x and y */

/** Defines needed for decimal conversion */
#define SQLZ_DYNLINK
#define TRUE 1
#define LEFT 1
#define RIGHT 0
#define FALSE 0
#define sqlrx_get_left_nibble(byte) (((unsigned char)(byte)) >> 4)

#define sqlrx_get_right_nibble(byte) (((unsigned char)(byte) & 'x0f'))
#define SQL_MAXDECIMAL 31
#define SQLRX_PREFERRED_PLUS 0x0c

/** Timer-necessary defines for portability */
#if (defined (SQLOS2) || defined (SQLWINT) || defined (SQLWIN) ||
defined (SQLDOS))
typedef struct timeb Timer_struct;
#elif (defined (SQLUNIX) || defined (SQLAIX) || defined (SQLHP))
/*TIMER jen*/
typedef struct timeval Timer_struct;
#else
#error Unknown operating system
#endif

/* sleep time between starting subsequent tpcdbatches running UF1 and UF2 */
#define UF1_SLEEP 1
#define UF2_SLEEP 1
#define UF_DEADLOCK_SLEEP 1 /* sleep between deadlock retries in
UF1,UF2 */

#define MAXWAIT 50 /* maximum retries for deadlock encounters */

#define DEBUG 0 /* to be set to 1 for diagnostic purposes if needed */
/* #define UF1DEBUG 1 */
/* #define UF2DEBUG 1 */

```

tpcdbatch.sqc

```

/*****
*****
*
* TPCDBATCH.SQC
*
* Revision History:
*
* 21 Dec 95 jen Corrected calculation of geometric mean to include in the
count of statements the update functions.
* 03 Jan 96 jen Corrected calculation of arithmetic mean to not include the
timings for the update functions. (only want query timings

```

```

* as part of arithmetic mean)
* 15 Jan 96 jen Added extra timestamps to the update functions.
* 22 Jan 96 jen Get rid of checking of short_time....we always use the long
timings.
* Fixed timings to print query/uf times rounded up to 0.1 seconds
and uses these rounded time values in subsequent calculations
* Fixed bug where last seed in mseedme file wasn't getting read
correctly - EOF processing done too soon.
*
* 22 Feb 96 kbs port to NT
* 26 Mar 96 kbs Fix to avoid counting UFs as queries for min max
* 27 Jun 97 wlc Temporarily fixed deadlock problems when doing UF1, UF2
* 30 Jul 97 wlc Add in support for load_update and TPCD_SPLIT_DELETES
* 13 Aug 97 wlc fixed UF1 log file formatting problem,
* using TPCD_TMP_DIR for temp files instead of /tmp,
* make summary table fit in 80-column,
* fixed UF2 # of deleted rows reporting problem
* 18 Aug 97 wlc added command line support for inlistmax
* 20 Aug 97 wlc added support for runthroughput without UF
* 27 Aug 97 aph Replaced hardcoded 'tpcaudit' with
getenv("TPCD_AUDIT_DIR")
* 05 Sep 97 wlc fixing free() problem in NT
* 26 Sep 97 kmw change FLOAT processing in echo_sqlda and print_headings
* 10 Oct 97 jen add lock table in share mode for staging tables
* 21 Oct 97 jen added explicit rollback on failure of ufl
* 27 Oct 97 jen don't update TPCD.xxxx.update.pair.num if not running UFs in
throughput run
* 01 Nov 97 jen temp code to do a prep then execute stmt in UFs so we can
get timings
* 03 Nov 97 jen realigned UF code for readability
* pushed UF2 commit into loop for inlistmax
* fixed UF2 code so rollback performed
* 04 Nov 97 jen Added code to handle vldb
* 06 Nov 97 jen Commented out temp code for prep then execute stmts using
TPCD_PREPARETIME def
* Updated version number to 2.2
* send all output during update function to output files, not
stderr
* 10 Nov 97 jen jenCI Updated version number to 2.3
* Added handling of TPCD_CONCURRENT_INSERTS. Change
control of
* chunk processing to use the concurrent_inserts value as the
control. Now the inserts will be run in
TPCD_CONCURRENT_INSERTS
* sets, each having concurrent_inserts/
* 13 Nov 97 jen jen DEADLOCK. Fixed bug that Alex found where deadlock
count
* (maxwait) was incremented on every execution of the stmt as
opposed to just when deadlock really happened.
* 14 Nov 97 jen jenSEM - fix up error reporting on semaphore failure
sem_op now returns failure to caller so caller can report where
failure has happened.
* Forced dbname to be upper case, an all other parts of update
pair number to be lowercase
* 15 Nov 97 jen SEED Reworked code to grab the seed from the seed file. Now
reusing seeds between runs, so power run will always use first
seed, throughput will use the 2nd - #stream+1 seeds
*
* 13 Jan 98 jen LONG Increase stmt_str to be able to hold inlists with larger
order key numbers
* 04 Mar 98 jen IMPORT added support for TPCD_UPDATE_IMPORT to chose
whether
* using import or load api's for loading data into the staging
tables
* 04 Mar 98 jen TIMER changed from using gettimeofday to gettimeofday for unix
* 01 Apr 98 jen Fixed IMPORT code to do the proper checking on strcmp (ie
!strcmp)
* 01 Apr 98 jen removed code to handle vldb - not needed
* Upgraded version to 2.4 for ( chunk
* 01 Apr 98 jen Fixed up import code on NT so the variable is recognized in the
children
* 25 May 98 sks Reworked some of the environment variable code so consolidate
as
* much as possible. Not all complete because of differences in

```

```

* the way nt and AIX calls (and starts stuff in background) for UFs
* 29 may 98 jen REUSE_STAGE Changed UF1 so we reuse the same staging
  tables
* instead of having a new set for each update pair
* 06 jul 98 jen Removed locking of staging tables since they are created with
  locksize table now
* 06 jul 98 jen 912RETRY - added code to retry query execution on 912 as well
  as 911
* 07 jul 98 jen Fixed summary_table() so 1000x adjustment not based on UF
  (setting
* of max and min pointers
* Added generic SleepSome function to handle NT vs AIX sleep
  differences
* 01 apr 98 djf Added change to permit the use of table functions for UF1.
* to enable this set TPCD_UPDATE_IMPORT to tf in TPCD.SETUP
  file.
* MERGED this into base copy on Jul 07
* 10 jul 98 jen haider's fix for 'outstream' var for error processing in
  runUF1_fn and runUF2_fn
* Updated version to 2.5
* 25 sep 98 jen Added stream number printing into mpqry* files and increases
  accuracy of timestamp in mpqry (and mts*qry*) files
* 06 oct 98 jen TIME_ACC Added accuracy of timestamp in mpqry (and
  mts*qry*)
* files. Cleaned up misuse of Sleep and flushed buffers on
  deadlocks
* 19 oct 98 kbs fix UF2_fn to correctly count rows deleted in case of deadlock
* 20 oct 98 kbs rewrite UF2 and UF2_fn for static SQL with staging table
* 23 oct 98 jen Cleaned up retrying of order/lineitem on lineitem deadlock in
  UF1
* 24 oct 98 jen Used load_ufl and load_ufl instead of general load_updates
* 26 oct 98 kbs inject the UF1 with a single staging table
* 02 nov 98 jen Fixed processing of multiple chunks in uf2 so don't duplicate
* 21 nov 98 kmw Fixed BIGINT
* 05 dec 98 aph Moved runUF1_fn() and runUF2_fn() into a separate file
  tpcdUF.sqc
* so that it can be bound separately with a different isolation level.
* 21 dec 98 aph Integrated Jennifer's QppD calculation (rounding & adjustment)
  fixes.
* 22 dec 98 aph For UFs during Throughput run, defer CONNECT until children
  launched.
* 28 dec 98 aph Removed error_check() call after CONNECT RESET
* 29 dec 98 aph For UFs do not COMMIT in tpcdbatch.sqc. COMMITs happen
  in tpcdUF.sqc.
* 18 jan 99 kal replaced header with #include "tpcdbatch.h"
* 27 may 99 bbeaton from (03 mar 99 jen) Fixed SUN fix that wasn't compatible
  with
* NT (using %D %T instead of %x %X for strftime)
* 16 jun 99 jen Added missing LPCTSTR cast of semaphore file name for NT
* 17 jun 99 jen SEMA Changes semaphore file for update functions to look for
  tpcd.setup
* not for the orders.*** update data file
* 21 jul 99 bbeaton Added semaphore control that allows runpower to be run as
  two
* separate streams (update and query). This involves the use of
  two semaphores to be used as it executes in three different
  sections. The first is the update inserts. The next is the query
  stream which is started with the update stream, but waits until
  the inserts are complete. The third section is the update deletes
  which execute after the queries are complete.
* 21 jul 99 bbeaton Added functions to handle semaphore creation, control, etc.
* 21 jul 99 bbeaton Modified output to mp*inter files. It now only outputs
  intermediate data that will be calculated by calcmetric.pl. This
  is a result of the runpower being split into two streams and thus
  tpcdbatch not having access to all data.
* 21 jul 99 bbeaton The start time for runpower UF2 now does not start until after
  the query stream is complete so that its wait time is not included
  NOTE: The wait time that the first UF1 in runthroughput still
  includes the wait period that occurs waiting on queries.
* 18 mar 02 kentond removed the need for list files. Instead of using the *.list
  files to determine the name of the output files, the tags for the
  source sql files are used.
*****
*****/

```

```

/* included in tpcdbatch.sqc and tpcdUF.sqc */

#include "tpcdbatch.h"

/*****
*****/
/* global structure containing elements passed between different functions */
/*****
*****/
struct global_struct
{
  struct stmt_info *s_info_ptr; /* ptr to stmt_info list */
  struct stmt_info *s_info_stop_ptr; /* ptr to last struct in list */
  struct comm_line_opt *c_l_opt; /* ptr to comm_line_opt struct */
  struct ctrl_flags *c_flags; /* ptr to ctrl_flags struct */
  Timer_struct stream_start_time; /* start time for stream TIME_ACC */
  Timer_struct stream_end_time; /* end time for stream TIME_ACC */
  char file_time_stamp[50]; /* time stamp for output files */
  double scale_factor; /* scale factor of database */
  char run_dir[150]; /* directory for output files */
  int copy_on_load; /* indication of whether or not */
  /* to do use a copy directory */
  /* (equiv to COPY YES) on load */
  /* default is FALSE */
  long lSeed; /* seed used to generate the */
  /* queries for this particular */
  /* run. */
  FILE *stream_list; /* ptr to query list file */
  char update_num_file[150]; /* name of file that keeps track */
  /* of which update pairs have run */
  char sem_file[150]; /* semaphore name */
  char sem_file2[150]; /* semaphore name bbe */
  FILE *stream_report_file; /* file to report start stop */
  /* progress of the stream */
};

/*****
*****/
/* New type declaration to store details about SQL statement */
/*****
*****/
struct stmt_info
{
  long max_rows_fetch;
  long max_rows_out;
  int query_block; /* @d30369 tlg */
  unsigned int stmt_num; /* @d24993 tlg */
  double elapse_time; /* @d24993 tlg */
  double adjusted_time;
  char start_stamp[50]; /* start time stamp for block */
  char end_stamp[50]; /* end time stamp for block */
  char tag[50]; /* block tag */
  char qry_description[100];
  struct stmt_info *next; /* @d24993 tlg */
};

/*****
*****/
/* Structure containing command line options */
/*****
*****/
struct comm_line_opt
{
  /* @d22275 tlg */
  /* kjd715 */
  /* char str_file_name[256]; /* output filename */
  /* kjd715 */
  char infile[256]; /* input filename */
  int intStreamNum; /* integer version of stream number */
  int a_commit; /* auto-commit flag */
};

```

```

int      short_time; /* time interval flag */
int      update;
int      outfile;
};

/*****
***/
/* Structure used to hold precision for decimal numbers */
/*****
***/
struct declen
{ /* kmw */
    unsigned char m; /* # of digits left of decimal */
    unsigned char n; /* # of digits right of decimal */
};

/*****
***/
/* Structure containing control flags passed between functions */
/*****
***/
struct ctrl_flags
{ /* @d25594 tlg */
    int eo_infile;
    int time_stamp;
    int eo_block; /* @d30369 tlg */
    int select_status;
};

/*****
***/
/* Function Prototypes */
/*****
***/
int SleepSome( int amount );
int get_env_vars(void);
int Get_SQL_stmt(struct global_struct *g_struct);

void print_headings (struct sqlda *sqlda, int *col_lengths); /* @d22817 tlg */
void echo_sqlda(struct sqlda *sqlda, int *col_lengths);
void allocate_sqlda(struct sqlda *sqlda);

void get_start_time(Timer_struct *start_time);
double get_elapsed_time (Timer_struct *start_time);

long error_check(void); /* @d28763 tlg */
void dumpCa(struct sqlca*); /* kmw */

void display_usage(void);
char *uppercase(char *string);
char *lowercase(char *string);
void comm_line_parse(int argc, char *argv[], struct global_struct *g_struct);
int sqlrx2a(char *decptr, char *asciiptr, short prec, short scal);
void init_setup(int argc, char *argv[], struct global_struct *g_struct);
void runUF1( struct global_struct *g_struct, int updatePair );
void runUF2( struct global_struct *g_struct, int updatePair );

/* These need to be extern because they're in another SQC file.  aph 981205 */
/*extern void runUF1_fn( int updatePair, int i );*/ /* aph 981205 */
/*extern void runUF2_fn( int updatePair, int i, int numChunks );*/ /* aph 981205 */
/*
/* Added four new arguments because SQL host vars can't be global.  aph 981205 */
extern void runUF1_fn ( int updatePair, int i, char *dbname, char *userid, char
*passwd );
extern void runUF2_fn ( int updatePair, int thisConcurrentDelete, int numChunks,
char *dbname, char *userid, char *passwd );

int sem_op (int semid, int semnum, int value);

```

```

char *get_time_stamp(int form, Timer_struct *timer_pointer); /* TIME_ACC
jen */
void summary_table (struct global_struct *g_struct);
void free_sqlda (struct sqlda *sqlda, int select_status); /* @d30369 tlg */
void output_file(struct global_struct *g_struct);
int PreSQLprocess(struct global_struct *g_struct, Timer_struct *start_time);
void SQLprocess(struct global_struct *g_struct);
int PostSQLprocess(struct global_struct *g_struct, Timer_struct *start_time);
int cleanup(struct global_struct *g_struct);

/* Semaphore control functions */
void create_semaphores(struct global_struct *g_struct);
void throughput_wait(struct global_struct *g_struct);
void runpower_wait(struct global_struct *g_struct, int sem_num);
void release_semaphore(struct global_struct *g_struct, int sem_num);
#ifdef SQLWINT
HANDLE open_semaphore(struct global_struct *g_struct, int num);
#else
int open_semaphore(struct global_struct *g_struct);
#endif

EXEC SQL INCLUDE SQLCA;

/*****
***/
/* Declare the SQL host variables. */
/*****
***/
EXEC SQL BEGIN DECLARE SECTION;

char  stmt_str1[4000] = "\0"; /* Assume max SQL statment
of 4000 char */
struct { /* jen LONG */
    short len;
    char data[32700];
} stmt_str; /* jen LONG */
char  dbname[9] = "\0";
char  userid[9] = "\0";
char  passwd[9] = "\0";
char  sourcefile[256]; /* used for semaphores and table functions? */
sqlint32 chunk = 0; /* jenCI counter for within the set of chunks */

EXEC SQL END DECLARE SECTION;

/*****
***/
/* Declare the global variables. */
/*****
***/
struct sqlda *sqlda; /* SQL Descriptor area */

/* Global environment variables (sks May 25 98) */
char env_tpcd_dbname[100];
char env_user[100];
char env_tpcd_audit_dir[150];
char env_tpcd_path_delim[2];
char env_tpcd_tmp_dir[150];
char env_tpcd_run_on_multiple_nodes[10];
char env_tpcd_copy_dir[150];
char env_tpcd_update_import[10];

/* Other globals */
FILE *instream, *outstream; /* File pointers */
int verbose = 0; /* Verbose option flag */
int semcontrol = 1; /* allows/disallows smaphores usage */
int updatePairStart; /* update pair to start at */
int currentUpdatePair; /* update pair running */
int updatePairStop; /* update pair to stop before */
char newtime[50] = "\0"; /* Des - moved from get_time_stamp */
char outstreamfilename[256]; /* store filename of outstream
wlc 081397 */
int inlistmax = 400; /* define # of keys to delete at a time

```



```

wlc 081897 */
int      sqlda_allocated = 0; /* fixing free() problem in NT
wlc 090597 */
int      iImportStagingTbl=0; /* IMPORT use import or load (default) */
char      temp_time_stamp[50]; /* holds end timestamp to be copied into
start_time_stamp of next query bbeaton */
Timer_struct temp_time_struct; /* holds end time value to be copied into
start_time of next query bbeaton */

/* constants for the semaphores used; 1 for throughput and 2 for power */
#define INSERT_POWER_SEM 1
#define QUERY_POWER_SEM 2
#define THROUGHPUT_SEM 1

/*****
**/
/* Start main program processing. */
/*****
**/
int main(int argc, char *argv[])
{
/* kjd715 */
/*struct comm_line_opt c_l_opt = { "\0", "\0", 0, 1, 0, 0, 0 };*/ /* kjd715 */
struct comm_line_opt c_l_opt = { "\0", 0, 1, 0, 0, 0 };
/* kjd715 */
/* command line options */
Timer_struct start_time; /* start point for elapsed time */

struct stmt_info s_info = { -1, -1, 0, 1, -1, -1, "\0", "\0", "\0", "\0", NULL };
/* first stmt_info structure */

struct ctrl_flags c_flags = { 0, 1, 0, TPCDBATCH_SELECT };
/* structure holding ctrl flags
passed between functions */

/* TIME_ACC jen start */
#if defined (SQLUNIX) || defined (SQLAIX)
struct global_struct g_struct =
{ NULL, NULL, NULL, NULL, {0,0}, {0,0}, "\0", 0.1, "\0", FALSE, 0,
NULL, "\0", "\0", "\0", NULL };

#elif (defined (SQLOS2) || defined (SQLWINT) || defined (SQLWIN) ||
defined (SQLDOS))
struct global_struct g_struct =
{ NULL, NULL, NULL, NULL, {0,0,0,0}, {0,0,0,0}, "\0", 0.1, "\0", FALSE, 0,
NULL, "\0", "\0", "\0", NULL };
#else
#error Unknown operating system
#endif
/* TIME_ACC jen end */

/* Get environment variables */
if (get_env_vars() != 0)
return -1;

/* perform setup and initialization and get process id of agent */
outstream = stdout;
g_struct.c_flags = &c_flags;

g_struct.s_info_ptr = &s_info;
g_struct.c_l_opt = &c_l_opt;

init_setup(argc,argv,&g_struct); /* @d22275 tjg */

if ((g_struct.c_l_opt->update == 1) && (semcontrol == 1))
/* runpower: wait for insert function to complete */
/* waiting on the INSERT_POWER_SEM semaphore */
runpower_wait(&g_struct, INSERT_POWER_SEM);

strcpy(temp_time_stamp, "0");

```

```

/*****
*****
*
* This is the transition from the "driver" to the "SUT"
*
*****
*****/

/*****
*****/
/* Read in each statement, prepare, execute, and send output to file. */
/*****
*****/

while (!c_flags.eof_infile) { /* Check to see if there's no more input */

c_flags.eof_block = 0;

if (c_l_opt.outfile)
output_file(&g_struct); /* determine appropriate name for output files */
if ((g_struct.c_l_opt->update != 3) && (g_struct.c_l_opt->update != 4))
{
if (!strcmp(temp_time_stamp, "0")) /* if first query, get timestamp */
{
get_start_time(&start_time);
strcpy(g_struct.s_info_ptr->start_stamp,
get_time_stamp(T_STAMP_FORM_3,&start_time)); /* TIME_ACC
jen*/
}
else /* else get the end timestamp of previous query */
{
strcpy(g_struct.s_info_ptr->start_stamp, temp_time_stamp);
start_time = temp_time_struct;
}
/* write the start timestamp to the file...if this is not a qualification */
/* run, then write the seed used as well */

fprintf( outstream,"Start timestamp %*.s \n",
T_STAMP_3LEN,T_STAMP_3LEN, /* TIME_ACC jen*/
g_struct.s_info_ptr->start_stamp);
if (c_l_opt.intStreamNum >= 0)
{
if (g_struct.lSeed == -1)
{
fprintf( outstream,"Using default qgen seed file");
}
else
fprintf( outstream,"Seed used = %ld",g_struct.lSeed);

fprintf( outstream,"\n");
}
}
do { /* Loop through these statements as long as we haven't reached
the end of the input file or the end of a block of statements
*/

/* Read in the next statment */
c_flags.select_status=Get_SQL_stmt(&g_struct);

if (PreSQLprocess(&g_struct, &start_time) == FALSE)
/* if after reading the next statement we see that we should
exit this loop (i.e. eof, update functions, etc...), get out
*/
break;

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

```

```

* The SQLprocess function implements the implementation specific layer.
*
* It can handle arbitrary SQL statements.
*
*****
*****/

/* If we've got up to here then processing
a regular SQL statement */
SQLprocess(&g_struct);

} while ((!c_flags.eo_block) && (!c_flags.eo_infile)); /* @d30369 tlg */

if (PostSQLprocess(&g_struct,&start_time) == FALSE)
/* if we've reached the end of the input file, then get out
of this loop (i.e. no more statements). Otherwise get
elapsed times and display info about rows */
break;

} /* end of for loop for multiple SQL statements */

g_struct.s_info_ptr = &s_info; /* set the global pointer to start of
linked list */

cleanup(&g_struct); /* finish some semaphore stuff, cleanup files,
and print out summary table */

/*****
*****
*
* In cleanup we make the transition back from the "SUT" to the "driver"
*
*****
*****/

return(0);

} /* end of main */

/*****
*****/
/* Generic form of Sleep */
int SleepSome( int amount)
{
#ifdef SQLWINT
sleep (amount);
#else
// Sleep (amount*100); /* 10x for NT DJD Changed "sleep" to "Sleep" */
Sleep (amount); /* 10x for NT DJD Changed "sleep" to "Sleep" */
#endif
return 0;
}

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

/*****
***/
/* Get environment variables. (sks May 25 98) */
/*****
***/

int get_env_vars(void) {
if (strcpy(env_tpcd_dbname, getenv("TPCD_DBNAME")) == NULL) {
fprintf(stderr, "\n The environment variable $TPCD_DBNAME is not setup
correctly.\n");
return -1;
}
if (strcpy(env_user, getenv("USER")) == NULL) {

```

```

fprintf(stderr, "\n The environment variable $USER is not setup correctly.\n");
return -1;
}
if (strcpy(env_tpcd_audit_dir, getenv("TPCD_AUDIT_DIR")) == NULL) {
fprintf(stderr, "\n The environment variable $TPCD_AUDIT_DIR is not setup
correctly.\n");
return -1;
}
if (strcpy(env_tpcd_tmp_dir, getenv("TPCD_TMP_DIR")) == NULL) {
fprintf(stderr, "\n The environment variable $TPCD_TMP_DIR is not setup
correctly.\n");
return -1;
}
#endif
if (strcpy(env_tpcd_path_delim, getenv("TPCD_PATH_DELIM")) == NULL ||
(strcmp(env_tpcd_path_delim, "/") && strcmp(env_tpcd_path_delim, "\\"))){
fprintf(stderr, "\n The environment variable $TPCD_PATH_DELIM is not
setup correctly , env_tpcd_path_delim'%s'.\n", env_tpcd_path_delim);

return -1;
}
}
#endif
strcpy( env_tpcd_path_delim , "/" ); /*kmw*/
if (strcpy(env_tpcd_run_on_multiple_nodes,
getenv("TPCD_RUN_ON_MULTIPLE_NODES")) == NULL) {
fprintf(stderr, "\n The environment variable
$TPCD_RUN_ON_MULTIPLE_NODES");
fprintf(stderr, "\n is not setup correctly.\n");
return -1;
}
if (strcpy(env_tpcd_copy_dir, getenv("TPCD_COPY_DIR")) == NULL) {
fprintf(stderr, "\n The environment variable $TPCD_COPY_DIR is not setup
correctly.\n");
return -1;
}
/* If TPCD_UPDATE_IMPORT is not set then, the default is set to false, */
/* which is done in init_setup subroutine */
strcpy(env_tpcd_update_import, getenv("TPCD_UPDATE_IMPORT"));

return 0;
}

/*****
***/
/* Get the SQL statement and any control statements from input. */
/*****
***/

int Get_SQL_stmt(struct global_struct *g_struct)

{
char input_ln[256] = "\0"; /* buffer for 1 line of text */
char temp_str[4000] = "\0"; /* temp string for SQL stmt */
char control_str[256] = "\0"; /* control string */

char *test_semi; /* ptr to test for semicolon */
char *control_opt; /* ptr used in control_str parsing */
char *select_status; /* ptr to first word in query */
char *temp_ptr; /* general purpose temp ptr */

int good_sql = 0; /* good-sql stmt flag @d23684 tlg */
int stmt_num_flag = 1; /* first line of SQL stmt flag */
int eostmt = 0; /* flag to signal end of statement */

stmt_str.data[0]='\0'; /* Initialize statement buffer */

if (verbose)
fprintf (stderr, "\n-----\n");
fprintf (outstream, "\n-----\n");

do {
/** Read in lines from input one at a time */
fscanf(instream, "\n%[^\n]\n", input_ln);

```

```

if (strstr(input_ln, "--") == input_ln) { /* Skip all -- comments */

    if (strstr(input_ln, "--SET") == input_ln) {
        /* Store control string but
           keep going to find SQL stmt */
        strcpy(control_str, input_ln);
        if (verbose)
            fprintf(stderr, "%s\n", uppercase(control_str));
        fprintf(outstream, "%s\n", uppercase(control_str));

        /** Start parsing control str. and update appropriate vars. **/
        control_opt = strtok(control_str, " ");
        while (control_opt != NULL) {
            if (strcmp(control_opt, "--SET") == 0) { /* Skip the #SET token */
                if (!strcmp(control_opt, "ROWS_FETCH"))
                    g_struct->s_info_ptr->max_rows_fetch = atoi(strtok(NULL, " "));

                if (!strcmp(control_opt, "ROWS_OUT"))
                    g_struct->s_info_ptr->max_rows_out = atoi(strtok(NULL, " "));
            }

            control_opt = strtok(NULL, " ");
        }

        /* if the block option has been set, then check if we've
           reached the end of a block of statements */
        if (g_struct->s_info_ptr->query_block) /* @d30369 tlg */
            if (strstr(input_ln, "--EOBLK") == input_ln) {
                g_struct->c_flags->eo_block = 1;
                return TPCDBATCH_EOBLOCK;
            }

        if (strstr(input_ln, "-- Query") == input_ln)
            strcpy(g_struct->s_info_ptr->qry_description, input_ln);

        if (strstr(input_ln, "--#TAG") == input_ln)
            strcpy(g_struct->s_info_ptr->tag, (input_ln + sizeof("--#TAG")));

        /* if we're using update functions, return that info
           appropriately */
        if (g_struct->c_l_opt->update != 0) {
            if (strstr(input_ln, "--#INSERT") == input_ln)
                return TPCDBATCH_INSERT;

            if (strstr(input_ln, "--#DELETE") == input_ln)
                return TPCDBATCH_DELETE;
        }

        if (strstr(input_ln, "--#COMMENT") == input_ln) { /* @d25594 tlg */
            temp_ptr = (input_ln + 11); /* User-specified comments go to
                the outfile */
            if (verbose)
                fprintf(stderr, "%s\n", temp_ptr);
            fprintf(outstream, "%s\n", temp_ptr);
        }

        eostmt = 0;
    }

    /* Need this hack here to check if there's any more empty lines left
       in the input file. Continue only if there are aren't any */
    else if (strcmp(input_ln, "0")) /* HACK */ { /* A regular SQL statement */
        if (stmt_num_flag) { /* print this out only if it's the first line
            of the SQL statement. We only want this
            line to appear once per statement */
            if (verbose)
                fprintf(stderr, "\n%s\n", g_struct->s_info_ptr->qry_description);
            fprintf(outstream, "\n%s\n", g_struct->s_info_ptr->qry_description);

            if (verbose)
                fprintf(stderr, "\nTag: %-5.5s Stream: %d Sequence number: %d\n",
                    g_struct->s_info_ptr->tag, g_struct->c_l_opt->intStreamNum,
                    g_struct->s_info_ptr->stmt_num); /*jen0925*/
            fprintf(outstream, "\nTag: %-5.5s Stream: %d Sequence number: %d\n",

```

```

                    g_struct->s_info_ptr->tag, g_struct->c_l_opt->intStreamNum,
                    g_struct->s_info_ptr->stmt_num); /*jen0925*/

        /* Turn off this flag once the number has been printed */
        stmt_num_flag = 0;

    } /** Print out this heading the first time you encounter a
        non-comment statement **/

    /* Test to see if we've reached the end of a statement */
    good_sql = TRUE; /* @d23684 tlg */
    test_semi = strstr(input_ln, ";");
    if (test_semi == NULL) { /* if there's no semi-colon keep on going */
        strcat(stmt_str.data, input_ln); /* jen LONG */
        strcat(stmt_str.data, " "); /* jen LONG */
        stmt_str.len = strlen(stmt_str.data); /* jen LONG */
        eostmt = 0;
    }

    else { /* else replace the ; with a \0 and continue */
        *test_semi = '\0';
        strcat(stmt_str.data, input_ln); /* jen LONG */
        stmt_str.len = strlen(stmt_str.data); /* jen LONG */
        eostmt = 1;
    }

    fprintf(outstream, "\n%s", input_ln);
    if (verbose)
        fprintf(stderr, "\n%s", input_ln);
}

/** Test to see if we've reached the EOF. Get out if that's the case **/
if (feof(instream)) {
    eostmt = TRUE;
    g_struct->c_flags->eo_infile = TRUE; /* @d22275 tlg */
}

} while (!eostmt);

fprintf(outstream, "\n");
if (verbose)
    fprintf(stderr, "\n");

/** erase the old control string **/
strcpy(control_str, "\0");

/** Determine whether statement is a SELECT or other SQL **/
if (good_sql) {
    strcpy(temp_str, stmt_str.data); /* jen LONG */
    uppercase(temp_str); /* Make sure that select is made to SELECT */
    select_status = strtok(temp_str, " ");
    if ((stmt_str.data[0] == '(') || (!strcmp(select_status, "SELECT"))) ||
        (!strcmp(select_status, "VALUES")) ||
        (!strcmp(select_status, "WITH")))
        return TPCDBATCH_SELECT;
    else
        return TPCDBATCH_NONSELECT;
}

/** If you go through a file with just comments or control statments
    with no SQL, there's nothing to process...Exit TPCDBATCH **/

else /* @d23684 tlg */
    return TPCDBATCH_NONSQL;
} /* Get_SQL_stmt */

/*****
**/
/* allocate_sqlda -- This routine allocates space for the SQLDA. */

```

```

/*****
**/

void allocate_sqlda(struct sqlda *sqlda)
{
    int loopvar;          /* Loop counter */

    for (loopvar=0; loopvar<sqlda->sqld; loopvar++)
    {
        switch (sqlda->sqlvar[loopvar].sqltype)
        {
            case SQL_TYP_INTEGER:          /* INTEGER */
            case SQL_TYP_NINTEGER:
                if ((sqlda->sqlvar[loopvar].sqldata=
                    (TPCDBATCH_CHAR *)malloc(sizeof(sqlint32))) == NULL)
                    mem_error("allocating INTEGER");
                break;
            case SQL_TYP_BIGINT:             /* BIGINT */ /*kmwBIGINT*/
            case SQL_TYP_NBIGINT:
                /*#ifdef SQLWINT */
                /* if ((sqlda->sqlvar[loopvar].sqldata= */
                /* (TPCDBATCH_CHAR *)malloc(sizeof(__int64))) == NULL)*/
                /*#else */
                if ((sqlda->sqlvar[loopvar].sqldata=
                    (TPCDBATCH_CHAR *)malloc(sizeof(sqlint64))) == NULL)
                /*#endif*/
                    mem_error("allocating BIGINT");
                break;
            case SQL_TYP_CHAR:              /* CHAR */
            case SQL_TYP_NCHAR:
                if ((sqlda->sqlvar[loopvar].sqldata=
                    (TPCDBATCH_CHAR *)calloc(256,sizeof(char))) == NULL)
                    mem_error("allocating CHAR/VARCHAR");
                break;
            case SQL_TYP_VARCHAR:           /* VARCHAR */
            case SQL_TYP_NVARCHAR:
                if ((sqlda->sqlvar[loopvar].sqldata=
                    (TPCDBATCH_CHAR *)calloc(4002,sizeof(char))) == NULL)
                    mem_error("allocating CHAR/VARCHAR");
                break;
            case SQL_TYP_LONG:              /* LONG VARCHAR */
            case SQL_TYP_NLONG:
                if ((sqlda->sqlvar[loopvar].sqldata=
                    (TPCDBATCH_CHAR *)calloc(32702,sizeof(char))) == NULL)
                    mem_error("allocating VARCHAR/LONG VARCHAR");
                break;
            case SQL_TYP_FLOAT:             /* FLOAT */
            case SQL_TYP_NFLOAT:
                if ((sqlda->sqlvar[loopvar].sqldata=
                    (TPCDBATCH_CHAR *)malloc(sizeof(double))) == NULL)
                    mem_error("allocating FLOAT");
                break;
            case SQL_TYP_SMALL:             /* SMALLINT */
            case SQL_TYP_NSMALL:
                if ((sqlda->sqlvar[loopvar].sqldata=
                    (TPCDBATCH_CHAR *)malloc(sizeof(short))) == NULL)
                    mem_error("allocating SMALLINT");
                break;
            case SQL_TYP_DECIMAL:           /* DECIMAL */
            case SQL_TYP_NDECIMAL:
                if ((sqlda->sqlvar[loopvar].sqldata=
                    (TPCDBATCH_CHAR *)malloc(20)) == NULL)
                    mem_error("allocating DECIMAL");
                break;
            case SQL_TYP_CSTR:              /* VARCHAR (null terminated) */
            case SQL_TYP_NCSTR:
                if ((sqlda->sqlvar[loopvar].sqldata=
                    (TPCDBATCH_CHAR *)calloc(4001,sizeof(char))) == NULL)
                    mem_error("allocating CHAR/VARCHAR");
                break;
            case SQL_TYP_DATE:             /* DATE */
            case SQL_TYP_NDATE:
                if ((sqlda->sqlvar[loopvar].sqldata=
                    (TPCDBATCH_CHAR *)calloc(13,sizeof(char))) == NULL)

```

```

                mem_error("allocating DATE");
                break;
            case SQL_TYP_TIME:              /* TIME */
            case SQL_TYP_NTIME:
                if ((sqlda->sqlvar[loopvar].sqldata=
                    (TPCDBATCH_CHAR *)calloc(11,sizeof(char))) == NULL)
                    mem_error("allocating TIME");
                break;
            case SQL_TYP_STAMP:             /* TIMESTAMP */
            case SQL_TYP_NSTAMP:
                if ((sqlda->sqlvar[loopvar].sqldata=
                    (TPCDBATCH_CHAR *)calloc(29,sizeof(char))) == NULL)
                    mem_error("allocating TIMESTAMP");
                break;
        }
        if ((sqlda->sqlvar[loopvar].sqlind=
            (short *)calloc(1,sizeof(short))) == NULL)
            mem_error("allocating indicator");
    }
    sqlda_allocated = 1; /* fix free() problem on NT
                        wlc 090597 */
    return; /* allocate_sqlda */
}

```

```

/*****
*****
/* echo_sqlda -- This routine displays the contents of an SQLDA. */
*****
*****

```

```

void echo_sqlda(struct sqlda *sqlda, int *col_lengths)
{
    int col;          /* Column counter */

    int col_type;     /* Type of column */

    char temp_string[100] = "\0"; /* Temporary string */
    char decimal_string[100] = "\0"; /* String holding decimals */
    char *temp_ptr;

    TPCDBATCH_CHAR m,n; /* precision and accuracy
                        for decimal conversion */

    for (col=0; col<sqlda->sqld; col++) /* Loop through column count */
    {
        col_type=sqlda->sqlvar[col].sqltype; /* @d22817 tjg */

        if (*sqlda->sqlvar[col].sqlind) /* @d30369 tjg */
            fprintf(outstream, "%* n/a ",(col_lengths[col]-3));
        else
            switch (col_type)
            {
                case SQL_TYP_INTEGER:
                case SQL_TYP_NINTEGER:

                    fprintf(outstream, "%*ld ",col_lengths[col],
                        *(sqlint32 *)sqlda->sqlvar[col].sqldata);
                    break;

                case SQL_TYP_BIGINT: /*kmwBIGINT*/
                case SQL_TYP_NBIGINT:
                /*#ifdef SQLWINT*/
                /* fprintf(outstream, "%*I64d ",col_lengths[col],*/
                /* *(__int64 *)sqlda->sqlvar[col].sqldata);*/
                /*#else*/
                fprintf(outstream, "%*lld ",col_lengths[col],
                    *(sqlint64 *)sqlda->sqlvar[col].sqldata);
                /*#endif*/
                break;
            }
    }
}

```

```

case SQL_TYP_CHAR:
case SQL_TYP_NCHAR:

    fprintf(outstream, "%-*s ", col_lengths[col], sqlda->sqlvar[col].sqldata);
    break;
case SQL_TYP_VARCHAR:
case SQL_TYP_NVARCHAR:
case SQL_TYP_LONG:
case SQL_TYP_NLONG: /* @d30369 tlg */
    ((struct sqlchar *)sqlda->sqlvar[col].sqldata)->
        data[((struct sqlchar *)sqlda->sqlvar[col].sqldata)->length] = '\0';
    fprintf(outstream, "%-*s ",
        col_lengths[col],
        ((struct sqlchar *)sqlda->sqlvar[col].sqldata)->data);
    break;
case SQL_TYP_FLOAT:
case SQL_TYP_NFLOAT:
{ /* kmw */
    if ( fabs(*(double *) (sqlda->sqlvar[col].sqldata))
        < TPCDBATCH_PRINT_FLOAT_MAX )
        fprintf(outstream, "%#.3f ", col_lengths[col],
            *(double *) (sqlda->sqlvar[col].sqldata));
    else
        fprintf(outstream, "%*e ", col_lengths[col],
            *(double *) (sqlda->sqlvar[col].sqldata));
    break;
}

case SQL_TYP_SMALL:
case SQL_TYP_NSMAIL:

    fprintf(outstream, "%*hd ", col_lengths[col],
        *(short *) (sqlda->sqlvar[col].sqldata));
    break;
case SQL_TYP_DECIMAL:
case SQL_TYP_NDECIMAL:

    m=(*(struct declen *)&sqlda->sqlvar[col].sqlen).m;
    n=(*(struct declen *)&sqlda->sqlvar[col].sqlen).n;
    if (sqlrxd2a((char *)sqlda->sqlvar[col].sqldata,temp_string,m,n) != 0)
    {
        fprintf(stderr, "\nThe decimal value could not be converted.\n");
        exit (-1);
    }
    else {

        temp_ptr = temp_string;

        if (*temp_ptr == '-')
            strcpy(decimal_string, "-");

        else
            strcpy(decimal_string, "");

        for (temp_ptr = temp_string + 1; *temp_ptr == '0'; temp_ptr++)
            ;

        strcat(decimal_string,temp_ptr);
        fprintf(outstream, "%*s ", col_lengths[col], decimal_string);
    }

    break;

case SQL_TYP_CSTR:
case SQL_TYP_NCSTR:
case SQL_TYP_DATE:
case SQL_TYP_NDATE:
case SQL_TYP_TIME:
case SQL_TYP_NTIME:
case SQL_TYP_STAMP:
case SQL_TYP_NSTAMP:
    sqlda->sqlvar[col].sqldata[sqlda->sqlvar[col].sqlen+1]='\0';
    strcpy(temp_string,(char *)sqlda->sqlvar[col].sqldata);
    fprintf(outstream, "%-*s ", (col_lengths[col]), temp_string);

```

```

        break;

        default:
            fprintf(stderr, "--Unknown column type (%d). Aborting.\n", col_type);
            break;
    }
}

fprintf(outstream, "\n");

return;
}

/*****
/* Calculate the elapsed time. */
*****/

void get_start_time(Timer_struct *start_time)
{
    int rc = 0;

#ifdef (SQLOS2) || defined (SQLWINT) || defined (SQLWIN) || defined (SQLDOS)
    /*@d33143aha*/
    ftime(start_time);
#elif defined (SQLSNI)
    rc = gettimeofday(start_time);
#elif defined (SQLPTX)
    gettimeofday_mapped(start_time);
    rc = 0; /* gettimeofday_mapped returns void */
#elif defined (SQLUNIX) || defined (SQLAIX) /*TIMER jen*/
    rc = gettimeofday(start_time, NULL);
#else
#error Unknown operating system
#endif

    if (rc != 0) {
        fprintf(stderr, "Timer call failed, aborting test\nExiting tpcdbatch..\n");
        exit(-1);
    }
}

/*****
*****/
/* Calculate and return the elapsed time given a starting time. */
/*****
*****/

double get_elapsed_time ( Timer_struct *start_time)
{
    int status = 0;
    Timer_struct end_time;
    double result = -1.0;
#ifdef SQLWINT
    long int result_sec;
    long int result_usec;
#endif

#ifdef (SQLSNI)
    status = gettimeofday(&end_time);
#elif defined (SQLPTX)
    gettimeofday_mapped(&end_time);
    status = 0; /* gettimeofday_mapped returns void */
#elif defined (SQLUNIX) || defined (SQLAIX)
    status = gettimeofday(&end_time, NULL); /*TIMER jen*/
#elif defined (SQLOS2) || defined (SQLWINT) || defined (SQLWIN) || defined (SQLDOS)
    ftime(&end_time);
#else
    /** If another operating system **/
#error Unknown operating system
#endif

```

```

if (status != 0)
    fprintf(stderr, "Bad return from gettimeofday, don't trust timer results...\n");

else
{
#ifdef defined (SQLUNIX) || defined (SQLAIX)
    result_sec = end_time.tv_sec - start_time->tv_sec;
    result = (double) result_sec;
    /* TIMER used micro seconds with timeval (not nanoseconds) */
    if ((start_time->tv_usec > 0) && \
        (start_time->tv_usec < 1000000) && \
        (end_time.tv_usec > 0) && \
        (end_time.tv_usec < 1000000))
    {
        result_usec = end_time.tv_usec - start_time->tv_usec;
        result = (double) result_sec + ((double) result_usec/1000000);
    }
#elif defined (SQLOS2) || defined (SQLWINT) || defined (SQLWIN) ||
defined (SQLDOS))
    result = (double) (end_time.time - start_time->time);
    result = result * 1000 + (end_time.millitm - start_time->millitm);
    result = result/1000;
#else
#error Unknown operating system
#endif

}

/*
 * translate the time to that rounded to the CLOSEST 0.1 seconds as
 * required by the TPC-D spec.  ROUNDING
 */
/* result = (double)((long)((result + 0.099999) * 10))/10.0; */
result = (double)((long)((result + 0.05) * 10))/10.0;
return (result);
}

void dumpCa(struct sqlca *ca)
{
    int i;
    fprintf(outstream, "***** DUMP OF SQLCA\n");
    fprintf(outstream, "SQLCAID : %s\n", ca->sqlcaid);
    fprintf(outstream, "SQLCABC : %d\n", ca->sqlcabc);
    fprintf(outstream, "SQLCODE : %d\n", ca->sqlcode);
    fprintf(outstream, "SQLERRML : %d\n", ca->sqlerrml);
    fprintf(outstream, "SQLERRMC : %s\n", ca->sqlerrmc);
    fprintf(outstream, "SQLERRP : %s\n", ca->sqlerrp);

    for (i = 0; i < 6; i++)
    {
        fprintf(outstream, "SQLERRD[%d]: %d\n", i, ca->sqlerrd[i]);
    }
    fprintf(outstream, "SQLWARN : %s\n", ca->sqlwarn);
    fprintf(outstream, "SQLSTATE : %s\n", ca->sqlstate);
    fprintf(outstream, "***** END OF SQLCA DUMP\n");
    return;
}

/* *****
 * error_check
 * This function prints the contents of the sqlca error information
 * structure.
 * *****
long error_check(void)
{
    char    buffer[512]="\0";
    unsigned short i;

```

```

struct sqlca temp_sqlca; /* temporary sqlca */ /* @d30369 tjt */

temp_sqlca.sqlcode = 0; /* initialize the temporary sqlca to
                          avoid any memory problems */

if (sqlca.sqlcode != 0) {
    sqlaintp(buffer, sizeof(buffer), 80, &sqlca);
    fprintf(stderr, "\n%0.200s\n", buffer);
    fprintf(outstream, "\n%0.200s\n", buffer);

    /* Decode the SQLCA in more detail  KBS 98/09/28 */
    if ((sqlca.sqlerrml) /* there's one or more tokens */
        && (sqlca.sqlerrml < sizeof(sqlca.sqlerrmc)) /* and field not full */
    )
    {
        char *tokptr;
        int tokl;
        *(sqlca.sqlerrmc + sqlca.sqlerrml) = '\0'; /* prevent strtok from scanning
        beyond end */
        fprintf(stderr, "\n SQLCA: tokens:\n");
        fprintf(outstream, "\n SQLCA: tokens:\n");
        tokptr=strtok(sqlca.sqlerrmc, "\xff");
        while ( tokptr &&
            ((tokl = (sizeof(sqlca.sqlerrmc) - (tokptr-sqlca.sqlerrmc))) > 0)
        )
        {
            fprintf(stderr, "%.*s\n", tokl, tokptr);
            fprintf(outstream, "%.*s\n", tokl, tokptr);
            tokptr=strtok(NULL, "\xff");
        }
        fprintf(stderr, "\n SQLCA: errp= %s, errd 1-6= %d %d %d %d %d\n",
            sqlca.sqlerrp, sqlca.sqlerrd[0], sqlca.sqlerrd[1], sqlca.sqlerrd[2],
            sqlca.sqlerrd[3], sqlca.sqlerrd[4], sqlca.sqlerrd[5]);
        fprintf(outstream, "\n SQLCA: errp= %s, errd 1-6= %d %d %d %d %d\n",
            sqlca.sqlerrp, sqlca.sqlerrd[0], sqlca.sqlerrd[1], sqlca.sqlerrd[2],
            sqlca.sqlerrd[3], sqlca.sqlerrd[4], sqlca.sqlerrd[5]);

        temp_sqlca = sqlca; /* Make a copy of sqlca in case it gets changed
                             in the next statement below */ /* @d30369 tjt */

        /** Determine if the error is critical or a connection can be made **/

        EXEC SQL CONNECT ; /* @d28763 tjt */

        if (sqlca.sqlcode == SQLE_RC_NOSUDB ) { /* no connection exists */

            /*Print out header for DUMP*/
            fprintf(outstream, "*****\n");
            fprintf(outstream, "* CONTENTS OF SQLCA *\n");
            fprintf(outstream, "*****\n");

            /*Print out contents of SQLCA variables*/
            fprintf(outstream, "SQLCABC = %d\n", temp_sqlca.sqlcabc);
            fprintf(outstream, "SQLCODE = %d\n", temp_sqlca.sqlcode);
            fprintf(outstream, "SQLERRMC = %s\n", temp_sqlca.sqlerrmc);
            fprintf(outstream, "SQLERRP = %s\n", temp_sqlca.sqlerrp);

            for (i = 0; i < 6; i++)
            {
                fprintf(outstream, "sqlerrd[%d] = %lu\n", i, temp_sqlca.sqlerrd[i]);
            }

            fprintf(outstream, "SQLWARN = %s\n", temp_sqlca.sqlwarn);
            fprintf(outstream, "SQLSTATE = %s\n", temp_sqlca.sqlstate);

            fprintf(stderr, "\nCritical SQLCODE. Exiting TPCDBATCH\n");
            exit(-1);
        }
    }
    return (temp_sqlca.sqlcode);
}

```

```

} /* error_check */

/*****
/* Displays a help screen */
*****/
void display_usage()
{
    printf("\ntpcdbatch -- version %s",TPCDBATCH_VERSION);
    printf("\n\nSyntax is:\n");
    printf("\ntpcdbatch [-d dbname] [-f file_name] [-l file_name] [-r on/off]");
    printf("\n      [-v on/off] [-b on/off] [-u p/t/t1/t2]");
    printf("\n      [-s scale_factor] [-n stream_num] [-m inlistmax] [-h]\n");
    printf("\n where: -d Database name");
    printf("\n      Default - dbname set in $DB2DBDFT");
    printf("\n      -f Input file containing SQL statements");
    printf("\n      Default - stdin");
    printf("\n      -r Create set of output files containing query results");
    printf("\n      Default - off");
    printf("\n      -v Verbose. Sends information to stderr during");
    printf("\n      query processing");
    printf("\n      Default - off");
    printf("\n      -b Process groups of statements as blocks");
    printf("\n      instead of individually.");
    printf("\n      Default - off");
    printf("\n      -u Update streams: p - for power test");
    printf("\n      t - for throughput test without");
    printf("\n      UFs (run this instead of t2)");
    printf("\n      t1 - for throughput test step 1");
    printf("\n      only running queries");
    printf("\n      t2 - for throughput test step 2");
    printf("\n      running update functions");
    printf("\n      -s Scale factor");
    printf("\n      Default - 0.1");
    printf("\n      -n Stream number");
    printf("\n      Default - 0");
    printf("\n      Qualification - -1");
    printf("\n      Power - 0");
    printf("\n      Throughput - >= 1 (actual number depends on the current");
    printf("\n      query stream)");
    printf("\n      -m Maximum number of keys to delete at a time");
    printf("\n      Default - 400");
    printf("\n      -h Display this help screen");
    printf("\n      -p turns smeaphores on or off");
    printf("\n      Default - off");

    printf("\n\nControl statements specifying output and performance details");
    printf("\nncan be included before SQL statements; they will apply for");
    printf("\nthat and subsequent statements until updated.");

    printf("\n\nSyntax: --SET <control option> <value>");
    printf("\n option value default");
    printf("\nROWS_FETCH -1 to n -1 (all rows fetched from answer set)");
    printf("\nROWS_OUT -1 to n -1 (all fetched rows sent to output)");
    printf("\n\n--TAG tag (user specified tag name for sequence#)");
    printf("\n\n--COMMENT comment (user specified comments for");
    printf("\n      output)");
    printf("\n\nNote: All statements executed with ISOLATION LEVEL RR");
    printf("\n      and must be terminated with semi-colons.\n");
    exit (1);
}

/*****
/* Converts a string to upper case characters */
*****/
char *uppercase( char *string )
{
    char *c; /* temp char used to convert word to upper case */

    for ( c = string; *c != '\0'; c++)
        *c = (char) toupper( (int) *c );
}

```

```

    return (string);
}

/*****
/* Converts a string to lower case characters */
*****/
char *lowercase( char *string )
{
    char *c; /* temp char used to convert word to lower case */

    for ( c = string; *c != '\0'; c++)
        *c = (char) tolower( (int) *c );

    return (string);
}

/*****
/* Parses and processes command line options. */
*****/
void comm_line_parse(int argc, char *argv[], struct global_struct *g_struct)
{
    char authentic_info[40] = "\0";
    char *testptr;
    int loopvar = 0;

    int comm_opt = 0;
#ifdef PARALLEL_UPDATES
    int running_updates=0;
    int updatePair=-1;
    int updateStream=-1;
    int function;
    int copyOnOrOff;
    int deleteChunk=0; /*DELjen */
#endif

    while ((loopvar < argc) && (argc != 1)) {

        if (*argv[loopvar] == '-') {

            switch(*(argv[loopvar]+1)) {

                case 'f': /* @d26350 tlg */
                case 'F':
                    strcpy(g_struct->c_l_opt->infile,argv[++loopvar]);
                    break;

                /* kjd715 */
                case 'l':
                case 'L': loopvar+=1;
                    /*
                    strcpy(g_struct->c_l_opt->str_file_name,argv[++loopvar]);
                    */
                    break;

                /* kjd715 */
                case 'r': /* @d26350 tlg */
                case 'R':
                    if (!strcmp(uppercase(argv[++loopvar]),"ON"))
                        g_struct->c_l_opt->outfile=1;
                    else
                        g_struct->c_l_opt->outfile=0;
                    break;

                case 'd': /* @d26350 tlg */
                case 'D':
                    strcpy(dbname,argv[++loopvar]);
                    break;

                case 'v': /* @d26350 tlg */
                case 'V':
                    if (!strcmp(uppercase(argv[++loopvar]),"ON"))
                        verbose=1;
                    else

```

```

        verbose=0;
        break;

case 'u' :                /* @d26350 tlg */
case 'U' :
    g_struct->c_l_opt->update=-1; /* init to invalid number */
    if (!strcmp(uppercase(argv[++loopvar]), "P1"))
        g_struct->c_l_opt->update=1; /* power query stream */
    if (!strcmp(uppercase(argv[loopvar]), "P2"))
        g_struct->c_l_opt->update=3; /* power update with updates */
    if (!strcmp(uppercase(argv[loopvar]), "P"))
        g_struct->c_l_opt->update=4; /* power update without updates */
    if (!strcmp(uppercase(argv[loopvar]), "T1"))
        g_struct->c_l_opt->update=0; /* throughput query stream */
    if (!strcmp(uppercase(argv[loopvar]), "T2"))
        g_struct->c_l_opt->update=2; /* throughput update with updates */
    if (!strcmp(uppercase(argv[loopvar]), "T"))
        g_struct->c_l_opt->update=5; /* throughput update without updates */

    break;

case 'b' :                /* @d26350 tlg */
case 'B' :
    if (!strcmp(uppercase(argv[++loopvar]), "ON"))
        g_struct->s_info_ptr->query_block=1;
    else
        g_struct->s_info_ptr->query_block=0;
    break;

case 'n' :                /* @d26350 tlg */
case 'N' :
    g_struct->c_l_opt->intStreamNum = atoi(argv[++loopvar]);
    break;

case 's' :                /* @d26350 tlg */
case 'S' :    g_struct->scale_factor=atof(argv[++loopvar]); break;

case 'h':
case 'H' :                /* @d26350 tlg */
    display_usage();
    break;

case 'm' :
case 'M' :
    inlistmax = atoi(argv[++loopvar]); /* wlc 081897 */
    break;

case 'p' :
case 'P' :
    if (!strcmp(uppercase(argv[++loopvar]), "ON")) /* bbe 072599 */
        semcontrol = 1;
    else
        semcontrol = 0;
    break;

#ifdef PARALLEL_UPDATES
case 'i':
    updatePair = atoi(argv[++loopvar]);
#endif
#ifdef UF2DEBUG
    fprintf(stderr, "updatePair = %d\n", updatePair);
    fflush(stderr);
#endif
    break;

case 'j':
    function = atoi(argv[++loopvar]);
#ifdef UF2DEBUG
    fprintf(stderr, "function = %d\n", function);
    fflush(stderr);
#endif
    break;

case 'k':

```

```

        updateStream = atoi(argv[++loopvar]);
#ifdef UF2DEBUG
    fprintf(stderr, "updateStream = %d\n", updateStream);
    fflush(stderr);
#endif
    break;

case 'x':                /* DEL jen -x is chunk */
    deleteChunk = atoi(argv[++loopvar]); /* to delete for this */
#ifdef UF2DEBUG
    fprintf(stderr, "DelChunk = %d\n", deleteChunk);
    fflush(stderr);
#endif
    break;                /* invocation */

case 'z':
    running_updates = 1;
    break;
#endif

default :
    fprintf(stderr, "An invalid option has been set\n");
    display_usage();
    break;

} /* end switch */
} /* end if */

loopvar ++;
} /* end while */

/* checking if -u option is set */
if (g_struct->c_l_opt->update == -1) {
    fprintf(stderr, "-u option is not set, exiting ...\n");
    exit(-1);
}

#ifdef PARALLEL_UPDATES
if (running_updates) {
    if (updatePair == -1) {
        fprintf(stderr, "The parameters to tpcedbatch have not been passed correctly\n");
        exit(-1);
    }
    else {
        /* check to see if we are to use copy on for the load */
        if ((getenv("TPCD_LOG") != NULL) &&
            (!strcmp(uppercase(getenv("TPCD_LOG")), "YES")))
        {
            /* okay, we have set LOG_RETAIN on so we need to use copy directory
            */
            copyOnOrOff = TRUE;
        }
        else
        {
            /* log retain off don't use copy directory */
            copyOnOrOff = FALSE;
        }
    }

    if (function == 1)
        /* runUF1_fn (updatePair, updateStream); aph 981205 */
        runUF1_fn (updatePair, updateStream, dbname, userid, passwd);
    else
        if (function == 2) {
            //DJD fprintf(stderr, "A-Calling runUF2_fn %d %d %d...\n",
            //DJD updatePair, updateStream, deleteChunk);
            /* runUF2_fn (updatePair, updateStream, deleteChunk); aph 981205 */
            runUF2_fn (updatePair, updateStream, deleteChunk, dbname, userid,
            passwd);
        }
        else {
            fprintf(stderr, "Wrong function to tpcedbatch\n");
            exit(-1);
        }
    }
}

```



```

        exit (0);
    }
}
#endif /* PARALLEL_UPDATES */

/* If no database name is given, then use the one specified in the
environment variable DB2DBDFT, otherwise error */
if (!strcmp(dbname,"0")) {
    testptr = getenv("DB2DBDFT");
    if (testptr == NULL) {
        fprintf(stderr, "\nNo database name has been specified on command ");
        fprintf(stderr, "line\n\nin environment variable DB2DBDFT.");
        display_usage();
    }
    else
        strcpy(dbname,testptr);
}

/* kjd715 */
/*
if (g_struct->c_l_opt->outfile) &&
!strcmp(g_struct->c_l_opt->str_file_name,"0")) {
    fprintf(stderr, "\nMust specify input file for statement list.\n");
    display_usage();
}
*/
/* kjd715 */
}

/*****
/* Converts DECIMAL values to ASCII text */
/*****
int sqlrxd2a(
/*kmw*/
/* C++ */char *decptr,
/* C++ */char *asciiptr,
short prec,
short scal)
{
/*
int allzero = TRUE;
/* C++ */char *srcptr;
unsigned char sign;
/* C++ */char *targptr, decimal_point = '.';
int rc = 0; /*kmw*/
int tmpint, src_nibble;
int count, j, limit[3];

targptr = &asciiptr[ prec + 1];
*(1 + targptr) = '\0';
srcptr = decptr + prec/2;

/* Validity check sign nibble */
if (((sign = sqlrx_get_right_nibble( *srcptr )) < 0x0a)
|| (prec > SQL_MAXDECIMAL) || (prec < scal ))
{
    goto exit;
}
/** end end if invalid sign value */

limit[ 0 ] = scal; limit[ 1 ] = prec - scal; limit[ 2 ] = 0;
src_nibble = LEFT;
for(j = 0 ; j < 2 ; j++)
{
    for( count = limit[ j ] ; count > 0 ; count-- )
    {
        tmpint = ( (src_nibble == LEFT)?
            sqlrx_get_left_nibble( *srcptr-- ) :
            sqlrx_get_right_nibble( *srcptr ) );
        if( tmpint > 9 )
        {
            goto exit;
        }
    }
}

```

```

    else
        *targptr-- = (/* C++ */char)tmpint + '0';
    src_nibble = ((src_nibble == LEFT) ? RIGHT : LEFT);
    if ( tmpint != 0 ) allzero = FALSE;
} /** end for scal > 0 */

if( j == 0 )
    *targptr-- = decimal_point;
else
    *targptr = (/* C++ */char)((allzero
        || (sign == SQLRX_PREFERRED_PLUS)
        || (sign == 0x0a)
        || (sign == 0x0e)
        || (sign == 0x0f)) ?
        '+' : '-' );
} /** end for limit[ j++ ] > 0 */

exit :
if( rc < 0 )
{
    printf ("The decimal conversion has failed\n");
    exit (-1);
}

return(rc);
} /** sqlrxd2a */

/*****
/
/* Does some setup and initialization like parsing command line */
/* and connecting to database. Returns process id of agent. */
/*****
void init_setup(int argc, char *argv[], struct global_struct *g_struct)
{
    int connect=0;
#ifdef SQLWINT
    char *pid;
#endif
    char temparray[256]="\0";
    int loopvar=0;
    FILE *updateFP;
    FILE *fpSeed;
    char file_name[256] = "\0";
    short seedEntry;
    long lSeed;
    int i;

    /** Parse and process command line options */
    comm_line_parse (argc,argv,g_struct);

/*****
*****/
/* Start the mainline report processing. */
/*****
*****/
if (!strcmp(g_struct->c_l_opt->infile,"0")) {
    instream=stdin;
}
else {
    instream=NULL;
    if ( ( instream = fopen(g_struct->c_l_opt->infile, READMODE)) == NULL ) {
        /* kjd715 */
        fprintf(outstream, "XXThe input file could not be opened.\n\n");
        /* kjd715 */
        printf(stdout,"Make sure that the filename is correct.\n");
        printf(stdout,"filename = %s\n",g_struct->c_l_opt->infile);
        exit(-1);
    }
    /** open the input file if specified */
}
}

```

```

/* IMPORT (begin) - determine whether we should use the IMPORT api or */
/* LOAD api for loading into the staging tables, default is load */
if (env_tpcd_update_import != NULL)
{
    if (!strcmp(uppercase(env_tpcd_update_import),"TRUE"))
    {
        iImportStagingTbl = 1; /* use import */
    }
    /* DJD */
    else if (!strcmp(uppercase(env_tpcd_update_import),"TF"))
    {
        iImportStagingTbl = 2; /* Table Functions */
    }
}

/* IMPORT (end) */

/* we want to print the seed in the output files to show what seed was */
/* used to generate the queries. */
/* if intStreamNum is -1 then we are running a qualification database */
/* and the default seed has been used so skip this section */
if (g_struct->c_l_opt->intStreamNum >= 0)
{
    /* check to make sure the TPCD_RUNNUMBER environment variable is set.
We */
    /* use this and the stream number to determine which seed was used to */
    /* generate the current set of queries */
    if (getenv("TPCD_RUNNUMBER") == NULL)
    {
        fprintf(stderr,"nThe TPCD_RUNNUMBER environment variable is not
set");
        fprintf(stderr,"....exiting\n");
        exit(-1);
    }
    if (getenv("TPCD_NUMSTREAM") == NULL)
    {
        fprintf(stderr,"nThe TPCD_NUMSTREAM environment variable is not
set");
        fprintf(stderr,"....exiting\n");
        exit(-1);
    }
}

/*****
*****
* SEED jen
* we want to print the seed used in the output files. For the seed usage
* we can now reuse the seeds from run to run, therefore all the power runs
* will use the 1st seed in the file, and the throughput streams will use
* the 2nd to #streams+1 seeds.
* determine the seed to use...e.g. given 3 streams will have the following:
*
*      Entry in seed file
*
*      TEST      Stream Number  Run 1  Run 2
*      power      0              1      1
*      throughput 1              2      2
*      2          3              3
*      3          4              4
*
*****
*****/

seedEntry = g_struct->c_l_opt->intStreamNum + 1;
/* end SEED jen */
/* open the generated seed file...if not there, try the default */

sprintf(file_name, "%s%sauditruns%sseedme", env_tpcd_audit_dir,
env_tpcd_path_delim, env_tpcd_path_delim);

if ((fpSeed = fopen(file_name,READMODE)) == NULL )
{
    fprintf(stderr,"nCannot open the seed file, please ensure that\n");
    fprintf(stderr,"the file exists. filename = %s\n",file_name);

```

```

    exit(-1);
}
for (i = 1; i <= seedEntry; i++)
{
    if (feof(fpSeed))
    {
        lSeed = -1; /* seed not available for some reason */
    }
    fscanf(fpSeed,"%ld\n",&lSeed);
}
g_struct->lSeed = lSeed;
fclose(fpSeed);
}

/* check to see if we are to use copy on for the load */
if ((getenv("TPCD_LOG") != NULL ) &&
(!strcmp(uppercase(getenv("TPCD_LOG")), "YES")))
{
    /* okay, we have set LOG_RETAIN on so we need to use copy directory */
    g_struct->copy_on_load = TRUE;
}
else
{
    /* log retain off don't use copy directory */
    g_struct->copy_on_load = FALSE;
}

/*****
/
/* Make sure that DB2 is started. */
/* CONNECT now unless this is a UF stream for a Throughput test. */
/* (aph 98/12/22) */
/*****
/

if (g_struct->c_l_opt->update > 1)
{
    /* This is an update function stream in a throughput run. */
    /* Just make sure that DB2 is started. Each UF child will CONNECT itself. */
    if (verbose) fprintf(stderr,"nStarting the DB2 Database Manager Now\n");
    sqlstar ();
}
else
{
    /* In all other cases, CONNECT to the target database. */
    do
    {
        if (!strcmp(userid,"0")) /* No authentication provided */
            EXEC SQL CONNECT TO :dbname;
        else EXEC SQL CONNECT TO :dbname USER :userid USING :passwd;
        if (sqlca.sqlcode == SQLE_RC_NOSTARTG) {
            if (verbose)
                fprintf(stderr,"nStarting the DB2 Database Manager Now\n");
            sqlstar ();
            connect=0;
        }
        else connect=1;
    } while (!connect);
    error_check();
}

/*****
*****
* All session initialization is performed at connect time or immediately *
* following and is complete before starting the stream. *
*****
*****/

/* Get start timestamp for stream */
get_start_time(&(g_struct->stream_start_time)); /* TIME_ACC jen*/
strcpy(g_struct->file_time_stamp,
get_time_stamp(T_STAMP_FORM_2,&(g_struct->stream_start_time))); /*
TIME_ACC jen*/

```

```

if (getenv("TPCD_RUN_DIR") != NULL)
    strcpy(g_struct->run_dir, getenv("TPCD_RUN_DIR"));
else
    strcpy(g_struct->run_dir, ".");

/* if we are running a throughput test, then we must report the */
/* stream count information...we will report one file per stream */
/* and amalgamate them after all streams have completed */
/* if the number of streams is greater than 0 then this is a throughput test */
switch (g_struct->c_l_opt->update)
{
    case (2):
    case (5):
        /* update throughput function stream */
        sprintf(file_name, "%s%sstrentuf.%s", g_struct->run_dir,
            env_tpcd_path_delim, g_struct->file_time_stamp);
        break;
    case (3):
    case (4):
        /* update power function stream */
        sprintf(file_name, "%s%spstrentuf.%s", g_struct->run_dir,
            env_tpcd_path_delim, g_struct->file_time_stamp);
        break;
    case (1):
        /* power query stream */
        sprintf(file_name, "%s%spstrent%d.%s", g_struct->run_dir,
            env_tpcd_path_delim,
            g_struct->c_l_opt->intStreamNum, g_struct->file_time_stamp);
        break;
    case (0):
        /* throughput query stream */
        sprintf(file_name, "%s%ssstrent%d.%s", g_struct->run_dir,
            env_tpcd_path_delim,
            g_struct->c_l_opt->intStreamNum, g_struct->file_time_stamp);
        break;
}

if (g_struct->stream_report_file = fopen(file_name, WRITEMODE)) == NULL
)
{
    fprintf(stderr, "\nThe output file for the stream count information\n");
    fprintf(stderr, "could not be opened, make sure the filename is correct\n");
    fprintf(stderr, "filename = %s\n", file_name);
    exit(-1);
}

if (g_struct->c_l_opt->update > 1)
{
    /* update function stream */
    fprintf(g_struct->stream_report_file,
        "Update function stream starting at %*. *s\n",
        T_STAMP_3LEN, T_STAMP_3LEN, /* TIME_ACC jen */
        get_time_stamp(T_STAMP_FORM_3, &(g_struct->stream_start_time)));
    /* TIME_ACC jen */
}
else
{
    /* query stream */
    fprintf(g_struct->stream_report_file,
        "Stream number %d starting at %*. *s\n",
        g_struct->c_l_opt->intStreamNum,
        T_STAMP_3LEN, T_STAMP_3LEN, /* TIME_ACC jen */
        get_time_stamp(T_STAMP_FORM_3, &(g_struct->stream_start_time)));
    /* TIME_ACC jen */
}

#ifdef LINUX

    fclose(g_struct->stream_report_file);

#endif

/* set up the update_num_file name so that if we do use semaphores, */
/* we will have a filename to generate the semkey */

```

```

    sprintf(g_struct->update_num_file, "%s%s%s.%s.update.pair.num",
        env_tpcd_audit_dir,
        env_tpcd_path_delim, uppercase(env_tpcd_dbname),
        lowercase(env_user));
    sprintf(g_struct->sem_file, "%s.%s.semfile", env_tpcd_dbname, env_user);
    if (g_struct->c_l_opt->intStreamNum == 0)
    {
        sprintf(g_struct->sem_file2, "%s.%s.semfile2", env_tpcd_dbname, env_user);
    }
    if (verbose) { /* print out the update pair number file for debugging */
        fprintf(stderr, "\n init_setup: strem %d update pair numb file = %s\n",
            g_struct->c_l_opt->intStreamNum, g_struct->update_num_file);
    }

    /* update the
$TPCD_AUDIT_DIR/$TPCD_DBNAME.$USER.update.pair.num file */
    /* update pairs have been run */
    if ((g_struct->c_l_opt->update >= 1) && (g_struct->c_l_opt->update < 4))
        /* on or onl, but not */ /* bbe or > 1 */
    {
        updateFP = fopen(g_struct->update_num_file, "r");
        if (updateFP != NULL )
        {
            fscanf(updateFP, "%d", &updatePairStart);
            fclose(updateFP);
            if (g_struct->c_l_opt->intStreamNum == 0) /* on, 1 update pair */
                updatePairStop = updatePairStart + 1;
            else /* only, multiple update pairs, stream number will be total */
                updatePairStop = updatePairStart + g_struct->c_l_opt->intStreamNum;
            currentUpdatePair = updatePairStart;

            if (updatePairStart <= 0)
            {
                fprintf(stderr, "updatePairStart is bogus!");
                exit(-1);
            }
        }
        else
        {
            fprintf(stderr, "\n %s not set up, set this \n", g_struct->update_num_file);
            fprintf(stderr, "file to contain the number of the update pair to \n");
            fprintf(stderr, "run and resubmit\n");
            exit(-1);
        }
    }

    return ;
}

/*****
*****/
/* A function to print out the column titles for a returned set */
/*****
*****/
void print_headings (struct sqllda *sqllda, int *col_lengths)
{
    int col = 0; /* Column number */
    int col_width = 0; /* width of column */
    int max_col_width = 0; /* maximum column width */
    int col_name_length = 0; /* sizeof column name string */
    int col_type = 0; /* column type */

    int total_length = 0; /* accumulator var. for
length of column headings */

    int loopvar = 0;

    char col_name[256] = "\0";
    unsigned char m, n; /* precision and accuracy
for decimal conversion */

    fprintf (outstream, "\n");

```

```

/** loop through for each column in solution set
and determine the maximum column width */

for (col = 0; col < sqlda->sqlvar[0].sqlname.length; col++) {
    col_name_length=sqllda->sqlvar[col].sqlname.length;
    col_type = sqllda->sqlvar[col].sqltype;
    col_width = sqllda->sqlvar[col].sqlllen;
    strncpy(col_name,(char *)sqllda->sqlvar[col].sqlname.data,col_name_length) ;

    switch (col_type)
    {
    case SQL_TYP_SMALL:
    case SQL_TYP_NSMALL: /* @d30369 tlg */
        col_lengths[col] = TPCDBATCH_MAX (col_name_length,6);
        break;
    case SQL_TYP_INTEGER:
    case SQL_TYP_NINTEGER:
        col_lengths[col] = TPCDBATCH_MAX (col_name_length,11);
        break;
    case SQL_TYP_BIGINT: /*kmwBIGINT*/
    case SQL_TYP_NBIGINT:
        col_lengths[col] = TPCDBATCH_MAX (col_name_length,19);
        break;
    case SQL_TYP_CSTR:
    case SQL_TYP_NCSTR:
    case SQL_TYP_DATE:
    case SQL_TYP_NDATE:
    case SQL_TYP_TIME:
    case SQL_TYP_NTIME:
    case SQL_TYP_STAMP:
    case SQL_TYP_NSTAMP:
    case SQL_TYP_CHAR:
    case SQL_TYP_NCHAR:
    case SQL_TYP_VARCHAR:
    case SQL_TYP_NVARCHAR:
    case SQL_TYP_LONG:
    case SQL_TYP_NLONG:
        col_lengths[col] = TPCDBATCH_MAX (col_name_length,col_width);
        break;

    case SQL_TYP_FLOAT:
    case SQL_TYP_NFLOAT:
        /* kmw - note: TPCDBATCH_PRINT_FLOAT_WIDTH > max long
identifier */
        col_lengths[col] = TPCDBATCH_PRINT_FLOAT_WIDTH;
        break;

    case SQL_TYP_DECIMAL:
    case SQL_TYP_NDECIMAL:

        m=*(struct declen *)&sqllda->sqlvar[col].sqlllen).m;
        n=*(struct declen *)&sqllda->sqlvar[col].sqlllen).n;

        col_lengths[col] = TPCDBATCH_MAX ((int)(m+n), col_name_length);
        /* Special handling for DECIMAL */ /* @d26350 tlg */
        break;

    default:
        fprintf(stderr,"--Unknown column type (%d). Aborting.\n",col_type);
        break;
    }

    fprintf(outstream,"%-*s ",col_lengths[col],col_name_length,col_name);

    total_length += (col_lengths[col] + 2); /* 2 is from padding spaces */
}

fprintf(outstream,"\n");
for (loopvar=0; loopvar < total_length; loopvar++)
    fprintf(outstream,"-");
fprintf(outstream,"\n");
}

```

```

/*****
**/
/* Gets the current system time and prints it out */
/*****
**/
char *get_time_stamp(int form, Timer_struct *time_pointer)
{
    Timer_struct temp_stamp; /* TIME_ACC jen */
    struct tm *tp;
    size_t timeLength = 0;

    /* TIME_ACC jen start */
    if (time_pointer == (Timer_struct *)NULL)
        get_start_time(&temp_stamp);
    else
        temp_stamp = *time_pointer;

    #if defined (SQLUNIX) || defined (SQLAIX)
        tp = localtime((time_t *)&(temp_stamp.tv_sec));
    #elif (defined (SQLOS2) || defined (SQLWINT) || defined (SQLWIN) ||
defined (SQLDOS))
        tp = localtime(&(temp_stamp.time));
    #else
    #error Unknown operating system
    #endif
    /* TIME_ACC jen stop*/

    if ((form == T_STAMP_FORM_1) || (form == T_STAMP_FORM_3))
    {
        /* SUN fix bbe start */
        #if (defined (SQLWINT) || defined (SQLWIN) || defined (SQLOS2) ||
defined (SQLDOS))
            timeLength = strftime(newtime,50,"%x %X",tp);
        #elif (defined (SQLUNIX) || defined (SQLAIX))
            timeLength = strftime(newtime,50,"%D %T",tp); /* SUN ...test this */
        #else
        #error Unknown operating system
        #endif
        /* SUN fix bbe stop */
        /* TIME_ACC jen start*/
        if (form == T_STAMP_FORM_3)
        {
            /* concatenate the microsecond/milliseconds on the end of the */
            /*timestamp jen1006 */
            #if defined (SQLUNIX) || defined (SQLAIX)
                sprintf(newtime+timeLength,"%0.6d",temp_stamp.tv_usec);
            #elif (defined (SQLOS2) || defined (SQLWINT) || defined (SQLWIN) ||
defined (SQLDOS))
                sprintf(newtime+timeLength,"%0.3d",temp_stamp.millitm);
            #else
            #error Unknown operating system
            #endif
            /* TIME_ACC jen stop*/
        }
    }
    else
    {
        if (form == T_STAMP_FORM_2)
            strftime(newtime,50,"%y%m%d-%H%M%S",tp);

        return (newtime);
    }
}

/*****
**/
/* Handle all the processing for the summary table */
/*****
**/

void summary_table (struct global_struct *g_struct)
{

```

```

double arith_mean = 0;
double geo_mean = 0;
int num_stmt = 0;
int num_stmt_for_geo_mean = 0;

double adjusted_a_mean = 0;
double adjusted_g_mean = 0;
double adjusted_g_mean_intern;
double adjusted_max_time = 0;

double Ts = 0; /* different TPC-D metrics */
double Ts1;
double Ts2;
/* double QppD = 0; MARK
double QthD = 0;
double QphD = 0; */

double db_size_frac_part = 0; /* stores the fractional part of db size */
double db_size = 0; /* size in numbers */
char db_size_qualifier[3] = "0"; /* MB, GB or TB */

struct stmt_info
*s_info_ptr,
*s_info_head_ptr,
*max,
*min;

/* Determine the size of the database from the scale factor (1 SF = 1GB) */
if (g_struct->scale_factor < 1.0) {
    db_size = g_struct->scale_factor * 1000;
    strcpy(db_size_qualifier, "MB");
} else if (g_struct->scale_factor >= 1000.0) {
    db_size = g_struct->scale_factor / 1000;
    strcpy(db_size_qualifier, "TB");
} else {
    db_size = g_struct->scale_factor;
    strcpy(db_size_qualifier, "GB");
}

/* computes the fractional part of db_size */
db_size_frac_part = db_size - (int) db_size;

s_info_ptr = g_struct->s_info_ptr; /* Just use a local copy */
s_info_head_ptr = s_info_ptr;

max = s_info_head_ptr;
/* ensure that we are not already setting max to the UF timings */
while ( strstr(max->tag, "UF") != NULL )
    max = max->next;
min = max;

if (g_struct->c_1_opt->outfile) /* create the appropriate output file */
    output_file(g_struct);

/* write the seed used for this run unless it is a qualification run */
/* (qualification runs use the default seed for their queries) or */
/* unless it is the update function stream (no seeds used for this) */
/* (this is an update stream iff update is 2) */
if ((g_struct->c_1_opt->intStreamNum >= 0) &&
    (g_struct->c_1_opt->update != 2)) {
    {
        if (g_struct->lSeed == -1)
        {
            fprintf( outstream, "\nUsing default qgen seed file");
        }
        else
            fprintf( outstream, "\nSeed used for current run = %ld", g_struct->lSeed);
        fprintf( outstream, "\n");
    }
}

/* print out the stream number if we are in a throughput stream and if */
/* this is not the update stream portion of the throughput test */

```

```

if ( (g_struct->c_1_opt->intStreamNum > 0) &&
    (g_struct->c_1_opt->update != 2) )
{
    fprintf( outstream, "Stream number =
%d\n", g_struct->c_1_opt->intStreamNum);
}
/* print the stream start timestamp to the inter file */
fprintf( outstream, "Stream start time stamp %*.*s\n",
    T_STAMP_3LEN, T_STAMP_3LEN, /* TIME_ACC jen*/
    get_time_stamp(T_STAMP_FORM_3, &(g_struct->stream_start_time)));
/* TIME_ACC jen*/
/* print the stream stop timestamp to the inter file */
fprintf( outstream, "Stream stop time stamp %*.*s\n",
    T_STAMP_3LEN, T_STAMP_3LEN, /* TIME_ACC jen*/
    get_time_stamp(T_STAMP_FORM_3, &(g_struct->stream_end_time)));
/* TIME_ACC jen*/

fprintf( outstream, "\n\nSummary of Results\n===== \n");
fprintf( outstream,
    "\nSequence # Elapsed Time Adjusted Time Start Timestamp End
Timestamp\n\n");

/* Go through the linked list and determine which statement had the
highest and lowest elapsed times */
while ( (s_info_ptr != NULL) && (s_info_ptr != g_struct->s_info_stop_ptr) ) {

    /* check if we are in an update function...if so, we do not want to */
    /* consider the update function times as the min or max time */
    if ( strstr(s_info_ptr->tag, "UF") == NULL )
    {
        /* we are not in an update function */
        if (s_info_ptr->elapsed_time > max->elapsed_time)
            max = s_info_ptr;
        else
            if ((s_info_ptr->elapsed_time < min->elapsed_time)
                && (s_info_ptr->elapsed_time > -1))
                min = s_info_ptr;
    }

    s_info_ptr = s_info_ptr->next;
}

s_info_ptr = s_info_head_ptr;

/** Start from the first structure and go through until the stop
pointer is reached **/
while ( (s_info_ptr != NULL) && (s_info_ptr != g_struct->s_info_stop_ptr) ) {

    if (s_info_ptr->elapsed_time != -1) {
        s_info_ptr->adjusted_time = s_info_ptr->elapsed_time;
        /* determine whether the elapsed times have to be adjusted or not */
        /* if this is an update function, we do not adjust the elapsed time */
        if ( strstr(s_info_ptr->tag, "UF") == NULL )
        {
            /* this is not an update function, adjust time if necessary */
            if (max->elapsed_time/min->elapsed_time > 1000)
            {
                /* jmc fix geo_mean calculation...round adjusted time properly
ROUNDING*/
                adjusted_max_time = max->elapsed_time/1000;
                if (s_info_ptr->elapsed_time < adjusted_max_time)
                {
                    s_info_ptr->adjusted_time =
                        (double)((long)((adjusted_max_time + 0.05) * 10))/10.0;
                    if (s_info_ptr->adjusted_time < 0.1)
                        s_info_ptr->adjusted_time = 0.1;
                }
                /* jmc fix geo_mean calculation...round adjusted time properly
ROUNDING end*/
            }
        }
    }
}

/* a value was calculated */

```

```

fprintf (outstream,
        "%-5d %-5.5s %15.1f %15.1f %*.*s %*.*s\n",
        s_info_ptr->stmt_num, s_info_ptr->tag,
        s_info_ptr->elapsed_time, s_info_ptr->adjusted_time,
        T_STAMP_1LEN, T_STAMP_1LEN, s_info_ptr->start_stamp, /*
TIME_ACC jen*/
        T_STAMP_1LEN, T_STAMP_1LEN, s_info_ptr->end_stamp); /*
TIME_ACC jen*/

/* Only update arithmetic mean for queries not update functions */
if ( strstr(s_info_ptr->tag, "UF") == NULL )
{
    arith_mean += s_info_ptr->elapsed_time;
    adjusted_a_mean += s_info_ptr->adjusted_time;
}

if (s_info_ptr->elapsed_time > 0) { /* don't bother finding log of
    numbers < 0 */
    geo_mean += log(s_info_ptr->elapsed_time);
    adjusted_g_mean += log(s_info_ptr->adjusted_time);
}

/* Only update num_stmt for queries not update functions */
if ( strstr(s_info_ptr->tag, "UF") == NULL )
    num_stmt++;
num_stmt_for_geo_mean++;
}

else
    fprintf (outstream, "%-5d %-5.5s %-15s %-15s\n",
            s_info_ptr->stmt_num,
            s_info_ptr->tag, "Not Collected", "Not Collected");

if (s_info_ptr != g_struct->s_info_stop_ptr)
    s_info_ptr = s_info_ptr->next;
}

fprintf(outstream, "\n\nNumber of statements: %d\n\n", s_info_ptr->stmt_num -
1);
/* Calculate the arithmetic and geometric means */

if (geo_mean != 0) { /* Used to test if arith_mean != 0
    Don't bother doing any of this if the
    elapsed time mean is 0 */
    arith_mean = arith_mean / num_stmt;
    adjusted_a_mean = adjusted_a_mean / num_stmt;
    geo_mean = exp(geo_mean / num_stmt_for_geo_mean);
    adjusted_g_mean_intern = adjusted_g_mean; /* MARK */
    adjusted_g_mean = exp(adjusted_g_mean / num_stmt_for_geo_mean);
}

/* print out all the appropriate information including the
different TPC-D metrics */
/* do not bother with this if we are in an update only stream */
fprintf (outstream, "\nGeom. mean queries %7.3f %15.3f\n", \
        geo_mean, adjusted_g_mean);
if (g_struct->c_l_opt->update < 2)
{
    fprintf (outstream, "Arith. mean queries %7.3f %15.3f\n", \
            arith_mean, adjusted_a_mean);

    fprintf (outstream,
            "\n\nMax Qry %-3.3s %15.1f %15.1f %*.*s %*.*s\n",
            max->tag, max->elapsed_time, max->adjusted_time,
            T_STAMP_1LEN, T_STAMP_1LEN, max->start_stamp, /* TIME_ACC
jen*/
            T_STAMP_1LEN, T_STAMP_1LEN, max->end_stamp); /* TIME_ACC
jen*/
    fprintf (outstream,

```

```

        "Min Qry %-3.3s %15.1f %15.1f %*.*s %*.*s\n",
        min->tag, min->elapsed_time, min->adjusted_time,
        T_STAMP_1LEN, T_STAMP_1LEN, min->start_stamp, /* TIME_ACC
jen*/
        T_STAMP_1LEN, T_STAMP_1LEN, min->end_stamp); /* TIME_ACC
jen*/
    }

if (g_struct->c_l_opt->intStreamNum == 0) {
    /* fprintf (outstream, "\n\nMetrics\n===== \n\n"); */

    /* Increase the Ts measurement by one second since the accuracy of our */
    /* timestamps is only to 1 second and if the start was at 1.01 seconds, */
    /* and the end was at 5.99 seconds, we get a free second ... this will */
    /* be made explicit in the upcoming revision of the spec (after 1.0.1) */
    /* TIME_ACC jen start */

    /* NOTE this can probably be better coded by changing get_elapsed_time */
    /* to just calculate the elapsed time give a start and an end time, and */
    /* to also give a precision for the calculation (sec, 10ths....). The */
    /* call then will grab a timestamp before calling. Then we can get rid */
    /* of the if def...and just call get_elapsed_time (which can handle the */
    /* os differences on its own */

#ifdef (SQLUNIX) || defined (SQLAIX)
    Ts = g_struct->stream_end_time.tv_sec - g_struct->stream_start_time.tv_sec +
1;
    Ts1 = (double)g_struct->stream_start_time.tv_sec +
((double)g_struct->stream_start_time.tv_usec/1000000);
    Ts2 = (double)g_struct->stream_end_time.tv_sec +
((double)g_struct->stream_end_time.tv_usec/1000000);

#elif (defined (SQLOS2) || defined (SQLWINT) || defined (SQLWIN) ||
defined (SQLDOS))
    Ts = g_struct->stream_end_time.time - g_struct->stream_start_time.time + 1;
    Ts1 = (double)g_struct->stream_start_time.time +
((double)g_struct->stream_start_time.millitm/1000);
    Ts2 = (double)g_struct->stream_end_time.time +
((double)g_struct->stream_end_time.millitm/1000);

#else
#error Unknown operating system
#endif

    /* TIME_ACC jen stop */

    /* MARK
    ##Now do in calcmetrics.pl##
    QppD = (3600 * g_struct->scale_factor) / adjusted_g_mean;
    QthD = (num_stmt * 3600 * g_struct->scale_factor) / Ts;
    QphD = sqrt(QppD*QthD);
    */

    /* if the decimal part has some meaningful value then print the database size
    with decimal part; otherwise just print the integer part */

    fprintf (outstream,
            "\nGeometric mean interim value = %10.3f\n\nStream Ts %11 =
%10.0f\n\nStream start int representation %11 = %f\n\nStream stop int
representation %11 = %f",
            adjusted_g_mean_intern, Ts, Ts1, Ts2);
    }
}

/*****
/* free up all the elements of the sqlda after done processing */
/*****
void free_sqlda (struct sqlda *sqlda, int select_status) /* @d30369 tlg */
{
    int loopvar;

    if (select_status == TPCDBATCH_SELECT)
        for (loopvar=0; loopvar<sqlda->sqld; loopvar++) {
            free(sqlda->sqlvar[loopvar].sqldata);

```

```

    free(sqlda->sqlvar[loopvar].sqlind);
}

free(sqlda);
sqlda_allocated = 0; /* fix free() problem on NT
                    wlc 090597 */
}

/*****
/* processing to run the insert update function */
*****/
void runUF1 ( struct global_struct *g_struct, int updatePair )
{

    char statement[3000];
    char sourcedir[256];

    int split_updates = 2; /* no. of ways update records are split */
    int concurrent_inserts = 2; /* jenCI no of concurrent updates to be */
    /* jenCI run at once */
    int loop_updates = 1; /* jenCI no of updates to be run in one */
    /* jenCI "concurrent" invocation. should */
    /* jenCI be split_updates / concurrent_inserts */

    int i;
    int streamNum;
#ifdef SQLWINT
    /* PROCESS_INFORMATION childprocess[100]; */
    char commandline[256];
    HANDLE su_hSem;
    char UF1_semfile[256];
#else
    int childpid[100];
    int su_semid; /* semaphore for controlling split updates */
    key_t su_semkey; /* key to generate semid */
#endif
    if (g_struct->c_l_opt->intStreamNum == 0)
        streamNum = 0;
    else
        streamNum = currentUpdatePair - updatePairStart + 1;

    fprintf( outstream, "UF1 for update pair %d, stream %d, starting\n", updatePair,
streamNum);

    /* Start by loading the data into the staging table at each node */
    /* The orderkeys were split earlier by the split_updates program */
    if (env_tpcd_audit_dir != NULL)
        strcpy(sourcedir, env_tpcd_audit_dir);
    else
        strcpy(sourcedir, ".");

    /* Load the orderkeys into the staging table */
    /* In SMP environments one could use a load command but by using a */
    /* script we can keep the code common */
#ifdef SQLWINT
    sprintf (statement, "perl %s\\tools\\ploaduf1 %d\n", sourcedir, updatePair);
#else
    sprintf (statement, "perl %s/tools/ploaduf1 %d 1", sourcedir, updatePair);
#endif
    if (system(statement))
    {
        fprintf (stderr, "ploaduf1 failed for UF1, examine UF1.log for cause.
Exiting.\n");
        if (verbose)
            fprintf (stderr,
                "ploaduf1 failed for UF1, examine UF1.log for cause. Exiting.\n");
        exit (-1);
    }

    fprintf (outstream, "load_update finished for UF1.\n");

    if (getenv ("TPCD_SPLIT_UPDATES") != NULL)
        split_updates = atoi (getenv ("TPCD_SPLIT_UPDATES"));

```

```

    if (getenv ("TPCD_CONCURRENT_INSERTS") != NULL)
/*jenCI*/
        concurrent_inserts = atoi (getenv ("TPCD_CONCURRENT_INSERTS"));
/*jenCI*/
    loop_updates = split_updates / concurrent_inserts; /*jenCI*/

#ifdef SQLWINT
    /* we will use the tpcd.setup file to generate the semaphore key */
    if (getenv ("TPCD_AUDIT_DIR") != NULL) /*begin SEMA */
    {
        /* this is assuming that you will be running this from 0th node */
        sprintf(sourcefile, "%s%ctools%ctpcd.setup",
            getenv("TPCD_AUDIT_DIR"), PATH_DELIM, PATH_DELIM);
    }
    else
    {
        fprintf (stderr, "runUF1 Can't open UF1 semaphore file,TPCD_AUDIT_DIR
is not defined.\n");
        exit (-1);
    }
/*end SEMA */
    su_semkey = ftok (sourcefile, 'J');
    if ((su_semid = semget (su_semkey, 1, IPC_CREAT|S_IRUSR|S_IWUSR)) <
0)
    {
        fprintf (stderr, "Cannot get semaphore! semget failed: errno = %d\n", errno);
        exit (-1);
    }
#else /* SQLWINT */
    sprintf (UF1_semfile, "%s.%s.UF1.semfile", env_tpcd_dbname, env_user);
    su_hSem = CreateSemaphore(NULL, 0,
        concurrent_inserts, /*jenCI*/
        (LPCTSTR)(UF1_semfile));
    if (su_hSem == NULL)
    {
        fprintf(stderr,
            "CreateSemaphore (ready semaphore) failed, GetLastError: %d,
quitting\n",
            GetLastError());
        exit(-1);
    }
#endif /* SQLWINT */
    if (verbose) fprintf(stderr, "Semaphore created successfully!\n");

    fclose(outstream); /* to prevent multiple header caused by forking
                    wlc 081397 */

    for (i=0; i < concurrent_inserts; i++) /*jenCI*/
    {
#ifdef SQLWINT
        if ((childpid[i] = fork()) == 0)
        {
            /* runUF1_fn (updatePair, i); aph 981205 */
            runUF1_fn (updatePair, i, dbname, userid, passwd);
        }
        else
        {
            /* This is the parent */
            if (verbose)
                fprintf (stderr, "stream #%d started with pid %d\n", i, childpid[i]);
        }
#else /* SQLWINT */
        sprintf (commandline,
            "start /b %s\\auditruns\\tpcdbatch.exe -z -d %s -i %d -j 1 -k %d",
            env_tpcd_audit_dir, dbname, updatePair, i ); /* aph 082797 */

        system (commandline);
#endif /* SQLWINT */
        //DJD 021004 sleep (UF1_SLEEP);
    }

    /* All children have been created, now wait for them to finish */
#ifdef SQLWINT
    if (sem_op (su_semid, 0, concurrent_inserts * -1) != 0) /*jenCI*/

```

```

        /*jenSEM*/
        fprintf(stderr,
            "Failure to wait on insert semaphore with %d of children\n",
            concurrent_inserts);
        exit(1);
    }
    /*jenSEM*/
    semctl(su_sem, 0, IPC_RMID, 0);
#else
    for (i = 0; i < concurrent_inserts; i++)
        /*jenCI*/
        {
            if (verbose)
            {
                fprintf(stderr, "About to wait again ...Sets to wait for %d\n",
                    concurrent_inserts - i);
                /*jenCI*/
            }
            if (WaitForSingleObject(su_hSem, INFINITE) == WAIT_FAILED)
            {
                fprintf(stderr,
                    "WaitForSingleObject (su_hSem) failed in runUF1 on set %d, error:
%d, quitting\n",
                    i, GetLastError());
                exit(-1);
            }
        }
    if (!CloseHandle(su_hSem))
    {
        fprintf(stderr,
            "RunUF1 Close Sem failed - Last Error: %d\n", GetLastError());
        /* no exit here */
    }
#endif

    if ( (outstream = fopen(outstreamfilename, APPENDMODE)) == NULL )
    {
        fprintf(stderr, "\nThe output file could not be opened. ");
        fprintf(stderr, "Make sure that the filename is correct.\n");
        fprintf(stderr, "filename = %s\n", outstreamfilename);
        exit(-1);
    }

    fprintf(outstream, "UF1 for update pair %d complete\n", updatePair);
}

/* runUF1_fn() moved to another SQC file          aph 981205 */

/*****
/* processing to run the delete update function */
*****/
void runUF2 ( struct global_struct *g_struct, int updatePair )
{
    char statement[3000];
    char sourcedir[256];

    int split_deletes = 1; /* no. of ways update records are split @dxxxxxhar */
    int concurrent_deletes = 1; /* number of database partitions DELjen */
    int chunks_per_concurrent_delete = 1;

    int i;
    int streamNum;
#ifdef SQLWINT
    char commandline[256];
    HANDLE su_hSem;
    char UF2_semfile[256];
#else
    int childpid[100];
    char sourcefile[256];
    int su_sem; /* semaphore for controlling split updates */
    key_t su_semkey; /* key to generate semid */
#endif
    if (g_struct->c_l_opt->intStreamNum == 0)
        streamNum = 0;
    else

```

```

        streamNum = currentUpdatePair - updatePairStart + 1;

    fprintf(outstream, "UF2 for update pair %d, stream %d, starting\n", updatePair,
        streamNum);

    /* We need to know both how many chunks there are and how many chunks */
    /* are to be executed by each concurrent UF2 process. More chunks means */
    /* both smaller transactions (less deadlock) and more potential concurrency */

    /* How many "chunks" have the orderkeys been divided into? */
    if (getenv("TPCD_SPLIT_DELETES") != NULL)
        split_deletes = atoi(getenv("TPCD_SPLIT_DELETES"));
    /* How many deletes should run concurrently */
    if (getenv("TPCD_CONCURRENT_DELETES") != NULL)
        concurrent_deletes = atoi(getenv("TPCD_CONCURRENT_DELETES"));
    /* How many chunks in each concurrently running delete process */
    chunks_per_concurrent_delete = split_deletes / concurrent_deletes;

    /* Start by loading the data into the staging table at each node */
    /* The orderkeys were split earlier by the split_updates program */
    if (env_tpcd_audit_dir != NULL)
        strcpy(sourcedir, env_tpcd_audit_dir);
    else
        strcpy(sourcedir, ".");

    /* Load the orderkeys into the staging table */
    /* In SMP environments one could use a load command but by using a */
    /* script we can keep the code common */

#ifdef SQLWINT
    sprintf(statement, "perl %s\\tools\\ploaduf2 %d\n", sourcedir, updatePair);
#else
    sprintf(statement, "perl %s/tools/ploaduf2 %d 2", sourcedir, updatePair);
#endif
    if (system(statement))
    {
        fprintf(stderr, "ploaduf2 failed for UF2, examine UF2.log for cause.
Exiting.\n");
        exit(-1);
    }
    fprintf(outstream, "ploaduf2 finished for UF2.\n");

    fclose(outstream); /* to prevent multiple header caused by forking
                        wlc 081397 */

    /* Next we need to get ready to launch a bunch of concurrent processes */
#ifdef SQLWINT
    /* we will use the tpcd.setup file to generate the semaphore key begin
SEMA */
    if (getenv("TPCD_AUDIT_DIR") != NULL)
    {
        sprintf(sourcefile, "%s\\tools\\ctpcd.setup",
            getenv("TPCD_AUDIT_DIR"), PATH_DELIM, PATH_DELIM);
    }
    else
    {
        fprintf(stderr, "runUF2 Can't open UF2 semaphore file, TPCD_AUDIT_DIR
is not defined.\n");
        exit(-1);
    }

    su_semkey = ftok(sourcefile, 'D'); /* use D for deletes */
    /* end SEMA */
    if ( (su_sem = semget(su_semkey, 1, IPC_CREAT | S_IRUSR | S_IWUSR)) <
0)
    {
        fprintf(stderr, "UF2 Can't get semaphore! semget failed: errno = %d\n",
            errno);
        exit(-1);
    }
#else
    sprintf(UF2_semfile, "%s.%s.UF2.semfile", env_tpcd_dbname, env_user);

```



```

//DJD fprintf(stderr,"UF2 semfile = %s\n",UF2_semfile);
su_hSem = CreateSemaphore(NULL, 0,
    concurrent_deletes,
    (LPCTSTR)(UF2_semfile));
if (su_hSem == NULL)
{
    fprintf(stderr,
        "CreateSemaphore (ready semaphore) failed, GetLastError: %d,
quitting\n",
        GetLastError());
    exit(-1);
}
fprintf(stderr,"Semaphore created successfully!\n");
#endif

for (i=0; i < concurrent_deletes; i++)
{
#ifdef SQLWINT
    if ((childpid[i] = fork()) == 0)
    {
        fprintf(stderr, "B-Calling runUF2_fn %d %d %d ...\n",
            updatePair, i, chunks_per_concurrent_delete);
        /* runUF2_fn (updatePair, i, chunks_per_concurrent_delete);  aph 981205
        */
        runUF2_fn (updatePair, i, chunks_per_concurrent_delete, dbname, userid,
passwd);
    }
    else
    {
        /* This is the parent */
        if (verbose)
            fprintf (stderr, "stream #%d started with pid %d\n", i, childpid[i]);
    }
}
#else
{
    /* SECURITY_ATTRIBUTES sec_process;
    SECURITY_ATTRIBUTES sec_thread; */
    /* NEED TO FIX THIS UP - KBS 98/10/20 */

    sprintf (commandline,
        "start /b %s\\auditruns\\tpcdbatch.exe -z -d %s -i %d -j 2 -k %d -x %d",
        env_tpcd_audit_dir, dbname, updatePair, i,
chunks_per_concurrent_delete ); /* aph */
    /* the -x parm should be passed at 0...not 100% sure of this jen */
    //DJD fprintf(stderr, "commandline= %s\n", commandline);
    system (commandline);
    //DJD 021004 sleep (UF2_SLEEP);
}
#endif
}

/* All children have been created, now wait for them to finish */
#ifdef SQLWINT
    fprintf(stderr, "About to wait on the semaphore...\n");
    if (sem_op (su_semid, 0, concurrent_deletes * -1) != 0) /*jenSEM*/
    {
        /*jenSEM*/
        fprintf(stderr,
            "Failure to update wait on delete semaphore with %d children\n",
concurrent_deletes);
        exit(1);
    }
    /*jenSEM*/
    semctl (su_semid, 0, IPC_RMID, 0);
#else
// for (i = 0; i < split_deletes; i++) //DJD Waits forever.....
for (i = 0; i < concurrent_deletes; i++)
{
    if (verbose)
    {
        fprintf(stderr,"About to wait again ...Sets to wait for %d\n",
split_deletes - i);
        fprintf(stderr,"About to wait again ...Sets to wait for %d\n",
concurrent_deletes - i);
    }
}
if (WaitForSingleObject(su_hSem, INFINITE) == WAIT_FAILED)

```

```

{
    fprintf(stderr,
        "WaitForSingleObject (su_hSem) failed on set %d, error: %d,
quitting\n",
        i, GetLastError());
    exit(-1);
}
}
if (! CloseHandle(su_hSem))
{
    fprintf(stderr, "Close Sem failed - Last Error: %d\n", GetLastError());
    /* no exit here */
}
}
#endif

if( (outstream = fopen(outstreamfilename, APPENDMODE)) == NULL )
{
    fprintf(stderr,"The output file could not be opened. ");
    fprintf(stderr,"Make sure that the filename is correct.\n");
    fprintf(stderr,"filename = %s\n",outstreamfilename);
    exit(-1);
}

fprintf(ostream,"UF2 for update pair %d complete\n",updatePair);
}

/* runUF2_fn() moved to another SQC file                                aph 981205 */

/*-----*/
/*    General semaphore function.                                     */
/*-----*/
#ifdef SQLWINT
int sem_op (int semid, int semnum, int value)
{
    struct sembuf sembuf; /* = {semnum, value, 0}; */
    sembuf.sem_num = semnum;
    sembuf.sem_op = value;
    sembuf.sem_flg = 0;

    if (semop(semid,&sembuf,1) < 0)
    {
        fprintf(stderr,"ERROR*** sem_op errorno = %d\n", errno);
        return(-1);
        /* exit(1); */
    }
    return (0); /* successful return jenSEM */
}
#endif

/*****
*/
/* Determines the proper name for the output file to
be generated for a particular TPC-D query, update function, or
interval summary */
/*****
*/
void output_file(struct global_struct *g_struct)
{
    char file_name[256] = "\0";
    char run_dir[150] = "\0";
    char time_stamp[50] = "\0";
    char delim[2] = "\0";
    int qnum=0, found=0; /* kjd715 */
    char input_ln[256] = "\0"; /* kjd715 */
    char tag[128] = "\0"; /* kjd715 */

    strcpy(run_dir,g_struct->run_dir);
    sprintf(delim,"%s",env_tpcd_path_delim);
    strcpy(time_stamp,g_struct->file_time_stamp);
    /* kjd715 */
    if (g_struct->stream_list == NULL)

```

```

{
    if((g_struct->stream_list =
        fopen(g_struct->c_l_opt->infile, READMODE)) == NULL)
    {
        fprintf(stderr, "\nThe input file could not be opened.");
        fprintf(stderr, "Make sure that the filename is correct.\n");
        exit(-1);
    }
}
found = 0;
do {
    fscanf(g_struct->stream_list, "%n%[\n]\n", input_ln);
    if (strstr(input_ln, "--#TAG") == input_ln)
    {
        found = 1;
        strcpy(tag, (input_ln + sizeof("--#TAG")));
        if (strcmp(tag, "UF", 2) == 0)
            qnum = atoi(tag + 2) * (-1);
        else if (strcmp(tag, "Q", 1) == 0)
        {
            /* for query 15a the 'a' must be trimmed */
            /* off before converting to integer */
            if (strlen(tag) > 3)
                tag[3] = '\0';
            qnum = atoi(tag + 1);
        }
    }

    if (feof(g_struct->stream_list))
        found = 1;
} while (!found);
/*
if ((g_struct->stream_list =
    fopen(g_struct->c_l_opt->str_file_name, READMODE)) == NULL)
{
    fprintf(stderr, "\nThe stream list file could not be opened.");
    fprintf(stderr, "Make sure that the filename is correct.\n");
    exit(-1);
}

fscanf(g_struct->stream_list, "%d", &qnum);
*/
/* kjd715 */

switch (g_struct->c_l_opt->intStreamNum)
{
case -1: /* qualifying */
    sprintf(file_name, "%s%sqryqual%02d.%s", run_dir, delim, qnum, time_stamp);
    break;
case 0: /* power tests */
    if (qnum < 0) /* update functions */
        sprintf(file_name, "%s%smps00uf%d.%02d.%s", run_dir, delim, abs(qnum), \
            currentUpdatePair, time_stamp);
    else
        sprintf(file_name, "%s%smpqry%02d.%s", run_dir, delim, qnum, time_stamp);
    break;

default:
    /* if (qnum < 0) - replaced by berni 96/03/26 */
    if (g_struct->c_l_opt->update == 2 ||
        g_struct->c_l_opt->update == 5)
        sprintf(file_name, "%s%smts%02duf%d.%02d.%s", run_dir, delim, \
            currentUpdatePair - updatePairStart + 1, abs(qnum), \
            currentUpdatePair, time_stamp);
    else
        sprintf(file_name, "%s%smts%dqry%02d.%s", run_dir, delim, \
            g_struct->c_l_opt->intStreamNum, qnum, time_stamp);
    break;
}

if (g_struct->c_flags->eo_infile)

```

```

if (g_struct->c_l_opt->update == 2 ||
    g_struct->c_l_opt->update == 5)
    sprintf(file_name, "%s%smtufinter.%s", run_dir, delim, time_stamp);
else
    switch (g_struct->c_l_opt->intStreamNum) {
        case -1:
            sprintf(file_name, "%s%sqryqualinter.%s", run_dir, delim, time_stamp);
            break;
        case 0:
            /* sprintf(file_name, "%s%smpinter.%s", run_dir, delim, time_stamp); */
            if (g_struct->c_l_opt->update == 1)
                sprintf(file_name, "%s%smpqinter.%s", run_dir, delim, time_stamp);
            else
                sprintf(file_name, "%s%smpufinter.%s", run_dir, delim, time_stamp);
            break;
        default:
            if (g_struct->c_l_opt->intStreamNum > 0)
                sprintf(file_name, "%s%smts%dinter.%s", run_dir, delim, g_struct->c_l_opt->intStreamNum, time_stamp);
            else
                fprintf(stderr, "Invalid stream number specified\n");
            break;
    }

strcpy(outstreamfilename, file_name); /* wlc 081397 */

if (!feof(instream) || g_struct->c_flags->eo_infile)
    /* Only create an output file if there are input
    statements left to process, or if we're all done
    and want to print out the summary table file */
    if (outstream = fopen(file_name, WRITEMODE)) == NULL ) {
        fprintf(stderr, "\nThe output file could not be opened. ");
        fprintf(stderr, "Make sure that the filename is correct.\n");
        fprintf(stderr, "filename = %s\n", file_name);
        exit(-1);
    }

return;
}

/*****
*/
/* Determine whether or not we should break out of the block loop
because of an end of file, end of block, or update function.
Also handle some semaphore stuff for update functions */
/*****
*/
int PreSQLprocess(struct global_struct *g_struct, Timer_struct *start_time)
{
    int rc = 1;
    FILE *updateFP;
#ifdef SQLWINT
    int semid; /* semaphore for controlling UFs */
    key_t semkey; /* key to generate semid */
#else
    int SemTimeout = 600000; /* Des time out period of 1 minute */
#endif

    switch (g_struct->c_flags->select_status)
    {
        case TPCDBATCH_NONSQL:
            g_struct->s_info_stop_ptr = g_struct->s_info_ptr;
            /* if we're at the end of the input file, set the stop
            pointer to this structure */
            rc = FALSE;
            break;
        case TPCDBATCH_EOBLOCK:
            rc = FALSE;
            break;
        case TPCDBATCH_INSERT:
            /* we have to check whether or not this is a throughput */
            /* test, and if it is, we have to set up a semaphore to */

```

```

/* control when the update functions are run. We want */
/* them to be run after all the query streams have finished. */
/* What we do is set up the semaphore here, decrement it */
/* in the query streams, and wait for it to get cleared */
/* before we allow the UFs to run. */
/* Note: we only set up the semaphore if: */
/* 1. we are running the throughput test (num of */
/* streams > 0) */
/* 2. we are at the first UF1 (i.e. this is the */
/* case where currentUpdatePair = updatePairStart */
/* we also want to check the sem_on element in the global */
/* structure to see if we want to use semaphores or let */
/* the calling script do the synchronization of the update */
/* stream */
if ( semcontrol == 1 )
{
    /* yes we are to be using semaphores */
    /* is this the 1st time into update function 1 (uf1)? */
    if (currentUpdatePair == updatePairStart )
    {
        /* create the semaphores */
        create_semaphores(g_struct);
        if (g_struct->c_l_opt->intStreamNum != 0)
            /* wait period for runthroughput updates */
            throughput_wait(g_struct);
    }
    /* otherwise continue to run*/
}
if ((g_struct->c_l_opt->update == 3) || (g_struct->c_l_opt->update == 4))
{
    get_start_time(start_time);
    strcpy(g_struct->s_info_ptr->start_stamp,
           get_time_stamp(T_STAMP_FORM_3,start_time)); /* TIME_ACC
jen*/
    /* write the start timestamp to the file...if this is not a qualification */
    /* run, then write the seed used as well */
    fprintf( outstream,"Start timestamp %*.s \n",
            T_STAMP_3LEN,T_STAMP_3LEN, /* TIME_ACC jen*/
            g_struct->s_info_ptr->start_stamp);
    if (g_struct->c_l_opt->intStreamNum >= 0)
    {
        if (g_struct->lSeed == -1)
        {
            fprintf( outstream,"Using default qgen seed file");
        }
        else
            fprintf( outstream,"Seed used = %ld",g_struct->lSeed);
        fprintf( outstream,"\n");
    }
}
if (g_struct->c_l_opt->update < 4){
    /* run only if updates are enabled */
    runUF1(g_struct, currentUpdatePair);
}

rc = FALSE;
if ((g_struct->c_l_opt->intStreamNum == 0) && (semcontrol == 1))
    /* RUNPOWER: release first semaphore so the queries can run */
    release_semaphore(g_struct, INSERT_POWER_SEM);
break;
case TPCDBATCH_DELETE:
    if ((g_struct->c_l_opt->intStreamNum == 0) && (semcontrol == 1))
    {
        /* RUNPOWER: wait for queries to finish */
        /* waiting on QUERY_POWER_SEM semaphore */
        runpower_wait(g_struct, QUERY_POWER_SEM);
    }
    if ((g_struct->c_l_opt->update == 3) || (g_struct->c_l_opt->update == 4))
    {
        get_start_time(start_time);
        strcpy(g_struct->s_info_ptr->start_stamp,
               get_time_stamp(T_STAMP_FORM_3,start_time)); /* TIME_ACC
jen*/
        /* write the start timestamp to the file...if this is not a qualification */

```

```

/* run, then write the seed used as well */
fprintf( outstream,"Start timestamp %*.s \n",
        T_STAMP_3LEN,T_STAMP_3LEN, /* TIME_ACC jen*/
        g_struct->s_info_ptr->start_stamp);
if (g_struct->c_l_opt->intStreamNum >= 0)
{
    if (g_struct->lSeed == -1)
    {
        fprintf( outstream,"Using default qgen seed file");
    }
    else
        fprintf( outstream,"Seed used = %ld",g_struct->lSeed);
    fprintf( outstream,"\n");
}
}
if (g_struct->c_l_opt->update < 4){
    /* run only if updates are enabled */
    runUF2(g_struct, currentUpdatePair);
    if (g_struct->c_l_opt->intStreamNum == 0)
        /* RUNPOWER */
        fprintf(stderr, "UF2 completed\n");
}
currentUpdatePair += 1;
/* update the update.pair.num file to reflect the successfully completed */
/* update pair */
if (g_struct->c_l_opt->update < 4)
{
    /*jen*/
#ifdef NO_INCREMENT
    /* don't update the pair, only for my testing - Haider */
    updateFP = fopen(g_struct->update_num_file,"w");
    fprintf(updateFP,"%d\n",currentUpdatePair);
    fclose(updateFP);
#endif
}
/*jen*/
rc = FALSE;
break;
}
return(rc);
}

/*****
*****/
/* Handles actual processing of SQL statement. Initializes the SQLDA
for returned rows, does PREPARE, DECLARE, and OPEN statements and
executed multiple FETCHes as needed. If not a SELECT statement,
goes into EXECUTE IMMEDIATE section */
/*****
*****/
void SQLprocess(struct global_struct *g_struct)
{
    int rc = 0; /* 912RETRY */
    int rows_fetch = 0;
    long sqlcode = SQL_RC_E911; /* Temporary sqlcode to test
for deadlocks */
    int max_wait = 1; /* Maximum number of retries
for deadlock scenario */

    int col_lengths[TPCDBATCH_MAX_COLS]; /* array containing widths of
columns in returned set */
    struct stmt_info *s_info_ptr;

    s_info_ptr = g_struct->s_info_ptr;
/*****
*****/
    /* grab storage for the SQLDA */
/*****
*****/
    if ((sqlda=(struct sqlda *)malloc(SQLDASIZE(100))) == NULL)
        mem_error("allocating sqlda");

    sqlda->sqln = TPCDBATCH_MAX_COLS; /* @d30369 tlg */

```

```

/* Error-recovery code for errors resulting from multi-stream errors */

while (((sqlcode == SQL_RC_E911) ||
      (sqlcode == SQL_RC_E912) ||
      (sqlcode == SQL_RC_E901)) &&
      (max_wait < MAXWAIT) &&
      (rc==0))
{

    sqlcode = 0;      /* Re-initialize sqlcode to avoid infinite-loop */
    if (g_struct->c_flags->select_status == TPCDBATCH_SELECT)
    {
        /* Enter this loop if SQL stmt is a SELECT */
        EXEC SQL PREPARE STMT1 INTO :sqllda FROM :stmt_str;

        sqlcode = error_check();
        if (sqlcode < 0)
        {
            fprintf(stderr, "\nPrepare failed. Stopping this query.\n");
            rc = -1;
        }
        else /* print out the column headings for the answer set */
        {
            print_headings(sqllda, col_lengths);      /* @d22817 tlg */

            allocate_sqllda(sqllda); /* This is where we set storage for the */
                                     /* SQLDA based on the column types in */
                                     /* the answer set table. */

            EXEC SQL DECLARE DYNCUR CURSOR FOR STMT1;

            EXEC SQL OPEN DYNCUR;
            sqlcode = error_check();

            if (sqlcode < 0) /* we ran into an error of some kind KBS 98/09/28 */
            {
                max_wait++;
                fprintf(stderr, "\nAn error has been detected on open...Retrying...\n");
                SleepSome(10);
            }
            else
            {

                /* Fetch appropriate number of rows and determine whether or not to
                */

                /* send them to file. */

                /* Fetch appropriate number of rows and determine whether or not to
                */

                rows_fetch = 0;

                do
                {
                    /* Keep fetching as long as we haven't finished reading
                    all the rows and we haven't gone past the limits set
                    in the control string */

                    EXEC SQL FETCH DYNCUR USING DESCRIPTOR :sqllda;
                    if (sqlca.sqlcode == 100)
                    {
                        sqlcode = sqlca.sqlcode;
                    }
                    else
                    {
                        sqlcode = error_check();
                    }
                }
                if (sqlcode == 0)
                {
                    rows_fetch++;
                }
            }
        }
    }
}

```

```

        if ((rows_fetch <= s_info_ptr->max_rows_out) ||
            (s_info_ptr->max_rows_out == -1))
            echo_sqllda(sqllda, col_lengths);
    }
    else if (sqlcode < 0)
    {
        max_wait++;
        fprintf(stderr, "\nAn error has been detected on
fetch...Retrying...\n");
        SleepSome(10);
    }
    while ((sqlcode == 0) && \
           ((s_info_ptr->max_rows_fetch == -1) || \
            (rows_fetch < s_info_ptr->max_rows_fetch)))
    { /* end of successful open */
    } /* end of successful prepare */
} /* End of block for handling SELECT statements */

else
{
    /* SQL statement is not a SELECT */
    EXEC SQL EXECUTE IMMEDIATE :stmt_str;
    sqlcode = error_check();

    if (sqlcode < 0)
    {
        max_wait++;
        fprintf(stderr, "\nAn error has been detected on execute
immediate...Retrying...\n");
        SleepSome(10);
    }
} /* end of block for handling NON-select statements */

if ((sqlcode >= 0) &&
    (g_struct->c_flags->select_status == TPCDBATCH_SELECT))
{
    /* we opened a cursor before */
    EXEC SQL CLOSE DYNCUR;
    sqlcode = error_check();

    if ((s_info_ptr->max_rows_fetch == -1) ||
        (rows_fetch < s_info_ptr->max_rows_fetch))
    #ifndef SQLPTX
        fprintf(outstream, "\n\nNumber of rows retrieved is: %6d",
            rows_fetch);
    else
        fprintf(outstream, "\n\nNumber of rows retrieved is: %6d",
            s_info_ptr->max_rows_fetch);
    #else
        fprintf(outstream, "\n\nNumber of rows retrieved is: %6d",
            rows_fetch);
    else
        fprintf(outstream, "\n\nNumber of rows retrieved is: %6d",
            s_info_ptr->max_rows_fetch);
    #endif
} /* @d28763 tlg */

if (s_info_ptr->query_block == FALSE) /* if block is off don't loop */
    g_struct->c_flags->eo_block = TRUE;

} /* end of while loop to retry if needed */

} /* end of SQLprocess */

/* performs some operations after a statement has been processed,
including doing a COMMIT if necessary, and calculating the
elapsed time. Also initializes a new stmt_info structure
for the next block of statements */

int PostSQLprocess(struct global_struct *g_struct, Timer_struct *start_time)
{
    struct stmt_info *s_info_ptr;

```

```

Timer_struct    end_t;    /* end point for elapsed time */

#if DEBUG
    fprintf (outstream, "In PostgreSQLprocess\n");
#endif

    s_info_ptr = g_struct->s_info_ptr;

    if (g_struct->c_flags->select_status == TPCDBATCH_NONSQL)
        return FALSE; /* get out if we've reached the end of input file */

    if (g_struct->c_l_opt->update > 1)
    {
        /* This is an update function stream. There is no need to COMMIT. */
        /* Each UF child will COMMIT its own transactions. */
        ;
    }
    else
    { /* For non-UF cases, COMMIT now. */
        if (g_struct->c_l_opt->a_commit) {
            EXEC SQL COMMIT WORK;
            error_check();          /* @d22275 tlg */
        }
    }

    fflush(outstream);

    s_info_ptr->elapsed_time = get_elapsed_time(start_time);

    if (g_struct->c_flags->time_stamp == TRUE)          /* @d25594 tlg */
        get_start_time(&end_t); /* Get the end time */
        strcpy(s_info_ptr->end_stamp,
            get_time_stamp(T_STAMP_FORM_3,&end_t));
        /*get_time_stamp(T_STAMP_FORM_3,(time_t)NULL));*/

    /* BBE: Pass on time stamp values for the next query */
    temp_time_struct = end_t;
    strcpy(temp_time_stamp, s_info_ptr->end_stamp);

    /* write the start timestamp to the file */
    fprintf( outstream, "\n\nStop timestamp %*.s\n",
        T_STAMP_3LEN, T_STAMP_3LEN, /* TIME_ACC jen */
        s_info_ptr->end_stamp);

    /* DJD print elapsed time in seconds */
    fprintf( outstream, "Query Time = %15.1f secs\n", s_info_ptr->elapsed_time);

    /** Allocate space for a new stmt_info structure */ /* @d24993 tlg */
    s_info_ptr->next =
        (struct stmt_info *) malloc(sizeof(struct stmt_info));
    if (s_info_ptr->next != NULL) {
        memset(s_info_ptr->next, '0', sizeof(struct stmt_info));
        /** Transfer details from one structure to another for
            to apply for the next statement */
        s_info_ptr->next->stmt_num = s_info_ptr->stmt_num + 1;
        s_info_ptr->next->max_rows_fetch = s_info_ptr->max_rows_fetch;
        s_info_ptr->next->max_rows_out = s_info_ptr->max_rows_out;

        s_info_ptr->next->query_block = s_info_ptr->query_block;
        s_info_ptr->next->elapsed_time = -1;

        s_info_ptr = s_info_ptr->next;
    }
    else {
        mem_error("allocating next stmt structure. Exiting\n");
        exit(-1);
    }

    /** Set the stop and travelling pointer to the current info structure */
    g_struct->s_info_stop_ptr = g_struct->s_info_ptr = s_info_ptr;

```

```

    if (sqllda_allocated)
        free_sqllda(sqllda, g_struct->c_flags->select_status);
    /* fix free() problem on NT
       wlc 090597 */

    if (g_struct->c_l_opt->outfile != 0)
        fclose(outstream);

    return (TRUE);
}

/*****
*****
*/
/* Does some cleaning up once all the statements are processed. Disconnects
from the database, cleans up some semaphore stuff from the update functions,
prints out the summary table, and closes all file handles. */
/*****
*****
*/
int cleanup(struct global_struct *g_struct)
{
#ifdef SQLWINT
    int          semid;          /* semaphore for controlling UFs */
    key_t        semkey;        /* key to generate semid */
#endif
    char file_name[256] = "\0";

    /** End timestamp for stream */
    /*g_struct->stream_end_time = time(NULL);*/
    get_start_time(&(g_struct->stream_end_time)); /* TIME_ACC jen */

    switch (g_struct->c_l_opt->update)
    {
        case (2):
        case (5):
            /* update throughput function stream */
            sprintf(file_name, "%s%sstrcntuf.%s", g_struct->run_dir,
                env_tpcd_path_delim, g_struct->file_time_stamp);
            break;
        case (3):
        case (4):
            /* update power function stream */
            sprintf(file_name, "%s%spspstrcntuf.%s", g_struct->run_dir,
                env_tpcd_path_delim, g_struct->file_time_stamp);
            break;
        case (1):
            /* power query stream */
            sprintf(file_name, "%s%spspstrcnt%d.%s", g_struct->run_dir,
                env_tpcd_path_delim,
                g_struct->c_l_opt->intStreamNum, g_struct->file_time_stamp);
            break;
        case (0):
            /* throughput query stream */
            sprintf(file_name, "%s%sstrcnt%d.%s", g_struct->run_dir,
                env_tpcd_path_delim,
                g_struct->c_l_opt->intStreamNum, g_struct->file_time_stamp);
            break;
    }

#ifdef LINUX

    if (g_struct->stream_report_file = fopen(file_name, APPENDMODE)) ==
        NULL )
    {
        fprintf(stderr, "\n\nThe output file for the stream count information\n");
        fprintf(stderr, "could not be opened, make sure the filename is correct\n");
        fprintf(stderr, "filename = %s\n", file_name);
        exit(-1);
    }

#endif

    /* print out the stream stop time in the stream count information file */
    if (g_struct->c_l_opt->update > 1)
    {

```

```

/* update function stream */
fprintf(g_struct->stream_report_file,
        "Update function stream stopping at %s.*s\n",
        T_STAMP_3LEN,T_STAMP_3LEN, /* TIME_ACC jen*/
        get_time_stamp(T_STAMP_FORM_3,&(g_struct->stream_end_time)));
/* TIME_ACC jen*/
}
else
{
/* query stream(s) */
fprintf(g_struct->stream_report_file,
        "Stream number %d stopping at %s.*s\n",
        g_struct->c_l_opt->intStreamNum,
        T_STAMP_3LEN,T_STAMP_3LEN, /* TIME_ACC jen*/
        get_time_stamp(T_STAMP_FORM_3,&(g_struct->stream_end_time)));
/* TIME_ACC jen*/
}
fclose(g_struct->stream_report_file);

/* No need to check for errors here.
Also, the UF stream in a Throughput run
has no connection in tpcdbatch.sqc.      aph 98/12/26
error_check();
*/

/* if we are in a query stream AND this is a throughput test, then need */
/* do to some semaphore stuff (0 implies update functions are off) */
/* AND we are supposed to be using semaphores */

if ( ( semcontrol == 1 ) &&
    ( g_struct->c_l_opt->update < 2))
/* only queries need to release the semaphore at this point */
{
    if (g_struct->c_l_opt->intStreamNum == 0)
        release_semaphore(g_struct, QUERY_POWER_SEM); /* power stream */
    else
        release_semaphore(g_struct, THROUGHPUT_SEM); /* throughput stream
*/

EXEC SQL CONNECT RESET;
#ifdef SQLWINT
    if (verbose)
    {
        fprintf(stderr,
            "cleanup: semkey = %ld, semid = %d, file = %s, stream = %d\n",
            semkey,semid,g_struct->update_num_file,
            g_struct->c_l_opt->intStreamNum);
    }
#endif
}

/** Summary table processing **/          /* @d24993 tlg */
summary_table(g_struct);

fprintf (outstream, "\n\n");

fclose(outstream);      /* Close the output data stream. */
fclose(instream);       /* Close the SQL input stream. */

return (TRUE);
}

void create_semaphores(struct global_struct *g_struct)
{
#ifdef SQLWINT
    int      semid;      /* semaphore for controlling UFs*/
    key_t    semkey;     /* key to generate semid */
#else
    HANDLE    hSem;
    HANDLE    hSem2;

```

```

    int      SemTimeout = 600000; /* Des time out period of 1 minute */
#endif
    fprintf(stderr,"numstreams = %d\n",g_struct->c_l_opt->intStreamNum);
    fprintf(stderr,"Update stream creating semaphore(s) for update and query
sequencing\n");
#ifdef SQLWINT

    fprintf(stderr,"semfile = %s\n",g_struct->sem_file);
    if (g_struct->c_l_opt->intStreamNum == 0)
/*RUNPOWER*/
    {
        fprintf(stderr,"semfile2 = %s\n",g_struct->sem_file2);
        hSem = CreateSemaphore(NULL,
0,1,(LPCTSTR)(g_struct->sem_file));
        hSem2 = CreateSemaphore(NULL,
0,1,(LPCTSTR)(g_struct->sem_file2));
        if ((hSem == NULL) || (hSem2 == NULL))
        {
            fprintf(stderr,
                "CreateSemaphores (ready semaphore) failed, GetLastError: %d,
quitting\n",
                GetLastError());
            exit(-1);
        }
        fprintf(stderr,"Semaphores created successfully!\n");
    }
    else
    {
/* RUNTHROUGHPUT creates semaphores based on the number of query
streams while the number of streams for runpower is constant */
        hSem = CreateSemaphore(NULL, 0,
            g_struct->c_l_opt->intStreamNum,
            (LPCTSTR)(g_struct->sem_file));

        if (hSem == NULL)
        {
            fprintf(stderr,
                "CreateSemaphore (ready semaphore) failed, GetLastError: %d,
quitting\n",
                GetLastError());
            exit(-1);
        }
        fprintf(stderr,"Semaphore created successfully!\n");
    }
}
#else /* AIX, SUN, etc. */
/* create a semaphore key...use the name of a file that */
/* you know exists */
fprintf(stderr,"semfile = %s\n", g_struct->update_num_file);
semkey = ftok(g_struct->update_num_file,'J');
if (g_struct->c_l_opt->intStreamNum == 0)
/* RUNPOWER */
{
    if ( ( semid = semget(semkey,2,IPC_CREAT|S_IRUSR|S_IWUSR)) < 0)
    {
        fprintf(stderr,
            "Throughput can't get initial semaphore! semget failed errno =
%d\n",
            errno);
        exit(1);
    }
}
else
/* THROUGHPUT */
{
    if ( ( semid = semget(semkey,1,IPC_CREAT|S_IRUSR|S_IWUSR)) < 0)
    {
        fprintf(stderr,
            "Throughput can't get initial semaphore! semget failed errno =
%d\n",
            errno);
        exit(1);
    }
}

```

```

    }
    if (verbose)
    {
        fprintf(stderr,
            "insert: semkey = %ld, semid = %d, file = %s, value = %d\n",
            semkey, semid, g_struct->update_num_file,
            (g_struct->c_l_opt->intStreamNum * -1));
    }
}

#endif

/*throughput update */
void throughput_wait(struct global_struct *g_struct)
{
#ifdef SQLWINT
    int      semid;          /* semaphore for controlling UFs*/
    key_t     semkey;        /* key to generate semid */
#else
    HANDLE     hSem;
    int        j;
    int        SemTimeout = 600000; /* Des time out period of 1 minute */
#endif

#ifdef SQLWINT
    hSem = open_semaphore(g_struct, THROUGHPUT_SEM);
    for (j = 0; j < g_struct->c_l_opt->intStreamNum; j++)
    {
        if (verbose)
            fprintf(stderr, "About to wait again ...\n");
        if (WaitForSingleObject(hSem, INFINITE) == WAIT_FAILED)
        {
            fprintf(stderr,
                "WaitForSingleObject (hSem) failed on stream %d, error: %d,
quitting\n",
                j, GetLastError());
            exit(-1);
        }
        if (verbose)
            fprintf(stderr, "Streams to wait for %d\n", j);
    }
    fprintf(stderr, "finished waiting on stream semaphore! Ready to run
updates!\n");
    /* close the semaphore handle */
    if (! CloseHandle(hSem)) {
        fprintf(stderr, "Close Sem failed - Last Error: %d\n", GetLastError());
        /* no exit here */
    }
#else
    semid = open_semaphore(g_struct);
    /* call the sem_op routine to decrement the semaphore by */
    /* however many streams .... by calling this function with*/
    /* a negative number, this stream is forced to wait until */
    /* the semaphore gets back to 0 */
    if (sem_op(semid, 0, (g_struct->c_l_opt->intStreamNum * -1)) != 0)
    {
        /*jenSEM*/
        fprintf(stderr,
            "Failure to wait on throughput semaphore for %d streams\n",
            g_struct->c_l_opt->intStreamNum);
        exit(1);
    }
    /*jenSEM*/
    fprintf(stderr, "finished waiting on stream semaphore! Ready to run
updates!\n");
    semctl(semid, 0, IPC_RMID, 0); /* we've finished waiting, now */
    /* remove the semaphore */
#endif

}

void runpower_wait(struct global_struct *g_struct, int sem_num)
{
    char semfile[150];
#ifdef SQLWINT

```

```

    HANDLE hSem;

    if (sem_num == 1)
        strcpy (semfile, g_struct->sem_file);
    else
        strcpy (semfile, g_struct->sem_file2);

#else /* AIX */
    int      semid;          /* semaphore for controlling UFs*/
    key_t     semkey;        /* key to generate semid */

    strcpy (semfile, g_struct->update_num_file);

#endif

    if (g_struct->c_l_opt->update == 1)
        fprintf(stderr, "querystream waiting for update stream (UF1) to signal
semaphore based on %s\n", semfile);
    else
        fprintf(stderr, "updatestream (UF2) waiting on querystream semaphore to signal
semaphore based on %s\n", semfile);

#ifdef SQLWINT

    hSem = open_semaphore(g_struct, sem_num);
    if (verbose)
        fprintf(stderr, "Runpower queries about to wait ...\n");
    if (WaitForSingleObject(hSem, INFINITE) == WAIT_FAILED)
    {
        fprintf(stderr,
            "WaitForSingleObject (hSem) failed on stream 0, error: %d, quitting\n",
            GetLastError());
        exit(-1);
    }
    if (! CloseHandle(hSem))
    {
        fprintf(stderr, "Close Sem failed - Last Error: %d\n", GetLastError());
        /* no exit here */
    }
#else
    semid = open_semaphore(g_struct);

    /* call the sem_op routine to decrement the semaphore by */
    /* however many streams .... by calling this function with*/
    /* a negative number, this stream is forced to wait until */
    /* the semaphore gets back to 0 */
    /* aix semaphores start at 0, not 1, so sem_num -1 is used */
    if (sem_op(semid, sem_num - 1, -1) != 0)
    {
        /*jenSEM*/
        fprintf(stderr,
            "Failure to wait on runpower semaphore for %d streams\n",
            g_struct->c_l_opt->intStreamNum);
        exit(1);
    }
    /*jenSEM*/
#endif

    if (g_struct->c_l_opt->update == 1)
        fprintf(stderr, "querystream finished waiting on updatestream semaphore\n");
    else
        fprintf(stderr, "updatestream finished waiting on querystream semaphore\n");
}

void release_semaphore(struct global_struct *g_struct, int sem_num)
{
#ifdef SQLWINT
    int      semid;          /* semaphore for controlling UFs*/
    key_t     semkey;        /* key to generate semid */
#else
    HANDLE     hSem;
    int        SemTimeout = 600000; /* Des time out period of 1 minute */
#endif

```

```

#ifdef SQLWINT
    hSem = open_semaphore(g_struct, sem_num); /* query */
    if (! ReleaseSemaphore(hSem,
        1,
        (LPLONG)(NULL)))
    {
        fprintf(stderr, "ReleaseSemaphore failed, Sem#: %d LastError: %d,
quit\n",
            sem_num, GetLastError());
        exit(-1);
    }
#else
    semid = open_semaphore(g_struct); /* query */
    /* aix semaphores start at 0, not 1, so sem_num - 1 is used */
    if (sem_op(semid, sem_num - 1, 1) != 0) /*jenSEM*/
    {
        /*jenSEM*/
        fprintf(stderr,
            "Failed to increment semaphore %d for throughput stream %d\n",
            sem_num, g_struct->c_l_opt->intStreamNum);
        fprintf(stderr,
            "file for generation of semaphore is: %s\n",
            g_struct->update_num_file);
        exit(1);
    }
#endif

if (g_struct->c_l_opt->intStreamNum == 0)
{
    /* RUNPOWER */
    if (sem_num == 1)
    {
        fprintf(stderr, "UF1 completed.\n");
    }
    else
    {
        fprintf(stderr, "query stream completed.\n");
    }
}

#ifdef SQLWINT /* Compile only in NT */
HANDLE open_semaphore(struct global_struct *g_struct, int num)
{
    HANDLE hSem;
    LPCTSTR semfile;

    if (num == 1)
        semfile = (LPCTSTR)g_struct->sem_file;
    else
        semfile = (LPCTSTR)g_struct->sem_file2;

    while ((hSem = OpenSemaphore(SEMAPHORE_ALL_ACCESS |
        SEMAPHORE_MODIFY_STATE |
        SYNCHRONIZE,
        TRUE,
        semfile))
        == (HANDLE)(NULL))
    {
        /*
        ** if cannot open the semaphore, wait for 0.1 second
        */
        fprintf(stderr, "Retry Open semaphore %s\n", semfile);

        Sleep(1000);
    }
    return hSem;
}

#else /* Compile only in non-NT (i.e. AIX) */
int open_semaphore(struct global_struct *g_struct)
{
    int semid; /* semaphore for controlling UFs */
    key_t semkey; /* key to generate semid */
    int num;

```

```

    if (g_struct->c_l_opt->intStreamNum == 0)
        num = 2;
    else
        num = 1;

    semkey = ftok(g_struct->update_num_file, 'J');
    while ((semid = semget(semkey, num, 0)) < 0)
    {
        if (errno == ENOENT)
        {
            sleep(2);
            fprintf(stderr, "cleanUp: looping for access to semaphore stream %d
",
                g_struct->c_l_opt->intStreamNum);
            fprintf(stderr, "semkey=%ld semid = %d file=%s\n", semkey, semid,
                g_struct->update_num_file);
        }
        else
        {
            fprintf(stderr, "query stream %d semget failed errno = %d\n",
                g_struct->c_l_opt->intStreamNum, errno);
            exit(1);
        }
    }
    return semid;
}
#endif

```

tpcdUF.sqc

```

/*****
*****
*
* TPCDUF.SQC
*
* Revision History:
*
* 05 dec 98 aph Created tpcdUF.sqc containing runUF1_fn() and runUF2_fn()
* so that it can be bound separately with a different isolation level.
* 15 may 99 bbe Added cast (short) for type conversion between a long and a
short.
* 16 jun 99 jen Added in proper connect reset code for UF functions (mistakenly
removed
* 17 jun 99 jen SEMA Changes semaphore file for update functions to look for
tpcd.setup
* not for the orders.*** update data file (AIX only )
* 21 jul 99 bbe Commented out conditions in SQL statments that searched on
fields
* other than app_id.
*
*****
*****/

#define UF1DEBUG
#define UF2DEBUG

#if (defined(SQLPTX) && defined(SQLSUN))
#define exit(rc) _exit(rc)
#else
#define exit(rc) exit(rc)
#endif /* SQLPTX & SQLSUN */

#include "tpcdbatch.h"
/* EXEC SQL INCLUDE SQLCA; */

#include "sqlca.h"
extern struct sqlca sqlca;

/*****
*****
*/
/* Function Prototypes
*/

```



```

/*****
*****/
extern int SleepSome( int amount );
extern long error_check(void); /* @d28763 tjg */
extern void dumpCa(struct sqlca*); /*kmw*/
extern int sem_op( int semid, int semnum, int value);
extern char *get_time_stamp(int form, Timer_struct *timer_pointer); /*
TIME_ACC jen */

/*****
*****/
/* Declare the SQL host variables. */
/*****
*****/
EXEC SQL BEGIN DECLARE SECTION;
char UF_dbname[9] = "\0";
char UF_userid[9] = "\0";
char UF_passwd[9] = "\0";
sqlint32 UF_chunk = 0;
short month = 0;
EXEC SQL END DECLARE SECTION;

/*****
*****/
/* Declare the global variables. */
/*****
*****/
extern char env_tpcd_tmp_dir[150];
extern FILE *instream, *outstream; /* File pointers */
extern char sourcefile[256]; /* Used for semaphores and table functions?*/
extern struct {
    short len;
    char data[32700];
} stmt_str; /* jen LONG */

/*****
*****/
/* UF1 child */
/* (i is the application number.) */
/*****
*****/
void runUF1_fn ( int updatePair, int i, char *dbname, char *userid, char *passwd )
{
    int rc = 0;
    int split_updates = 2; /* no. of ways update records are split */
    int concurrent_inserts = 2; /* jenCI no of concurrent updates to be */
    /* jenCI run at once*/
    int loop_updates = 1; /* jenCI no of updates to be run in one */
    /* jenCI "concurrent" invocation. should*/
    /* jenCI be split_updates / concurrent_inserts*/
    int startChunk = 0; /* jenCI number of first chunk to insert for */
    /* jenCI this child */
    int stopChunk = 0; /* jenCI number of last chunk to insert for */
    /* jenCI this child */
    long insertedLineitem = 0; /*kmw*/
    long insertedOrders = 0; /*kmw*/
    long saveInsertedOrders = 0; /*kbs*/

    long sqlcode;
    int maxwait;

#ifdef SQLWINT
    int su_semId;
    key_t su_semkey;
#else
    HANDLE su_hSem;
    char UF1_semfile[256];
#endif

    char myoutstreamfile[256];
    FILE *myoutstream;

    strcpy(UF_dbname, dbname);

```

```

    strcpy(UF_userid, userid);
    strcpy(UF_passwd, passwd);

    /* Get ready to start logging diagnostic output */
    sprintf(myoutstreamfile, UF1OUTSTREAMPATTERN, env_tpcd_tmp_dir,
    PATH_DELIM,
        updatePair, i);
    if ( (myoutstream = fopen(myoutstreamfile, WRITEMODE)) == NULL)
    {
        fprintf(stderr, "\nThe output file '%s' for update pair %d set %d could not be
opened. runUF1_fn\n",
            myoutstreamfile, updatePair, i);
        rc=-1;
        goto UF1_exit;
    }
    outstream=myoutstream; /* initialize outstream for error_check dxxxxhar*/

    fprintf(myoutstream, "\nUF1 for update pair %d set %d starting at %*.s\n",
        updatePair, i,
        T_STAMP_1LEN, T_STAMP_1LEN, /* TIME_ACC jen*/
        get_time_stamp(T_STAMP_FORM_1, (Timer_struct *)NULL)); /*
TIME_ACC jen*/

    if (getenv("TPCD_SPLIT_UPDATES") != NULL)
        split_updates = atoi(getenv("TPCD_SPLIT_UPDATES"));
    if (getenv("TPCD_CONCURRENT_INSERTS") != NULL)
/*jenCI*/
        concurrent_inserts = atoi(getenv("TPCD_CONCURRENT_INSERTS"));
/*jenCI*/
        loop_updates = split_updates / concurrent_inserts; /*jenCI*/

    /* determine the starting and stopping point of the chunks that this jenCI*/
    /* invocation will apply. i is starting chunk number with range 0 jenCI*/
    /* through (concurrent_inserts -1) jenCI*/
    startChunk = i * loop_updates; /*jenCI*/
    stopChunk = startChunk + (loop_updates - 1); /*jenCI*/

    /* Establish a connection to the database */
    if (!strcmp(userid, "\0")) /* No authentication provided */
        EXEC SQL CONNECT TO :UF_dbname;
    else
        EXEC SQL CONNECT TO :UF_dbname USER :UF_userid USING
:UF_passwd;
    error_check();
    if (sqlca.sqlcode < 0)
    {
        rc=-1;
        goto UF1_exit;
    }

    /* Start processing each chunk in my range */
#ifdef UF1DEBUG
    fprintf(myoutstream, "Before loop_a startChunk = %d, stopChunk = %d\n",
startChunk, stopChunk);
    fflush(myoutstream);
#endif
    for ( UF_chunk = startChunk; UF_chunk <= stopChunk; UF_chunk++ )
/*jenCI*/
    {
        /* wlc 062797 */
        sqlcode = SQL_RC_E911;
        month = (short)UF_chunk; /* Cast 'short' added bbe */
        maxwait = 1;
        rc = 0;

#ifdef UF1DEBUG
        fprintf(myoutstream, "Before While_a Chunk= %d\n", UF_chunk);
        fflush(myoutstream);
#endif
        /* Loop to handle any deadlocks */
        while (sqlcode == SQL_RC_E911 && maxwait <= MAXWAIT && rc==0)
        {

```

```

        sqlcode = 0;
#ifdef UF1DEBUG
        fprintf(myostream, "in loop before orders exec sql\n");
        fflush(myostream);
#endif
        EXEC SQL INSERT INTO TPCD.ORDERS
            SELECT
O_ORDERKEY,O_CUSTKEY,O_ORDERSTATUS,O_TOTALPRICE,

O_ORDERDATE,O_ORDERPRIORITY,O_CLERK,O_SHIPPRIORITY,O_CO
MMENT
            FROM TPCDTEMP.ORDERS_NEW
            WHERE APP_ID = :UF_chunk;
/*AND
12*(YEAR(O_ORDERDATE)-1992)+MONTH(O_ORDERDATE)-01 =
:month;*/

        if (sqlca.sqlcode < 0)
            sqlcode = error_check();

        if (sqlcode == SQL_RC_E911)
        {
            /* we've hit a deadlock */
            fprintf(myostream,
                "\nDeadlock detected inserting from tpcdtemp.orders_new for chunk
%d for pair %d..Retrying...\n",UF_chunk,updatePair);
            SleepSome(UF_DEADLOCK_SLEEP);
            maxwait++;
            /* jen DEADLOCK */
        }
        else if (sqlcode < 0)
        {
            fprintf(myostream,
                "Insert into orders pair %d chunk %d failed sqlcode=%d\n",
                updatePair,UF_chunk,sqlcode);
            dumpCa(&sqlca);
            rc = -1;
        }
        else
        {
            /* Everything worked with ORDERS, proceed with LINEITEM */
            saveInsertedOrders = sqlca.sqlerrd[2];

            sqlcode = 0;
#ifdef UF1DEBUG
            fprintf(myostream, "in lineitem for update pair number %d set %d
chunk %d\n",
                updatePair, i,UF_chunk);
            fflush(myostream);
#endif

            EXEC SQL INSERT INTO TPCD.LINEITEM
            SELECT
L_ORDERKEY,L_PARTKEY,L_SUPPKEY,L_LINENUMBER,L_QUANTITY,
L_EXTENDEDPRISE,L_DISCOUNT,L_TAX,

L_RETURNFLAG,L_LINESTATUS,L_SHIPDATE,L_COMMITDATE,L_REC
EIPTDATE,
            L_SHIPINSTRUCT,L_SHIPMODE,L_COMMENT
            FROM TPCDTEMP.LINEITEM_NEW WHERE APP_ID = :UF_chunk;
/*(AND L_ORDERKEY IN
(SELECT O_ORDERKEY FROM TPCD.ORDERS
WHERE
12*(YEAR(O_ORDERDATE)-1992)+MONTH(O_ORDERDATE)-01 =
:month);*/

            if (sqlca.sqlcode < 0)
                sqlcode = error_check();

            if (sqlcode == SQL_RC_E911)
            {
                /* we've hit a deadlock */
                fprintf(myostream,
                    "\nA deadlock has been detected inserting from
tpcdtemp.lineitem%d_%d..Retrying...\n",
                    updatePair, UF_chunk);

```

```

            SleepSome(UF_DEADLOCK_SLEEP);
            maxwait++;
            /* jen DEADLOCK */
        }
        else if (sqlcode < 0)
        {
            fprintf(myostream,
                "Insert into lineitem pair %d chunk %d failed sqlcode=%d\n",
                updatePair,UF_chunk,sqlcode);
            dumpCa(&sqlca);
            rc = -1;
        }
        else
        {
#ifdef UF1DEBUG
            fprintf(myostream, "lineitem insert succeeded\n");
            fflush(myostream);
#endif
            /* accumulate the number of row inserted */
            /* Order count ONLY updated if both orders and lineitem */
            /* go through */
            insertedOrders += saveInsertedOrders;
            /* kbs */
            insertedLineitem += sqlca.sqlerrd[2];
            rc=0;
            EXEC SQL COMMIT WORK;
            error_check();

#ifdef UF1DEBUG
            /* report the number of row inserted */
            fprintf(myostream, " interim %ld rows for chunk %d into
TPCD.ORDERS at %*.s\n",
                insertedOrders,UF_chunk,T_STAMP_1LEN,T_STAMP_1LEN, /*
TIME_ACC jen*/
                get_time_stamp(T_STAMP_FORM_1,(Timer_struct *)NULL)); /*
TIME_ACC jen*/
            /* report the number of row deleted *s inserted */
            fprintf(myostream,
                " interim %ld rows for chunk %d into TPCD.LINEITEM at
%*.s\n",
                insertedLineitem,UF_chunk,
                T_STAMP_1LEN,T_STAMP_1LEN, /* TIME_ACC jen*/
                get_time_stamp(T_STAMP_FORM_1,
                    (Timer_struct *)NULL)); /* TIME_ACC jen*/

            fprintf(myostream,
                " inserts for update pair %d chunk %d complete at %*.s\n\n",
                updatePair, UF_chunk,
                T_STAMP_1LEN,T_STAMP_1LEN, /* TIME_ACC jen*/
                get_time_stamp(T_STAMP_FORM_1,
                    (Timer_struct *)NULL)); /* TIME_ACC jen*/
#endif
        }
        /* process lineitem INSERTs */
        /* while loop for deadlocks */
        /* while processing chunks */

        /* report the number of row deleted */
        fprintf(myostream, "%ld rows inserted into TPCD.ORDERS at %*.s\n",
            insertedOrders,T_STAMP_1LEN,T_STAMP_1LEN, /* TIME_ACC jen*/
            get_time_stamp(T_STAMP_FORM_1,(Timer_struct *)NULL)); /*
TIME_ACC jen*/
        fprintf(myostream, "%ld rows inserted into TPCD.LINEITEM at %*.s\n",
            insertedLineitem,T_STAMP_1LEN,T_STAMP_1LEN, /* TIME_ACC
jen*/
            get_time_stamp(T_STAMP_FORM_1,(Timer_struct *)NULL)); /*
TIME_ACC jen*/

        if (sqlcode < 0)
        {
            if (sqlcode == SQL_RC_E911)
            {
                fprintf(myostream, "# of deadlocks exceeds %i\n", MAXWAIT);
            }
            rc=-1;

```

```

EXEC SQL ROLLBACK WORK;
error_check();          /* @d22275 tjc */

goto UF1_exit;
}

/* UF1_conn_reset: */
EXEC SQL CONNECT RESET;
error_check();          /* @d22275 tjc */

UF1_exit:
fclose(myostream);
/* exiting, increment the semaphore */

/* we used the first flat file to generate the semaphore key */

#ifdef SQLWINT
/* we will use the tpcd.setup file to generate the semaphore key begin SEMA
*/
if (getenv("TPCD_AUDIT_DIR") != NULL)
{
/* this is assuming that you will be running this from 0th node */
sprintf(sourcefile, "%s%ctools%ctpcd.setup",
getenv("TPCD_AUDIT_DIR"), PATH_DELIM, PATH_DELIM);
}
else
{
fprintf(stderr, "Can't open UF1 semaphore file TPCD_AUDIT_DIR is not
defined.\n");
exit(-1);
}
/* end SEMA */

su_semkey = ftok(sourcefile, 'J');
while ((su_sem_id = semget(su_semkey, 1, 0)) < 0)
{
if (errno == ENOENT) {
sleep(2);
}
else {
fprintf(stderr, "update set %d: semget failed errno = %d\n",
i, errno);
exit(1);
}
}
if (sem_op(su_sem_id, 0, 1) != 0)          /*jen SEM*/
{
fprintf(stderr, "Failure to increment semaphore UF1 set %d\n", i);
fprintf(stderr, " semaphore sourcefile = %s su_sem_id =
su_sem_id\n", sourcefile);
exit(1);
}          /*jenSEM*/

#else /* SQLWINT */
sprintf(UF1_semfile, "%s.%s.UF1.semfile",
getenv("TPCD_DBNAME"), getenv("USER"));
//DJD fprintf(stderr, "UF1 semfile = %s\n", UF1_semfile);
while ((su_hSem = OpenSemaphore(SEMAPHORE_ALL_ACCESS |
SEMAPHORE_MODIFY_STATE |
SYNCHRONIZE,
TRUE,
UF1_semfile))
== (HANDLE)(NULL))
{
/*
** if cannot open the semaphore, wait for 0.1 second
*/
fprintf(stderr, "Retry Open semaphore %s\n", UF1_semfile);

sleep(1);
}

if (! ReleaseSemaphore(su_hSem,

```

```

1,
(LPLONG)(NULL)))
{
fprintf(stderr, "ReleaseSemaphore failed, LastError: %d, quit\n",
GetLastError());
exit(-1);
}
#endif /* SQLWINT */
exit(rc);          /* child exiting after finishing up */
}

/*****
**/
/* UF2 child */
/*****
**/
void runUF2_fn ( int updatePair, int thisConcurrentDelete, int numChunks, char
*dbname, char *userid, char *passwd )
{
int rc = 0;
long sqlcode;
int maxwait;
int startChunk = thisConcurrentDelete*numChunks; /* where do we start? */
long deletedLineitems = 0; /*kmw*/
long deletedOrders = 0; /*kmw*/
long savedDeletedLineitems = 0; /*kbs*/

#ifdef SQLWINT
int su_sem_id; /* semaphore for controlling split updates */
key_t su_semkey; /* key to generate semid */
#else
HANDLE su_hSem;
char UF2_semfile[256];
#endif

char myostreamfile[256];
FILE *myostream, *src_fh=NULL;

strcpy(UF_dbname, dbname);
strcpy(UF_userid, userid);
strcpy(UF_passwd, passwd);

/* Generate the unique filename for this concurrent delete process */
sprintf(myostreamfile, UF2OUTSTREAMPATTERN, env_tpcd_tmp_dir,
PATH_DELIM,
updatePair, thisConcurrentDelete);
if ((myostream = fopen(myostreamfile, WRITEMODE)) == NULL)
{
fprintf(stderr,
"\n\nThe output file '%s' for update pair %d set %d could not be opened
runUF2_fn.\n",
myostreamfile, updatePair, thisConcurrentDelete);
rc=-1;
goto UF2_exit;
}

outstream=myostream; /* initialize outstream for error_check dxxxxhar */

#ifdef UF2DEBUG
fprintf(myostream, "RunUF2 Called %d %d %d\n",
updatePair, thisConcurrentDelete, numChunks);
fflush(myostream);
#endif
fprintf(myostream,
"\nUF2 for update pair %d set %d starting at %*.s\n",
updatePair, thisConcurrentDelete, T_STAMP_1LEN, T_STAMP_1LEN,
/* TIME_ACC jen */
get_time_stamp(T_STAMP_FORM_1, (Timer_struct *)NULL)); /*
TIME_ACC jen */

#ifdef UF2DEBUG

```

```

fprintf(myostream, "before connect\n");
fflush(myostream);
#endif

if (!strcmp(userid, "\0")) /** No authentication provided **/
    EXEC SQL CONNECT TO :UF_dbname;
else
    EXEC SQL CONNECT TO :UF_dbname USER :UF_userid USING
:UF_passwd;
    error_check();

#ifdef UF2DEBUG
    fprintf(myostream, "after connect startchunk= %d, EndChunk = %d\n",
        startChunk, startChunk+numChunks);
    fflush(myostream);
#endif

/* Start processing each chunk in my range */
for ( UF_chunk = startChunk; UF_chunk < startChunk+numChunks;
UF_chunk++)
{

    /* Set things up for the loop which will retry if there is a deadlock */
    sqlcode = SQL_RC_E911;
    month = (short)UF_chunk;
    maxwait = 1;
    rc = 0;

#ifdef UF2DEBUG
    fprintf(myostream, "Chunk = %d\n", UF_chunk);
    fflush(myostream);
#endif
    while (sqlcode == SQL_RC_E911 && maxwait <= MAXWAIT && rc == 0)
    {

#ifdef UF2DEBUG
        fprintf(myostream, "in loop before orders exec sql\n");
        fflush(myostream);
#endif
        sqlcode = 0;

        EXEC SQL DELETE FROM TPCD.LINEITEM
        WHERE L_ORDERKEY IN
        (SELECT O_ORDERKEY FROM TPCDTEMP.ORDERS_DEL
        WHERE APP_ID = :UF_chunk);
        /*AND O_ORDERKEY IN
        (SELECT O_ORDERKEY FROM TPCD.ORDERS
        WHERE
        12*(YEAR(O_ORDERDATE)-1992)+MONTH(O_ORDERDATE)-01 =
        :month));*/
        if (sqlca.sqlcode < 0)
            sqlcode = error_check();

        if (sqlcode == SQL_RC_E911)
        {
            /* we've hit a deadlock */
            fprintf(myostream,
                "\nA deadlock detected while deleting from LINEITEM: update pair %d
set %d chunk %d. Retrying.\n",
                updatePair, thisConcurrentDelete, UF_chunk);
            dumpCa(&sqlca);
            SleepSome(UF_DEADLOCK_SLEEP);
            maxwait++; /* jen DEADLOCK */
        }
        else if (sqlcode < 0)
        {
            fprintf(myostream, "\n%s\n", stmt_str.data);
            fprintf(myostream, "\nsqlcode %d occurred deleting from
TPCD.LINEITEM\n", sqlca.sqlcode);
            dumpCa(&sqlca);
            fprintf(myostream,
                "for update pair number %d set %d chunk %d..Exiting\n",
                updatePair, thisConcurrentDelete, UF_chunk);
            rc=-1;
        }
    }
}

```

```

else
{
    /* accumulate the number of row deleted */
    savedDeletedLineitems = sqlca.sqlerrd[2]; /*kbs*/

#ifdef UF2DEBUG
    fprintf(myostream, "in loop for update pair number %d set %d chunk
%d\n",
        updatePair, thisConcurrentDelete, UF_chunk);
    fflush(myostream);
#endif

    /* delete the orders now */

    EXEC SQL DELETE FROM TPCD.ORDERS
    WHERE O_ORDERKEY IN
    (SELECT O_ORDERKEY FROM TPCDTEMP.ORDERS_DEL
    WHERE APP_ID = :UF_chunk);
    /*AND
    12*(YEAR(O_ORDERDATE)-1992)+MONTH(O_ORDERDATE)-01 =
    :month;*/

    if (sqlca.sqlcode < 0)
        sqlcode = error_check();

    if (sqlcode == SQL_RC_E911)
    {
        /* we've hit a deadlock */
#ifdef UF2DEBUG
        fprintf(myostream, "orders deadlocked\n");
        fflush(myostream);
#endif
        fprintf(myostream,
            "\nA deadlock detected while deleting from ORDERS: update pair %d
set %d chunk %d. Retrying.\n",
            updatePair, thisConcurrentDelete, UF_chunk);
        dumpCa(&sqlca);
        SleepSome(UF_DEADLOCK_SLEEP);
        maxwait++; /* jen DEADLOCK */
    }
    else if (sqlcode < 0)
    {
#ifdef UF2DEBUG
        fprintf(myostream, "orders failed\n");
        fflush(myostream);
#endif
        fprintf(myostream, "\nAn error %d occurred deleting from
TPCD.ORDERS\n", sqlca.sqlcode);
        dumpCa(&sqlca);
        fprintf(myostream, "for update pair number %d set %d chunk
%d..Exiting\n",
            updatePair, thisConcurrentDelete, UF_chunk);
        rc=-1;
    }
    else
    {
#ifdef UF2DEBUG
        fprintf(myostream, "orders succeeded\n");
        fflush(myostream);
#endif
        /* accumulate the number of row deleted */
        /* Order count ONLY updated if both orders and lineitem */
        /* go through */
        deletedLineitems += savedDeletedLineitems; /* kbs */
        deletedOrders += sqlca.sqlerrd[2];
        rc=0;
        EXEC SQL COMMIT WORK;
        error_check();
#ifdef UF2DEBUG
        /* report the number of rows deleted */
        fprintf(myostream, " interim %ld rows for chunk %d from
TPCD.ORDERS at %*s\n",
            deletedOrders, UF_chunk, T_STAMP_1LEN, T_STAMP_1LEN, /*
TIME_ACC jen*/

```

```

        get_time_stamp(T_STAMP_FORM_1,(Timer_struct *)NULL)); /*
TIME_ACC jen*/
        fprintf(myostream, "    interim %ld rows for chunk %d from
TPCD.LINEITEM at %*.s\n",
        deletedLineitems,UF_chunk,T_STAMP_1LEN,T_STAMP_1LEN,
/* TIME_ACC jen*/
        get_time_stamp(T_STAMP_FORM_1,(Timer_struct *)NULL)); /*
TIME_ACC jen*/
        fprintf( myostream,
        "    deletes for update pair %d chunk %d complete at %*.s\n\n",
        updatePair, UF_chunk,
        T_STAMP_1LEN,T_STAMP_1LEN, /* TIME_ACC jen*/
        get_time_stamp(T_STAMP_FORM_1,
        (Timer_struct *)NULL)); /* TIME_ACC jen*/
#endif
    }
    } /* process orders deletes */
    } /* while trying to delete one chunk loop */
    } /* while there are more chunks */

#ifdef UF2DEBUG
    fprintf(myostream,"after loop\n");
    fflush(myostream);
#endif
    /* report the number of row deleted */
    fprintf(myostream, "%ld rows deleted from TPCD.ORDERS at %*.s\n",
    deletedOrders,T_STAMP_1LEN,T_STAMP_1LEN, /* TIME_ACC jen*/
    get_time_stamp(T_STAMP_FORM_1,(Timer_struct *)NULL)); /*
TIME_ACC jen*/
    fprintf(myostream, "%ld rows deleted from TPCD.LINEITEM at %*.s\n",
    deletedLineitems,T_STAMP_1LEN,T_STAMP_1LEN, /* TIME_ACC
jen*/
    get_time_stamp(T_STAMP_FORM_1,(Timer_struct *)NULL)); /*
TIME_ACC jen*/

    if (sqlca.sqlcode < 0)
    {
        fprintf(myostream,"# of deadlocks %d exceeds %i\n",
maxwait,MAXWAIT);
        rc=-1;
        EXEC SQL ROLLBACK WORK;
        error_check(); /* @d22275 tjg */
    }

/* UF2_conn_reset: */ /*971101jen*/
EXEC SQL CONNECT RESET;
error_check(); /* @d22275 tjg */

UF2_exit:
    fclose(myostream);

    /* exiting, increment the semaphore */
#ifdef SQLWINT
    /* we used the tpcd.setup file to generate the semaphore key begin SEMA */
    if (getenv("TPCD_AUDIT_DIR") != NULL)
    {
        sprintf(sourcefile, "%s%ctools%ctpcd.setup",
        getenv("TPCD_AUDIT_DIR"), PATH_DELIM, PATH_DELIM);
    }
    else
    {
        fprintf(stderr, "Can't open UF2 semaphore file TPCD_AUDIT_DIR is not
defined.\n");
        exit (-1);
    }

    su_semkey = ftok (sourcefile, 'D'); /* use D for deletes */
/* end SEMA */
    while ((su_semid = semget(su_semkey,1,0)) < 0)
    {
        if (errno == ENOENT)
            sleep(2);
        else {

```

```

        fprintf(stderr,"UF2 update stream %d: semget failed errno = %d\n",
        updatePair, errno);
        exit(1);
    }
}
if (sem_op (su_semid, 0, 1) != 0 ) /*jenSEM*/
{
    /*jenSEM*/
    fprintf(stderr,"Failure to increment semaphore UF2 set %d\n",
thisConcurrentDelete);
    exit(1);
} /*jenSEM*/

#else
    sprintf (UF2_semfile, "%s.%s.UF2.semfile",
        getenv("TPCD_DBNAME"), getenv("USER"));
    //DJD fprintf(stderr,"UF2 semfile = %s\n",UF2_semfile);
    while ((su_hSem = OpenSemaphore(SEMAPHORE_ALL_ACCESS |
        SEMAPHORE_MODIFY_STATE |
        SYNCHRONIZE,
        TRUE,
        UF2_semfile))
        == (HANDLE)(NULL)) {
        /*
        ** if cannot open the semaphore, wait for 0.1 second
        */
        fprintf(stderr,"Retry Open semaphore %s\n", UF2_semfile);

        SleepSome(1);
    }

    if (! ReleaseSemaphore(su_hSem,
        1,
        (LPLONG)(NULL)))
    {
        fprintf(stderr, "ReleaseSemaphore failed, GetLastError: %d, quit\n",
        GetLastError());
        exit(-1);
    }
#endif

    exit(rc); /* child exiting after finishing up */
}

```

Appendix E: ACID Transaction Source Code

acid.h

```

/*****
*****/
/*   File: acid.h                               */
/*****
*****/

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

#ifdef SQLWINT
#include <windows.h>
#include <sys\timeb.h>
#include <sys\stat.h>
#include <stdlib.h>
#include <io.h>
#else
#include <unistd.h>
#include <sys/time.h>
#include <sys/timeb.h>
#endif

#include <string.h>
#include <math.h>

#define acidtime(tvsec,tvusec) tvsec*1000+tvusec/1000
#define TSLEN 20

#if 0 /* needed on NT, not on AIX */
typedef struct timeval {
    long   tv_sec;    /* seconds */
    long   tv_usec;   /* and microseconds */
};
#endif

struct update_struct {
    int    qnum;
};

struct acidQ_struct {
    int    tag;
    long   o_key;
    double l_extendedprice;
};

struct acidT_struct {
    int    termination;
    int    tag;
    int    logging;
    long   o_key;
    long   l_key;
    long   delta;
    long   l_partkey;
    long   l_suppkey;
    double l_quantity;
    double l_tax;
    double l_discount;
    double l_extendedprice;
    double o_totalprice;
};

/*
** in acid.sqc
*/

int updateQ (struct update_struct *us);
```

```

char del(void);

#ifdef SQLWINT
void sleep (int sec);
#endif
```

acid.sqc

```

/*****
*****/
/*   File: acid.sqc                               */
/*****
*****/

/* changes:
*
* 961109 jel  add EXEC SQL CLOSE for each cursor in acidT
*           to avoid bug in db2pe v1r2
* 980225 gav  port to NT
* 981103 kal  added ast_acidQ for isolation test 7
* 981103 kal  changed ast query to be the same as that used in
*           consistency tests. Fixed so the long lEprice is
*           cast to a double. Changed so uses 3 decimal points of
*           precision.
*/

#include "acid.h"

#if (defined(SQLPTX) || defined(SQLWINT) || defined(SQLSUN) ||
defined(Linux))
double nearest(double);
#endif /* SQLPTX */

#define DEADLOCK -911

/*
#define TRUNC2(d) ((floor((d)*100.0))/100.0)
*/
/*
#define TRUNC2(d) ((floor(nearest((d)*100.0)))*0.01)
*/
/*
#define TRUNC2(d) ((floor(nearest((d)*1000.0)/10.0)/100.0))
*/
#define TRUNC2(d) ((floor(nearest((d)*100000.0)/1000.0)/100.0))

void sqlerror(char *, struct sqlca *);

EXEC SQL INCLUDE SQLCA;
EXEC SQL BEGIN DECLARE SECTION;
char dbname[8]; /* = "tpcd"; */
EXEC SQL END DECLARE SECTION;

#ifdef SQLWINT

/*
** redefine gettimeofday so I don't have to
** change too much aix-specific code
*/
/*#typedef struct timeval { unsigned tv_sec; unsigned tv_usec; };
typedef struct timezone { int dummy; };
struct timeb timer;

void gettimeofday( struct timeval *tv, struct timezone *tz)
{
    ftime(&timer);
    tv->tv_sec = timer.time;
    tv->tv_usec = timer.millitm * 1000;
    tz->dummy = 0;
}
```

```

#endif

/*-----*/
/*      acidQ                               */
/*-----*/
int acidQ (struct acidQ_struct *acid)
{
    time_t timeT;
    FILE *out;
    char out_fn[50];
    struct timeval tv;
    struct timezone tz;
    int mypid;
    int rc = 0;

    EXEC SQL BEGIN DECLARE SECTION;
    sqlint32  okey;
    sqlint32  lEprice;
    double  eprice;
    EXEC SQL END DECLARE SECTION;

    okey = acid->o_key;

    /* mypid = getpid(); */
    mypid = acid->tag;

    sprintf(out_fn, "%s%cacidQ.out.%d",getenv("TPCD_TMP_DIR"),del(),mypid);
    out=fopen(out_fn,"a");
    if (out == NULL)
    {
        fprintf(stderr, "ERROR input file %s could not be appended to!!\n",out_fn);
    }

    gettimeofday(&tv, &tz);
    time(&timeT);
    fprintf(out, "\n----- START of acidQ tag: %d ----- \n\n",mypid);
    fprintf(out, "acidQ tag: %d, begin transaction time: (%0s %06uu) %s",
        mypid, tv.tv_sec, tv.tv_usec, ctime(&timeT));
    fprintf(out, "okey: %d\n", okey);

    gettimeofday(&tv, &tz);
    time(&timeT);
    fprintf(out, "acidQ tag: %d, before read of LINEITEM: (%0s %06uu) %s",
        mypid, tv.tv_sec, tv.tv_usec, ctime(&timeT));

    /*
    ** use the same sql code as used in the consistsql.pl to
    ** run the consistency acid queries. Note we assign an long int
    ** to lEprice (we make it 10s of pennies by * 1000). Then divide
    ** by 1000.0 and cast it to a double (eprice) for printing
    */

    EXEC SQL
    SELECT
        INTEGER(DECIMAL(SUM(DECIMAL(INTEGER(INTEGER(DECIMAL
            (INTEGER(100*DECIMAL(L_EXTENDEDPRI,20,3)), 20,3) *
            (1-L_DISCOUNT)) * (1+L_TAX)),20,3)/100.0),20,3) * 1000)
        into :lEprice
    FROM
        TPCD.LINEITEM
    WHERE
        L_ORDERKEY = :okey;

    if (sqlca.sqlcode != 0) {
        rc = sqlca.sqlcode;
        fprintf(out, "acidQ **ERROR** sqlcode = %d\n", sqlca.sqlcode);
        sqlerror("acidQ: select sum(l_extendedprice)", &sqlca);
        goto Qerror;
    }
    eprice = (double)lEprice / 1000.0; /* translate to double for printout*/

    gettimeofday(&tv, &tz);
    time(&timeT);

    fprintf(out, "ACID tag: %d, after read of LINEITEM: (%0s %06uu) %s",
        mypid, tv.tv_sec, tv.tv_usec, ctime(&timeT));
    fprintf(out, "okey: %d \t sum(l_extendedprice): %0.3f\n",
        okey, eprice);

    EXEC SQL COMMIT;
    if (sqlca.sqlcode != 0) {
        rc = sqlca.sqlcode;
        fprintf(out, "acidQ **ERROR** sqlcode = %d\n", sqlca.sqlcode);
        sqlerror("acidQ: COMMIT", &sqlca);
        goto Qerror;
    }
    acid->l_extendedprice = eprice;

    rc = 0;
    goto Qexit;

Qerror:
    EXEC SQL rollback work;
    if (sqlca.sqlcode != 0) sqlerror("acidQ: ROLLBACK FAILED", &sqlca);

Qexit:
    fprintf(out, "\n----- END of acidQ tag: %d ----- \n\n",mypid);
    fflush(out);fclose(out);
    return(rc);
}

/*-----*/
/*      ast_acidQ                               */
/*-----*/
int ast_acidQ (struct acidQ_struct *acid)
{
    time_t timeT;
    FILE *out;
    char out_fn[50];
    struct timeval tv;
    struct timezone tz;
    int mypid;
    int rc = 0;

    EXEC SQL BEGIN DECLARE SECTION;
    double  ast_lEprice;
    double  ast_eprice;
    EXEC SQL END DECLARE SECTION;

    /* mypid = getpid(); */
    mypid = acid->tag;

    sprintf(out_fn,
"%s%cast_acidQ.out.%d",getenv("TPCD_TMP_DIR"),del(),mypid);
    out=fopen(out_fn,"a");
    gettimeofday(&tv, &tz);
    time(&timeT);
    fprintf(out, "\n----- START of ast_acidQ tag: %d ----- \n\n",mypid);
    fprintf(out, "ast_acidQ tag: %d, begin transaction time: (%0s %06uu) %s",
        mypid, tv.tv_sec, tv.tv_usec, ctime(&timeT));

    gettimeofday(&tv, &tz);
    time(&timeT);
    fprintf(out, "ast_acidQ tag: %d, before read of LINEITEM: (%0s %06uu) %s",
        mypid, tv.tv_sec, tv.tv_usec, ctime(&timeT));

    /*
    ** use the same query acidQ except don't select for specific okey.
    ** this ensures that the ast will be used instead of the base table
    ** Have to use ast_lEprice as double since this sum is so big
    */

    EXEC SQL
    SELECT
        SUM ( L_EXTENDEDPRI*(1-L_DISCOUNT)*(1 + L_TAX))
        into :ast_lEprice
    FROM

```

```

TPCD.LINEITEM;

if (sqlca.sqlcode != 0) {
    rc = sqlca.sqlcode;
    fprintf(out, "ast_acidQ **ERROR** sqlcode = %d\n", sqlca.sqlcode);
    sqlerror("ast_acidQ: select sum(l_extendedprice)", &sqlca);
    goto Qerror;
}
ast_eprice = ast_IEprice; /* use ast_eprice for printout to be consistent*/

gettimeofday(&tv, &tz);
time(&timeT);
fprintf(out, "AST_ACID tag: %d, after read of LINEITEM: (%06u) %s",
        mypid, tv.tv_sec, tv.tv_usec, ctime(&timeT));
fprintf(out, "sum(l_extendedprice): %0.3f\n",
        ast_eprice);

EXEC SQL COMMIT;
if (sqlca.sqlcode != 0) {
    rc = sqlca.sqlcode;
    fprintf(out, "ast_acidQ **ERROR** sqlcode = %d\n", sqlca.sqlcode);
    sqlerror("ast_acidQ: COMMIT", &sqlca);
    goto Qerror;
}
acid->l_extendedprice = ast_eprice;

rc = 0;
goto Qexit;

Qerror:
EXEC SQL rollback work;
if (sqlca.sqlcode != 0) sqlerror("ast_acidQ: ROLLBACK FAILED", &sqlca);

Qexit:
fprintf(out, "\n----- END of ast_acidQ tag: %d ----- \n", mypid);
fflush(out); fclose(out);
return(rc);
}
/*-----*/
/*      acidT      */
/*-----*/
int acidT (struct acidT_struct *acid)
{

    time_t timeT;
    FILE *out;
    char out_fn[50];
    struct timeval tv;
    struct timezone tz;
    int mypid;
    int rc = 0;

    EXEC SQL BEGIN DECLARE SECTION;
    sqlint32  o_key, l_key, delta;
    sqlint32  l_partkey, l_suppkey;
    double    l_quantity, l_tax, l_discount, l_extendedprice;
    double    o_totalprice;
    double    new_quantity, rprice, cost, new_extprice, new_ototal, ototal;
    EXEC SQL END DECLARE SECTION;

    EXEC SQL DECLARE l_cursor CURSOR FOR
        SELECT l_partkey, l_suppkey, l_quantity,
               l_tax, l_discount,
               l_extendedprice
        FROM tpcd.lineitem
        WHERE l_orderkey = :o_key
        AND l_linenum = :l_key
        FOR UPDATE OF l_extendedprice, l_quantity;

    EXEC SQL DECLARE o_cursor CURSOR FOR
        SELECT o_totalprice
        FROM tpcd.orders
        WHERE o_orderkey = :o_key
        FOR UPDATE OF o_totalprice;

```

```

if (acid->termination < 0 || acid->termination > 3) acid->termination = 0;
o_key = acid->o_key;
l_key = acid->l_key;
delta = acid->delta;

if (acid->logging) {
    /* mypid = getpid(); */
    mypid = acid->tag;
    sprintf(out_fn,
"%s%06uacidT.out.%d", getenv("TPCD_TMP_DIR"), del(), mypid);
    out = fopen(out_fn, "a");
    gettimeofday(&tv, &tz);
    time(&timeT);
    fprintf(out, "\n----- START of acidT tag: %d ----- \n", mypid);
    fprintf(out, "acidT tag: %d, begin transaction time: (%06u) %s",
            mypid, tv.tv_sec, tv.tv_usec, ctime(&timeT));
    fprintf(out, "o_key: %d\tl_key: %d\tdelta: %d\n", o_key, l_key, delta);
}
#ifdef DEBUG
    printf("o_key: %d\tl_key: %d\tdelta: %d\n", o_key, l_key, delta);
#endif

retry_tran:

if (acid->logging) {
    gettimeofday(&tv, &tz);
    time(&timeT);
    fprintf(out, "acidT tag: %d, before read of LINEITEM: (%06u) %s",
            mypid, tv.tv_sec, tv.tv_usec, ctime(&timeT));
}

EXEC SQL OPEN l_cursor;
if (sqlca.sqlcode != 0) {
    if (sqlca.sqlcode == DEADLOCK) goto retry_tran;
    rc = sqlca.sqlcode;
    if (acid->logging) {
        fprintf(out, "acidT **ERROR** sqlcode = %d\n", sqlca.sqlcode);
    } else {
        fprintf(stderr, "acidT **ERROR** sqlcode = %d\n", sqlca.sqlcode);
    } /* endif */
    sqlerror("acidT: OPEN l_cursor", &sqlca);
    goto Terror;
}

EXEC SQL FETCH l_cursor INTO
    :l_partkey, :l_suppkey, :l_quantity, :l_tax,
    :l_discount, :l_extendedprice;
if (sqlca.sqlcode != 0) {
    if (sqlca.sqlcode == DEADLOCK) goto retry_tran;
    rc = sqlca.sqlcode;
    if (acid->logging) {
        fprintf(out, "acidT **ERROR** sqlcode = %d\n", sqlca.sqlcode);
    } else {
        fprintf(stderr, "acidT **ERROR** sqlcode = %d\n", sqlca.sqlcode);
    } /* endif */
    sqlerror("acidT: FETCH l_cursor", &sqlca);
    goto Terror;
}

#ifdef DEBUG
    printf("l_quantity = %0.3f\n", l_quantity);
    printf("l_tax = %0.3f\n", l_tax);
    printf("l_discount = %0.3f\n", l_discount);
    printf("l_extendedprice = %0.3f\n", l_extendedprice);
#endif

if (acid->logging) {
    gettimeofday(&tv, &tz);
    time(&timeT);
    fprintf(out, "acidT tag: %d, after read of LINEITEM: (%06u) %s",
            mypid, tv.tv_sec, tv.tv_usec, ctime(&timeT));
    fprintf(out, "l_partkey: %d l_suppkey: %d l_quantity: %0.3f\n\tl_tax: %0.3f\n\tl_discount: %0.3f l_extendedprice: %0.3f\n",

```



```

        l_partkey, l_supkey, l_quantity, l_tax, l_discount, l_extendedprice);
    }

    rprice = TRUNC2( l_extendedprice/l_quantity );
    cost = TRUNC2( rprice * delta );
    new_extprice = l_extendedprice + cost;
    new_quantity = l_quantity + delta;

#ifdef DEBUG
    printf("rprice = %0.3f\n", rprice );
    printf("cost = %0.3f\n", cost );
    printf("new_extprice = %0.3f\n", new_extprice );
    printf("new_quantity = %0.3f\n", new_quantity );
#endif

    EXEC SQL UPDATE tpcd.lineitem
        SET l_extendedprice = :new_extprice,
            l_quantity = :new_quantity
        WHERE CURRENT OF l_cursor;

    if (sqlca.sqlcode != 0) {
        if (sqlca.sqlcode == DEADLOCK) goto retry_tran;
        rc = sqlca.sqlcode;
        if (acid->logging) {
            fprintf(out, "acidT **ERROR** sqlcode = %d\n", sqlca.sqlcode);
        } else {
            fprintf(stderr, "acidT **ERROR** sqlcode = %d\n", sqlca.sqlcode);
        } /* endif */
        sqlerror("acidT: UPDATE l_cursor", &sqlca);
        goto Terror;
    }

    if (acid->logging) {
        gettimeofday(&tv, &tz);
        time(&timeT);
        fprintf(out, "acidT tag: %d, after update of LINEITEM: (%us %06uu) %s",
            mypid, tv.tv_sec, tv.tv_usec, ctime(&timeT));
        fprintf(out, "updated l_extendedprice: %0.3f\n", new_extprice );
        fprintf(out, "updated l_quantity: %0.3f\n", new_quantity );
    }

    /* if (acid->termination == 0) {
        EXEC SQL CLOSE l_cursor;
        EXEC SQL CLOSE o_cursor;
        EXEC SQL COMMIT;
        if (sqlca.sqlcode != 0) {
            if (sqlca.sqlcode == DEADLOCK) goto retry_tran;
            rc = sqlca.sqlcode;
            if (acid->logging) {
                fprintf(out, "acidT **ERROR** sqlcode = %d\n", sqlca.sqlcode);
            } else {
                fprintf(stderr, "acidT **ERROR** sqlcode = %d\n", sqlca.sqlcode);
            }
            sqlerror("acidT: COMMIT", &sqlca);
            goto Terror;
        }
    } */

    if (acid->logging) {
        gettimeofday(&tv, &tz);
        time(&timeT);
        fprintf(out, "acidT tag: %d, before read of ORDER: (%us %06uu) %s",
            mypid, tv.tv_sec, tv.tv_usec, ctime(&timeT));
    }

    EXEC SQL OPEN o_cursor;
    if (sqlca.sqlcode != 0) {
        if (sqlca.sqlcode == DEADLOCK) goto retry_tran;
        rc = sqlca.sqlcode;
        if (acid->logging) {
            fprintf(out, "acidT **ERROR** sqlcode = %d\n", sqlca.sqlcode);
        } else {
            fprintf(stderr, "acidT **ERROR** sqlcode = %d\n", sqlca.sqlcode);
        } /* endif */
    }

```

```

        sqlerror("acidT: OPEN o_cursor", &sqlca);
        goto Terror;
    }

    EXEC SQL FETCH o_cursor INTO :o_totalprice;
    if (sqlca.sqlcode != 0) {
        if (sqlca.sqlcode == DEADLOCK) goto retry_tran;
        rc = sqlca.sqlcode;
        if (acid->logging) {
            fprintf(out, "acidT **ERROR** sqlcode = %d\n", sqlca.sqlcode);
        } else {
            fprintf(stderr, "acidT **ERROR** sqlcode = %d\n", sqlca.sqlcode);
        }
        sqlerror("acidT: FETCH o_cursor", &sqlca);
        goto Terror;
    }

#ifdef DEBUG
    printf("o_totalprice = %0.3f\n", o_totalprice);
#endif

    if (acid->logging) {
        gettimeofday(&tv, &tz);
        time(&timeT);
        fprintf(out, "acidT tag: %d, after read of ORDER: (%us %06uu) %s",
            mypid, tv.tv_sec, tv.tv_usec, ctime(&timeT));
        fprintf(out, "o_totalprice: %0.3f\n", o_totalprice);
    }

#ifdef DEBUG
    {
        double zeroone = l_extendedprice * (1.0 - l_discount);
        double zeroonetimes = (l_extendedprice * (1.0 - l_discount)) * 100.0;
        double firstone = TRUNC2(l_extendedprice * (1.0 - l_discount));
        double notone = TRUNC2( l_extendedprice * (1.0 - l_discount) ) * (1.0 + l_tax);
        double secondone = TRUNC2( TRUNC2( l_extendedprice * (1.0 - l_discount) )
            * (1.0 + l_tax) );
        printf("firstone= %f\n", firstone);
        printf("zeroone= %f\n", zeroone);
        printf("zeroonetimes= %f\n", zeroonetimes);
        printf("notone= %f\n", notone);
        printf("secondone= %f\n", secondone);
    }
#endif

    ototal = o_totalprice -
        TRUNC2( TRUNC2( l_extendedprice * (1 - l_discount) ) * (1 + l_tax) );
    new_ototal = TRUNC2( new_extprice * (1.0 - l_discount) );
    new_ototal = TRUNC2( new_ototal * (1.0 + l_tax) );
    new_ototal = ototal + new_ototal;

#ifdef DEBUG
    printf("o_totalprice= %f\n", o_totalprice);
    printf("ototal= %0.3f\n", ototal);
    printf("ototal= %f\n", ototal);
    printf("new_ototal= %0.3f\n", new_ototal);
#endif

    EXEC SQL UPDATE tpcd.orders
        SET o_totalprice = :new_ototal
        WHERE CURRENT OF o_cursor;
    if (sqlca.sqlcode != 0) {
        if (sqlca.sqlcode == DEADLOCK) goto retry_tran;
        rc = sqlca.sqlcode;
        if (acid->logging) {
            fprintf(out, "acidT **ERROR** sqlcode = %d\n", sqlca.sqlcode);
        } else {
            fprintf(stderr, "acidT **ERROR** sqlcode = %d\n", sqlca.sqlcode);
        } /* endif */
        sqlerror("acidT: UPDATE o_cursor", &sqlca);
        goto Terror;
    }

```

```

if (acid->logging) {
    gettimeofday(&tv, &tz);
    time(&timeT);
    fprintf(out, "acidT tag: %d, after update of ORDER: (%us %06uu) %s",
        mypid, tv.tv_sec, tv.tv_usec, ctime(&timeT));
    fprintf(out, "updated o_totalprice: %0.3f\n", new_ototal);
}

/*
** why is this code in here? we don't want to
** commit until the history table has been updated as well
if (acid->termination == 0) {
    EXEC SQL CLOSE L_CURSOR;
    EXEC SQL CLOSE O_CURSOR;
    EXEC SQL COMMIT;
    if (sqlca.sqlcode != 0) {
        if (sqlca.sqlcode == DEADLOCK) goto retry_tran;
        rc = sqlca.sqlcode;
        if (acid->logging) {
            fprintf(out, "acidT **ERROR** sqlcode = %d\n", sqlca.sqlcode);
        } else {
            fprintf(stderr, "acidT **ERROR** sqlcode = %d\n", sqlca.sqlcode);
        }
        sqlerror("acidT: COMMIT", &sqlca);
        goto TError;
    }
}

if (acid->logging) {
    gettimeofday(&tv, &tz);
    time(&timeT);
    fprintf(out, "acidT tag: %d, before insert into HISTORY: (%us %06uu) %s",
        mypid, tv.tv_sec, tv.tv_usec, ctime(&timeT));
}

EXEC SQL INSERT INTO tpcd.history values
(:l_partkey, :l_supkey, :o_key, :l_key, :delta, CURRENT TIMESTAMP);
if (sqlca.sqlcode != 0) {
    if (sqlca.sqlcode == DEADLOCK) goto retry_tran;
    rc = sqlca.sqlcode;
    if (acid->logging) {
        fprintf(out, "acidT **ERROR** sqlcode = %d\n", sqlca.sqlcode);
    } else {
        fprintf(stderr, "acidT **ERROR** sqlcode = %d\n", sqlca.sqlcode);
    } /* endif */
    sqlerror("acidT: INSERT INTO history", &sqlca);
    goto TError;
}

if (acid->logging) {
    gettimeofday(&tv, &tz);
    time(&timeT);
    fprintf(out, "acidT tag: %d, after insert into HISTORY: (%us %06uu) %s",
        mypid, tv.tv_sec, tv.tv_usec, ctime(&timeT));
}

/* sleep for 1 second for 80% of the transactions */
#ifdef SQLWINT
    if ( ((rand() % (100)) + 1) < 80 ) sleep(1);
#else
    if ( ((random() % (100)) + 1) < 80 ) sleep(1);
#endif

switch (acid->termination) {
case 1:
    {
        if (acid->logging)
        {
            gettimeofday(&tv, &tz);
            time(&timeT);
            fprintf(out, "acidT tag: %d, wait before COMMIT: (%us %06uu) %s",
                mypid, tv.tv_sec, tv.tv_usec, ctime(&timeT));

```

```

        }
    }
    sleep(60);
case 0:
    if (acid->logging) {
        gettimeofday(&tv, &tz);
        time(&timeT);
        fprintf(out, "acidT tag: %d, immediately before COMMIT: (%us %06uu) %s",
            mypid, tv.tv_sec, tv.tv_usec, ctime(&timeT));
    }
    EXEC SQL CLOSE L_CURSOR;
    EXEC SQL CLOSE O_CURSOR;
    EXEC SQL COMMIT;
    if (sqlca.sqlcode != 0) {
        if (sqlca.sqlcode == DEADLOCK) goto retry_tran;
        rc = sqlca.sqlcode;
        if (acid->logging) {
            fprintf(out, "acidT **ERROR** sqlcode = %d\n", sqlca.sqlcode);
        } else {
            fprintf(stderr, "acidT **ERROR** sqlcode = %d\n", sqlca.sqlcode);
        } /* endif */
        sqlerror("acidT: COMMIT", &sqlca);
        goto TError;
    }
    if (acid->logging) {
        gettimeofday(&tv, &tz);
        time(&timeT);
        fprintf(out, "acidT tag: %d, after COMMIT: (%us %06uu) %s",
            mypid, tv.tv_sec, tv.tv_usec, ctime(&timeT));
    }
    break;
case 3:
    if (acid->logging) {
        gettimeofday(&tv, &tz);
        time(&timeT);
        fprintf(out, "acidT tag: %d, wait before ROLLBACK: (%us %06uu) %s",
            mypid, tv.tv_sec, tv.tv_usec, ctime(&timeT));
    }
    sleep(60);
case 2:
    if (acid->logging) {
        gettimeofday(&tv, &tz);
        time(&timeT);
        fprintf(out, "acidT tag: %d, immediately before ROLLBACK: (%us %06uu) %s",
            mypid, tv.tv_sec, tv.tv_usec, ctime(&timeT));
    }
    EXEC SQL CLOSE L_CURSOR;
    EXEC SQL CLOSE O_CURSOR;
    EXEC SQL rollback work;
    if (sqlca.sqlcode != 0) {
        if (sqlca.sqlcode == DEADLOCK) goto retry_tran;
        rc = sqlca.sqlcode;
        if (acid->logging) {
            fprintf(out, "acidT **ERROR** sqlcode = %d\n", sqlca.sqlcode);
        } else {
            fprintf(stderr, "acidT **ERROR** sqlcode = %d\n", sqlca.sqlcode);
        } /* endif */
        sqlerror("acidT: ROLLBACK", &sqlca);
        goto TError;
    }
    if (acid->logging) {
        gettimeofday(&tv, &tz);
        time(&timeT);
        fprintf(out, "acidT tag: %d, after ROLLBACK: (%us %06uu) %s",
            mypid, tv.tv_sec, tv.tv_usec, ctime(&timeT));
    }
    break;
}

acid->l_partkey = l_partkey;
acid->l_supkey = l_supkey;
acid->l_quantity = l_quantity;

```

```

acid->l_tax = l_tax;
acid->l_discount = l_discount;
acid->l_extendedprice = l_extendedprice;
acid->o_totalprice = o_totalprice;

rc = 0;
goto Texit;

Terror:
EXEC SQL CLOSE L_CURSOR;
EXEC SQL CLOSE O_CURSOR;
EXEC SQL rollback work;
if (sqlca.sqlcode != 0) sqlerror("acidT: ROLLBACK FAILED", &sqlca);

Texit:
if (acid->logging) {
    fprintf(out, "\n----- END of acidT tag: %d ----- \n\n", mypid);
    fflush(out); fclose(out);
}
return(rc);
}

/*-----*/
/*      updateQ      */
/*-----*/
int updateQ (struct update_struct *us)
{
    FILE *out;
    time_t timeT;
    struct timeval tv;
    struct timezone tz;
    int qnum;
    int rc = 0;
    int i;
    int secs2sleep;
    char buff[256];
    struct acidtype {int logging;} a, *acid;

EXEC SQL BEGIN DECLARE SECTION;
double  acctbal;
double  discount;
double  price;
sqlint32  availqty;
sqlint32  size;
EXEC SQL END DECLARE SECTION;

qnum = us->qnum;

acid = &a;
acid->logging= 1;

sprintf(buff, "%s%cupdate.out", getenv("TPCD_TMP_DIR"), del());
out=fopen(buff, "a");

gettimeofday(&tv, &tz);
time(&timeT);
fprintf(out, "\n----- START of update ----- \n\n");
fprintf(out, "update query number: %d, begin transaction time: (%us %06uu)
%s",
    qnum, tv.tv_sec, tv.tv_usec, ctime(&timeT));

sqlca.sqlcode = 0;
discount = 0.25;
price = 5000.50;
acctbal = 1000.00;
availqty = 10;
size = 5;

for (i=1; i <= 2; i++) {
    gettimeofday(&tv, &tz);
    time(&timeT);
    fprintf(out, "update query number: %d, pass %d, immediately before
UPDATE: (%us %06uu) %s",
        qnum, i, tv.tv_sec, tv.tv_usec, ctime(&timeT));

```

```

switch (qnum)
{
case 1:
{
EXEC SQL
UPDATE TPCD.LINEITEM set L_DISCOUNT = L_DISCOUNT +
:discount
WHERE L_ORDERKEY IN (326,512,928,995);
if (sqlca.sqlcode != 0) {
rc = sqlca.sqlcode;
if (acid->logging)
{
fprintf(out, "update query number: %d, pass %d, **ERROR** sqlcode =
%d\n",
        qnum, i, sqlca.sqlcode);
}
else
{
fprintf(stderr, "update query number: %d, pass %d, **ERROR** sqlcode
= %d\n",
        qnum, i, sqlca.sqlcode);
}
sqlerror("update query number 1", &sqlca);
goto Uerror;
}
discount = discount * (-1);
secs2sleep = 300;
break;
}
case 2:
{
EXEC SQL
UPDATE TPCD.SUPPLIER set S_ACCTBAL = S_ACCTBAL +
:acctbal
WHERE S_NAME in
('Supplier#000000647','Supplier#000000070','Supplier#000000802');
if (sqlca.sqlcode != 0) {
rc = sqlca.sqlcode;
if (acid->logging)
{
fprintf(out, "update query number: %d, pass %d, **ERROR** sqlcode =
%d\n",
        qnum, i, sqlca.sqlcode);
}
else
{
fprintf(stderr, "update query number: %d, pass %d, **ERROR** sqlcode
= %d\n",
        qnum, i, sqlca.sqlcode);
}
sqlerror("update query number 2", &sqlca);
goto Uerror;
}
acctbal = acctbal * (-1);
secs2sleep = 90;
break;
}
case 3:
{
EXEC SQL
UPDATE TPCD.LINEITEM set L_DISCOUNT = L_DISCOUNT +
:discount
WHERE L_ORDERKEY IN (260930, 402497, 457859, 509889, 58117,
538311, 588421, 416167, 97830, 90276);
if (sqlca.sqlcode != 0) {
rc = sqlca.sqlcode;
if (acid->logging)
{
fprintf(out, "update query number: %d, pass %d, **ERROR** sqlcode =
%d\n",
        qnum, i, sqlca.sqlcode);
}
else

```

```

        {
            fprintf(stderr,"update query number: %d, pass %d, **ERROR** sqlcode
= %d\n",
                qnum, i, sqlca.sqlcode);
        }
        sqlerror("update query number 3", &sqlca);
        goto Uerror;
    }
    discount = discount * (-1);
    secs2sleep = 300;
    break;
}
case 4:
{
    if ( i == 1 ) {
        EXEC SQL
            UPDATE TPCD.ORDERS set O_ORDERDATE = O_ORDERDATE -
6 MONTHS
            WHERE O_ORDERKEY = 67461;
        /* WHERE O_ORDERKEY IN
(22400,28515,34338,46596,67461,92644,98307);*/
    } else {
        EXEC SQL
            UPDATE TPCD.ORDERS set O_ORDERDATE = O_ORDERDATE +
6 MONTHS
            WHERE O_ORDERKEY = 67461;
    }
    if (sqlca.sqlcode != 0) {
        rc = sqlca.sqlcode;
        if (acid->logging)
        {
            fprintf(out,"update query number: %d, pass %d, **ERROR** sqlcode =
%d\n",
                qnum, i, sqlca.sqlcode);
        }
        else
        {
            fprintf(stderr,"update query number: %d, pass %d, **ERROR** sqlcode
= %d\n",
                qnum, i, sqlca.sqlcode);
        }
        sqlerror("update query number 4", &sqlca);
        goto Uerror;
    }
    secs2sleep = 300;
    break;
}
case 5:
{
    EXEC SQL
        UPDATE TPCD.LINEITEM set L_DISCOUNT = L_DISCOUNT +
:discount
        WHERE L_ORDERKEY IN (70976,566279,152897,84226,232483);
    if (sqlca.sqlcode != 0) {
        rc = sqlca.sqlcode;
        if (acid->logging)
        {
            fprintf(out,"update query number: %d, pass %d, **ERROR** sqlcode =
%d\n",
                qnum, i, sqlca.sqlcode);
        }
        else
        {
            fprintf(stderr,"update query number: %d, pass %d, **ERROR** sqlcode
= %d\n",
                qnum, i, sqlca.sqlcode);
        }
        sqlerror("update query number 5", &sqlca);
        goto Uerror;
    }
    discount = discount * (-1);
    secs2sleep = 300;
    break;
}

```

```

case 6:
{
    EXEC SQL
        UPDATE TPCD.LINEITEM set L_DISCOUNT = L_DISCOUNT +
:discount
        WHERE L_ORDERKEY in (33,131,161,195,229,230,231,323,353,356);
    if (sqlca.sqlcode != 0) {
        rc = sqlca.sqlcode;
        if (acid->logging)
        {
            fprintf(out,"update query number: %d, pass %d, **ERROR** sqlcode =
%d\n",
                qnum, i, sqlca.sqlcode);
        }
        else
        {
            fprintf(stderr,"update query number: %d, pass %d, **ERROR** sqlcode
= %d\n",
                qnum, i, sqlca.sqlcode);
        }
        sqlerror("update query number 6", &sqlca);
        goto Uerror;
    }
    discount = discount * (-1);
    secs2sleep = 300;
    break;
}
case 7:
{
    EXEC SQL
        UPDATE TPCD.LINEITEM set L_DISCOUNT = L_DISCOUNT +
:discount
        WHERE L_ORDERKEY IN
(562917,410659,16550,398401,157634,429920,45411);
    if (sqlca.sqlcode != 0) {
        rc = sqlca.sqlcode;
        if (acid->logging)
        {
            fprintf(out,"update query number: %d, pass %d, **ERROR** sqlcode =
%d\n",
                qnum, i, sqlca.sqlcode);
        }
        else
        {
            fprintf(stderr,"update query number: %d, pass %d, **ERROR** sqlcode
= %d\n",
                qnum, i, sqlca.sqlcode);
        }
        sqlerror("update query number 7", &sqlca);
        goto Uerror;
    }
    discount = discount * (-1);
    secs2sleep = 300;
    break;
}
case 8:
{
    EXEC SQL
        UPDATE TPCD.LINEITEM set L_DISCOUNT = L_DISCOUNT +
:discount
        WHERE L_ORDERKEY IN
(129569,343591,270242,254983,98500,28963);
    if (sqlca.sqlcode != 0) {
        rc = sqlca.sqlcode;
        if (acid->logging)
        {
            fprintf(out,"update query number: %d, pass %d, **ERROR** sqlcode =
%d\n",
                qnum, i, sqlca.sqlcode);
        }
        else
        {
            fprintf(stderr,"update query number: %d, pass %d, **ERROR** sqlcode
= %d\n",
                qnum, i, sqlca.sqlcode);
        }
    }
}

```

```

        qnum, i, sqlca.sqlcode);
    }
    sqlerror("update query number 8", &sqlca);
    goto Uerror;
}
discount = discount * (-1);
secs2sleep = 300;
break;
}
case 9:
{
    EXEC SQL
        UPDATE TPCD.LINEITEM set L_DISCOUNT = L_DISCOUNT +
:discount
        WHERE L_ORDERKEY IN
(113509,232997,246691,379233,448162,32134);
    if (sqlca.sqlcode != 0) {
        rc = sqlca.sqlcode;
        if (acid->logging)
        {
            fprintf(out,"update query number: %d, pass %d, **ERROR** sqlcode =
%d\n",
                qnum, i, sqlca.sqlcode);
        }
        else
        {
            fprintf(stderr,"update query number: %d, pass %d, **ERROR** sqlcode
= %d\n",
                qnum, i, sqlca.sqlcode);
        }
        sqlerror("update query number 9", &sqlca);
        goto Uerror;
    }
    discount = discount * (-1);
    secs2sleep = 300;
    break;
}
case 10:
{
    EXEC SQL
        UPDATE TPCD.LINEITEM set L_DISCOUNT = L_DISCOUNT +
:discount
        WHERE L_ORDERKEY IN
(516487,245411,265799,253025,6914,562020);
    if (sqlca.sqlcode != 0) {
        rc = sqlca.sqlcode;
        if (acid->logging)
        {
            fprintf(out,"update query number: %d, pass %d, **ERROR** sqlcode =
%d\n",
                qnum, i, sqlca.sqlcode);
        }
        else
        {
            fprintf(stderr,"update query number: %d, pass %d, **ERROR** sqlcode
= %d\n",
                qnum, i, sqlca.sqlcode);
        }
        sqlerror("update query number 10", &sqlca);
        goto Uerror;
    }
    discount = discount * (-1);
    secs2sleep = 300;
    break;
}
case 11:
{
    EXEC SQL
        UPDATE TPCD.PARTSUPP set PS_AVAILQTY = PS_AVAILQTY +
:availqty
        WHERE PS_PARTKEY IN
(12098,5134,13334,17052,3452,12552,1084,5797);
    if (sqlca.sqlcode != 0) {
        rc = sqlca.sqlcode;

```

```

        if (acid->logging)
        {
            fprintf(out,"update query number: %d, pass %d, **ERROR** sqlcode =
%d\n",
                qnum, i, sqlca.sqlcode);
        }
        else
        {
            fprintf(stderr,"update query number: %d, pass %d, **ERROR** sqlcode
= %d\n",
                qnum, i, sqlca.sqlcode);
        }
        sqlerror("update query number 11", &sqlca);
        goto Uerror;
    }
    availqty = availqty * (-1);
    secs2sleep = 180;
    break;
}
case 12:
{
    if (i == 1) {
        EXEC SQL
            UPDATE TPCD.LINEITEM set L_RECEIPTDATE =
L_RECEIPTDATE - 3 YEARS
            WHERE L_ORDERKEY IN (33,70,195,355,677,837,960,962,1028);
    } else {
        EXEC SQL
            UPDATE TPCD.LINEITEM set L_RECEIPTDATE =
L_RECEIPTDATE + 3 YEARS
            WHERE L_ORDERKEY IN (33,70,195,355,677,837,960,962,1028);
    }
    if (sqlca.sqlcode != 0) {
        rc = sqlca.sqlcode;
        if (acid->logging)
        {
            fprintf(out,"update query number: %d, pass %d, **ERROR** sqlcode =
%d\n",
                qnum, i, sqlca.sqlcode);
        }
        else
        {
            fprintf(stderr,"update query number: %d, pass %d, **ERROR** sqlcode
= %d\n",
                qnum, i, sqlca.sqlcode);
        }
        sqlerror("update query number 12", &sqlca);
        goto Uerror;
    }
    secs2sleep = 300;
    break;
}
case 13:
{
    EXEC SQL
        UPDATE TPCD.LINEITEM set L_DISCOUNT = L_DISCOUNT +
:discount
        WHERE L_ORDERKEY IN (263,9476,32355,34854,53445,56901);
    if (sqlca.sqlcode != 0) {
        rc = sqlca.sqlcode;
        if (acid->logging)
        {
            fprintf(out,"update query number: %d, pass %d, **ERROR** sqlcode =
%d\n",
                qnum, i, sqlca.sqlcode);
        }
        else
        {
            fprintf(stderr,"update query number: %d, pass %d, **ERROR** sqlcode
= %d\n",
                qnum, i, sqlca.sqlcode);
        }
        sqlerror("update query number 13", &sqlca);
        goto Uerror;
    }

```

```

    }
    discount = discount * (-1);
    secs2sleep = 90;
    break;
}
case 14:
{
    EXEC SQL
        UPDATE TPCD.LINEITEM set L_DISCOUNT = L_DISCOUNT +
:discount
        WHERE L_ORDERKEY IN (32,225,326,448,449,483,512);
    if (sqlca.sqlcode != 0) {
        rc = sqlca.sqlcode;
        if (acid->logging)
        {
            fprintf(out,"update query number: %d, pass %d, **ERROR** sqlcode =
%d\n",
                qnum, i, sqlca.sqlcode);
        }
        else
        {
            fprintf(stderr,"update query number: %d, pass %d, **ERROR** sqlcode
= %d\n",
                qnum, i, sqlca.sqlcode);
        }
        sqlerror("update query number 14", &sqlca);
        goto Uerror;
    }
    discount = discount * (-1);
    secs2sleep = 180;
    break;
}
case 15:
{
    EXEC SQL
        UPDATE TPCD.LINEITEM set L_DISCOUNT = L_DISCOUNT +
:discount
        WHERE L_ORDERKEY IN (1,4,7,35,135,131300);
    if (sqlca.sqlcode != 0) {
        rc = sqlca.sqlcode;
        if (acid->logging)
        {
            fprintf(out,"update query number: %d, pass %d, **ERROR** sqlcode =
%d\n",
                qnum, i, sqlca.sqlcode);
        }
        else
        {
            fprintf(stderr,"update query number: %d, pass %d, **ERROR** sqlcode
= %d\n",
                qnum, i, sqlca.sqlcode);
        }
        sqlerror("update query number 15", &sqlca);
        goto Uerror;
    }
    discount = discount * (-1);
    secs2sleep = 180;
    break;
}
case 16:
{
    EXEC SQL
        UPDATE TPCD.PART set P_SIZE = P_SIZE + :size
        WHERE P_PARTKEY IN (4,7,15,1313);
    if (sqlca.sqlcode != 0) {
        rc = sqlca.sqlcode;
        if (acid->logging)
        {
            fprintf(out,"update query number: %d, pass %d, **ERROR** sqlcode =
%d\n",
                qnum, i, sqlca.sqlcode);
        }
        else
        {

```

```

            fprintf(stderr,"update query number: %d, pass %d, **ERROR** sqlcode
= %d\n",
                qnum, i, sqlca.sqlcode);
        }
        sqlerror("update query number 16", &sqlca);
        goto Uerror;
    }
    size = size * (-1);
    secs2sleep = 180;
    break;
}
case 17:
{
    EXEC SQL
        UPDATE TPCD.LINEITEM set L_EXTENDEDPRICE =
L_EXTENDEDPRICE + :price
        WHERE L_ORDERKEY IN (4065,110372,165061,265702,87138);
    if (sqlca.sqlcode != 0) {
        rc = sqlca.sqlcode;
        if (acid->logging)
        {
            fprintf(out,"update query number: %d, pass %d, **ERROR** sqlcode =
%d\n",
                qnum, i, sqlca.sqlcode);
        }
        else
        {
            fprintf(stderr,"update query number: %d, pass %d, **ERROR** sqlcode
= %d\n",
                qnum, i, sqlca.sqlcode);
        }
        sqlerror("update query number 17", &sqlca);
        goto Uerror;
    }
    price = price * (-1);
    secs2sleep = 90;
    break;
}
default:
{
    fprintf(out,"ERROR: Invalid query number specified %d\n", qnum);
    rc = 1;
    goto Uexit;
}
}

gettimeofday(&tv, &tz);
time(&timeT);

if (acid->logging)
    fprintf(out,"update query number: %d, pass %d, after UPDATE: (%us
%06uu) %s",
        qnum, i, tv.tv_sec, tv.tv_usec, ctime(&timeT));
else
    fprintf(stderr,"update query number: %d, pass %d, after UPDATE: (%us
%06uu) %s",
        qnum, i, tv.tv_sec, tv.tv_usec, ctime(&timeT));

if ( i == 2 ) {
    gettimeofday(&tv, &tz);
    time(&timeT);
    fprintf(out,"update query number: %d, pass %d, sleeping for %d seconds:
(%us %06uu) %s",
        qnum, i, secs2sleep, tv.tv_sec, tv.tv_usec, ctime(&timeT));
    fflush(out);
    system("touch /tmp/tpcd/update.sync.sleep");
    sleep(secs2sleep);
}

gettimeofday(&tv, &tz);
time(&timeT);
fprintf(out,"update query number: %d, pass %d, immediately before
COMMIT: (%us %06uu) %s",
    qnum, i, tv.tv_sec, tv.tv_usec, ctime(&timeT));

```

```

EXEC SQL COMMIT;
if (sqlca.sqlcode != 0) {
    rc = sqlca.sqlcode;
    fprintf(out,"update pass %d, **ERROR** sqlcode = %d\n", i,
sqlca.sqlcode);
    sqlerror("update: COMMIT", &sqlca);
    goto Uerror;
}
gettimeofday(&tv, &tz);
time(&timeT);
if (acid->logging)
    fprintf(out,"update query number: %d, pass %d, after COMMIT: (%us
%06uu) %s",
        qnum, i, tv.tv_sec, tv.tv_usec, ctime(&timeT));
else
    fprintf(stderr,"update query number: %d, pass %d, after COMMIT: (%us
%06uu) %s",
        qnum, i, tv.tv_sec, tv.tv_usec, ctime(&timeT));
}

rc = 0;
goto Uexit;

Uerror:
EXEC SQL rollback work;
if (sqlca.sqlcode != 0) sqlerror("update: ROLLBACK FAILED", &sqlca);
system("touch /tmp/tpcd/update.sync.sleep");

Uexit:
fprintf(out,"n----- END of update -----n\n");
fflush(out);fclose(out);
return(rc);
}

/*-----*/
/*    connect_to_TM                                */
/*-----*/

void connect_to_TM( void )
{
    char *dbname_ptr;
    if ((dbname_ptr = getenv("TPCD_QUAL_DBNAME")) != NULL) {
        fprintf(stderr,"***** %s *****n",dbname_ptr);
        strcpy (dbname, dbname_ptr);
    }

    EXEC SQL CONNECT TO :dbname IN SHARE MODE;
    if (sqlca.sqlcode < 0) {
        fprintf(stderr, "CONNECT TO %s failed SQLCODE = %d\n", dbname,
sqlca.sqlcode);
        exit(-1);
    }
    return;
}

/*-----*/
/*    disconnect_from_TM                            */
/*-----*/

void disconnect_from_TM ( void )
{
    EXEC SQL CONNECT RESET;
    if (sqlca.sqlcode < 0) {
        fprintf(stderr, "DISCONNECT failed SQLCODE = %d\n", sqlca.sqlcode);
        exit(-1);
    }
    return;
}

/*-----*/
/*    sqlerror                                      */
/*-----*/

void sqlerror(char *msg, struct sqlca *psqlca)
{
    FILE *err_fp;

```

```

char err_fn[256];

int j,k;

sprintf(err_fn, "%s%cacid.sqlerrors",getenv("TPCD_TMP_DIR"),del());
err_fp=fopen(err_fn,"a");
fprintf(err_fp,"acid: sqlcode: %4d %s\n", psqlca->sqlcode, msg);
fprintf(stderr,"acid: sqlcode: %4d %s\n", psqlca->sqlcode, msg);
fflush(stderr);
if (psqlca->sqlerrmc[0] != ' ' || psqlca->sqlerrmc[1] != ' ') {
    fprintf(err_fp,"acid: slerrmc: ");
    for(j = 0; j < 5; j++)
    {
        for(k = 0; k < 14; k++) fprintf(err_fp,"%x ", psqlca->sqlerrmc[j*10+k]);
        fprintf(err_fp," ");
        for(k = 0; k < 14; k++) fprintf(err_fp,"%c", psqlca->sqlerrmc[j*10+k]);
        fprintf(err_fp,"n");
        if (j < 4) fprintf(err_fp," ");
    }
}

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

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

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

fprintf(err_fp,"n");
fflush(err_fp);fclose(err_fp);
}

#ifdef SQLWINT
void sleep(int sec)
{
    Sleep(sec * 1000);
}
#endif

char del(void)
{
#ifdef SQLWINT
    return '\\';
#else
    return '/';
#endif
}

#ifdef defined(SQLPTX) || defined(SQLWINT) || defined(SQLSUN) ||
defined(Linux)
/* added for PTX as this one is not there in libm */
double nearest(double x)
{
    double y, z;

    y = x;
    if (x < 0)
        y = -x;
    z = y - (int)y;
    if (z == 0.5) {
        if (((int)floor(y) % 2) {
            return((x < 0) ? -ceil(y) : ceil(y));
        } else {

```

```

        return((x < 0) ? -floor(y) : floor(y));
    }
} else if (z < 0.5)
    return((x < 0) ? -floor(y) : floor(y));
else
    return((x < 0) ? -ceil(y) : ceil(y));
}
#endif /* SQLPTX */

```

makefile

```

DBNAME =          $(TPCD_QUAL_DBNAME)

INCLUDE =         $(HOME)/sqllib/include

#CFLAGS =         -I$(INCLUDE) -g -Dpascal= -DLINT_ARGS \
#                -Dfar= -D_loadds= -DSQLA_NOLINES -qflag=i:i -qlanglvl=ansi

#LFLAGS =         -lm -lcurses -ls -ll -ly -liconv -lbsd
CFLAGS =         -I$(INCLUDE) -g -Dpascal= -DLINT_ARGS \
                -DSQLA_NOLINES -qflag=i:i -qlanglvl=ansi
# .. sun         -DSQLA_NOLINES

LFLAGS =         -lm -lbsd
# sun .... LFLAGS = -lm

LIB =            -L$(HOME)/sqllib/lib -ldb2

CC =             cc

HDR =            acid.h
C =              mainacid.c
SQC =            acid.sqc
SRC =            $(HDR)          $(C)          $(SQC)
OBJ =            acid.o
EXEC =           mainacid

TARGET =         $(EXEC) tsec

.SUFFIXES: .o .c .sqc .bnd

.c.o:
    $(CC) -c $< $(CFLAGS)

all:             $(TARGET)

mainacid: $(SRC) $(OBJ) mainacid.o
    $(CC) -o $@ $(CFLAGS) $(OBJ) mainacid.o $(LIB) $(LFLAGS)

acid.c: acid.sqc $(HDR)
    - db2 connect to $(DBNAME); \
    db2 prep acid.sqc BINDFILE ISOLATION RR NOLINEMACRO
PACKAGE; \
    db2 bind acid.bnd GRANT PUBLIC; \
    db2 connect reset; \
    db2 terminate

acid.o: acid.c
    $(CC) $(CFLAGS) -c acid.c -o acid.o

tsec: tsec.c
    $(CC) $(CFLAGS) $(LFLAGS) -o tsec tsec.c

clean:
    rm -f *.o *.bnd $(EXEC) tsec
    rm -f acid.c

```

Appendix F: Price Quotations



Protect Your Data - Grow Your Business

To:
Attention:
Phone:
Fax:
Email:

From: Alan Powers
Phone: (248)223-1020 x344
Fax: (248)223-1026
Email: apowers@compsat.com

QUOTE # : 4Wx365_051404
DATE: May 14, 2004

IBM x365 Configuration

		List Price		Compsat Discounted Price		
Part No.	Description	Qty	(per unit)	(quantity x unit price)	(per unit)	(quantity x unit price)
			US Dollar	US Dollar	US Dollar	US Dollar
x365 SERVER						
8862-6RX	x365, 2xXeon MP 3.0GHz/400MHz, 4MB, 2GB, O/Bay	1	\$ 15,599.00	\$ 15,599.00	\$13,103.16	\$13,103.16
96P2253	3 YR onsite repair 24x7x4 hour (x365)	1	\$ 1,000.00	\$ 1,000.00	\$880.00	\$880.00
13N0710	Xeon MP 3.0GHz/400MHz	2	\$ 5,749.00	\$ 11,498.00	\$5,174.10	\$10,348.20
73P9710	Memory Upgrade Card	1	\$ 999.00	\$999.00	\$899.10	\$899.10
73P2031	1GB PC2100 DDR ECC SDRAM RDIMM	16	\$ 659.00	\$ 10,544.00	\$593.10	\$9,489.60
32P0734	IBM 36.4GB 15K Ultra320 SCSI Hot-Swap SL HDD	2	\$ 495.00	\$ 990.00	\$445.50	\$891.00
633147N	E54 15" Colour Monitor (Stealth Grey)/MPRII	1	\$ 119.00	\$ 119.00	\$107.10	\$107.10
30L9183	3 YR onsite exch. 24x7x4 hour (E54 Monitor)	1	\$ 90.00	\$ 90.00	\$79.20	\$79.20
31P7415	IBM Preferred Pro Full-size Keyboard PS/2	1	\$ 29.00	\$ 29.00	\$26.10	\$26.10
28L3673	IBM Sleek 2-Button Mouse	1	\$ 14.95	\$ 14.95	\$13.46	\$13.46
32P0033	IBM ServeRAID-6M Ultra320 SCSI Adapter	4	\$ 999.00	\$ 3,996.00	\$899.10	\$3,596.40
03K9311	4.2M Ultra2 SCSI Cable	8	\$ 105.00	\$ 840.00	\$94.50	\$756.00
EXP400(s)						
17331RU	IBM EXP400 Storage Expansion Unit	8	\$ 3,099.00	\$ 24,792.00	\$2,603.16	\$20,825.28
41L2768	3 YR onsite repair 24x7x4 hour (EXP400)	8	\$ 760.00	\$ 6,080.00	\$668.80	\$5,350.40
32P0734	IBM 36.4 GB 15K Ultra320 SCSI Hot-Swap HDD	112	\$ 495.00	\$ 55,440.00	\$445.50	\$49,896.00
RACK and OPTIONS						
9306421	NetBAY42 SX Standard Expansion Rack Cabinet - includes Perforated front door	1	\$ 1,439.00	\$ 1,439.00	\$1,208.76	\$1,208.76
41L2758	3 YR onsite repair 24x7x4 hour (NetBAY Rack)	1	\$ 168.00	\$ 168.00	\$147.84	\$147.84
32P1020	APC Smart-UPS Model 1400	1	\$ 855.00	\$ 855.00	\$769.50	\$769.50
TOTAL =				\$134,492.95	TOTAL =	\$118,387.10

25330 Telegraph Road / Suite 200 Raleigh Officentre / Southfield, Michigan 48034
Phone: 248-223-1020 / Fax: 248-223-1026 / www.compsat.com

NOTE:

- This quote may include Compsat Technology consulting and configuration charges.
- Mfg. pricing is out of our control and could change without notice.
- Pricing good for 30 Days from date quoted.

Microsoft Corporation
One Microsoft Way
Redmond, WA 98052-6399

Tel 425 882 8080
Fax 425 936 7329
<http://www.microsoft.com/>

Microsoft

May 17, 2004

IBM Corporation
Tricia Thomas
D23U/B060/Office E127
3039 Cornwallis Road
Research Triangle Park,
NC 27709

Ms. Thomas:

Here is the information you requested regarding pricing for several Microsoft products to be used in conjunction with your TPC-H benchmark testing.

All pricing shown is in US Dollars (\$).

Part Number	Description	Unit Price	Quantity	Price
P72-00264	Windows Server 2003, Enterprise Edition <i>Server license only - No CALs</i> <i>Discount Schedule: Open Program - No Level</i> <i>Unit Price reflects a 40% discount from the</i> <i>retail unit price of \$3,999.</i>	\$2,399	1	\$2,399
PRO-PRORS-16U-01	Database Server Support Package <i>1 Year Term</i>	\$1,950	3	\$5,850

All products are currently orderable through Microsoft's normal distribution channels.

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: PHtrth0417057674

Please include this Reference ID in any correspondence regarding this price quote.

Shop in **Jewelry & Watches** (Beta-What is this?) 

amazon.com 

 [VIEW CART](#) | [WISH LIST](#) | [YOUR ACCOUNT](#) | [HELP](#) 

[WELCOME](#) [YOUR STORE](#) [BOOKS](#) [APPAREL & ACCESSORIES](#) [ELECTRONICS](#) [TOYS & GAMES](#) [MUSIC](#) [CELL PHONES & SERVICE](#) [SOFTWARE](#) [SEE MORE STORES](#) [Your Gold Box](#)

[BROWSE BRANDS & PRODUCTS](#) [TOP SELLERS](#) [NEW & FUTURE RELEASES](#) [CHILDREN'S SOFTWARE](#) [GAMES](#) [SOFTWARE DOWNLOADS](#) [TODAY'S DEALS](#) [OUTLET](#)

new styles for summer [shop NORDSTROM at Amazon.com](#)

SEARCH

Software 

ITEM INFORMATION

Explore this item

[buying info](#)[product description](#)[customer reviews](#)

See more by this manufacturer

[Microsoft](#)

Customers also bought

[these other items...](#)

Share your thoughts

[write a review](#)[write a So You'd Like to... guide](#)[e-mail a friend about this item](#)

Help us help others

[submit a manual](#)

RATE THIS ITEM

[\(Sign in to rate this item.\)](#)**Microsoft Visual C++ .NET Standard 2003**[Microsoft](#)**List Price:** ~~\$100.00~~**Price:** **\$89.99****You Save:** **\$19.01 (17%)****Availability:** Usually ships within 24 hoursThis item ships for **FREE** with **Super Saver Shipping**. [See details.](#)**3 used & new** from **\$89.99**

Platform:  Windows 2000 / 2003 Server / XP Professional / XP Home Edition / 2000 Server

Media: CD-ROM

Manufacturers, merchants, and enthusiasts: [Submit a product manual](#) for this item.**Amazon.com Sales Rank:** 263**Shipping:** [Amazon.com's shipping rates and policies](#)**Shipping:** Currently, item can be shipped only within the U.S.**ASIN:** B000089GKV**Item model number:** 254-00257**Date first available at Amazon:** February 2, 2003**Average Customer Review:**  Based on 6 reviews. [Write a review.](#)**Popular in:** [University of Illinois \(#17\)](#) , [Government \(#7\)](#) . [See more](#)**READY TO BUY?****Amazon.com****Price:** **\$89.99**Availability: **Usually ships in 24 hours****Add to Shopping Cart**

or

[Sign in](#) to turn on 1-Click ordering.**MORE BUYING CHOICES**

J&R Music and Computer World

Price: **\$99.99**

Availability: **Usually ships in 1-2 business days**

[Add to Cart](#)

Office Depot

Price: **\$102.95**

Availability: **Usually ships in 1-2 business days**

[Add to Cart](#)

3 used & new from **\$89.99**

Have one to sell?

[Sell yours here](#)[Add to Wish List](#)[Add to Wedding Registry](#)

Don't have one?
We'll set one up for you.

Visit the [Books Store](#)

Customers who bought this item also bought these items:

- [Microsoft Visual C# .NET Standard 2003](#) by Microsoft ([Rate it](#))
- [Microsoft Visual Basic .NET Standard 2003](#) by Microsoft Software ([Rate it](#))
- [Borland C++ Builder 6 Personal](#) by Borland International ([Rate it](#))
- [MICROSOFT VISUAL C++ NET DELUXE \(0-7356-1908-5 \)](#) by Microsoft Press ([Rate it](#))