TPC Benchmark™ H Full Disclosure Report

for

IBM® @server™ xSeries® 445

using

IBM DB2® Universal Database 8.1

Submitted for Review

June 30, 2003

First Edition - June 2003

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.

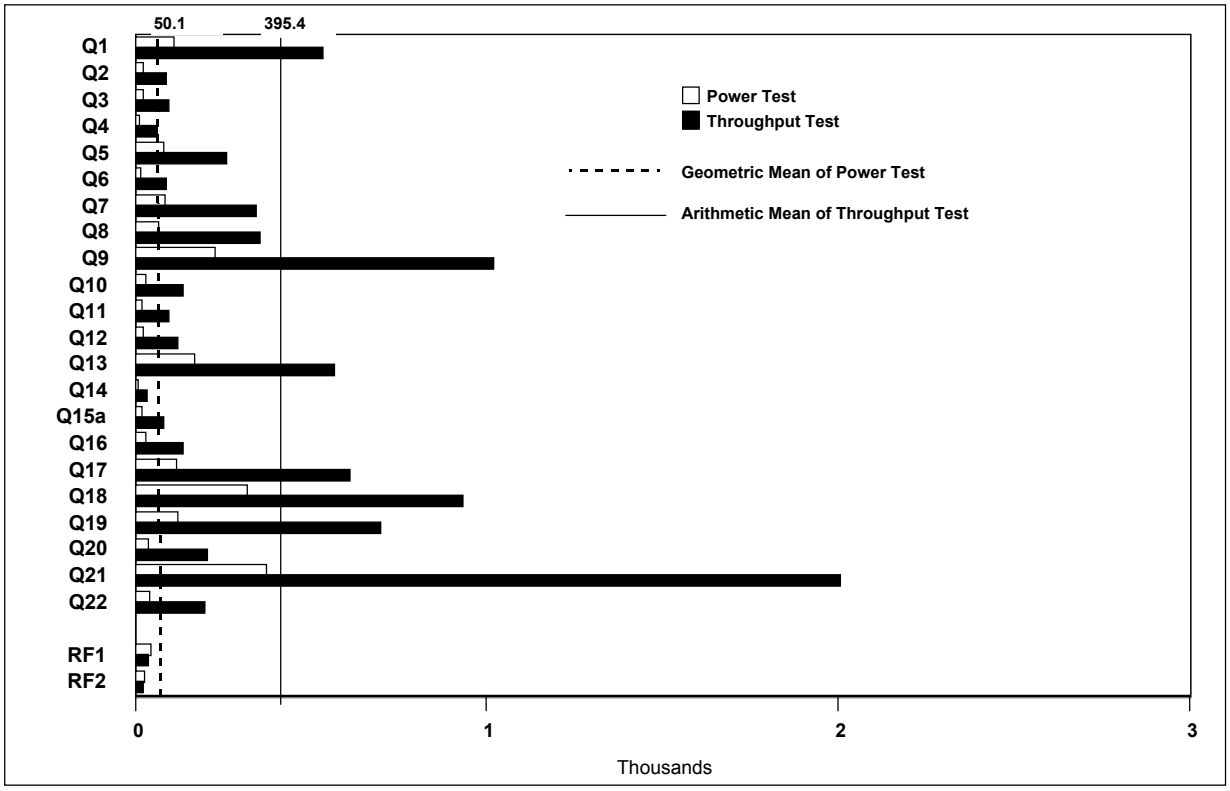
Notes

[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	IBM® @server™ xSeries® 445 with IBM DB2® UDB 8.1	TPC-H Rev 2.0 Report Date: June 30, 2003

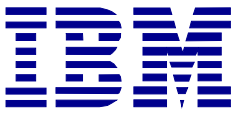
Total System Cost	Composite Query-per-Hour Metric	Price/Performance
$410,229	**5602.6** **QphH @ 100GB**	**$73** **per QphH@100GB**

Database Size	Database Manager	Operating System	Other Software	Availability Date
100GB	**IBM DB2 UDB 8.1**	**Microsoft® Windows® Server 2003 Enterprise Edition**	**Microsoft Visual Studio Professional**	**Dec. 31, 2003**



Database Load Time: 01:37:22	Load Included Backup: Y	Total Data Storage / Database Size: 24.39
RAID (Base Tables Only): N	**RAID (Base Tables and Auxiliary Data Structures): N**	**RAID (All): N**

Configuration		
Processors/Cores/Threads	**8/8/16**	**Intel Xeon MP 2.8GHz / 2MB ECC L3 Cache**
Memory	**32**	**512MB PC2100 DDR ECC SDRAM RDIMM**
Disk Controllers	**5**	**ServeRAID-6M Ultra320 SCSI Adapter**
Disk Drives	**140**	**18.2GB 15K Ultra160 SCSI Drive**
	2	**36.4GB 15K Ultra160 SCSI Drive**
Total Disk Storage		**2439.4GB**



IBM @server xSeries 445 with IBM DB2 UDB 8.1

TPC-H Revision 2.0

Report Date: June 30, 2003

Description	Part Number	Source	Unit Price	Qty	Ext. Price	3-Yr. Maint.
Server Hardware						
xSeries 445 w / 4 x 2.8GHz/2MB L3 Xeon MP and 4 x 512MB Memory	8870-7RX	1	40,799	1	40,799	3,390
2.8GHz / 2MB L3 Cache Xeon MP	02R2064	1	6,599	4	26,396	0
xSeries 445 SMP Expansion Module	02R1870	1	4,849	1	4,849	0
512MB PC2100 DDR ECC SDRAM	33L5038	1	305	28	8,540	0
ServeRAID-6M Ultra320 Adapter	32P0033	1	999	5	4,995	0
36.4GB 15K Ultra160 SCSI Drive	06P5768	1	549	2	1,098	0
4.2M Ultra2 SCSI Cable	03K9311	1	105	10	1,050	0
APC Smart-UPS Model 1400	32P1020	1	855	1	855	0
IBM Preferred Pro Full-Size Keyboard	31P7415	1	29	1	29	0
IBM Sleek 2-Button Mouse	28L3673	1	19	1	19	0
E54 15” (13.8” Viewable) Display	633147N	1	129	1	129	90
NetBAY42S Enterprise Rack	9306421	1	1,439	1	1,439	168
				Subtotal	$90,198	$3,648
Server Storage						
RXE-100 Remote Exp. Enclosure	8684-1RX	1	4,569	1	$4,569	$1,330
EXP300 Rack Storage Expansion Enclosure	35311RU	1	3,179	10	31,790	2,000
18.2GB 15K Ultra160 SCSI Drive	06P5767	1	329	140	46,060	0
				Subtotal	$82,419	$3,330
Server Software						
DB2 UDB ESE 8.1License/1-Yr Maint.	D518GLL	1	21,242	8	169,936	
DB2 UDB ESE 8.1 Support -1Yr/Proc.	E00BILL	1	4,721	16		75,536
Microsoft Windows Server 2003 Enterprise Edition	P72-00264	2	2,399	1	2,399	Incl Below
Visual Studio Professional	659-00844	2	1,079	1	1,079	Incl Below
Operating System Support Package	PROPROPS16U01	2	1,950	3		5,850
				Subtotal	$173,414	$81,386
IBM hardware discount of 14%.					$24,166	$0
				Total	$321,865	$88,364

Notes: Standard 3-Year and extended warranties upgraded to 7x24, 4-hour response time coverage. *10% or minimum 2 spares are added in place of on-site service; product has a 5-year return-to-vendor warranty.
Pricing: 1 - IBM List Prices - 919-254-0367; 2 - Microsoft Corp.
Audited by Brad Askins of InfoSizing, Inc.

3-Year Cost of Ownership:	**$410,229**
QphH @ 100GB:	**5,602.6**
$ / QphH @ 100GB:	**$73**

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 specification. If you find that stated prices are not available according to these terms, please inform the TPC at pricing@tpc.org. Thank you.

Measurement Results:

Database Scale Factor	100
Total Data Storage/Database Size	24.39
Start of Database Load	15:11:31
End of Database Load	16:48:53
Database Load Time	01:37:22
Query Streams for Throughput Test	5
TPC-H Power	7,187.7
TPC-H Throughput	4,367.0
TPC-H Composite Query-per-Hour (QphH@100GB)	5,602.6
Total System Price over 3 Years	$410,229
TPC-H Price/Performance Metric ($/QphH@100GB)	$73

Measurement Interval:

Measurement Interval in Throughput Test (Ts) = 9068 seconds

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	608164853	06/08/03 17:04:17 06/08/03 17:36:58	06/08/03 17:03:22 06/08/03 17:04:06	06/08/03 17:36:58 06/08/03 17:37:24	00:32:41
Stream 01	608164854	06/08/03 17:37:26 06/08/03 20:02:11	06/08/03 20:03:26 06/08/03 20:04:10	06/08/03 20:04:10 06/08/03 20:04:34	02:24:45
Stream 02	608164855	06/08/03 17:37:26 06/08/03 20:01:31	06/08/03 20:04:34 06/08/03 20:05:06	06/08/03 20:05:06 06/08/03 20:05:29	02:24:05
Stream 03	608164856	06/08/03 17:37:26 06/08/03 20:01:59	06/08/03 20:05:29 06/08/03 20:06:09	06/08/03 20:06:09 06/08/03 20:06:34	02:24:33
Stream 04	608164857	06/08/03 17:37:26 06/08/03 20:03:00	06/08/03 20:06:34 06/08/03 20:07:13	06/08/03 20:07:13 06/08/03 20:07:36	02:25:34
Stream 05	608164858	06/08/03 17:37:26 06/08/03 20:03:27	06/08/03 20:07:36 06/08/03 20:08:10	06/08/03 20:08:10 06/08/03 20:08:34	02:26:01

TPC-H Timing Intervals (in seconds):

Query	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12
Stream 00	111.9	23.5	15.1	12.5	84.0	17.1	86.5	65.8	229.2	32.1	19.5	25.4
Stream 01	581.2	70.2	64.1	25.6	309.2	79.2	320.4	274.2	978.5	109.8	85.0	139.3
Stream 02	494.9	112.2	72.2	64.4	239.8	91.0	402.7	415.8	1,123.9	126.7	103.3	119.5
Stream 03	565.4	108.4	50.2	53.3	240.5	76.9	300.1	302.9	980.3	186.3	89.0	92.3
Stream 04	538.2	74.0	49.8	114.2	246.0	116.8	289.8	510.8	1,215.1	113.8	90.0	153.8
Stream 05	491.7	84.0	242.5	57.9	272.3	87.3	420.7	282.5	812.0	152.3	110.8	111.8
Minimum	491.7	70.2	49.8	25.6	239.8	76.9	289.8	274.2	812.0	109.8	85.0	92.3
Average	534.3	89.8	95.8	63.1	261.6	90.2	346.7	357.2	1,022.0	137.8	95.6	123.3
Maximum	581.2	112.2	242.5	114.2	309.2	116.8	420.7	510.8	1,215.1	186.3	110.8	153.8

Stream ID	Q13	Q14	Q15a	Q16	Q17	Q18	Q19	Q20	Q21	Q22	RF1	RF2
Stream 00	170.1	10.9	21.6	32.8	118.8	318.2	121.4	39.7	374.9	41.3	44.1	25.7
Stream 01	606.7	26.1	80.5	208.4	731.0	964.6	648.4	222.2	1,916.0	244.0	44.0	23.8
Stream 02	498.0	30.4	128.0	95.6	611.1	961.9	578.1	212.3	1,965.6	197.7	32.1	22.2
Stream 03	533.7	29.6	85.1	125.3	546.7	1,030.1	579.9	231.6	2,258.6	206.5	40.3	25.2
Stream 04	504.2	48.6	60.9	97.8	641.1	790.8	877.2	87.8	1,961.5	151.5	38.8	23.0
Stream 05	694.5	41.2	61.7	161.2	538.5	927.2	810.4	278.8	1,933.7	187.5	34.1	23.9
Minimum	498.0	26.1	60.9	95.6	538.5	790.8	578.1	87.8	1,916.0	151.5	32.1	22.2
Average	567.4	35.2	83.2	137.7	613.7	934.9	698.8	206.5	2,007.1	197.4	37.9	23.6
Maximum	694.5	48.6	128.0	208.4	731.0	1,030.1	877.2	278.8	2,258.6	244.0	44.0	25.2

Table of Contents

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 diclosure report for benchmark testing of the IBM @server xSeries 445 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, exeat 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.

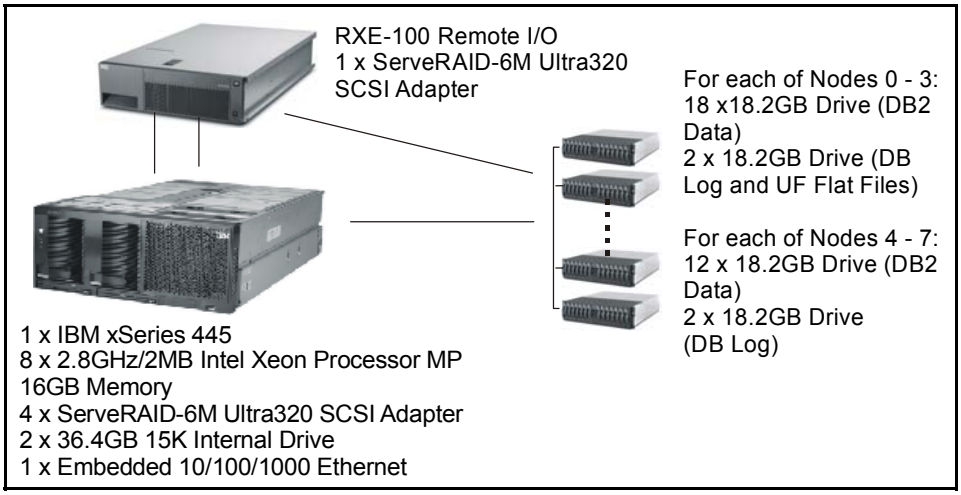
1.3 Configuration Diagrams

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 445 contained:

- Eight Intel Xeon MP 2.8GHz processors, each with 2MB of ECC L3 cache
- 16GB of memory
- One embedded dual-port 10/100/1000 Ethernet interface
- One embedded dual-channel Ultra320 SCSI interface
- Five ServeRAID-6M Ultra320 SCSI adapters
- One hundred forty (140) 18.2GB 15K Ultra160 SCSI disk drives
- Ten (10) EXP300 Storage Expansion Enclosures
- Two (2) 36.4GB 15K Ultra160 SCSI Disk Drives
- One RXE-100 Remote I/O Expansion Enclosure

The measured configuration was different in that it contained one additional EXP300 Storage Expansion unit, which held fourteen 18.2GB 15K Ultra160 SCSI drives that held raw data files. A sixth 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).

No replication was 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.1, "Qualification Queries," contains the output for each of the queries. The functional query definitions and variants used in this dicslosure 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 445 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 Atomiciy 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. Six 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 six 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

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. Six 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 an additional 100 transactions per streams.
4. The success file and the HISTORY table counts were compared and were found to match.
5. 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.

1. Six streams of ACID transactions were started. Each stream executed a minimum of 100 transactions.
2. Then the system was powered off.
3. When power was restored, the system rebooted and the database was restarted.
4. The database went through a recovery period.
5. The success file and the HISTORY table counts were compared and were found to match.

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	150,000,000
Lineitem	600,037,902
Customer	15,000,000
Part	20,000,000
Supplier	1,000,000
Partsupp	80,000,000
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	1 - 36.4GB	Physical Drive 0	33.9GB	Compiler, Operating System
	1 - 36.4GB	Physical Drive 1	33.9GB	DB2, TPC-H Kit
ServeRAID - 1	6 - 18.2GB RAID-0	Physical Drive 2; Logical Node 4	5860MB 1000MB 2730MB 800MB 11720MB 60000MB 1000MB 400MB 800MB 300MB 2440MB 16160MB	Lineitem Data Lineitem Indexes Other Table Data Other Table Indexes Temp Tables M: Stripeset Backup RF Raw Data Lineitem Data (Qual) Lineitem Indexes (Qual) Temp Tables (Qual) Other Table Indexes (Qual) Other Tables (Qual) Free Space
	6 - 18.2GB RAID-0	Physical Drive 3; Logical Node 5	5860MB 1000MB 2730MB 800MB 11720MB 60000MB 1000MB 400MB 800MB 300MB 2440MB 16160MB	Lineitem Data Lineitem Indexes Other Table Data Other Table Indexes Temp Tables M: Stripeset RF Raw Data Backup Lineitem Data (Qual) Lineitem Indexes (Qual) Temp Tables Data (Qual) Other Table Indexes (Qual) Other Tables (Qual) Free Space
	2 - 18.2GB RAID-1	Physical Drive 4; Logical Node 0	16940MB	NTFS E:\logs
	6 - 18.2GB RAID-0	Physical Drive 5; Logical Node 6	5860MB 1000MB 2730MB 800MB 11720MB 60000MB 21040MB	Lineitem Data Lineitem Indexes Other Tables Data Other Indexes Temp Tables M: Stripeset Backup RF Raw Data Free Space
	6 - 18.2GB RAID-0	Physical Drive 6; Logical Node 7	5860MB 1000MB 2730MB 800MB 11720MB 60000MB 21040MB	Lineitem Data Lineitem Indexes Other Table Data Other Table Indexes Temp Tables M: Stripeset Backup RF Raw Data Free Space
	2 - 18.2GB RAID-1	Physical Drive 7; Logical Node 1	16940MB	NTFS F:\logs

Controller	Drives	Partition*	Size	Use
ServeRAID - 2	6 - 18.2GB RAID-0	Physical Drive 8; Logical Node 4	5860MB 1000MB 2730MB 800MB 11720MB 79630MB	Lineitem Data Lineitem Indexes Other Table Data Other Table Indexes Temp Tables Free Space
	6 - 18.2GB RAID-0	Physical Drive 9; Logical Node 5	5860MB 1000MB 2730MB 800MB 11720MB 79630MB	Lineitem Data (Qual) Lineitem Indexes (Qual) Other Table Data (Qual) Other Table Indexes (Qual) Temp Tables (Qual) Free Space
	2 - 18.2GB RAID-1	Physical Drive 10; Logical Node 0	16940MB	Unused
	6 - 18.2GB RAID-0	Physical Drive 11; Logical Node 6	5860MB 1000MB 2730MB 800MB 11720MB 79630MB	Lineitem Data Lineitem Indexes Other Table Data Other Table Indexes Temp Tables Free Space
	6 - 18.2GB RAID-0	Physical Drive 12; Logical Node 7	5860MB 1000MB 2730MB 800MB 11720MB 79630MB	Lineitem Data (Qual) Lineitem Indexes (Qual) Other Table Data (Qual) Other Table Indexes (Qual) Temp Tables (Qual) Free Space
	2 - 18.2GB RAID-1	Physical Drive 13; Logical Node 0	16940MB	Unused

Controller	Drives	Partition*	Size	Use
ServeRAID - 3	6 - 18.2GB RAID-0	Physical Drive 14; Logical Node 4	5860MB 1000MB 2730MB 800MB 11720MB 60000MB 1000MB 400MB 800MB 300MB 2440MB 16160MB	Lineitem Data Lineitem Indexes Other Table Data Other Table Indexes Temp Tables N: Stripeset Backup Lineitem Data (Qual) Lineitem Indexes (Qual) Temp Tables (Qual) Other Table Indexes (Qual) Other Tables (Qual) Free Space
	6 - 18.2GB RAID-0	Physical Drive 15; Logical Node 5	5860MB 1000MB 2730MB 800MB 11720MB 60000MB 1000MB 400MB 800MB 300MB 2440MB 16160MB	Lineitem Data Lineitem Indexes Other Table Data Other Indexes Temp Tables N: Stripeset Backup Lineitem Data (Qual) Lineitem Indexes (Qual) Temp Tables (Qual) Other Table Indexes (Qual) Other Tables Free Space
	2 - 18.2GB RAID-1	Physical Drive 16; Logical Node 2	16940MB	NTFS G:\Logs
	6 - 18.2GB RAID-0	Physical Drive 17; Logical Node 6	5860MB 1000MB 2730MB 800MB 11720MB 60000MB 21040MB	Lineitem Data Lineitem Indexes Other Table Data Other Table Indexes Temp Tables N: Stripeset Backup Free Space
	6 - 18.2GB RAID-0	Physical Drive 18; Logical Node 7	5860MB 1000MB 2730MB 800MB 11720MB 60000MB 21040MB	Lineitem Data Lineitem Indexes Other Table Data Other Table Indexes Temp Tables N: Stripeset Backup Free Space
	2 - 18.2GB RAID-1	Physical Drive 19; Logical Node 3	16940MB	NTFS H:\Logs
ServeRAID - 4	14 - 18.2GB RAID-5	Physical Drive 20	22035MB	NTFS W: Raw Data

Controller	Drives	Partition*	Size	Use
ServeRAID - 5	6 - 18.2GB RAID-0	Physical Drive 21; Logical Node 0	5860MB 1000MB 2730MB 800MB 11720MB 60000MB 21040MB	Lineitem Data Lineitem Indexes Other Table Data Other Table Indexes Temp Tables P: Stripeset Backup Free Space
	6 - 18.2GB RAID-0	Physical Drive 22; Logical Node 1	5860MB 1000MB 2730MB 800MB 11720MB 60000MB 21040MB	Lineitem Data Lineitem Indexes Other Table Data Other Table Indexes Temp Tables P: Stripeset Backup Free Space
	2 - 18.2GB RAID-1	Physical Drive 23; Logical Node 6	16940MB	NTFS K:\Logs
	6 - 18.2GB RAID-0	Physical Drive 24; Logical Node 2	5860MB 1000MB 2730MB 800MB 11720MB 60000MB 21040MB	Lineitem Data Lineitem Indexes Other Table Data Other Table Indexes Temp Tables P: Stripeset Backup Free Space
	6 - 18.2GB RAID-0	Physical Drive 25; Logical Node 3	5860MB 1000MB 2730MB 800MB 11720MB 60000MB 21040MB	Lineitem Data Lineitem Indexes Other Table Data Other Table Indexes Temp Tables P: Stripeset Backup Free Space
	2 - 18.2GB RAID-1	Physical Drive 26; Logical Node 7	16940MB	NTFS L:\Logs

Controller	Drives	Partition*	Size	Use
ServeRAID - 6	6 - 18.2GB RAID-0	Physical Drive 27; Logical Node 0	5860MB 1000MB 2730MB 800MB 11720MB 60000MB 21040MB	Lineitem Data Lineitem Indexes Other Table Data Other Table Indexes Temp Tables O: Stripeset Backup Free Space
	6 - 18.2GB RAID-0	Physical Drive 28; Logical Node 1	5860MB 1000MB 2730MB 800MB 11720MB 60000MB 21040MB	Lineitem Data Lineitem Indexes Other Table Data Other Table Indexes Temp Tables O: Stripeset Backup Free Space
	2 - 18.2GB RAID-1	Physical Drive 29; Logical Node 4	16940MB	NTFS I:\Logs
	6 - 18.2GB RAID-0	Physical Drive 30; Logical Node 2	5860MB 1000MB 2730MB 800MB 11720MB 60000MB 21040MB	Lineitem Data Lineitem Indexes Other Table Data Other Table Indexes Temp Tables O: Stripeset Backup Free Space
	6 - 18.2GB RAID-0	Physical Drive 31; Logical Node 3	5860MB 1000MB 2730MB 800MB 11720MB 60000MB 21040MB	Lineitem Data Lineitem Indexes Other Table Data Other Table Indexes Temp Tables O: Stripeset Backup Free Space
	2 - 18.2GB RAID-1	Physical Drive 32; Logical Node 5	16940MB	NTFS J:\Logs

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

The priced configuration used 140 disks. An additional 14 disks were used to store the 100GB database files and were not priced.

5.3 Database Partition / Replication Mapping

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

Database partitioning/replication was not used.

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 D: drive 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
18.2GB 15K Ultra160 SCSI Drive	140	16.94GB	2371.6GB		
36.4GB 15K Ultra160 SCSI Drive	2	33.9GB	67.8GB		
Total			2439.4GB	100	24.39

The data storage ratio is 24.39, derived by dividing 24.39GB by the database size of 100GB.

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 node 0 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.

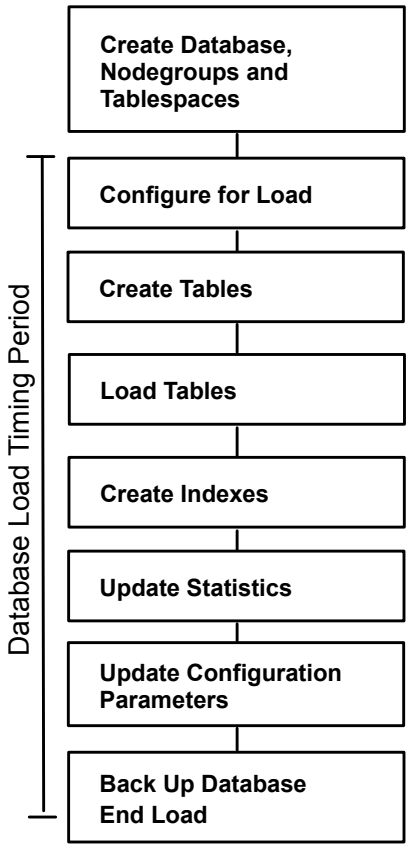


Figure 4-1. Database Load Procedure

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, 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.

Five 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 @ 100GB	QthH @ 100GB	QphH @ 100GB
Run1	7,187.7	4,367.0	5,602.6
Run2	7,488.1	4,365.1	5,717.2

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 will be generally available December 31, 2003.

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 Brad Askins of InfoSizing, Inc. Further information can be downloaded from www.tpc.org.

INFO SIZING

TPC TRANSACTION PROCESSING PERFORMANCE COUNCIL CERTIFIED AUDITOR

Benchmark Sponsors:

Chitra Balachandran
Mgr., xSeries Performance
IBM Systems Group
3039 Cornwallis Road
Research Triangle Park, NC 27709

Haider Rizvi
Mgr., DB2 Data Warehouse Performance
IBM Canada Ltd;
8200 Warden Avenue
Markham, Ontario L6G 1C7

June 18, 2003

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

Platform: **IBM @server xSeries 445**

Database Manager: **IBM DB2 UDB 8.1**

Operating System: **Microsoft Windows Server 2003 Enterprise Edition**

The results were:

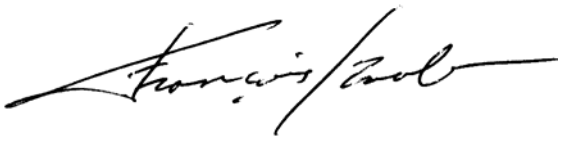
CPU (Speed)	Memory	Disks	**QphH@100GB**
IBM @server xSeries 445			
8 x Intel Xeon MP (2.8 GHz)	2MB ECC L3 Cache/cpu 16 GB Main	140 x 18.2 GB 2 x 36.4 GB	**5602.6**

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 100GB 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 1 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 5 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

Respectfully Yours,



François Raab, President

Bradley J. Askins, Auditor

1373 North Franklin Street • Colorado Springs, CO 80903-2527 • Office: 719/473-7555 • Fax: 719/473-7554

Appendix A: Tunable Parameters and System Configuration

DB2 UDB 8.1 Database and Database Manager Configuration

Database Configuration for Node 0

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 = NO

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) = 60000

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) = 8000

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) = 80

Number of asynchronous page cleaners (NUM_IOCLEANERS) = 4

Number of I/O servers (NUM_IOSERVERS) = 2

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\NODE0000\

Overflow log path (OVERFLOWLOGPATH) =

Mirror log path (MIRRORLOGPATH) =

First active log file =

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 chckpt (SOFTMAX) = 600

Log retain for recovery enabled (LOGRETAIN) = OFF

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) =

Database Configuration for Node 1

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 = NO

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) = 60000

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) = 8000

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) = 80

Number of asynchronous page cleaners (NUM_IOCLEANERS) = 4

Number of I/O servers (NUM_IOSERVERS) = 2

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\NODE0001\

Overflow log path (OVERFLOWLOGPATH) =

Mirror log path (MIRRORLOGPATH) =

First active log file =

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 chckpt (SOFTMAX) = 600

Log retain for recovery enabled (LOGRETAIN) = OFF

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) =

Database Configuration for Node 2

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 = NO

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) = 60000

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) = 8000

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) = 80

Number of asynchronous page cleaners (NUM_IOCLEANERS) = 4

Number of I/O servers (NUM_IOSERVERS) = 2

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\NODE0002\

Overflow log path (OVERFLOWLOGPATH) =

Mirror log path (MIRRORLOGPATH) =

First active log file =

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 chckpt (SOFTMAX) = 600

Log retain for recovery enabled (LOGRETAIN) = OFF

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) =

Database Configuration for Node 3

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                      = NO
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) = 60000
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) = 8000
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) = 80
Number of asynchronous page cleaners   (NUM_IOCLEANERS) = 4
```

Number of I/O servers (NUM_IOSERVERS) = 2

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\NODE0003\

Overflow log path (OVERFLOWLOGPATH) =

Mirror log path (MIRRORLOGPATH) =

First active log file =

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 chckpt (SOFTMAX) = 600

Log retain for recovery enabled (LOGRETAIN) = OFF

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) =

Database Configuration for Node 4

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          = NO
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) = 60000
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) = 8000
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) = 80
Number of asynchronous page cleaners    (NUM_IOCLEANERS) = 4
Number of I/O servers                   (NUM_IOSERVERS) = 2
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\NODE0004\
Overflow log path                       (OVERFLOWLOGPATH) =
Mirror log path                         (MIRRORLOGPATH) =
First active log file                   =
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 chckpt (SOFTMAX) = 600

Log retain for recovery enabled (LOGRETAIN) = OFF

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) =

Database Configuration for Node 5

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 = NO

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) = 60000

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) = 8000

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) = 80

Number of asynchronous page cleaners (NUM_IOCLEANERS) = 4

Number of I/O servers (NUM_IOSERVERS) = 2

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\NODE0005\

Overflow log path (OVERFLOWLOGPATH) =

Mirror log path (MIRRORLOGPATH) =

First active log file =

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 chckpt (SOFTMAX) = 600

Log retain for recovery enabled (LOGRETAIN) = OFF

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) =

Database Configuration for Node 6

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                  = NO
 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) = 60000
 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) = 8000
 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) = 80
 Number of asynchronous page cleaners   (NUM_IOCLEANERS) = 4
 Number of I/O servers                    (NUM_IOSERVERS) = 2
 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\NODE0006\

Overflow log path (OVERFLOWLOGPATH) =

Mirror log path (MIRRORLOGPATH) =

First active log file =

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 chckpt (SOFTMAX) = 600

Log retain for recovery enabled (LOGRETAIN) = OFF

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) =

Database Configuration for Node 7

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 = NO

```
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) = 60000
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) = 8000
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) = 80
Number of asynchronous page cleaners  (NUM_IOCLEANERS) = 4
Number of I/O servers                   (NUM_IOSERVERS) = 2
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                            = l:\logs\NODE0007\
Overflow log path                       (OVERFLOWLOGPATH) =
Mirror log path                         (MIRRORLOGPATH) =
First active log file                        =
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 chckpt (SOFTMAX) = 600
Log retain for recovery enabled         (LOGRETAIN) = OFF
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 Database Manager Configuration

DB2 Database Manager Configuration

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) =
2.000000e-005
Communications bandwidth (MB/sec) (COMM_BANDWIDTH) =
1.000000e+002

Max number of concurrently active databases (NUMDB) = 1
Data Links support (DATALINKS) = 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) = 3
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) =

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) (RQRIOBLK) = 32767
DOS requester I/O block size (bytes) (DOS_RQRIOBLK) = 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
Initialize fenced process with JVM (INITFENCED_JVM) = NO
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
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 Registry Variables

DB2_EXTENDED_OPTIMIZATION=YES
DB2_ANTIJOIN=Y
DB2BPVARS=d:\tpch\ddl\scattered_read
DB2ACCOUNTNAME=RUTHLESS-D\administrator
DB2INSTOWNER=RUTHLESS1
DB2PORTRANGE=60008:60011
DB2MEMMAXFREE=1000000000
DB2OPTIONS=-t -v +c
DB2NTNOCACHE=ON
DB2INSTPROF=D:\SQLLIB
DB2COMM=TCPIP
DB2_PARALLEL_IO=*

Microsoft Windows Server 2003 Enterprise Edition

Configuration Parameters

The following startup parmeters were selected via Boot.ini

[boot loader]
timeout=5
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" /fastdetect /PAE

SUT Hardware Information Report

System Information report written at: 06/17/03 15:48:02
System Name: RUTHLESS1
[System Summary]

Item Value
OS Name Microsoft(R) Windows(R) Server 2003, Enterprise Edition
Version 5.2.3785 Build 3785
OS Manufacturer Microsoft Corporation
System Name RUTHLESS1
System Manufacturer IBM
System Model eserver xSeries 445 -[88704RX]-
System Type X86-based PC
Processor x86 Family 15 Model 2 Stepping 5 GenuineIntel ~2799 Mhz
Processor x86 Family 15 Model 2 Stepping 5 GenuineIntel ~2799 Mhz
Processor x86 Family 15 Model 2 Stepping 5 GenuineIntel ~2799 Mhz
Processor x86 Family 15 Model 2 Stepping 5 GenuineIntel ~2799 Mhz
Processor x86 Family 15 Model 2 Stepping 5 GenuineIntel ~2800 Mhz
Processor x86 Family 15 Model 2 Stepping 5 GenuineIntel ~2800 Mhz
Processor x86 Family 15 Model 2 Stepping 5 GenuineIntel ~2799 Mhz
Processor x86 Family 15 Model 2 Stepping 5 GenuineIntel ~2799 Mhz
Processor x86 Family 15 Model 2 Stepping 5 GenuineIntel ~2799 Mhz
Processor x86 Family 15 Model 2 Stepping 5 GenuineIntel ~2799 Mhz
Processor x86 Family 15 Model 2 Stepping 5 GenuineIntel ~2799 Mhz
Processor x86 Family 15 Model 2 Stepping 5 GenuineIntel ~2800 Mhz
Processor x86 Family 15 Model 2 Stepping 5 GenuineIntel ~2799 Mhz
BIOS Version/Date IBM -[REE112EDI-1.00]-, 6/2/2003
SMBIOS Version 2.3
Windows Directory C:\WINDOWS
System Directory C:\WINDOWS\system32
Boot Device \Device\HarddiskVolume1
Locale United States
Hardware Abstraction Layer Version = "1.25.0000 built by: (x86chk_i)"
User NameRUTHLESS-D\tpch
Time ZoneEastern Daylight Time
Total Physical Memory 16,384.00 MB
Available Physical Memory 15.46 GB
Total Virtual Memory 33.75 GB
Available Virtual Memory 33.00 GB
Page File Space 17.76 GB
Page File C:\pagefile.sys

[Hardware Resources]

[Conflicts/Sharing]

Resource Device
I/O Port 0x00000000-0x00001FFF PCI bus
I/O Port 0x00000000-0x00001FFF Direct memory access controller

Memory Address 0xF8B00000-0xF8BFFFFF PCI bus
Memory Address 0xF8B00000-0xF8BFFFFF PCI standard PCI-to-PCI bridge

I/O Port 0x00002000-0x000027FF PCI bus
I/O Port 0x00002000-0x000027FF LSI Logic PCI-X Ultra320 SCSI Host Adapter

Memory Address 0xE8000000-0xE9FFFFFF PCI bus
Memory Address 0xE8000000-0xE9FFFFFF RAGE XL PCI Family (Microsoft Corporation)

Memory Address 0xF8A00000-0xF8AFFFFF PCI bus
Memory Address 0xF8A00000-0xF8AFFFFF PCI standard PCI-to-PCI bridge

Memory Address 0xED400000-0xED4FFFFF PCI bus

Memory Address 0xED400000-0xED4FFFFF LSI Logic PCI-X Ultra320 SCSI Host Adapter

Memory Address 0xED000000-0xED1FFFFF PCI bus
Memory Address 0xED000000-0xED1FFFFF PCI standard PCI-to-PCI bridge

IRQ 18 VIA Rev 5 or later USB Universal Host Controller
IRQ 18 VIA Rev 5 or later USB Universal Host Controller

Memory Address 0xA0000-0xA7FFF PCI bus
Memory Address 0xA0000-0xA7FFF RAGE XL PCI Family (Microsoft Corporation)

[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 RAGE XL PCI Family (Microsoft Corporation) OK
0x000003B0-0x000003BB RAGE XL PCI Family (Microsoft Corporation) OK
0x000003C0-0x000003DF RAGE XL PCI Family (Microsoft Corporation) OK
0x00001900-0x0000197F Other PCI Bridge Device OK
0x00000A79-0x00000A79 ISAPNP Read Data Port OK
0x00000279-0x00000279 ISAPNP Read Data Port OK
0x00000274-0x00000277 ISAPNP Read Data Port OK
0x00000430-0x00000437 Motherboard resources OK
0x00000438-0x00000439 Motherboard resources OK
0x0000002E-0x0000002F Motherboard resources OK
0x00000064-0x00000064 Standard 101/102-Key or Microsoft Natural PS/2 Keyboard OK
0x00000060-0x00000060 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
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
0x00000092-0x00000092 Motherboard resources OK
0x000000A8-0x000000A9 Motherboard resources OK
0x00000440-0x0000044F Motherboard resources OK
0x000004C0-0x000004C3 Motherboard resources OK
0x000004D0-0x000004D1 Motherboard resources OK
0x000004E0-0x000004FF Motherboard resources OK
0x00000500-0x0000057F Motherboard resources OK
0x00000700-0x0000070F VIA Bus Master 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
0x00001980-0x0000199F VIA Rev 5 or later USB Universal Host Controller OK
0x000019A0-0x000019BF VIA Rev 5 or later USB Universal Host Controller OK
0x00002000-0x000027FF PCI bus OK
0x00002000-0x000027FF LSI Logic PCI-X Ultra320 SCSI Host Adapter OK
0x00002100-0x000021FF LSI Logic PCI-X Ultra320 SCSI Host Adapter OK
0x0000E000-0x0000EFFF PCI bus OK
0x00003000-0x00003FFF PCI bus OK
0x00004000-0x00005FFF PCI bus OK
0x00006000-0x00007FFF PCI bus OK
0x00008000-0x00009FFF PCI bus OK
0x0000A000-0x0000BFFF PCI bus OK
0x0000C000-0x0000DFFF PCI bus OK

[IRQs]

Resource Device Status
IRQ 9 Microsoft ACPI-Compliant System OK
IRQ 39 RAGE XL PCI Family (Microsoft Corporation) OK
IRQ 4 Other PCI Bridge Device 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 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 15 Secondary IDE Channel OK
IRQ 18 VIA Rev 5 or later USB Universal Host Controller OK
IRQ 18 VIA Rev 5 or later USB Universal Host Controller OK
IRQ 40 LSI Logic PCI-X Ultra320 SCSI Host Adapter OK
IRQ 41 LSI Logic PCI-X Ultra320 SCSI Host Adapter OK
IRQ 42 Broadcom NetXtreme Gigabit Ethernet OK
IRQ 11 Broadcom NetXtreme Gigabit Ethernet #2 OK
IRQ 51 IBM ServeRAID 6M Controller OK
IRQ 55 IBM ServeRAID 6M Controller OK
IRQ 71 IBM ServeRAID 6M Controller OK
IRQ 67 IBM ServeRAID 6M Controller OK
IRQ 63 IBM ServeRAID 6M Controller OK
IRQ 116 IBM ServeRAID 6M Controller OK

[Memory]

Resource Device Status
0xA0000-0xA7FFF PCI bus OK
0xA0000-0xA7FFF RAGE XL PCI Family (Microsoft Corporation) OK
0xA8000-0xAFFFF PCI bus OK
0xB0000-0xB7FFF PCI bus OK
0xB8000-0xBFFFF PCI bus OK
0xE8000000-0xE9FFFFFF PCI bus OK
0xE8000000-0xE9FFFFFF RAGE XL PCI Family (Microsoft Corporation) OK
0xE9200000-0xE9200FFF RAGE XL PCI Family (Microsoft Corporation) OK
0xE9000000-0xE91FFFFF Other PCI Bridge Device OK
0x0400-0x04FF System board OK
0x100000-0xDFFAC33F Memory Module OK
0xED400000-0xED4FFFFF PCI bus OK
0xED400000-0xED4FFFFF LSI Logic PCI-X Ultra320 SCSI Host Adapter OK

0xED410000-0xED41FFFF LSI Logic PCI-X Ultra320 SCSI Host Adapter OK
0xED420000-0xED42FFFF LSI Logic PCI-X Ultra320 SCSI Host Adapter OK
0xED430000-0xED43FFFF LSI Logic PCI-X Ultra320 SCSI Host Adapter OK
0xED440000-0xED44FFFF Broadcom NetXtreme Gigabit Ethernet OK
0xED450000-0xED45FFFF Broadcom NetXtreme Gigabit Ethernet OK
0xED460000-0xED46FFFF Broadcom NetXtreme Gigabit Ethernet #2 OK
0xED470000-0xED47FFFF Broadcom NetXtreme Gigabit Ethernet #2 OK
0xED000000-0xED1FFFFF PCI bus OK
0xED000000-0xED1FFFFF PCI standard PCI-to-PCI bridge OK
0xED100000-0xED1FFFFF PCI standard PCI-to-PCI bridge OK
0xED108000-0xED108FFF IBM ServeRAID 6M Controller OK
0xED008000-0xED008FFF IBM ServeRAID 6M Controller OK
0xF8A00000-0xF8AFFFFF PCI bus OK
0xF8A00000-0xF8AFFFFF PCI standard PCI-to-PCI bridge OK
0xF8A08000-0xF8A08FFF IBM ServeRAID 6M Controller OK
0xF8B00000-0xF8BFFFFF PCI bus OK
0xF8B00000-0xF8BFFFFF PCI standard PCI-to-PCI bridge OK
0xF8B08000-0xF8B08FFF IBM ServeRAID 6M Controller OK
0xF8000000-0xF87FFFFF PCI bus OK
0xF9000000-0xF93FFFFF PCI bus OK
0xF8400000-0xF84FFFFF PCI standard PCI-to-PCI bridge OK
0xF8408000-0xF8408FFF IBM ServeRAID 6M Controller OK
0xEA000000-0xEA7FFFFF PCI bus OK
0xF2800000-0xF2BFFFFF PCI bus OK
0xEA400000-0xEA4FFFFF PCI standard PCI-to-PCI bridge OK
0xEA408000-0xEA408FFF IBM ServeRAID 6M Controller OK
0xEA800000-0xEAFFFFFF PCI bus OK
0xF0000000-0xF07FFFFF PCI bus OK
0xEB000000-0xEB7FFFFF PCI bus OK
0xF0800000-0xF0FFFFFF PCI bus OK
0xEB800000-0xEBFFFFFF PCI bus OK
0xF1000000-0xF17FFFFF PCI bus OK
0xEC000000-0xEC7FFFFF PCI bus OK
0xF1800000-0xF1FFFFFF PCI bus OK
0xEC800000-0xECFFFFFF PCI bus OK
0xF2000000-0xF27FFFFF PCI bus OK

[Components]

[Multimedia]

[Audio Codecs]

CODEC Manufacturer Description Status
File Version Size Creation Date
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) 3/9/2003 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) 3/9/2003 7:00 AM
c:\windows\system32\msg723.acm Microsoft Corporation
OK C:\WINDOWS\system32\MSG723.ACM
4.4.4000 116.00 KB (118,784 bytes) 4/29/2003 4:27 PM
c:\windows\system32\msgsm32.acm Microsoft Corporation
OK C:\WINDOWS\system32\MSGSM32.ACM
5.2.3785.0 (srv03_rtm.030308-1736) 20.50 KB (20,992 bytes)
3/9/2003 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) 3/9/2003 7:00 AM
c:\windows\system32\msadp32.acm Microsoft Corporation
OK C:\WINDOWS\system32\MSADP32.ACM
5.2.3785.0 (srv03_rtm.030308-1736) 14.50 KB (14,848 bytes)
3/9/2003 7:00 AM
c:\windows\system32\msg711.acm Microsoft Corporation
OK C:\WINDOWS\system32\MSG711.ACM
5.2.3785.0 (srv03_rtm.030308-1736) 10.00 KB (10,240 bytes)
3/9/2003 7:00 AM
c:\windows\system32\imaadp32.acm Microsoft Corporation
OK C:\WINDOWS\system32\IMAADP32.ACM
5.2.3785.0 (srv03_rtm.030308-1736) 15.50 KB (15,872 bytes)
3/9/2003 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) 3/9/2003 7:00 AM

[Video Codecs]

CODEC Manufacturer Description Status
File Version Size Creation Date
c:\windows\system32\msh261.drv Microsoft Corporation
OK C:\WINDOWS\system32\MSH261.DRV
4.4.4000 180.00 KB (184,320 bytes) 4/29/2003 4:27 PM
c:\windows\system32\msrle32.dll Microsoft Corporation
OK C:\WINDOWS\system32\MSRLE32.DLL 5.2.3785.0
(srv03_rtm.030308-1736) 10.50 KB (10,752 bytes)
3/9/2003 7:00 AM
c:\windows\system32\msyuv.dll Microsoft Corporation
OK C:\WINDOWS\system32\MSYUV.DLL 5.2.3785.0
(srv03_rtm.030308-1736) 16.50 KB (16,896 bytes)
3/8/2003 6:00 PM
c:\windows\system32\msvidc32.dll Microsoft Corporation
OK C:\WINDOWS\system32\MSVIDC32.DLL
5.2.3785.0 (srv03_rtm.030308-1736) 26.50 KB (27,136 bytes)
3/9/2003 7:00 AM
c:\windows\system32\msh263.drv Microsoft Corporation
OK C:\WINDOWS\system32\MSH263.DRV
4.4.4000 284.00 KB (290,816 bytes) 3/8/2003 5:57 PM
c:\windows\system32\tsbyuv.dll Microsoft Corporation
OK C:\WINDOWS\system32\TSBYUV.DLL 5.2.3785.0
(srv03_rtm.030308-1736) 8.00 KB (8,192 bytes) 3/8/2003 6:01 PM
c:\windows\system32\iyuv_32.dll Microsoft Corporation
OK C:\WINDOWS\system32\IYUV_32.DLL
5.2.3785.0 (srv03_rtm.030308-1736) 45.00 KB (46,080 bytes)
3/8/2003 6:00 PM

[CD-ROM]

Item Value
Drive Q:
Description CD-ROM Drive
Media Loaded No
Media Type CD-ROM
Name HL-DT-ST DVD-ROM GDR8081N
Manufacturer (Standard CD-ROM drives)
Status OK
Transfer Rate Not Available
SCSI Target ID 0

PNP Device ID
IDE\CDROMHL-DT-ST_DVD-ROM_GDR8081N_____________
0012____\5&CBC355F&0&0.0.0
Driver c:\windows\system32\drivers\cdrom.sys (5.2.3785.0
(srv03_rtm.030308-1736), 49.50 KB (50,688 bytes), 3/9/2003 7:00
AM)

[Sound Device]

Item Value

[Display]

Item Value
Name RAGE XL PCI Family (Microsoft Corporation)
PNP Device ID
PCI\VEN_1002&DEV_4752&SUBSYS_02A61014&REV_27\3&267
A616A&0&18
Adapter Type ATI RAGE XL PCI (B41), ATI Technologies
Inc. compatible
Adapter Description RAGE XL PCI Family (Microsoft Corporation)
Adapter RAM 8.00 MB (8,388,608 bytes)
Installed Drivers ati2drad.dll
Driver Version 5.10.3663.6013
INF File atiixpad.inf (ati2mpad section)
Color Planes 1
Color Table Entries 4294967296
Resolution 800 x 600 x 60 hertz
Bits/Pixel 32
Memory Address 0xE8000000-0xE9FFFFFF
I/O Port 0x00001800-0x000018FF
Memory Address 0xE9200000-0xE9200FFF
IRQ Channel IRQ 39
I/O Port 0x000003B0-0x000003BB
I/O Port 0x000003C0-0x000003DF
Memory Address 0xA0000-0xA7FFF
Driver c:\windows\system32\drivers\ati2mpad.sys
(5.10.3663.6013, 335.38 KB (343,424 bytes), 4/29/2003 12:11 PM)

[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&7FD7688&0
Number of Function Keys 12
IRQ Channel IRQ 1
I/O Port 0x00000064-0x00000064
I/O Port 0x00000060-0x00000060
Driver c:\windows\system32\drivers\i8042prt.sys (5.2.3785.0
(srv03_rtm.030308-1736), 69.00 KB (70,656 bytes), 3/9/2003 7:00
AM)

[Pointing Device]

Item Value
Hardware Type PS/2 Compatible Mouse
Number of Buttons 3
Status OK
PNP Device ID ACPI\PNP0F13\4&7FD7688&0
Power Management Supported No
Double Click Threshold 6
Handedness Right Handed Operation
IRQ Channel IRQ 12
Driver c:\windows\system32\drivers\i8042prt.sys (5.2.3785.0
(srv03_rtm.030308-1736), 69.00 KB (70,656 bytes), 3/9/2003 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 6/17/2003 12:55 PM
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 6/17/2003 12:55 PM
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 (5.2.3785.0
(srv03_rtm.030308-1736), 77.00 KB (78,848 bytes), 3/9/2003 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 6/17/2003 12:55 PM
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\raspptp.sys (5.2.3785.0 (srv03_rtm.030308-1736), 70.50 KB (72,192 bytes), 3/9/2003 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 6/17/2003 12:55 PM
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 (5.2.3785.0 (srv03_rtm.030308-1736), 38.00 KB (38,912 bytes), 3/9/2003 7:00 AM)

Name [00000005] Direct Parallel
Adapter Type Not Available
Product Type Direct Parallel
Installed Yes
PNP Device ID ROOT\MS_PTIMINIPORT\0000
Last Reset 6/17/2003 12:55 PM
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 (5.2.3785.0 (srv03_rtm.030308-1736), 18.50 KB (18,944 bytes), 3/9/2003 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 6/17/2003 12:55 PM
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 (5.2.3785.0 (srv03_rtm.030308-1736), 100.00 KB (102,400 bytes), 3/9/2003 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_02A61014&REV_02\3&13C0B0C5&0&20
Last Reset 6/17/2003 12:55 PM
Index 7
Service Name b57w2k
IP Address 192.168.200.100
IP Subnet 255.255.255.0
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:E6:02:AD
Memory Address 0xED440000-0xED44FFFF
Memory Address 0xED450000-0xED45FFFF
IRQ Channel IRQ 42
Driver c:\windows\system32\drivers\b57xp32.sys (6.34.0.0 built by: WinDDK, 166.88 KB (170,880 bytes), 2/17/2003 11:22 AM)

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_02A61014&REV_02\3&13C0B0C5&0&21
Last Reset 6/17/2003 12:55 PM
Index 8
Service Name b57w2k
IP Address 9.27.250.76
IP Subnet 255.255.255.128
Default IP Gateway 9.27.250.1
DHCP Enabled Yes
DHCP Server 9.44.5.76
DHCP Lease Expires 6/18/2003 12:57 AM
DHCP Lease Obtained 6/17/2003 12:57 PM
MAC Address 00:09:6B:16:02:AD
Memory Address 0xED460000-0xED46FFFF
Memory Address 0xED470000-0xED47FFFF
IRQ Channel IRQ 11
Driver c:\windows\system32\drivers\b57xp32.sys (6.34.0.0 built by: WinDDK, 166.88 KB (170,880 bytes), 2/17/2003 11:22 AM)

[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_{D0A36C1F-F005-4E54-843C-913F541E0E43}] 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_{D0A36C1F-F005-4E54-843C-913F541E0E43}] 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_{16DA7709-B2E3-468A-9606-027A6535DE58}] 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_{16DA7709-B2E3-468A-9606-027A6535DE58}] 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_{3713A903-F1F1-4B78-99ED-7FFBF8963AFC}] 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_{3713A903-F1F1-4B78-99ED-7FFBF8963AFC}] 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_{6C9CA6BD-CADB-427A-8D76-62EADCA31581}] 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_{6C9CA6BD-CADB-427A-8D76-62EADCA31581}] 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 5.2.3785.0 (srv03_rtm.030308-1736)

[Ports]

[Serial]

Item Value

[Parallel]

Item Value

[Storage]

[Drives]

Item Value

[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 8
SCSI Target ID 0

Sectors/Track 63
Size 101.70 GB (109,198,817,280 bytes)
Total Cylinders 13,276
Total Sectors 213,278,940
Total Tracks 3,385,380
Tracks/Cylinder 255
Partition Disk #21, Partition #0
Partition Size 101.70 GB (109,198,785,024 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 1
Sectors/Track 63
Size 101.70 GB (109,198,817,280 bytes)
Total Cylinders 13,276
Total Sectors 213,278,940
Total Tracks 3,385,380
Tracks/Cylinder 255
Partition Disk #22, Partition #0
Partition Size 101.70 GB (109,198,785,024 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 2
Sectors/Track 63
Size 16.94 GB (18,194,319,360 bytes)
Total Cylinders 2,212
Total Sectors 35,535,780
Total Tracks 564,060
Tracks/Cylinder 255
Partition Disk #23, Partition #0
Partition Size 16.94 GB (18,194,287,104 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 63
Size 101.70 GB (109,198,817,280 bytes)
Total Cylinders 13,276
Total Sectors 213,278,940
Total Tracks 3,385,380
Tracks/Cylinder 255
Partition Disk #24, Partition #0
Partition Size 101.70 GB (109,198,785,024 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 4
Sectors/Track 63
Size 101.70 GB (109,198,817,280 bytes)
Total Cylinders 13,276
Total Sectors 213,278,940
Total Tracks 3,385,380
Tracks/Cylinder 255
Partition Disk #25, Partition #0
Partition Size 101.70 GB (109,198,785,024 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 5
Sectors/Track 63
Size 16.94 GB (18,194,319,360 bytes)
Total Cylinders 2,212
Total Sectors 35,535,780
Total Tracks 564,060
Tracks/Cylinder 255
Partition Disk #26, Partition #0
Partition Size 16.94 GB (18,194,287,104 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 6
SCSI Target ID 0
Sectors/Track 63
Size 101.70 GB (109,198,817,280 bytes)
Total Cylinders 13,276
Total Sectors 213,278,940
Total Tracks 3,385,380
Tracks/Cylinder 255
Partition Disk #14, Partition #0
Partition Size 101.70 GB (109,198,785,024 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 6
SCSI Target ID 1
Sectors/Track 63
Size 101.70 GB (109,198,817,280 bytes)
Total Cylinders 13,276
Total Sectors 213,278,940
Total Tracks 3,385,380
Tracks/Cylinder 255
Partition Disk #15, Partition #0
Partition Size 101.70 GB (109,198,785,024 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 6
SCSI Target ID 2
Sectors/Track 63
Size 16.94 GB (18,194,319,360 bytes)
Total Cylinders 2,212
Total Sectors 35,535,780
Total Tracks 564,060
Tracks/Cylinder 255
Partition Disk #16, Partition #0
Partition Size 16.94 GB (18,194,287,104 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 6
SCSI Target ID 3
Sectors/Track 63
Size 101.70 GB (109,198,817,280 bytes)
Total Cylinders 13,276
Total Sectors 213,278,940
Total Tracks 3,385,380
Tracks/Cylinder 255
Partition Disk #17, Partition #0
Partition Size 101.70 GB (109,198,785,024 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 6
SCSI Target ID 4
Sectors/Track 63
Size 101.70 GB (109,198,817,280 bytes)
Total Cylinders 13,276
Total Sectors 213,278,940
Total Tracks 3,385,380
Tracks/Cylinder 255
Partition Disk #18, Partition #0
Partition Size 101.70 GB (109,198,785,024 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 6
SCSI Target ID 5
Sectors/Track 63
Size 16.94 GB (18,194,319,360 bytes)
Total Cylinders 2,212
Total Sectors 35,535,780
Total Tracks 564,060
Tracks/Cylinder 255
Partition Disk #19, Partition #0
Partition Size 16.94 GB (18,194,287,104 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 9
SCSI Target ID 0
Sectors/Track 63
Size 101.70 GB (109,198,817,280 bytes)
Total Cylinders 13,276
Total Sectors 213,278,940
Total Tracks 3,385,380
Tracks/Cylinder 255
Partition Disk #27, Partition #0
Partition Size 101.70 GB (109,198,785,024 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 9

SCSI Target ID 1
Sectors/Track 63
Size 101.70 GB (109,198,817,280 bytes)
Total Cylinders 13,276
Total Sectors 213,278,940
Total Tracks 3,385,380
Tracks/Cylinder 255
Partition Disk #28, Partition #0
Partition Size 101.70 GB (109,198,785,024 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 9
SCSI Target ID 2
Sectors/Track 63
Size 16.94 GB (18,194,319,360 bytes)
Total Cylinders 2,212
Total Sectors 35,535,780
Total Tracks 564,060
Tracks/Cylinder 255
Partition Disk #29, Partition #0
Partition Size 16.94 GB (18,194,287,104 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 9
SCSI Target ID 3
Sectors/Track 63
Size 101.70 GB (109,198,817,280 bytes)
Total Cylinders 13,276
Total Sectors 213,278,940
Total Tracks 3,385,380
Tracks/Cylinder 255
Partition Disk #30, Partition #0
Partition Size 101.70 GB (109,198,785,024 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 9
SCSI Target ID 4
Sectors/Track 63
Size 101.70 GB (109,198,817,280 bytes)
Total Cylinders 13,276
Total Sectors 213,278,940
Total Tracks 3,385,380
Tracks/Cylinder 255
Partition Disk #31, Partition #0
Partition Size 101.70 GB (109,198,785,024 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 9
SCSI Target ID 5
Sectors/Track 63
Size 16.94 GB (18,194,319,360 bytes)
Total Cylinders 2,212
Total Sectors 35,535,780
Total Tracks 564,060
Tracks/Cylinder 255
Partition Disk #32, Partition #0
Partition Size 16.94 GB (18,194,287,104 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 4
SCSI Target ID 0
Sectors/Track 63
Size 101.70 GB (109,198,817,280 bytes)
Total Cylinders 13,276
Total Sectors 213,278,940
Total Tracks 3,385,380
Tracks/Cylinder 255
Partition Disk #2, Partition #0
Partition Size 101.70 GB (109,198,785,024 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 4
SCSI Target ID 1
Sectors/Track 63
Size 101.70 GB (109,198,817,280 bytes)
Total Cylinders 13,276
Total Sectors 213,278,940
Total Tracks 3,385,380
Tracks/Cylinder 255
Partition Disk #3, Partition #0
Partition Size 101.70 GB (109,198,785,024 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 4
SCSI Target ID 2
Sectors/Track 63
Size 16.94 GB (18,194,319,360 bytes)
Total Cylinders 2,212
Total Sectors 35,535,780
Total Tracks 564,060
Tracks/Cylinder 255
Partition Disk #4, Partition #0
Partition Size 16.94 GB (18,194,287,104 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 4
SCSI Target ID 3
Sectors/Track 63
Size 101.70 GB (109,198,817,280 bytes)
Total Cylinders 13,276
Total Sectors 213,278,940
Total Tracks 3,385,380
Tracks/Cylinder 255
Partition Disk #5, Partition #0
Partition Size 101.70 GB (109,198,785,024 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 4
SCSI Target ID 4
Sectors/Track 63
Size 101.70 GB (109,198,817,280 bytes)
Total Cylinders 13,276
Total Sectors 213,278,940
Total Tracks 3,385,380
Tracks/Cylinder 255
Partition Disk #6, Partition #0
Partition Size 101.70 GB (109,198,785,024 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 4
SCSI Target ID 5
Sectors/Track 63
Size 16.94 GB (18,194,319,360 bytes)
Total Cylinders 2,212
Total Sectors 35,535,780
Total Tracks 564,060
Tracks/Cylinder 255
Partition Disk #7, Partition #0
Partition Size 16.94 GB (18,194,287,104 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 5
SCSI Target ID 0
Sectors/Track 63
Size 101.70 GB (109,198,817,280 bytes)
Total Cylinders 13,276
Total Sectors 213,278,940
Total Tracks 3,385,380
Tracks/Cylinder 255
Partition Disk #8, Partition #0
Partition Size 101.70 GB (109,198,785,024 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 5
SCSI Target ID 1
Sectors/Track 63
Size 101.70 GB (109,198,817,280 bytes)
Total Cylinders 13,276
Total Sectors 213,278,940
Total Tracks 3,385,380
Tracks/Cylinder 255
Partition Disk #9, Partition #0
Partition Size 101.70 GB (109,198,785,024 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 5
SCSI Target ID 2
Sectors/Track 63
Size 16.94 GB (18,194,319,360 bytes)
Total Cylinders 2,212
Total Sectors 35,535,780
Total Tracks 564,060
Tracks/Cylinder 255
Partition Disk #10, Partition #0
Partition Size 16.94 GB (18,194,287,104 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 5
SCSI Target ID 3
Sectors/Track 63
Size 101.70 GB (109,198,817,280 bytes)
Total Cylinders 13,276
Total Sectors 213,278,940
Total Tracks 3,385,380
Tracks/Cylinder 255
Partition Disk #11, Partition #0
Partition Size 101.70 GB (109,198,785,024 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 5
SCSI Target ID 4
Sectors/Track 63
Size 101.70 GB (109,198,817,280 bytes)
Total Cylinders 13,276
Total Sectors 213,278,940
Total Tracks 3,385,380
Tracks/Cylinder 255
Partition Disk #12, Partition #0
Partition Size 101.70 GB (109,198,785,024 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 5
SCSI Target ID 5
Sectors/Track 63
Size 16.94 GB (18,194,319,360 bytes)
Total Cylinders 2,212
Total Sectors 35,535,780
Total Tracks 564,060
Tracks/Cylinder 255
Partition Disk #13, Partition #0
Partition Size 16.94 GB (18,194,287,104 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 7
SCSI Target ID 0
Sectors/Track 63
Size 220.35 GB (236,600,179,200 bytes)
Total Cylinders 28,765
Total Sectors 462,109,725
Total Tracks 7,335,075
Tracks/Cylinder 255
Partition Disk #20, Partition #0
Partition Size 220.35 GB (236,600,146,944 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 2
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 2
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 PCI-X Ultra320 SCSI Host Adapter
Manufacturer LSI Logic Inc.
Status OK
PNP Device ID
PCI\VEN_1000&DEV_0030&SUBSYS_02921014&REV_07\3&13C0B0C5&0&18
I/O Port 0x00002000-0x000027FF
Memory Address 0xED400000-0xED4FFFFF
Memory Address 0xED410000-0xED41FFFF
IRQ Channel IRQ 40
Driver c:\windows\system32\drivers\symmpi.sys (1.08.18.00 (NT.021001-2000), 25.88 KB (26,496 bytes), 3/9/2003 7:00 AM)

Name LSI Logic PCI-X Ultra320 SCSI Host Adapter
Manufacturer LSI Logic Inc.
Status OK
PNP Device ID
PCI\VEN_1000&DEV_0030&SUBSYS_02921014&REV_07\3&13C0B0C5&0&19
I/O Port 0x00002100-0x000021FF
Memory Address 0xED420000-0xED42FFFF
Memory Address 0xED430000-0xED43FFFF
IRQ Channel IRQ 41
Driver c:\windows\system32\drivers\symmpi.sys (1.08.18.00 (NT.021001-2000), 25.88 KB (26,496 bytes), 3/9/2003 7:00 AM)

Name IBM ServeRAID 6M Controller
Manufacturer IBM Corporation
Status OK
PNP Device ID
PCI\VEN_9005&DEV_0250&SUBSYS_02791014&REV_00\4&1C91D199&0&4008
Memory Address 0xED108000-0xED108FFF
IRQ Channel IRQ 51
Driver c:\windows\system32\drivers\nfrd6m.sys (6.09.02 built by: WinDDK, 28.50 KB (29,184 bytes), 1/14/2003 2:59 PM)

Name IBM ServeRAID 6M Controller
Manufacturer IBM Corporation
Status OK
PNP Device ID
PCI\VEN_9005&DEV_0250&SUBSYS_02791014&REV_00\4&392398EF&0&4010
Memory Address 0xED008000-0xED008FFF
IRQ Channel IRQ 55
Driver c:\windows\system32\drivers\nfrd6m.sys (6.09.02 built by: WinDDK, 28.50 KB (29,184 bytes), 1/14/2003 2:59 PM)

Name IBM ServeRAID 6M Controller
Manufacturer IBM Corporation
Status OK
PNP Device ID
PCI\VEN_9005&DEV_0250&SUBSYS_02791014&REV_00\4&180CD7A4&0&4020
Memory Address 0xF8A08000-0xF8A08FFF
IRQ Channel IRQ 71
Driver c:\windows\system32\drivers\nfrd6m.sys (6.09.02 built by: WinDDK, 28.50 KB (29,184 bytes), 1/14/2003 2:59 PM)

Name IBM ServeRAID 6M Controller
Manufacturer IBM Corporation
Status OK
PNP Device ID
PCI\VEN_9005&DEV_0250&SUBSYS_02791014&REV_00\4&14F365D1&0&4018
Memory Address 0xF8B08000-0xF8B08FFF
IRQ Channel IRQ 67
Driver c:\windows\system32\drivers\nfrd6m.sys (6.09.02 built by: WinDDK, 28.50 KB (29,184 bytes), 1/14/2003 2:59 PM)

Name IBM ServeRAID 6M Controller
Manufacturer IBM Corporation
Status OK
PNP Device ID
PCI\VEN_9005&DEV_0250&SUBSYS_02791014&REV_00\4&8C37DC5&0&4010
Memory Address 0xF8408000-0xF8408FFF
IRQ Channel IRQ 63
Driver c:\windows\system32\drivers\nfrd6m.sys (6.09.02 built by: WinDDK, 28.50 KB (29,184 bytes), 1/14/2003 2:59 PM)

Name IBM ServeRAID 6M Controller
Manufacturer IBM Corporation
Status OK
PNP Device ID
PCI\VEN_9005&DEV_0250&SUBSYS_02791014&REV_02\4&1CCD9D43&0&4008
Memory Address 0xEA408000-0xEA408FFF
IRQ Channel IRQ 116
Driver c:\windows\system32\drivers\nfrd6m.sys (6.09.02 built by: WinDDK, 28.50 KB (29,184 bytes), 1/14/2003 2:59 PM)

[IDE]

Item Value
Name VIA Bus Master IDE Controller
Manufacturer VIA Technologies, Inc.
Status OK
PNP Device ID
PCI\VEN_1106&DEV_0571&SUBSYS_02A61014&REV_06\3&267A616A&0&29
I/O Port 0x00000700-0x0000070F
Driver c:\windows\system32\drivers\viaide.sys (1.00.01.00, 7.00 KB (7,168 bytes), 3/9/2003 7:00 AM)

Name Primary IDE Channel
Manufacturer (Standard IDE ATA/ATAPI controllers)
Status OK
PNP Device ID PCIIDE\IDECHANNEL\4&29582549&0&0
I/O Port 0x000001F0-0x000001F7
I/O Port 0x000003F6-0x000003F6
IRQ Channel IRQ 14
Driver c:\windows\system32\drivers\atapi.sys (5.2.3785.0 (srv03_rtm.030308-1736), 89.00 KB (91,136 bytes), 3/9/2003 7:00 AM)

Name Secondary IDE Channel
Manufacturer (Standard IDE ATA/ATAPI controllers)
Status OK
PNP Device ID PCIIDE\IDECHANNEL\4&29582549&0&1
I/O Port 0x00000170-0x00000177
I/O Port 0x00000376-0x00000376
IRQ Channel IRQ 15
Driver c:\windows\system32\drivers\atapi.sys (5.2.3785.0 (srv03_rtm.030308-1736), 89.00 KB (91,136 bytes), 3/9/2003 7:00 AM)

[Printing]

Name Driver Port Name Server Name

[Problem Devices]

Device PNP Device ID Error Code
Not Available ACPI\IBM37D4\2&DABA3FF&0
The drivers for this device are not installed.
Other PCI Bridge Device
PCI\VEN_1014&DEV_010F&SUBSYS_01131014&REV_00\3&267
A616A&0&20 The drivers for this device are not installed.

[USB]

Device PNP Device ID
VIA Rev 5 or later USB Universal Host Controller
PCI\VEN_1106&DEV_3038&SUBSYS_02A61014&REV_16\3&267
A616A&0&2A
USB Root Hub USB\ROOT_HUB\4&226DFD17&0
VIA Rev 5 or later USB Universal Host Controller
PCI\VEN_1106&DEV_3038&SUBSYS_02A61014&REV_16\3&267
A616A&0&2B
USB Root Hub USB\ROOT_HUB\4&3527ADBC&0

[Software Environment]

[System Drivers]

Name Description File Type Started
Start ModeState 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
asyncmac RAS Asynchronous Media Driver
c:\windows\system32\drivers\asyncmac.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
ati2mpad ati2mpad c:\windows\system32\drivers\ati2mpad.sys
Kernel Driver Yes Manual Running OK
Ignore No Yes
atmarpc ATM ARP Client Protocol
c:\windows\system32\drivers\atmarpc.sys Kernel Driver No
Manual Stopped OK Normal No No
audstub Audio Stub Driver
c:\windows\system32\drivers\audstub.sys Kernel Driver
Yes Manual Running OK Normal No
Yes
b57w2k Broadcom NetXtreme Gigabit Ethernet
c:\windows\system32\drivers\b57xp32.sys Kernel Driver
Yes Manual Running OK Normal No
Yes
beep Beep c:\windows\system32\drivers\beep.sys
Kernel Driver Yes System Running OK
Normal No Yes
cbidf2k cbidf2k c:\windows\system32\drivers\cbidf2k.sys
Kernel Driver No Disabled Stopped OK
Normal No No
cd20xrnt cd20xrnt Not Available Kernel Driver No
Disabled Stopped OK Normal No No
cdfs Cdfs c:\windows\system32\drivers\cdfs.sys
File System Driver Yes Disabled Running OK
Normal No Yes
cdrom CD-ROM Driver
c:\windows\system32\drivers\cdrom.sys Kernel Driver
Yes System Running OK Normal No
Yes
changer Changer Not Available Kernel Driver No
System Stopped OK Ignore No No
clusdisk Cluster Disk Driver
c:\windows\system32\drivers\clusdisk.sys Kernel Driver No
Disabled Stopped OK Normal No No
cmdide CmdIde Not Available Kernel Driver No
Disabled Stopped OK Normal No No
cpqarray Cpqarray Not Available Kernel Driver No
Disabled Stopped OK Normal No No
cpqarry2 cpqarry2 Not Available Kernel Driver No
Disabled Stopped OK Normal No No
cpqcissm cpqcissm Not Available Kernel Driver No
Disabled Stopped OK Normal No No
cpqfcalm cpqfcalm Not Available Kernel Driver No
Disabled Stopped OK Normal No No
crcdisk CRC Disk Filter Driver
c:\windows\system32\drivers\crcdisk.sys Kernel Driver
Yes Boot Running OK Normal No
Yes
dac960nt dac960nt Not Available Kernel Driver No
Disabled Stopped OK Normal No No
dellcerc dellcerc Not Available Kernel Driver No
Disabled Stopped OK Normal No No
dfsdriver DfsDriver c:\windows\system32\drivers\dfs.sys
File System Driver Yes Boot Running OK
Normal No Yes
disk Disk Driver
c:\windows\system32\drivers\disk.sys Kernel Driver
Yes Boot Running OK Normal No
Yes
dmboot dmboot c:\windows\system32\drivers\dmboot.sys
Kernel Driver Yes Disabled Running OK
Normal No Yes
dmio Logical Disk Manager Driver
c:\windows\system32\drivers\dmio.sys Kernel Driver
Yes Boot Running OK Normal No
Yes

```
dmload dmload c:\windows\system32\drivers\dmload.sys
Kernel Driver Yes Boot Running OK
Normal No Yes
dpti2o dpti2o Not Available Kernel Driver No
Disabled Stopped OK Normal No No
fastfat Fastfat c:\windows\system32\drivers\fastfat.sys
File System Driver No Disabled Stopped OK
Normal No No
fdc Floppy Disk Controller Driver
c:\windows\system32\drivers\fdc.sys Kernel Driver
Yes Manual Running OK Normal No
Yes
fips Fips c:\windows\system32\drivers\fips.sys
Kernel Driver Yes System Running OK
Normal No Yes
flpydisk Floppy Disk Driver
c:\windows\system32\drivers\flpydisk.sys Kernel Driver
Yes Manual Running OK Normal No
Yes
ftdisk Volume Manager Driver
c:\windows\system32\drivers\ftdisk.sys Kernel Driver
Yes Boot Running OK Normal No
Yes
gpc Generic Packet Classifier
c:\windows\system32\drivers\msgpc.sys Kernel Driver
Yes Manual Running OK Normal No
Yes
hpn hpn Not Available Kernel Driver No
Disabled Stopped OK Normal No No
hpt3xx hpt3xx Not Available Kernel Driver No
Disabled Stopped OK Normal No No
http HTTP c:\windows\system32\drivers\http.sys
Kernel Driver No Manual Stopped OK
Normal No No
i2omgmt i2omgmt Not Available Kernel Driver No
System Stopped OK Normal No No
i2omp i2omp Not Available Kernel Driver No
Disabled Stopped OK Normal No No
i8042prt i8042 Keyboard and PS/2 Mouse Port Driver
c:\windows\system32\drivers\i8042prt.sys Kernel Driver
Yes System Running OK Normal No
Yes
iirsp iirsp Not Available Kernel Driver No
Disabled Stopped OK Normal No No
imapi CD-Burning Filter Driver
c:\windows\system32\drivers\imapi.sys Kernel Driver No
System Stopped OK Normal No No
intelide IntelIde Not Available Kernel Driver No
Disabled Stopped OK Normal No No
ipfilterdriver IP Traffic Filter Driver
c:\windows\system32\drivers\ipfltdrv.sys Kernel Driver No
Manual Stopped OK Normal No No
ipinip IP in IP Tunnel Driver
c:\windows\system32\drivers\ipinip.sys Kernel Driver No
Manual Stopped OK Normal No No
ipnat IP Network Address Translator
c:\windows\system32\drivers\ipnat.sys Kernel Driver No
Manual Stopped OK Normal No No
ipsec IPSEC driver
c:\windows\system32\drivers\ipsec.sys Kernel Driver
Yes System Running OK Normal No
Yes
ipsraidn ipsraidn Not Available Kernel Driver No
Disabled Stopped OK Normal No No
isapnp PnP ISA/EISA Bus Driver
c:\windows\system32\drivers\isapnp.sys Kernel Driver
Yes Boot Running OK Critical No
Yes
kbdclass Keyboard Class Driver
c:\windows\system32\drivers\kbdclass.sys Kernel Driver
Yes System Running OK Normal No
Yes
ksecdd KSecDD c:\windows\system32\drivers\ksecdd.sys
Kernel Driver Yes Boot Running OK
Normal No Yes
lp6nds35 lp6nds35 Not Available Kernel Driver No
Disabled Stopped OK Normal No No
mnmdd mnmdd c:\windows\system32\drivers\mnmdd.sys
Kernel Driver Yes System Running OK
Ignore No Yes
modem Modem c:\windows\system32\drivers\modem.sys
Kernel Driver No Manual Stopped OK
Ignore No No
mouclass Mouse Class Driver
c:\windows\system32\drivers\mouclass.sys Kernel Driver
Yes System Running OK Normal No
Yes
mountmgr Mount Point Manager
c:\windows\system32\drivers\mountmgr.sys Kernel Driver
Yes Boot Running OK Normal No
Yes
mraid35x mraid35x Not Available Kernel Driver No
Disabled Stopped OK Normal No No
mrxdav WebDav Client Redirector
c:\windows\system32\drivers\mrxdav.sys File System Driver No
Manual Stopped OK Normal No No
mrxsmb MRXSMB c:\windows\system32\drivers\mrxsmb.sys
File System Driver Yes System Running OK
Normal No Yes
msfs Msfs c:\windows\system32\drivers\msfs.sys
File System Driver Yes System Running OK
Normal No Yes
mup Mup c:\windows\system32\drivers\mup.sys
File System Driver Yes Boot Running OK
Normal No Yes
ndis NDIS System Driver
c:\windows\system32\drivers\ndis.sys Kernel Driver
Yes Boot Running OK Normal No
Yes
ndistapi Remote Access NDIS TAPI Driver
c:\windows\system32\drivers\ndistapi.sys Kernel Driver
Yes Manual Running OK Normal No
Yes
ndisuio NDIS Usermode I/O Protocol
c:\windows\system32\drivers\ndisuio.sys Kernel Driver No
Manual Stopped OK Normal No No
ndiswan Remote Access NDIS WAN Driver
c:\windows\system32\drivers\ndiswan.sys Kernel Driver
Yes Manual Running OK Normal No
Yes
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
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  IBM ServeRAID 4M/4Mx/4L/4Lx/5i/6M/6i Device Driver
c:\windows\system32\drivers\nfrd960.sys  Kernel Driver
Yes  Boot  Running  OK  Normal  No
Yes
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
numaqdrv  numaqdrv  \??\d:\goodwin\numaqdrv.sys  Kernel Driver
No  Manual  Stopped  OK  Normal  No  No
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  Not Available  Kernel Driver  No
Disabled  Stopped  OK  Normal  No  No
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
pdreli  PDRELI  Not Available  Kernel Driver  No
Manual  Stopped  OK  Ignore  No  No
pdrframe  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\raspptp.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)
c:\windows\system32\drivers\rasl2tp.sys  Kernel Driver
Yes  Manual  Running  OK  Normal  No
Yes
raspppoe  Remote Access PPPOE Driver
c:\windows\system32\drivers\raspppoe.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  No
Manual  Stopped  OK  Normal  No  No
serial  Serial port driver
c:\windows\system32\drivers\serial.sys  Kernel Driver  No
System  Stopped  OK  Ignore  No  No
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
usbhub USB2 Enabled Hub
c:\windows\system32\drivers\usbhub.sys Kernel Driver
Yes Manual Running OK Normal No
Yes
usbstor USB Mass Storage Driver
c:\windows\system32\drivers\usbstor.sys Kernel Driver No
Manual Stopped OK Normal No No
usbuhci Microsoft USB Universal Host Controller Miniport Driver
c:\windows\system32\drivers\usbuhci.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 ViaIde c:\windows\system32\drivers\viaide.sys
Kernel Driver Yes Boot Running OK
Normal No Yes
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 HTREE\ROOT\0
ACPI Multiprocessor Node No COMPUTER
1.25.0.0 2/24/2002 (Node-based Computers) oem0.inf
Not Available ROOT\ACPI_HAL\0000
Microsoft ACPI-Compliant System No SYSTEM
5.2.3785.0 10/1/2002 Microsoft acpi.inf Not Available
ACPI_HAL\PNP0C08\0
Not Available Not Available Not Available
Not Available Not Available Not Available
Not Available Not Available
ACPI\IBM37D4\2&DABA3FF&0
PCI bus No SYSTEM 5.2.3785.0 10/1/2002 (Standard
system devices) machine.inf Not Available
ACPI\PNP0A03\0
PCI standard host CPU bridge No SYSTEM 5.2.3785.0
10/1/2002 (Standard system devices) machine.inf
Not Available
PCI\VEN_1014&DEV_0302&SUBSYS_00000000&REV_05\3&267
A616A&0&00
RAGE XL PCI Family (Microsoft Corporation) No
DISPLAY 5.10.2600.6014 8/8/2001 ATI Technologies Inc.
atiixpad.inf Not Available
PCI\VEN_1002&DEV_4752&SUBSYS_02A61014&REV_27\3&267
A616A&0&18
Default Monitor No MONITOR 5.1.2001.0
6/6/2001 (Standard monitor types) monitor.inf
Not Available
DISPLAY\DEFAULT_MONITOR\4&82045C7&0&80000000&00&
03
Other PCI Bridge Device Not Available UNKNOWN
Not Available Not Available Not Available
Not Available Not Available
PCI\VEN_1014&DEV_010F&SUBSYS_01131014&REV_00\3&267
A616A&0&20
VIA Tech PCI to ISA bridge No SYSTEM 5.2.3785.0
10/1/2002 VIA machine.inf Not Available
PCI\VEN_1106&DEV_0686&SUBSYS_00000000&REV_40\3&267
A616A&0&28
ISAPNP Read Data Port No SYSTEM 5.2.3785.0
10/1/2002 (Standard system devices) machine.inf
Not Available ISAPNP\READDATAPORT\0
Motherboard resources No SYSTEM 5.2.3785.0
10/1/2002 (Standard system devices) machine.inf
Not Available ACPI\PNP0C02\2
Standard 101/102-Key or Microsoft Natural PS/2 Keyboard No
KEYBOARD 5.2.3785.0 10/1/2002 (Standard keyboards)
keyboard.inf Not Available
ACPI\PNP0303\4&7FD7688&0
PS/2 Compatible Mouse No MOUSE 5.2.3785.0
10/1/2002 Microsoft msmouse.inf Not Available
ACPI\PNP0F13\4&7FD7688&0
Standard floppy disk controller No FDC 5.2.3785.0
10/1/2002 (Standard floppy disk controllers) fdc.inf
Not Available ACPI\PNP0700\4&7FD7688&0
Floppy disk drive No FLOPPYDISK 5.2.3785.0
10/1/2002 (Standard floppy disk drives) flpydisk.inf
Not Available
FDC\GENERIC_FLOPPY_DRIVE\5&17D92A40&0&0
Advanced programmable interrupt controller No SYSTEM
5.2.3785.0 10/1/2002 (Standard system devices) machine.inf
Not Available ACPI\PNP0003\4&7FD7688&0
```

Direct memory access controller No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available ACPI\PNP0200\4&7FD7688&0
System timer No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available ACPI\PNP0100\4&7FD7688&0
System CMOS/real time clock No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available ACPI\PNP0B00\4&7FD7688&0
System speaker No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available ACPI\PNP0800\4&7FD7688&0
Numeric data processor No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available ACPI\PNP0C04\4&7FD7688&0
Motherboard resources No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available ACPI\PNP0C02\3
VIA Bus Master IDE Controller No HDC 5.2.3785.0 10/1/2002 VIA Technologies, Inc. mshdc.inf Not Available PCI\VEN_1106&DEV_0571&SUBSYS_02A61014&REV_06\3&267A616A&0&29
Primary IDE Channel No HDC 5.2.3785.0 10/1/2002 (Standard IDE ATA/ATAPI controllers) mshdc.inf Not Available PCIIDE\IDECHANNEL\4&29582549&0&0
CD-ROM Drive No CDROM 5.2.3785.0 10/1/2002 (Standard CD-ROM drives) cdrom.inf Not Available IDE\CDROMHL-DT-ST_DVD-ROM_GDR8081N_____0012____\5&CBC355F&0&0.0.0
Secondary IDE Channel No HDC 5.2.3785.0 10/1/2002 (Standard IDE ATA/ATAPI controllers) mshdc.inf Not Available PCIIDE\IDECHANNEL\4&29582549&0&1
VIA Rev 5 or later USB Universal Host Controller No USB 5.2.3785.0 10/1/2002 VIA Technologies usbport.inf Not Available PCI\VEN_1106&DEV_3038&SUBSYS_02A61014&REV_16\3&267A616A&0&2A
USB Root Hub No USB 5.2.3785.0 10/1/2002 (Standard USB Host Controller) usbport.inf Not Available USB\ROOT_HUB\4&226DFD17&0
VIA Rev 5 or later USB Universal Host Controller No USB 5.2.3785.0 10/1/2002 VIA Technologies usbport.inf Not Available PCI\VEN_1106&DEV_3038&SUBSYS_02A61014&REV_16\3&267A616A&0&2B
USB Root Hub No USB 5.2.3785.0 10/1/2002 (Standard USB Host Controller) usbport.inf Not Available USB\ROOT_HUB\4&3527ADBC&0
VIA Tech Power Management controller No SYSTEM 5.2.3785.0 10/1/2002 VIA machine.inf Not Available PCI\VEN_1106&DEV_3057&SUBSYS_02A61014&REV_40\3&267A616A&0&2C
System board No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available ACPI\PNP0C01\1
ACPI Fixed Feature Button No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available ACPI\FIXEDBUTTON\2&DABA3FF&0
Processor No PROCESSOR 5.2.3785.0 10/1/2002 (Standard processor types) cpu.inf Not Available ACPI\GENUINEINTEL_-_X86_FAMILY_15_MODEL_2_0
Processor No PROCESSOR 5.2.3785.0 10/1/2002 (Standard processor types) cpu.inf Not Available ACPI\GENUINEINTEL_-_X86_FAMILY_15_MODEL_2_1
Processor No PROCESSOR 5.2.3785.0 10/1/2002 (Standard processor types) cpu.inf Not Available ACPI\GENUINEINTEL_-_X86_FAMILY_15_MODEL_2_2
Processor No PROCESSOR 5.2.3785.0 10/1/2002 (Standard processor types) cpu.inf Not Available ACPI\GENUINEINTEL_-_X86_FAMILY_15_MODEL_2_3
Processor No PROCESSOR 5.2.3785.0 10/1/2002 (Standard processor types) cpu.inf Not Available ACPI\GENUINEINTEL_-_X86_FAMILY_15_MODEL_2_4
Processor No PROCESSOR 5.2.3785.0 10/1/2002 (Standard processor types) cpu.inf Not Available ACPI\GENUINEINTEL_-_X86_FAMILY_15_MODEL_2_5
Processor No PROCESSOR 5.2.3785.0 10/1/2002 (Standard processor types) cpu.inf Not Available ACPI\GENUINEINTEL_-_X86_FAMILY_15_MODEL_2_6
Processor No PROCESSOR 5.2.3785.0 10/1/2002 (Standard processor types) cpu.inf Not Available ACPI\GENUINEINTEL_-_X86_FAMILY_15_MODEL_2_7
Processor No PROCESSOR 5.2.3785.0 10/1/2002 (Standard processor types) cpu.inf Not Available ACPI\GENUINEINTEL_-_X86_FAMILY_15_MODEL_2_8
Processor No PROCESSOR 5.2.3785.0 10/1/2002 (Standard processor types) cpu.inf Not Available ACPI\GENUINEINTEL_-_X86_FAMILY_15_MODEL_2_9
Processor No PROCESSOR 5.2.3785.0 10/1/2002 (Standard processor types) cpu.inf Not Available ACPI\GENUINEINTEL_-_X86_FAMILY_15_MODEL_2\10
Processor No PROCESSOR 5.2.3785.0 10/1/2002 (Standard processor types) cpu.inf Not Available ACPI\GENUINEINTEL_-_X86_FAMILY_15_MODEL_2\11
Processor No PROCESSOR 5.2.3785.0 10/1/2002 (Standard processor types) cpu.inf Not Available ACPI\GENUINEINTEL_-_X86_FAMILY_15_MODEL_2\12
Processor No PROCESSOR 5.2.3785.0 10/1/2002 (Standard processor types) cpu.inf Not Available ACPI\GENUINEINTEL_-_X86_FAMILY_15_MODEL_2\13
Processor No PROCESSOR 5.2.3785.0 10/1/2002 (Standard processor types) cpu.inf Not Available ACPI\GENUINEINTEL_-_X86_FAMILY_15_MODEL_2\14
Processor No PROCESSOR 5.2.3785.0 10/1/2002 (Standard processor types) cpu.inf Not Available ACPI\GENUINEINTEL_-_X86_FAMILY_15_MODEL_2\15
Memory Module No MEMORY 5.2.3785.0 10/1/2002 Microsoft memory.inf Not Available ACPI\PNP0C80\0
Memory Module No MEMORY 5.2.3785.0 10/1/2002 Microsoft memory.inf Not Available ACPI\PNP0C80\2
PCI bus No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available ACPI\PNP0A03\1
PCI standard host CPU bridge No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available PCI\VEN_1014&DEV_0302&SUBSYS_00000000&REV_05\3&13C0B0C5&0&00
LSI Logic PCI-X Ultra320 SCSI Host Adapter No SCSIADAPTER 5.2.3785.0 10/1/2002 LSI Logic Inc. pnpscsi.inf Not Available PCI\VEN_1000&DEV_0030&SUBSYS_02921014&REV_07\3&13C0B0C5&0&18
Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available SCSI\DISK&VEN_IBM-ESXS&PROD_ST336753LC____FN&REV_B855\4&3CF91A&0&000
Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available SCSI\DISK&VEN_IBM-ESXS&PROD_ST336753LC____FN&REV_B855\4&3CF91A&0&010
SCSI Processor Device No SYSTEM 5.2.3785.0 10/1/2002 IBM scsidev.inf Not Available

SCSI\PROCESSOR&VEN_IBM&PROD_25P3495A_S320__1&REV_1\4&3CF91A&0&080
IBM Dummy Device No SYSTEM 5.2.3785.0 10/1/2002 IBM scsidev.inf Not Available
SCSI\BRIDGE&VEN_IBM&PROD_DUMMY_DEVICE&REV_4.80\4&3CF91A&0&100
LSI Logic PCI-X Ultra320 SCSI Host Adapter No SCSIADAPTER 5.2.3785.0 10/1/2002 LSI Logic Inc. pnpscsi.inf Not Available
PCI\VEN_1000&DEV_0030&SUBSYS_02921014&REV_07\3&13C0B0C5&0&19
IBM Dummy Device No SYSTEM 5.2.3785.0 10/1/2002 IBM scsidev.inf Not Available
SCSI\BRIDGE&VEN_IBM&PROD_DUMMY_DEVICE&REV_4.80\4&12F6A9BF&0&100
Broadcom NetXtreme Gigabit Ethernet No NET 6.34.0.0 2/17/2003 Broadcom oem2.inf Not Available
PCI\VEN_14E4&DEV_1648&SUBSYS_02A61014&REV_02\3&13C0B0C5&0&20
Broadcom NetXtreme Gigabit Ethernet No NET 6.34.0.0 2/17/2003 Broadcom oem2.inf Not Available
PCI\VEN_14E4&DEV_1648&SUBSYS_02A61014&REV_02\3&13C0B0C5&0&21
PCI bus No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available
ACPI\PNP0A03\2
PCI standard host CPU bridge No SYSTEM 5.2.3785.0 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 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available
PCI\VEN_1014&DEV_01A7&SUBSYS_00000000&REV_02\3&1070020&0&08
IBM ServeRAID 6M Controller No SCSIADAPTER 6.9.3.0 3/24/2003 IBM Corporation oem3.inf Not Available
PCI\VEN_9005&DEV_0250&SUBSYS_02791014&REV_00\4&1C91D199&0&4008
Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&56BAFDE&0&000
Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&56BAFDE&0&010
Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&56BAFDE&0&020
Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&56BAFDE&0&030
Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&56BAFDE&0&040
Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&56BAFDE&0&050
SCSI Processor Device No SYSTEM 5.2.3785.0 10/1/2002 IBM scsidev.inf Not Available
SCSI\PROCESSOR&VEN_IBM&PROD_EXP300___S160&REV_D014\5&56BAFDE&0&1F0
SCSI Processor Device No SYSTEM 5.2.3785.0 10/1/2002 IBM scsidev.inf Not Available
SCSI\PROCESSOR&VEN_IBM&PROD_EXP300___S160&REV_D014\5&56BAFDE&0&2F0
IBM Dummy Device No SYSTEM 5.2.3785.0 10/1/2002 IBM scsidev.inf Not Available
SCSI\BRIDGE&VEN_IBM&PROD_DUMMY_DEVICE&REV_6.00\5&56BAFDE&0&300
PCI standard PCI-to-PCI bridge No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available
PCI\VEN_1014&DEV_01A7&SUBSYS_00000000&REV_02\3&1070020&0&10
IBM ServeRAID 6M Controller No SCSIADAPTER 6.9.3.0 3/24/2003 IBM Corporation oem3.inf Not Available
PCI\VEN_9005&DEV_0250&SUBSYS_02791014&REV_00\4&392398EF&0&4010
Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&31E2F8CD&0&000
Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&31E2F8CD&0&010
Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&31E2F8CD&0&020
Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&31E2F8CD&0&030
Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&31E2F8CD&0&040
Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&31E2F8CD&0&050
SCSI Processor Device No SYSTEM 5.2.3785.0 10/1/2002 IBM scsidev.inf Not Available
SCSI\PROCESSOR&VEN_IBM&PROD_EXP300___S160&REV_D014\5&31E2F8CD&0&1F0
SCSI Processor Device No SYSTEM 5.2.3785.0 10/1/2002 IBM scsidev.inf Not Available
SCSI\PROCESSOR&VEN_IBM&PROD_EXP300___S160&REV_D014\5&31E2F8CD&0&2F0
IBM Dummy Device No SYSTEM 5.2.3785.0 10/1/2002 IBM scsidev.inf Not Available
SCSI\BRIDGE&VEN_IBM&PROD_DUMMY_DEVICE&REV_6.00\5&31E2F8CD&0&300
PCI bus No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available
ACPI\PNP0A03\3
PCI standard host CPU bridge No SYSTEM 5.2.3785.0 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 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available
PCI\VEN_1014&DEV_01A7&SUBSYS_00000000&REV_02\3&29E81982&0&20
IBM ServeRAID 6M Controller No SCSIADAPTER 6.9.3.0 3/24/2003 IBM Corporation oem3.inf Not Available

PCI\VEN_9005&DEV_0250&SUBSYS_02791014&REV_00\4&180CD7A4&0&4020
Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&13B4ABC2&0&000
Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&13B4ABC2&0&010
Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&13B4ABC2&0&020
Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&13B4ABC2&0&030
Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&13B4ABC2&0&040
Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&13B4ABC2&0&050
SCSI Processor Device No SYSTEM 5.2.3785.0 10/1/2002 IBM scsidev.inf Not Available SCSI\PROCESSOR&VEN_IBM&PROD_EXP300___S160&REV_D014\5&13B4ABC2&0&1F0
SCSI Processor Device No SYSTEM 5.2.3785.0 10/1/2002 IBM scsidev.inf Not Available SCSI\PROCESSOR&VEN_IBM&PROD_EXP300___S160&REV_D014\5&13B4ABC2&0&2F0
IBM Dummy Device No SYSTEM 5.2.3785.0 10/1/2002 IBM scsidev.inf Not Available SCSI\BRIDGE&VEN_IBM&PROD_DUMMY_DEVICE&REV_6.00\5&13B4ABC2&0&300
PCI bus No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available ACPI\PNP0A03\4
PCI standard host CPU bridge No SYSTEM 5.2.3785.0 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 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available PCI\VEN_1014&DEV_01A7&SUBSYS_00000000&REV_02\3&172E68DD&0&18
IBM ServeRAID 6M Controller No SCSIADAPTER 6.9.3.0 3/24/2003 IBM Corporation oem3.inf Not Available PCI\VEN_9005&DEV_0250&SUBSYS_02791014&REV_00\4&14F365D1&0&4018
Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&42703DC&0&000
Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&42703DC&0&010
Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&42703DC&0&020
Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&42703DC&0&030
Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&42703DC&0&040
Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&42703DC&0&050
SCSI Processor Device No SYSTEM 5.2.3785.0 10/1/2002 IBM scsidev.inf Not Available SCSI\PROCESSOR&VEN_IBM&PROD_EXP300___S160&REV_D014\5&42703DC&0&1F0
SCSI Processor Device No SYSTEM 5.2.3785.0 10/1/2002 IBM scsidev.inf Not Available SCSI\PROCESSOR&VEN_IBM&PROD_EXP300___S160&REV_D014\5&42703DC&0&2F0
IBM Dummy Device No SYSTEM 5.2.3785.0 10/1/2002 IBM scsidev.inf Not Available SCSI\BRIDGE&VEN_IBM&PROD_DUMMY_DEVICE&REV_6.00\5&42703DC&0&300
PCI bus No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available ACPI\PNP0A03\5
PCI standard host CPU bridge No SYSTEM 5.2.3785.0 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 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available PCI\VEN_1014&DEV_01A7&SUBSYS_00000000&REV_02\3&474B838&0&10
IBM ServeRAID 6M Controller No SCSIADAPTER 6.9.3.0 3/24/2003 IBM Corporation oem3.inf Not Available PCI\VEN_9005&DEV_0250&SUBSYS_02791014&REV_00\4&8C37DC5&0&4010
Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&E503D66&0&000
SCSI Processor Device No SYSTEM 5.2.3785.0 10/1/2002 IBM scsidev.inf Not Available SCSI\PROCESSOR&VEN_IBM&PROD_EXP300___S160&REV_D014\5&E503D66&0&1F0
SCSI Processor Device No SYSTEM 5.2.3785.0 10/1/2002 IBM scsidev.inf Not Available SCSI\PROCESSOR&VEN_IBM&PROD_EXP300___S160&REV_D014\5&E503D66&0&2F0
IBM Dummy Device No SYSTEM 5.2.3785.0 10/1/2002 IBM scsidev.inf Not Available SCSI\BRIDGE&VEN_IBM&PROD_DUMMY_DEVICE&REV_6.00\5&E503D66&0&300
Motherboard resources No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available ACPI\PNP0C02\10
PCI bus No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available ACPI\PNP0A03\80
PCI standard host CPU bridge No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available PCI\VEN_1014&DEV_0302&SUBSYS_00000000&REV_03\3&389E99D&0&00

PCI standard PCI-to-PCI bridge No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available PCI\VEN_1014&DEV_01A7&SUBSYS_00000000&REV_02\3&389E99D&0&08

IBM ServeRAID 6M Controller No SCSIADAPTER 6.9.3.0 3/24/2003 IBM Corporation oem3.inf Not Available PCI\VEN_9005&DEV_0250&SUBSYS_02791014&REV_02\4&1CCD9D43&0&4008

Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&95FC660&0&000

Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&95FC660&0&010

Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&95FC660&0&020

Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&95FC660&0&030

Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&95FC660&0&040

Disk drive No DISKDRIVE 5.2.3785.0 10/1/2002 (Standard disk drives) disk.inf Not Available SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.00\5&95FC660&0&050

SCSI Processor Device No SYSTEM 5.2.3785.0 10/1/2002 IBM scsidev.inf Not Available SCSI\PROCESSOR&VEN_IBM&PROD_EXP300___S160&REV_D014\5&95FC660&0&1F0

SCSI Processor Device No SYSTEM 5.2.3785.0 10/1/2002 IBM scsidev.inf Not Available SCSI\PROCESSOR&VEN_IBM&PROD_EXP300___S160&REV_D014\5&95FC660&0&2F0

IBM Dummy Device No SYSTEM 5.2.3785.0 10/1/2002 IBM scsidev.inf Not Available SCSI\BRIDGE&VEN_IBM&PROD_DUMMY_DEVICE&REV_6.00\5&95FC660&0&300

PCI bus No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available ACPI\PNP0A03\81

PCI standard host CPU bridge No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available PCI\VEN_1014&DEV_0302&SUBSYS_00000000&REV_03\3&F2FC708&0&00

PCI bus No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available ACPI\PNP0A03\82

PCI standard host CPU bridge No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available PCI\VEN_1014&DEV_0302&SUBSYS_00000000&REV_03\3&21E977AD&0&00

Motherboard resources No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available ACPI\PNP0C02\20

PCI bus No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available ACPI\PNP0A03\88

PCI standard host CPU bridge No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available PCI\VEN_1014&DEV_0302&SUBSYS_00000000&REV_03\3&32219A6E&0&00

PCI bus No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available ACPI\PNP0A03\89

PCI standard host CPU bridge No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available PCI\VEN_1014&DEV_0302&SUBSYS_00000000&REV_03\3&1F67E9C9&0&00

PCI bus No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available ACPI\PNP0A03\8A

PCI standard host CPU bridge No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available PCI\VEN_1014&DEV_0302&SUBSYS_00000000&REV_03\3&3ACAD158&0&00

Motherboard resources No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available ACPI\PNP0C02\21

Logical Disk Manager No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available ROOT\DMIO\0000

Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{85B46ECC-C804-4BCA-893D-F523EC80034B}

Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{AE45D532-E0FE-4149-8D47-42D3F5A9C3E1}

Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{06925C30-9607-480C-9CE6-55403A05802E}

Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{5A9D6586-54C6-4ED6-8700-CA2B8DB1CA8A}

Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{097C0558-143D-464A-886C-990555F89578}

Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{91B7504F-F2E9-4FA0-9489-284EB1F54022}

Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{7FF51DA7-1BF5-45C7-A7FD-6173CAC9A0E1}

Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{05316228-8F47-4865-8801-D08A43BA2C96}

Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{E9E4DB72-3772-40B9-A258-D8ADE158879E}

Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{40D8398D-0822-4A44-AC65-409109196E8A}

Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{45604376-B54D-492C-BB18-29B1DC3E86F0}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{FD4EA80D-0FEA-41BB-A685-68BA0F1061C0}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{A962F0C9-FEFA-4379-A335-0FCA99C78457}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{53D6AB25-7A37-40EF-B78B-08222F396005}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{88D69FF2-FBF6-4920-B2B3-2C7C19FD83ED}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{98CC75B7-A2EF-44F7-8CBD-78966FF3EF0C}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{587DFD5F-80F3-4EA8-B9B7-1FE2091EEEAE}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{495AE5C7-A35F-404A-98EA-45636C4A283F}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{7A4B508B-C9ED-4215-AE5A-B54E50EC2CCF}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{2308797E-E6E4-40C2-8C15-AC9685656388}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{362C8EFF-5307-4E2C-BA6D-615372B17077}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{F4EB6872-4EE4-456B-8FB4-9AB10652E48E}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{CE1E6CCF-D084-4F9F-9F6B-43F5BBB1307A}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{0C9D37DB-3497-4EA8-932A-7C4073F8FBD8}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{239CB6A1-A8B9-4EA4-9C4F-E56EB50F8800}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{373428E3-FC5F-4BC7-AB65-EEBEAC1E3DB2}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{FD6F15F5-1779-4A1D-945D-675E15CD9540}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{BCAB080E-A67D-4617-BE69-64CDE9136420}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{3D3E6CCE-540A-44B1-8F2E-B9E06E05E2A9}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{798EF2F2-EF48-4D17-BD05-B3965CBBBC8D}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{6DC96683-060C-4706-874A-D452386DB4DE}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{3D26E4C7-DB6F-4F3F-83D6-3B365C5A54CA}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{AED6F0ED-7F2F-4BE9-8818-E03767D5E288}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{94973EF0-9AAF-4FD2-B942-B7160127CAB4}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{6338DB72-C23D-42AE-A967-8E8CFEB18548}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{D1A5E5A0-03FD-4D05-A489-0C543CFDB49A}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{AB9D0879-7C39-40F4-B74A-EDB4705DC481}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{2087A1FB-FA2D-4497-B5B8-74D918A33351}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{494F5D5C-8263-46DF-B6DC-E322E43898A7}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{658C182D-F2F8-49ED-BCEF-5096C29771E5}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{25CD7670-DBA7-4B24-9928-AA0A6A283023}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{01FDB1CA-A263-4C55-9E25-0F3A18027A88}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{650E84D6-74CE-4402-8DAB-294D55CFCD23}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{02EF0438-B221-4A27-AFB2-5710089FAB02}

Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{D2EC7B5C-020B-40DE-85EF-9FEE08105F7C}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{734ED439-D30F-4A0D-8389-36B79DA45E9C}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{5204DA7A-4C4B-431D-984E-06013567DD16}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{9BA34873-2897-44EC-85F5-0E0EC6FA82E1}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{6861C000-B219-4BD5-9A21-505036FD6F10}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{7B560392-0235-4D47-8D4C-359D57DD44DF}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{02074A3E-0D84-432B-8A92-845E32E10B18}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{1985F874-E1C5-4D34-AF12-62779A1D8DC1}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{7871A29C-77AE-48B7-B565-6AF87001ACF0}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{45505383-14E0-421F-8AFA-AEE7C890F339}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{131E6F17-0237-4697-A16A-22C554C96890}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{04932315-2093-46A6-AE6F-3ADE77A08D89}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{4D0ADD82-9FA6-4951-A886-11CBC017F47F}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{89BCB55F-78BA-461B-BE3C-F01A6213EDB5}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{0E08C7BE-CDC6-4093-9AFD-BC7DF7849CCF}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{12CEB001-75D3-4346-A60D-8C23AD793639}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{30527BDE-4E3C-43AA-9F38-C682FED62668}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{68830C43-FA43-4A6B-A52A-A3EA93A34447}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{4D09461C-97EA-436C-AFA7-701E9A29BF70}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{8739ABD5-6602-4672-9353-3F9D1587CA19}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{23FA54E4-4165-4F41-8899-088053CF166D}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{2704E62B-0735-4C3D-AE35-958E0A09647C}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{4307221F-2FAD-46F6-B6D2-6BF54594BE1C}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{8011FC6C-321C-4C20-BBCF-06D7FF83AE74}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{4322CE43-A960-4C0A-9AA3-F7DB65D5A6D3}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{AB875DA1-1301-49E6-87E6-2D282569F8E3}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{EF00272D-85E6-4463-A65E-A8EB0C706CB1}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{7EDD0248-C338-40B8-8B94-1F8A74B67BFF}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{5F391906-C83E-4A6D-B08A-BD7F3B9B192D}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{84175CE2-1321-47A8-858F-956846E21245}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{5862CA85-D6DA-43A4-B915-01727B7CD5A9}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{4C63720F-90BD-4C48-AB44-3BB47219A92A}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{B22E0410-BAD8-43B4-9CA2-525C015566BE}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{BC1FAC61-4FAC-463D-BED8-356BEFF2B4FD}

Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{D9EAD399-FBA6-4BDD-8AA5-4BBD4BA5C286}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{45241808-82B5-42B1-9FD9-6AB17E2A512A}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{F39F1D17-5CBB-44CE-A6B5-8B4538AC7F47}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{E78D0E4C-30AC-4447-9445-19BDEBA6C987}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{32208FF0-A96F-4B80-ABDC-BA62A1C4A4B5}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{57FCF03F-1B39-47B4-9547-9D96E8BB4D87}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{8DA9418D-EFD5-4DF8-8F2C-FDB579B83D67}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{4B3C4510-3909-4CFE-8C15-5D7B8DC3AD1A}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{DBEED7FC-A18F-4CCA-A882-2E02C2790284}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{3446131A-25B0-437D-8694-EDA3CB0AA817}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{CAF44E3A-9FC6-486F-B04B-5BA7FCC3EA6B}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{A2AE2E3D-CD60-4B27-8E31-74556686965D}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{F93C4FD2-6D96-4D64-9768-01D4C4175A41}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{3AC79872-39EE-4CCC-B34E-F5FE94745D41}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{87C52419-EB70-464B-8C64-AC70A9D63A33}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{5045D40F-6F87-4376-8C7E-2EA4B4844342}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{DC40DA4F-6A3E-4869-A2BA-E114950B639A}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{164B4A37-8456-4F88-A59A-C1668C50028B}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{ACC75F2E-9A60-49AF-A822-F9A10459E28E}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{21754E7C-673E-4721-A0DA-A658B81BB8F4}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{D6A0711E-6F5D-4772-A9E5-E7B413ABF395}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{0A998CC9-349D-4457-A4E4-133D9EF68125}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{1A813EC4-98B1-4ACE-9CC7-E6A562884A23}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{965ABC9C-4330-4D01-85C9-127F249DCDC7}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{16AD8694-40C2-43B0-95AB-6BC0DB1363A6}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{5E6F854E-D85C-4786-AC59-A450EF914B96}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{0FD999FC-0F29-4F3F-92EB-C2AB85F3BAA9}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{30E11905-858C-46D9-9A1C-52CD1CBEAA0F}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{CBFE8CB7-373A-4E3A-87F4-EBFBF7B94798}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{161B6622-BBE6-4841-BB14-F470D0A5478F}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{3A4FAAF6-61D1-49A1-A7F8-B9B3604EDB81}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{0B2EF3D4-0E54-44D3-9128-653E1B7469E3}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{59BFDBD7-16DE-4288-ADAB-48CEA64551B6}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{C088323D-43AB-4564-ABA6-66B66D6268AC}

Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{AAA89DCF-99A8-49AB-A3A6-47E4F2676E7B}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{04E6493B-E8F6-445B-BB72-FFE4B2638D79}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{D7F95C59-FDCF-433F-BBE1-8731B72D2E9C}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{622DDF56-8A63-4EA7-9B48-47BAF7BF9F47}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{B66A5599-CAA7-40A7-BCBC-62400C976E8E}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{5F8CB461-1446-429E-99B7-E61B9459608F}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{1265F9FB-07D1-4F17-99A0-17F36AA6C3BD}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{40D6D582-F361-4899-ABE1-7E879B675C93}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{7EAAA8DD-195A-4FD8-92F2-164CC8FD6CE2}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{07D5DE92-E2E5-462F-B511-16C0923E195D}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{0B566E0D-5A7D-4B14-B69E-697E31830C0A}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{B1647A80-4216-4867-B684-0407932D2566}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{CA9DB58C-88A5-4A59-B946-2299EBA6D06A}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{EA7FF052-C8A1-4F82-9A18-B5F1F78DDBFD}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{005876A3-1C5E-4A1C-B1F5-756BE9278112}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{20172404-4BC5-44CE-BD61-579E44B89E10}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{7A6B57BF-D95D-47B7-8CA9-1B9A6BF1BB12}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{13B99F1C-041D-49C2-B13E-462A9E9E593A}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{9F91E9CD-44C1-42B9-A402-E553C86113AA}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{7DE2ACCA-6AB4-47A5-98C1-F3156228AD58}
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&3735C57B&0&LDM#{92607B24-1CBE-4CBC-8E83-6DF5B1A261C0}
Volume Manager No SYSTEM 5.2.3785.0 10/1/2002 (Standard system devices) machine.inf Not Available ROOT\FTDISK\0000
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&30A96598&0&SIGNATUREDB0BDB0BOFFSET7E00LENGTH878EE1200
Generic volume No VOLUME 5.2.3785.0 10/1/2002 Microsoft volume.inf Not Available STORAGE\VOLUME\1&30A96598&0&SIGNATUREDC31DC31OFFSET7E00LENGTH8796B9400
AFD Networking Support Environment Not Available LEGACYDRIVER Not Available Not Available Not Available Not Available Not Available ROOT\LEGACY_AFD\0000
Beep Not Available LEGACYDRIVER Not Available Not Available Not Available Not Available Not Available ROOT\LEGACY_BEEP\0000
CRC Disk Filter Driver Not Available LEGACYDRIVER Not Available Not Available Not Available Not Available Not Available ROOT\LEGACY_CRCDISK\0000
dmboot Not Available LEGACYDRIVER Not Available Not Available Not Available Not Available Not Available ROOT\LEGACY_DMBOOT\0000
dmload Not Available LEGACYDRIVER Not Available Not Available Not Available Not Available Not Available ROOT\LEGACY_DMLOAD\0000
Fips Not Available LEGACYDRIVER Not Available Not Available Not Available Not Available Not Available ROOT\LEGACY_FIPS\0000
Generic Packet Classifier Not Available LEGACYDRIVER Not Available Not Available Not Available Not Available Not Available ROOT\LEGACY_GPC\0000
IP Network Address Translator Not Available LEGACYDRIVER Not Available Not Available Not Available Not Available Not Available ROOT\LEGACY_IPNAT\0000
IPSEC driver Not Available LEGACYDRIVER Not Available Not Available Not Available Not Available Not Available ROOT\LEGACY_IPSEC\0000
ksecdd Not Available LEGACYDRIVER Not Available Not Available Not Available Not Available Not Available ROOT\LEGACY_KSECDD\0000
mnmdd Not Available LEGACYDRIVER Not Available Not Available Not Available Not Available Not Available ROOT\LEGACY_MNMDD\0000
mountmgr Not Available LEGACYDRIVER Not Available Not Available Not Available Not Available Not Available ROOT\LEGACY_MOUNTMGR\0000

NDIS System Driver Not Available LEGACYDRIVER
Not Available Not Available Not Available
Not Available Not Available
ROOT\LEGACY_NDIS\0000
Remote Access NDIS TAPI Driver Not Available
LEGACYDRIVER Not Available Not Available
Not Available Not Available Not Available
ROOT\LEGACY_NDISTAPI\0000
NDIS Usermode I/O Protocol Not Available
LEGACYDRIVER Not Available Not Available
Not Available Not Available Not Available
ROOT\LEGACY_NDISUIO\0000
NDProxy Not Available LEGACYDRIVER Not Available
Not Available Not Available Not Available
Not Available ROOT\LEGACY_NDPROXY\0000
NetBios over Tcpip Not Available LEGACYDRIVER
Not Available Not Available Not Available
Not Available Not Available
ROOT\LEGACY_NETBT\0000
IBM ServeRAID 4M/4Mx/4L/4Lx/5i/6M/6i Device Driver
Not Available LEGACYDRIVER Not Available
Not Available Not Available Not Available
Not Available ROOT\LEGACY_NFRD960\0000
Null Not Available LEGACYDRIVER Not Available
Not Available Not Available Not Available
Not Available ROOT\LEGACY_NULL\0000
numaqdrv Not Available LEGACYDRIVER Not Available
Not Available Not Available Not Available
Not Available ROOT\LEGACY_NUMAQDRV\0000
Partition Manager Not Available LEGACYDRIVER
Not Available Not Available Not Available
Not Available Not Available
ROOT\LEGACY_PARTMGR\0000
Remote Access Auto Connection Driver Not Available
LEGACYDRIVER Not Available Not Available
Not Available Not Available Not Available
ROOT\LEGACY_RASACD\0000
RDPCDD Not Available LEGACYDRIVER Not Available
Not Available Not Available Not Available
Not Available ROOT\LEGACY_RDPCDD\0000
RDPWD Not Available LEGACYDRIVER Not Available
Not Available Not Available Not Available
Not Available ROOT\LEGACY_RDPWD\0000
TCP/IP Protocol Driver Not Available
LEGACYDRIVER Not Available Not Available
Not Available Not Available Not Available
ROOT\LEGACY_TCPIP\0000
TDTCP Not Available LEGACYDRIVER Not Available
Not Available Not Available Not Available
Not Available ROOT\LEGACY_TDTCP\0000
VGA Display Controller. Not Available
LEGACYDRIVER Not Available Not Available
Not Available Not Available Not Available
ROOT\LEGACY_VGASAVE\0000
volsnap Not Available LEGACYDRIVER Not Available
Not Available Not Available Not Available
Not Available ROOT\LEGACY_VOLSNAP\0000
Remote Access IP ARP Driver Not Available
LEGACYDRIVER Not Available Not Available
Not Available Not Available Not Available
ROOT\LEGACY_WANARP\0000
Audio Codecs No MEDIA 5.2.3785.0 10/1/2002
(Standard system devices) wave.inf Not Available
ROOT\MEDIA\MS_MMACM
Legacy Audio Drivers No MEDIA 5.2.3785.0 10/1/2002
(Standard system devices) wave.inf Not Available
ROOT\MEDIA\MS_MMDRV
Media Control Devices No MEDIA 5.2.3785.0
10/1/2002 (Standard system devices) wave.inf Not Available
ROOT\MEDIA\MS_MMMCI
Legacy Video Capture Devices No MEDIA 5.2.3785.0
10/1/2002 (Standard system devices) wave.inf Not Available
ROOT\MEDIA\MS_MMVCD
Video Codecs No MEDIA 5.2.3785.0 10/1/2002
(Standard system devices) wave.inf Not Available
ROOT\MEDIA\MS_MMVID
WAN Miniport (L2TP) No NET 5.2.3785.0
10/1/2002 Microsoft netrasa.inf Not Available
ROOT\MS_L2TPMINIPORT\0000
WAN Miniport (IP) No NET 5.2.3785.0 10/1/2002
Microsoft netrasa.inf Not Available
ROOT\MS_NDISWANIP\0000
WAN Miniport (PPPOE) No NET 5.2.3785.0
10/1/2002 Microsoft netrasa.inf Not Available
ROOT\MS_PPPOEMINIPORT\0000
WAN Miniport (PPTP) No NET 5.2.3785.0
10/1/2002 Microsoft netrasa.inf Not Available
ROOT\MS_PPTPMINIPORT\0000
Direct Parallel No NET 5.2.3785.0 10/1/2002
Microsoft netrasa.inf Not Available
ROOT\MS_PTIMINIPORT\0000
Terminal Server Device Redirector No SYSTEM
5.2.3785.0 10/1/2002 (Standard system devices) machine.inf
Not Available ROOT\RDPDR\0000
Terminal Server Keyboard Driver No SYSTEM
5.2.3785.0 10/1/2002 (Standard system devices) machine.inf
Not Available ROOT\RDP_KBD\0000
Terminal Server Mouse Driver No SYSTEM 5.2.3785.0
10/1/2002 (Standard system devices) machine.inf
Not Available ROOT\RDP_MOU\0000
Plug and Play Software Device Enumerator No SYSTEM
5.2.3785.0 10/1/2002 (Standard system devices) machine.inf
Not Available ROOT\SYSTEM\0000
Microcode Update Device No SYSTEM 5.2.3785.0
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_licen
se_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:\Program Files\Microsoft Visual
Studio\VC98\atl\include;D:\Program Files\Microsoft Visual
Studio\VC98\mfc\include;D:\Program Files\Microsoft Visual
Studio\VC98\include; <SYSTEM>
LIB D:\sqllib\LIB;D:\Program Files\Microsoft Visual
Studio\VC98\mfc\lib;D:\Program Files\Microsoft Visual
Studio\VC98\lib <SYSTEM>
NUMBER_OF_PROCESSORS 16 <SYSTEM>
OS Windows_NT <SYSTEM>
Path
%SystemRoot%\system32;%SystemRoot%;%SystemRoot%\System32
\Wbem;D:\sqllib\BIN;D:\sqllib\FUNCTION;D:\sqllib\SAMPLES\REP
L;d:\tools;d:\nttools;D:\Program Files\Microsoft Visual
Studio\VC98\Bin;D:\Program Files\Microsoft Visual
Studio\Common\Tools\WinNT;D:\Program Files\Microsoft Visual
Studio\Common\MSDev98\Bin;D:\Program Files\Microsoft Visual

Studio\Common\Tools;D:\Program Files\Microsoft Visual Studio\VC98\bin <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 5, GenuineIntel <SYSTEM>
PROCESSOR_LEVEL 15 <SYSTEM>
PROCESSOR_REVISION 0205 <SYSTEM>
RAH_DO_ALL_LOGICAL_NODES TRUE <SYSTEM>
RAHSLEEPTIME 999999 <SYSTEM>
TEMP %SystemRoot%\TEMP <SYSTEM>
TMP %SystemRoot%\TEMP <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 RUTHLESS-D\tpch
TMP %USERPROFILE%\Local Settings\Temp RUTHLESS-D\tpch

[Print Jobs]

Document Size Owner Notify Status Time Submitted Start Time Until Time Elapsed Time 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
system Not Available 4 8 0 1413120 Not Available Not Available Not Available Not Available
smss.exe Not Available 348 11 204800 1413120 6/17/2003 12:56 PM Not Available Not Available Not Available
csrss.exe Not Available 520 13 Not Available Not Available 6/17/2003 12:56 PM Not Available Not Available Not Available
winlogon.exe c:\windows\system32\winlogon.exe 544 13 204800 1413120 6/17/2003 12:56 PM 5.2.3785.0 (srv03_rtm.030308-1736) 537.00 KB (549,888 bytes) 3/9/2003 7:00 AM
services.exe c:\windows\system32\services.exe 588 9 204800 1413120 6/17/2003 12:56 PM 5.2.3785.0 (srv03_rtm.030308-1736) 102.00 KB (104,448 bytes) 3/9/2003 7:00 AM
lsass.exe c:\windows\system32\lsass.exe 600 9 204800 1413120 6/17/2003 12:56 PM 5.2.3785.0 (srv03_rtm.030308-1736) 13.00 KB (13,312 bytes) 3/9/2003 7:00 AM
svchost.exe c:\windows\system32\svchost.exe 756 8 204800 1413120 6/17/2003 12:56 PM 5.2.3785.0 (srv03_rtm.030308-1736) 13.00 KB (13,312 bytes) 3/9/2003 7:00 AM
svchost.exe c:\windows\system32\svchost.exe 832 8 204800 1413120 6/17/2003 12:56 PM 5.2.3785.0 (srv03_rtm.030308-1736) 13.00 KB (13,312 bytes) 3/9/2003 7:00 AM
svchost.exe Not Available 964 8 Not Available Not Available 6/17/2003 12:56 PM Not Available Not Available Not Available
svchost.exe Not Available 1016 8 Not Available Not Available 6/17/2003 12:56 PM Not Available Not Available Not Available
svchost.exe c:\windows\system32\svchost.exe 1052 8 204800 1413120 6/17/2003 12:56 PM 5.2.3785.0 (srv03_rtm.030308-1736) 13.00 KB (13,312 bytes) 3/9/2003 7:00 AM
msdtc.exe Not Available 1676 8 Not Available Not Available 6/17/2003 12:57 PM Not Available Not Available
dfssvc.exe c:\windows\system32\dfssvc.exe 1740 8 204800 1413120 6/17/2003 12:57 PM 5.2.3785.0 (srv03_rtm.030308-1736) 130.50 KB (133,632 bytes) 3/9/2003 7:00 AM
dns.exe c:\windows\system32\dns.exe 1760 8 204800 1413120 6/17/2003 12:57 PM 5.2.3785.0 (srv03_rtm.030308-1736) 412.00 KB (421,888 bytes) 4/30/2003 4:13 PM
ismserv.exe c:\windows\system32\ismserv.exe 1820 8 204800 1413120 6/17/2003 12:57 PM 5.2.3785.0 (srv03_rtm.030308-1736) 35.50 KB (36,352 bytes) 3/9/2003 7:00 AM
ntfrs.exe c:\windows\system32\ntfrs.exe 1900 8 204800 1413120 6/17/2003 12:57 PM 5.2.3785.0 (srv03_rtm.030308-1736) 748.50 KB (766,464 bytes) 3/9/2003 7:00 AM
tcpsvcs.exe c:\windows\system32\tcpsvcs.exe 372 8 204800 1413120 6/17/2003 12:57 PM 5.2.3785.0 (srv03_rtm.030308-1736) 21.00 KB (21,504 bytes) 3/9/2003 7:00 AM
wmiprvse.exe Not Available 2432 8 Not Available Not Available 6/17/2003 12:58 PM Not Available Not Available Not Available
explorer.exe c:\windows\explorer.exe 2588 8 204800 1413120 6/17/2003 12:58 PM 6.00.3785.0 (srv03_rtm.030308-1736) 1,008.50 KB (1,032,704 bytes) 3/9/2003 7:00 AM
notepad.exe c:\windows\system32\notepad.exe 2932 8 204800 1413120 6/17/2003 12:59 PM 5.2.3785.0 (srv03_rtm.030308-1736) 66.50 KB (68,096 bytes) 3/9/2003 7:00 AM
cmd.exe c:\windows\system32\cmd.exe 3060 8 204800 1413120 6/17/2003 1:02 PM 5.2.3785.0 (srv03_rtm.030308-1736) 373.50 KB (382,464 bytes) 3/9/2003 7:00 AM
db2rcmd.exe Not Available 3432 8 Not Available Not Available 6/17/2003 1:03 PM Not Available Not Available Not Available
ntvdm.exe c:\windows\system32\ntvdm.exe 1972 8 204800 1413120 6/17/2003 1:05 PM 5.2.3785.0 (srv03_rtm.030308-1736) 408.00 KB (417,792 bytes) 3/9/2003 7:00 AM

csrss.exe Not Available 2192 13 Not Available
Not Available 6/17/2003 1:30 PM Not Available
Not Available Not Available
winlogon.exe c:\windows\system32\winlogon.exe
2212 13 204800 1413120 6/17/2003 1:30 PM
5.2.3785.0 (srv03_rtm.030308-1736) 537.00 KB (549,888
bytes) 3/9/2003 7:00 AM
rdpclip.exe c:\windows\system32\rdpclip.exe 2660 8
204800 1413120 6/17/2003 1:30 PM 5.2.3785.0
(srv03_rtm.030308-1736) 53.00 KB (54,272 bytes)
4/29/2003 4:23 PM
explorer.exe c:\windows\explorer.exe 2720 8
204800 1413120 6/17/2003 1:30 PM 6.00.3785.0
(srv03_rtm.030308-1736) 1,008.50 KB (1,032,704 bytes)
3/9/2003 7:00 AM
cmd.exe c:\windows\system32\cmd.exe 2836 8
204800 1413120 6/17/2003 1:30 PM 5.2.3785.0
(srv03_rtm.030308-1736) 373.50 KB (382,464 bytes)
3/9/2003 7:00 AM
cmd.exe c:\windows\system32\cmd.exe 684 8
204800 1413120 6/17/2003 1:32 PM 5.2.3785.0
(srv03_rtm.030308-1736) 373.50 KB (382,464 bytes)
3/9/2003 7:00 AM
logon.scr c:\windows\system32\logon.scr 3140 4
204800 1413120 6/17/2003 1:35 PM 5.2.3785.0
(srv03_rtm.030308-1736) 497.00 KB (508,928 bytes)
3/9/2003 7:00 AM
rxapi.exe d:\nttools\rxapi.exe 4092 8 204800
1413120 6/17/2003 3:25 PM 1. 0. 2. 2 22.50 KB (23,040 bytes)
5/2/2003 9:21 AM
helpctr.exe c:\windows\pchealth\helpctr\binaries\helpctr.exe
2272 8 204800 1413120 6/17/2003 3:43 PM
5.2.3785.0 (srv03_rtm.030308-1736) 764.00 KB (782,336
bytes) 4/29/2003 4:27 PM
helpsvc.exe
c:\windows\pchealth\helpctr\binaries\helpsvc.exe 2840 8
204800 1413120 6/17/2003 3:43 PM 5.2.3785.0
(srv03_rtm.030308-1736) 720.00 KB (737,280 bytes)
4/29/2003 4:27 PM
wmiprvse.exe Not Available 2844 8
Not Available Not Available 6/17/2003 3:43 PM
Not Available Not Available Not Available

[Loaded Modules]

Name Version Size File Date Manufacturer
Path
winlogon 5.2.3785.0 (srv03_rtm.030308-1736) 537.00 KB
(549,888 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\winlogon.exe
ntdll 5.2.3785.0 (srv03_rtm.030308-1736) 723.00 KB
(740,352 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\ntdll.dll
kernel32 5.2.3785.0 (srv03_rtm.030308-1736) 965.00 KB
(988,160 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\kernel32.dll
msvcrt 7.0.3785.0 (srv03_rtm.030308-1736) 319.50 KB
(327,168 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\msvcrt.dll
advapi32 5.2.3785.0 (srv03_rtm.030308-1736) 559.50 KB
(572,928 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\advapi32.dll
rpcrt4 5.2.3785.0 (srv03_rtm.030308-1736) 644.50 KB
(659,968 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\rpcrt4.dll
user32 5.2.3785.0 (srv03_rtm.030308-1736) 562.50 KB
(576,000 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\user32.dll
gdi32 5.2.3785.0 (srv03_rtm.030308-1736) 263.00 KB
(269,312 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\gdi32.dll
userenv 5.2.3785.0 (srv03_rtm.030308-1736) 733.50 KB
(751,104 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\userenv.dll
nddeapi 5.2.3785.0 (srv03_rtm.030308-1736) 16.00 KB
(16,384 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\nddeapi.dll
crypt32 5.131.3785.0 (srv03_rtm.030308-1736) 598.00 KB
(612,352 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\crypt32.dll
msasn1 5.2.3785.0 (srv03_rtm.030308-1736) 58.00 KB
(59,392 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\msasn1.dll
secur32 5.2.3785.0 (srv03_rtm.030308-1736) 63.00 KB
(64,512 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\secur32.dll
winsta 5.2.3785.0 (srv03_rtm.030308-1736) 51.00 KB
(52,224 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\winsta.dll
netapi32 5.2.3785.0 (srv03_rtm.030308-1736) 317.00 KB
(324,608 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\netapi32.dll
profmap 5.2.3785.0 (srv03_rtm.030308-1736) 22.00 KB
(22,528 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\profmap.dll
regapi 5.2.3785.0 (srv03_rtm.030308-1736) 48.50 KB
(49,664 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\regapi.dll
ws2_32 5.2.3785.0 (srv03_rtm.030308-1736) 85.50 KB
(87,552 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\ws2_32.dll
ws2help 5.2.3785.0 (srv03_rtm.030308-1736) 19.50 KB
(19,968 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\ws2help.dll
psapi 5.2.3785.0 (srv03_rtm.030308-1736) 21.50 KB
(22,016 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\psapi.dll
version 5.2.3785.0 (srv03_rtm.030308-1736) 17.00 KB
(17,408 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\version.dll
setupapi 5.2.3785.0 (srv03_rtm.030308-1736) 1,014.50 KB
(1,038,848 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\setupapi.dll
msgina 5.2.3785.0 (srv03_rtm.030308-1736) 1.14 MB
(1,191,936 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\msgina.dll
shsvcs 6.00.3785.0 (srv03_rtm.030308-1736) 121.50 KB
(124,416 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\shsvcs.dll
shlwapi 6.00.3785.0 (srv03_rtm.030308-1736) 281.00 KB
(287,744 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\shlwapi.dll
sfc 5.2.3785.0 (srv03_rtm.030308-1736) 4.50 KB
(4,608 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\sfc.dll
sfc_os 5.2.3785.0 (srv03_rtm.030308-1736) 133.00 KB
(136,192 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\sfc_os.dll
wintrust 5.131.3785.0 (srv03_rtm.030308-1736) 163.50 KB
(167,424 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\wintrust.dll
ole32 5.2.3785.0 (srv03_rtm.030308-1736) 1.13 MB
(1,187,328 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\ole32.dll

imagehlp 5.2.3785.0 (srv03_rtm.030308-1736) 142.50 KB (145,920 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\imagehlp.dll
comctl32 6.0 (srv03_rtm.030308-1736) 907.00 KB (928,768 bytes) 4/29/2003 12:06 PM Microsoft Corporation c:\windows\winsxs\x86_microsoft.windows.common-controls_6595b64144ccf1df_6.0.100.0_x-ww_8417450b\comctl32.dll
winscard 5.2.3785.0 (srv03_rtm.030308-1736) 98.50 KB (100,864 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\winscard.dll
wtsapi32 5.2.3785.0 (srv03_rtm.030308-1736) 17.50 KB (17,920 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\wtsapi32.dll
sxs 5.2.3785.0 (srv03_rtm.030308-1736) 733.00 KB (750,592 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\sxs.dll
winmm 5.2.3785.0 (srv03_rtm.030308-1736) 166.00 KB (169,984 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\winmm.dll
shell32 6.00.3785.0 (srv03_rtm.030308-1736) 7.79 MB (8,164,352 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\shell32.dll
wsock32 5.2.3785.0 (srv03_rtm.030308-1736) 22.00 KB (22,528 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\wsock32.dll
iphlpapi 5.2.3785.0 (srv03_rtm.030308-1736) 83.00 KB (84,992 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll
icmp 5.2.3785.0 (srv03_rtm.030308-1736) 4.50 KB (4,608 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\icmp.dll
mprapi 5.2.3785.0 (srv03_rtm.030308-1736) 81.00 KB (82,944 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\mprapi.dll
activeds 5.2.3785.0 (srv03_rtm.030308-1736) 189.00 KB (193,536 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\activeds.dll
adsldpc 5.2.3785.0 (srv03_rtm.030308-1736) 142.50 KB (145,920 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\adsldpc.dll
wldap32 5.2.3785.0 (srv03_rtm.030308-1736) 158.00 KB (161,792 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\wldap32.dll
credui 5.2.3785.0 (srv03_rtm.030308-1736) 159.00 KB (162,816 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\credui.dll
atl 3.05.2283 83.00 KB (84,992 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\atl.dll
oleaut32 5.2.3785.0 486.00 KB (497,664 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\oleaut32.dll
rtutils 5.2.3785.0 (srv03_rtm.030308-1736) 32.00 KB (32,768 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\rtutils.dll
samlib 5.2.3785.0 (srv03_rtm.030308-1736) 49.00 KB (50,176 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\samlib.dll
rsaenh 5.2.3785.0 (srv03_rtm.030308-1736) 176.83 KB (181,072 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\rsaenh.dll
mswsock 5.2.3785.0 (srv03_rtm.030308-1736) 254.00 KB (260,096 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\mswsock.dll
rasadhlp 5.2.3785.0 (srv03_rtm.030308-1736) 6.50 KB (6,656 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\rasadhlp.dll
kerberos 5.2.3785.0 (srv03_rtm.030308-1736) 314.00 KB (321,536 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\kerberos.dll
cryptdll 5.2.3785.0 (srv03_rtm.030308-1736) 34.50 KB (35,328 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\cryptdll.dll
wshtcpip 5.2.3785.0 (srv03_rtm.030308-1736) 18.00 KB (18,432 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\wshtcpip.dll
ntdsapi 5.2.3785.0 (srv03_rtm.030308-1736) 76.00 KB (77,824 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntdsapi.dll
dnsapi 5.2.3785.0 (srv03_rtm.030308-1736) 147.50 KB (151,040 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\dnsapi.dll
cscdll 5.2.3785.0 (srv03_rtm.030308-1736) 99.00 KB (101,376 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\cscdll.dll
wlnotify 5.2.3785.0 (srv03_rtm.030308-1736) 87.50 KB (89,600 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\wlnotify.dll
winspool 5.2.3785.0 (srv03_rtm.030308-1736) 140.00 KB (143,360 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\winspool.drv
mpr 5.2.3785.0 (srv03_rtm.030308-1736) 56.00 KB (57,344 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\mpr.dll
comctl32 5.82 (srv03_rtm.030308-1736) 561.00 KB (574,464 bytes) 4/29/2003 12:06 PM Microsoft Corporation c:\windows\winsxs\x86_microsoft.windows.common-controls_6595b64144ccf1df_5.82.0.0_x-ww_8a69ba05\comctl32.dll
uxtheme 6.00.3785.0 (srv03_rtm.030308-1736) 196.00 KB (200,704 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\uxtheme.dll
msv1_0 5.2.3785.0 (srv03_rtm.030308-1736) 127.00 KB (130,048 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\msv1_0.dll
clbcatq 2001.12.4715.0 (srv03_rtm.030308-1736) 481.00 KB (492,544 bytes) 4/29/2003 4:23 PM Microsoft Corporation c:\windows\system32\clbcatq.dll
comres 2001.12.4715.0 (srv03_rtm.030308-1736) 778.00 KB (796,672 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\comres.dll
cscui 5.2.3785.0 (srv03_rtm.030308-1736) 305.00 KB (312,320 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\cscui.dll
drprov 5.2.3785.0 (srv03_rtm.030308-1736) 12.50 KB (12,800 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\drprov.dll
ntlanman 5.2.3785.0 (srv03_rtm.030308-1736) 41.00 KB (41,984 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntlanman.dll
netui0 5.2.3785.0 (srv03_rtm.030308-1736) 75.50 KB (77,312 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\netui0.dll
netui1 5.2.3785.0 (srv03_rtm.030308-1736) 184.00 KB (188,416 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\netui1.dll
davclnt 5.2.3785.0 (srv03_rtm.030308-1736) 23.50 KB (24,064 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\davclnt.dll
mprui 5.2.3785.0 (srv03_rtm.030308-1736) 49.00 KB (50,176 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\mprui.dll
netui2 5.2.3785.0 (srv03_rtm.030308-1736) 309.50 KB (316,928 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\netui2.dll

comdlg32 6.00.3785.0 (srv03_rtm.030308-1736) 261.00 KB (267,264 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\comdlg32.dll
netmsg 5.2.3785.0 (srv03_rtm.030308-1736) 178.00 KB (182,272 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\netmsg.dll
ntmarta 5.2.3785.0 (srv03_rtm.030308-1736) 114.00 KB (116,736 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntmarta.dll
wbemprox 5.2.3785.0 (srv03_rtm.030308-1736) 17.50 KB (17,920 bytes) 4/29/2003 4:23 PM Microsoft Corporation c:\windows\system32\wbem\wbemprox.dll
wbemcomn 5.2.3785.0 (srv03_rtm.030308-1736) 212.00 KB (217,088 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\wbem\wbemcomn.dll
wbemsvc 5.2.3785.0 (srv03_rtm.030308-1736) 42.50 KB (43,520 bytes) 4/29/2003 4:23 PM Microsoft Corporation c:\windows\system32\wbem\wbemsvc.dll
fastprox 5.2.3785.0 (srv03_rtm.030308-1736) 443.00 KB (453,632 bytes) 4/29/2003 4:23 PM Microsoft Corporation c:\windows\system32\wbem\fastprox.dll
msvcp60 6.05.2144.0 388.00 KB (397,312 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\msvcp60.dll
cabinet 5.2.3785.0 (srv03_rtm.030308-1736) 61.00 KB (62,464 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\cabinet.dll
services 5.2.3785.0 (srv03_rtm.030308-1736) 102.00 KB (104,448 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\services.exe
scesrv 5.2.3785.0 (srv03_rtm.030308-1736) 316.50 KB (324,096 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\scesrv.dll
authz 5.2.3785.0 (srv03_rtm.030308-1736) 67.50 KB (69,120 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\authz.dll
umpnpmgr 5.2.3785.0 (srv03_rtm.030308-1736) 121.50 KB (124,416 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\umpnpmgr.dll
ncobjapi 5.2.3785.0 (srv03_rtm.030308-1736) 35.00 KB (35,840 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\ncobjapi.dll
eventlog 5.2.3785.0 (srv03_rtm.030308-1736) 60.00 KB (61,440 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\eventlog.dll
esent 5.2.3785.0 (srv03_rtm.030308-1736) 952.50 KB (975,360 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\esent.dll
lsass 5.2.3785.0 (srv03_rtm.030308-1736) 13.00 KB (13,312 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\lsass.exe
lsasrv 5.2.3785.0 (srv03_rtm.030308-1736) 781.00 KB (799,744 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\lsasrv.dll
samsrv 5.2.3785.0 (srv03_rtm.030308-1736) 453.00 KB (463,872 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\samsrv.dll
msprivs 5.2.3785.0 (srv03_rtm.030308-1736) 46.50 KB (47,616 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\msprivs.dll
netlogon 5.2.3785.0 (srv03_rtm.030308-1736) 409.50 KB (419,328 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\netlogon.dll
w32time 5.2.3785.0 (srv03_rtm.030308-1736) 216.00 KB (221,184 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\w32time.dll
schannel 5.2.3785.0 (srv03_rtm.030308-1736) 149.50 KB (153,088 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll
wdigest 5.2.3785.0 (srv03_rtm.030308-1736) 61.00 KB (62,464 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\wdigest.dll
ntdsa 5.2.3785.0 (srv03_rtm.030308-1736) 1.58 MB (1,652,224 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntdsa.dll
ntdsatq 5.2.3785.0 (srv03_rtm.030308-1736) 32.00 KB (32,768 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntdsatq.dll
ntdsmsg 5.2.3785.0 (srv03_rtm.030308-1736) 540.00 KB (552,960 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntdsmsg.dll
ntdsbsrv 5.2.3785.0 (srv03_rtm.030308-1736) 50.00 KB (51,200 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntdsbsrv.dll
vssapi 5.2.3785.0 (srv03_rtm.030308-1736) 528.00 KB (540,672 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\vssapi.dll
kdcsvc 5.2.3785.0 (srv03_rtm.030308-1736) 221.00 KB (226,304 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\kdcsvc.dll
rassfm 5.2.3785.0 (srv03_rtm.030308-1736) 20.50 KB (20,992 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\rassfm.dll
scecli 5.2.3785.0 (srv03_rtm.030308-1736) 179.50 KB (183,808 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\scecli.dll
dsrestor 5.2.3785.0 (srv03_rtm.030308-1736) 9.50 KB (9,728 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\dsrestor.dll
pwdssp 5.2.3785.0 (srv03_rtm.030308-1736) 12.50 KB (12,800 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\pwdssp.dll
msapsspc 6.00.7755 78.25 KB (80,128 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\msapsspc.dll
msvcrt40 5.2.3785.0 (srv03_rtm.030308-1736) 60.00 KB (61,440 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\msvcrt40.dll
ntdskcc 5.2.3785.0 (srv03_rtm.030308-1736) 113.00 KB (115,712 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntdskcc.dll
w32topl 5.2.3785.0 (srv03_rtm.030308-1736) 27.00 KB (27,648 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\w32topl.dll
ipsecsvc 5.2.3785.0 (srv03_rtm.030308-1736) 163.00 KB (166,912 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\ipsecsvc.dll
oakley 5.2.3785.0 (srv03_rtm.030308-1736) 324.50 KB (332,288 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\oakley.dll
winipsec 5.2.3785.0 (srv03_rtm.030308-1736) 34.00 KB (34,816 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\winipsec.dll
pstorsvc 5.2.3785.0 (srv03_rtm.030308-1736) 24.00 KB (24,576 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\pstorsvc.dll
psbase 5.2.3785.0 (srv03_rtm.030308-1736) 81.00 KB (82,944 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\psbase.dll
dssenh 5.2.3785.0 (srv03_rtm.030308-1736) 131.33 KB (134,480 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\dssenh.dll

wlbsctrl 5.2.3785.0 (srv03_rtm.030308-1736) 78.00 KB
(79,872 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\wlbsctrl.dll
es 2001.12.4715.0 (srv03_rtm.030308-1736) 221.50 KB
(226,816 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\es.dll
winrnr 5.2.3785.0 (srv03_rtm.030308-1736) 15.00 KB
(15,360 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\winrnr.dll
svchost 5.2.3785.0 (srv03_rtm.030308-1736) 13.00 KB
(13,312 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\svchost.exe
rpcss 5.2.3785.0 (srv03_rtm.030308-1736) 276.00 KB
(282,624 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\rpcss.dll
termsrv 5.2.3785.0 (srv03_rtm.030308-1736) 216.50 KB
(221,696 bytes) 4/29/2003 4:23 PM Microsoft Corporation
c:\windows\system32\termsrv.dll
icaapi 5.2.3785.0 (srv03_rtm.030308-1736) 10.50 KB
(10,752 bytes) 4/29/2003 4:23 PM Microsoft Corporation
c:\windows\system32\icaapi.dll
mstlsapi 5.2.3785.0 (srv03_rtm.030308-1736) 104.50 KB
(107,008 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\mstlsapi.dll
rdpwsx 5.2.3785.0 (srv03_rtm.030308-1736) 80.13 KB
(82,056 bytes) 4/29/2003 4:23 PM Microsoft Corporation
c:\windows\system32\rdpwsx.dll
wkssvc 5.2.3785.0 (srv03_rtm.030308-1736) 125.00 KB
(128,000 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\wkssvc.dll
wiarpc 5.2.3785.0 (srv03_rtm.030308-1736) 30.00 KB
(30,720 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\wiarpc.dll
srvsvc 5.2.3785.0 (srv03_rtm.030308-1736) 89.00 KB
(91,136 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\srvsvc.dll
dmserver 5.2.3785.0 (srv03_rtm.030308-1736) 24.00 KB
(24,576 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\dmserver.dll
mprdim 5.2.3785.0 (srv03_rtm.030308-1736) 49.50 KB
(50,688 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\mprdim.dll
rasapi32 5.2.3785.0 (srv03_rtm.030308-1736) 227.50 KB
(232,960 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\rasapi32.dll
rasman 5.2.3785.0 (srv03_rtm.030308-1736) 56.50 KB
(57,856 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\rasman.dll
tapi32 5.2.3785.0 (srv03_rtm.030308-1736) 175.00 KB
(179,200 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\tapi32.dll
wmi 5.2.3785.0 (srv03_rtm.030308-1736) 6.50 KB
(6,656 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\wmi.dll
seclogon 5.2.3785.0 (srv03_rtm.030308-1736) 16.50 KB
(16,896 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\seclogon.dll
sens 5.2.3785.0 (srv03_rtm.030308-1736) 35.50 KB
(36,352 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\sens.dll
iprtrmgr 5.2.3785.0 (srv03_rtm.030308-1736) 164.50 KB
(168,448 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\iprtrmgr.dll
rtm 5.2.3785.0 (srv03_rtm.030308-1736) 101.50 KB
(103,936 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\rtm.dll
iprtprio 5.2.3785.0 (srv03_rtm.030308-1736) 5.00 KB
(5,120 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\iprtprio.dll
wmisvc 5.2.3785.0 (srv03_rtm.030308-1736) 131.00 KB
(134,144 bytes) 4/29/2003 4:23 PM Microsoft Corporation
c:\windows\system32\wbem\wmisvc.dll
igmpv2 5.2.3785.0 (srv03_rtm.030308-1736) 128.00 KB
(131,072 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\igmpv2.dll
netman 5.2.3785.0 (srv03_rtm.030308-1736) 209.00 KB
(214,016 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\netman.dll
wzcsvc 5.2.3785.0 (srv03_rtm.030308-1736) 272.50 KB
(279,040 bytes) 3/8/2003 6:01 PM Microsoft Corporation
c:\windows\system32\wzcsvc.dll
dhcpcsvc 5.2.3785.0 (srv03_rtm.030308-1736) 101.50 KB
(103,936 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\dhcpcsvc.dll
wzcsapi 5.2.3785.0 (srv03_rtm.030308-1736) 24.50 KB
(25,088 bytes) 3/8/2003 6:01 PM Microsoft Corporation
c:\windows\system32\wzcsapi.dll
comsvcs 2001.12.4715.0 (srv03_rtm.030308-1736) 1.14 MB
(1,199,616 bytes) 4/29/2003 4:23 PM Microsoft Corporation
c:\windows\system32\comsvcs.dll
ntlsapi 5.2.3785.0 (srv03_rtm.030308-1736) 8.00 KB
(8,192 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\ntlsapi.dll
wbemcore 5.2.3785.0 (srv03_rtm.030308-1736) 457.00 KB
(467,968 bytes) 4/29/2003 4:23 PM Microsoft Corporation
c:\windows\system32\wbem\wbemcore.dll
esscli 5.2.3785.0 (srv03_rtm.030308-1736) 235.50 KB
(241,152 bytes) 4/29/2003 4:23 PM Microsoft Corporation
c:\windows\system32\wbem\esscli.dll
wmiutils 5.2.3785.0 (srv03_rtm.030308-1736) 90.50 KB
(92,672 bytes) 4/29/2003 4:23 PM Microsoft Corporation
c:\windows\system32\wbem\wmiutils.dll
repdrvfs 5.2.3785.0 (srv03_rtm.030308-1736) 165.00 KB
(168,960 bytes) 4/29/2003 4:23 PM Microsoft Corporation
c:\windows\system32\wbem\repdrvfs.dll
wmiprvsd 5.2.3785.0 (srv03_rtm.030308-1736) 405.50 KB
(415,232 bytes) 4/29/2003 4:23 PM Microsoft Corporation
c:\windows\system32\wbem\wmiprvsd.dll
wbemess 5.2.3785.0 (srv03_rtm.030308-1736) 256.50 KB
(262,656 bytes) 4/29/2003 4:23 PM Microsoft Corporation
c:\windows\system32\wbem\wbemess.dll
ncprov 5.2.3785.0 (srv03_rtm.030308-1736) 43.00 KB
(44,032 bytes) 4/29/2003 4:23 PM Microsoft Corporation
c:\windows\system32\wbem\ncprov.dll
netshell 5.2.3785.0 (srv03_rtm.030308-1736) 1.67 MB
(1,747,456 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\netshell.dll
clusapi 5.2.3785.0 (srv03_rtm.030308-1736) 56.00 KB
(57,344 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\clusapi.dll
hnetcfg 5.2.3785.0 (srv03_rtm.030308-1736) 243.50 KB
(249,344 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\hnetcfg.dll
wininet 6.00.3785.0 (srv03_rtm.030308-1736) 609.00 KB
(623,616 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\wininet.dll
rasdlg 5.2.3785.0 (srv03_rtm.030308-1736) 642.00 KB
(657,408 bytes) 3/9/2003 7:00 AM Microsoft Corporation
c:\windows\system32\rasdlg.dll
pchsvc 5.2.3785.0 (srv03_rtm.030308-1736) 31.50 KB
(32,256 bytes) 4/29/2003 4:27 PM Microsoft Corporation
c:\windows\pchealth\helpctr\binaries\pchsvc.dll

wbemcons 5.2.3785.0 (srv03_rtm.030308-1736) 69.00 KB (70,656 bytes) 4/29/2003 4:23 PM Microsoft Corporation c:\windows\system32\wbem\wbemcons.dll
netcfgx 5.2.3785.0 (srv03_rtm.030308-1736) 726.00 KB (743,424 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\netcfgx.dll
dfssvc 5.2.3785.0 (srv03_rtm.030308-1736) 130.50 KB (133,632 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\dfssvc.exe
resutils 5.2.3785.0 (srv03_rtm.030308-1736) 59.00 KB (60,416 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\resutils.dll
mfc42u 6.05.3014.0 960.00 KB (983,040 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\mfc42u.dll
dns 5.2.3785.0 (srv03_rtm.030308-1736) 412.00 KB (421,888 bytes) 4/30/2003 4:13 PM Microsoft Corporation c:\windows\system32\dns.exe
ismserv 5.2.3785.0 (srv03_rtm.030308-1736) 35.50 KB (36,352 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\ismserv.exe
ismip 5.2.3785.0 (srv03_rtm.030308-1736) 24.00 KB (24,576 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\ismip.dll
ismsmtp 5.2.3785.0 (srv03_rtm.030308-1736) 55.50 KB (56,832 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\ismsmtp.dll
ntfrs 5.2.3785.0 (srv03_rtm.030308-1736) 748.50 KB (766,464 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntfrs.exe
dbghelp 5.2.3785.0 (srv03_rtm.030308-1736) 620.00 KB (634,880 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\dbghelp.dll
ntfrsapi 5.2.3785.0 (srv03_rtm.030308-1736) 56.00 KB (57,344 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntfrsapi.dll
tcpsvcs 5.2.3785.0 (srv03_rtm.030308-1736) 21.00 KB (21,504 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\tcpsvcs.exe
dhcpssvc 5.2.3785.0 (srv03_rtm.030308-1736) 253.50 KB (259,584 bytes) 4/30/2003 4:10 PM Microsoft Corporation c:\windows\system32\dhcpssvc.dll
dsauth 5.2.3785.0 (srv03_rtm.030308-1736) 24.50 KB (25,088 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\dsauth.dll
binlsvc 5.2.3785.0 (srv03_rtm.030308-1736) 106.50 KB (109,056 bytes) 4/30/2003 4:10 PM Microsoft Corporation c:\windows\system32\binlsvc.dll
explorer 6.00.3785.0 (srv03_rtm.030308-1736) 1,008.50 KB (1,032,704 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\explorer.exe
browseui 6.00.3785.0 (srv03_rtm.030308-1736) 1.01 MB (1,056,768 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\browseui.dll
shdocvw 6.00.3785.0 (srv03_rtm.030308-1736) 1.33 MB (1,393,664 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\shdocvw.dll
apphelp 5.2.3785.0 (srv03_rtm.030308-1736) 122.00 KB (124,928 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\apphelp.dll
themeui 6.00.3785.0 (srv03_rtm.030308-1736) 360.50 KB (369,152 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\themeui.dll
msimg32 5.2.3785.0 (srv03_rtm.030308-1736) 4.50 KB (4,608 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\msimg32.dll
actxprxy 6.00.3785.0 (srv03_rtm.030308-1736) 95.00 KB (97,280 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\actxprxy.dll
linkinfo 5.2.3785.0 (srv03_rtm.030308-1736) 16.50 KB (16,896 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\linkinfo.dll
ntshrui 6.00.3785.0 (srv03_rtm.030308-1736) 136.00 KB (139,264 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntshrui.dll
urlmon 6.00.3785.0 (srv03_rtm.030308-1736) 501.50 KB (513,536 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\urlmon.dll
webcheck 6.00.3785.0 (srv03_rtm.030308-1736) 261.50 KB (267,776 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\webcheck.dll
stobject 5.2.3785.0 (srv03_rtm.030308-1736) 117.50 KB (120,320 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\stobject.dll
batmeter 6.00.3785.0 (srv03_rtm.030308-1736) 28.50 KB (29,184 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\batmeter.dll
powrprof 6.00.3785.0 (srv03_rtm.030308-1736) 14.50 KB (14,848 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\powrprof.dll
browselc 6.00.3785.0 (srv03_rtm.030308-1736) 62.00 KB (63,488 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\browselc.dll
duser 5.2.3785.0 (srv03_rtm.030308-1736) 270.00 KB (276,480 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\duser.dll
shdoclc 6.00.3785.0 (srv03_rtm.030308-1736) 588.50 KB (602,624 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\shdoclc.dll
zipfldr 6.00.3785.0 (srv03_rtm.030308-1736) 316.00 KB (323,584 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\zipfldr.dll
printui 5.2.3785.0 (srv03_rtm.030308-1736) 536.50 KB (549,376 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\printui.dll
cfgmgr32 5.2.3785.0 (srv03_rtm.030308-1736) 17.50 KB (17,920 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\cfgmgr32.dll
notepad 5.2.3785.0 (srv03_rtm.030308-1736) 66.50 KB (68,096 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\notepad.exe
cmd 5.2.3785.0 (srv03_rtm.030308-1736) 373.50 KB (382,464 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\cmd.exe
ntvdm 5.2.3785.0 (srv03_rtm.030308-1736) 408.00 KB (417,792 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntvdm.exe
ntvdmd 5.2.3785.0 (srv03_rtm.030308-1736) 14.50 KB (14,848 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntvdmd.dll
rdpsnd 5.2.3785.0 (srv03_rtm.030308-1736) 18.00 KB (18,432 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\rdpsnd.dll
scredir 5.2.3785.0 (srv03_rtm.030308-1736) 27.00 KB (27,648 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\scredir.dll
msacm32 5.2.3785.0 (srv03_rtm.030308-1736) 21.00 KB (21,504 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\msacm32.drv
msacm32 5.2.3785.0 (srv03_rtm.030308-1736) 67.50 KB (69,120 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\msacm32.dll

imaadp32 5.2.3785.0 (srv03_rtm.030308-1736) 15.50 KB (15,872 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\imaadp32.acm
msadp32 5.2.3785.0 (srv03_rtm.030308-1736) 14.50 KB (14,848 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\msadp32.acm
msg711 5.2.3785.0 (srv03_rtm.030308-1736) 10.00 KB (10,240 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\msg711.acm
msgsm32 5.2.3785.0 (srv03_rtm.030308-1736) 20.50 KB (20,992 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\msgsm32.acm
tssoft32 1.01 9.50 KB (9,728 bytes) 3/9/2003 7:00 AM DSP GROUP, INC. c:\windows\system32\tssoft32.acm
tsd32 1.03 16.50 KB (16,896 bytes) 3/9/2003 7:00 AM DSP GROUP, INC. c:\windows\system32\tsd32.dll
msg723 4.4.4000 116.00 KB (118,784 bytes) 4/29/2003 4:27 PM Microsoft Corporation c:\windows\system32\msg723.acm
msaud32 8.00.00.4487 288.00 KB (294,912 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\msaud32.acm
sl_anet 3.02 84.00 KB (86,016 bytes) 3/9/2003 7:00 AM Sipro Lab Telecom Inc. c:\windows\system32\sl_anet.acm
l3codeca 1, 9, 0, 0305 284.00 KB (290,816 bytes) 3/9/2003 7:00 AM Fraunhofer Institut Integrierte Schaltungen IIS c:\windows\system32\l3codeca.acm
rdpclip 5.2.3785.0 (srv03_rtm.030308-1736) 53.00 KB (54,272 bytes) 4/29/2003 4:23 PM Microsoft Corporation c:\windows\system32\rdpclip.exe
mmcshext 5.2.3785.0 (srv03_rtm.030308-1736) 50.00 KB (51,200 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\mmcshext.dll
hhsetup 5.2.3785.0 (srv03_rtm.030308-1736) 38.00 KB (38,912 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\hhsetup.dll
logon 5.2.3785.0 (srv03_rtm.030308-1736) 497.00 KB (508,928 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\logon.scr
rxapi 1. 0. 2. 2 22.50 KB (23,040 bytes) 5/2/2003 9:21 AM IBM Corporation d:\nttools\rxapi.exe
rexxapi 1. 0. 2. 2 64.00 KB (65,536 bytes) 5/2/2003 9:21 AM IBM Corporation d:\nttools\rexxapi.dll
helpctr 5.2.3785.0 (srv03_rtm.030308-1736) 764.00 KB (782,336 bytes) 4/29/2003 4:27 PM Microsoft Corporation c:\windows\pchealth\helpctr\binaries\helpctr.exe
hcappres 5.2.3785.0 (srv03_rtm.030308-1736) 6.50 KB (6,656 bytes) 4/29/2003 4:27 PM Microsoft Corporation c:\windows\pchealth\helpctr\binaries\hcappres.dll
itss 5.2.3785.0 (srv03_rtm.030308-1736) 119.50 KB (122,368 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\itss.dll
msxml3 8.40.9419.0 1.28 MB (1,337,344 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\msxml3.dll
pchshell 5.2.3785.0 (srv03_rtm.030308-1736) 100.50 KB (102,912 bytes) 4/29/2003 4:27 PM Microsoft Corporation c:\windows\pchealth\helpctr\binaries\pchshell.dll
mlang 6.00.3785.0 (srv03_rtm.030308-1736) 570.00 KB (583,680 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\mlang.dll
mshtml 6.00.3785.0 (srv03_rtm.030308-1736) 2.78 MB (2,915,328 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\mshtml.dll
msimtf 5.2.3785.0 (srv03_rtm.030308-1736) 149.00 KB (152,576 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\msimtf.dll
msctf 5.2.3785.0 (srv03_rtm.030308-1736) 287.00 KB (293,888 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\msctf.dll
jscript 5.6.0.8515 436.00 KB (446,464 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\jscript.dll
msls31 3.10.349.0 147.00 KB (150,528 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\msls31.dll
imm32 5.2.3785.0 (srv03_rtm.030308-1736) 105.50 KB (108,032 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\imm32.dll
mshtmled 6.00.3785.0 (srv03_rtm.030308-1736) 443.00 KB (453,632 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\mshtmled.dll
vbscript 5.6.0.8515 404.00 KB (413,696 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\vbscript.dll
mfc42 6.05.3014.0 960.00 KB (983,040 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\mfc42.dll
msinfo 5.2.3785.0 (srv03_rtm.030308-1736) 358.50 KB (367,104 bytes) 4/29/2003 4:27 PM Microsoft Corporation c:\windows\pchealth\helpctr\binaries\msinfo.dll
riched32 5.2.3785.0 (srv03_rtm.030308-1736) 3.50 KB (3,584 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\riched32.dll
riched20 5.31.23.1218 406.00 KB (415,744 bytes) 3/9/2003 7:00 AM Microsoft Corporation c:\windows\system32\riched20.dll
helpsvc 5.2.3785.0 (srv03_rtm.030308-1736) 720.00 KB (737,280 bytes) 4/29/2003 4:27 PM Microsoft Corporation c:\windows\pchealth\helpctr\binaries\helpsvc.exe

[Services]

Display Name Name State Start Mode Service Type Path Error Control Start Name Tag 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\dllhost.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 Manual Own Process d:\sqllib\\bin\db2dasrrm.exe Normal RUTHLESS-D\tpch 0
DB2 Governor DB2GOVERNOR Stopped Manual Own Process d:\sqllib\bin\db2govds.exe Normal LocalSystem 0
DB2 JDBC Applet Server DB2JDS Stopped Manual Own Process d:\sqllib\bin\db2jds.exe Normal LocalSystem 0
DB2 License Server DB2LICD Stopped Manual Own Process d:\sqllib\bin\db2licd.exe Ignore LocalSystem 0
DB2 Security Server DB2NTSECSERVER Stopped Manual Own Process d:\sqllib\bin\db2sec.exe Normal LocalSystem 0
DB2 Remote Command Server DB2REMOTECMD Running Manual Own Process d:\sqllib\bin\db2rcmd.exe Ignore RUTHLESS-D\tpch 0
Distributed File System Dfs Running Auto Own Process c:\windows\system32\dfssvc.exe Normal LocalSystem 0
DHCP Client Dhcp Running Auto Share Process c:\windows\system32\svchost.exe -k networkservice Normal NT AUTHORITY\NetworkService 0
DHCP Server DHCPServer Running Auto Share Process c:\windows\system32\tcpsvcs.exe Normal LocalSystem 0
Logical Disk Manager Administrative Service dmadmin Stopped Manual Share Process c:\windows\system32\dmadmin.exe /com Normal LocalSystem 0
Logical Disk Managerdmserver Running Auto Share Process c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0
DNS Server DNS Running Auto Own Process c:\windows\system32\dns.exe 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
IMAPI CD-Burning COM Service ImapiService Stopped Disabled Own Process c:\windows\system32\imapi.exe Normal LocalSystem 0
Intersite Messaging IsmServ Running Auto Own Process c:\windows\system32\ismserv.exe Normal LocalSystem 0
Kerberos Key Distribution Center kdc Running Auto 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\llssrv.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
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 0
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 Running Auto 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 c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0
File Replication Service NtFrs Running Auto Own Process c:\windows\system32\ntfrs.exe Ignore LocalSystem 0
NT LM Security Support Provider NtLmSsp Stopped Manual Share 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:\sqllib\bin\db2syscs.exe Normal RUTHLESS-D\tpch 0
DB2 - QUAL-1 QUAL-1 Stopped Manual Own Process d:\sqllib\bin\db2syscs.exe Normal RUTHLESS-D\tpch 0
Remote Access Auto Connection Manager RasAuto Stopped Manual Share Process c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0
Remote Access Connection Manager RasMan Stopped Manual Share 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 Running Auto Share 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 -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 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 Stopped Manual 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
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 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 c:\windows\system32\svchost.exe -k imgsvc Normal NT AUTHORITY\LocalService 0
Microsoft Software Shadow Copy Provider swprv Stopped Manual Own Process c:\windows\system32\svchost.exe -k swprv Normal LocalSystem 0
Performance Logs and Alerts SysmonLog Stopped Manual Own Process c:\windows\system32\smlogsvc.exe Normal NT Authority\NetworkService 0
Telephony TapiSrv Stopped 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
Distributed Link Tracking Server TrkSvr Stopped Disabled Share Process c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0
Distributed Link Tracking Client TrkWks Stopped Manual Share Process c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0
Terminal Services Session Directory Tssdis Stopped Disabled Own 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 Running Auto Share Process c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0
WebClient WebClient Stopped Disabled Share Process c:\windows\system32\svchost.exe -k localservice 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 winmgmt Running Auto Share 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\wmiapsrv.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
Accessories Default User:Accessories Default User
Accessories\Accessibility Default User:Accessories\Accessibility Default User
Accessories\Entertainment Default User:Accessories\Entertainment Default User
Startup Default User:Startup Default User
Accessories All Users:Accessories All Users
Accessories\Accessibility All Users:Accessories\Accessibility All Users
Accessories\Communications All Users:Accessories\Communications All Users
Accessories\Entertainment All Users:Accessories\Entertainment All Users
Accessories\System Tools All Users:Accessories\System Tools All Users
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
IBM DB2\Development Tools All Users:IBM DB2\Development Tools All Users
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
IBM DB2\Set-up Tools All Users:IBM DB2\Set-up Tools All Users
Microsoft Visual C++ 6.0 All Users:Microsoft Visual C++ 6.0 All Users
Microsoft Visual C++ 6.0\Microsoft Visual C++ 6.0 Tools All Users:Microsoft Visual C++ 6.0\Microsoft Visual C++ 6.0 Tools All Users
ServeRAID Manager All Users:ServeRAID Manager All Users
Startup All Users:Startup All Users
Accessories 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 RUTHLESS-D\tpch:Accessories RUTHLESS-D\tpch
Accessories\Accessibility RUTHLESS-D\tpch:Accessories\Accessibility RUTHLESS-D\tpch
Accessories\Entertainment RUTHLESS-D\tpch:Accessories\Entertainment RUTHLESS-D\tpch
Administrative Tools RUTHLESS-D\tpch:Administrative Tools RUTHLESS-D\tpch
Startup RUTHLESS-D\tpch:Startup RUTHLESS-D\tpch

[Startup Programs]

Program Command User NameLocation
desktop desktop.ini NT AUTHORITY\SYSTEM Startup
desktop desktop.ini RUTHLESS-D\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
6/17/2003 12:56 PM dsrestor The DSRestore filter started successfully.

[Internet Settings]

[Internet Explorer]

[Following are sub-categories of this main category]
[Summary]

Item Value
Version 6.0.3785.0
Build 63785
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 6.0.3785.0 95 KB 3/9/2003 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 6.0.3785.0 94 KB 3/9/2003 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 6.0.3785.0 90 KB 3/9/2003 8:00:00 AM C:\WINDOWS\system32 Microsoft Corporation
browselc.dll 6.0.3785.0 62 KB 3/9/2003 8:00:00 AM C:\WINDOWS\system32 Microsoft Corporation
browseui.dll 6.0.3785.0 1,032 KB 3/9/2003 8:00:00 AM C:\WINDOWS\system32 Microsoft Corporation
cdfview.dll 6.0.3785.0 144 KB 3/9/2003 8:00:00 AM C:\WINDOWS\system32 Microsoft Corporation
comctl32.dll 5.82.3785.0 561 KB 3/9/2003 8:00:00 AM C:\WINDOWS\system32 Microsoft Corporation
dxtrans.dll 6.3.3785.0 198 KB 3/9/2003 8:00:00 AM C:\WINDOWS\system32 Microsoft Corporation
dxtmsft.dll 6.3.3785.0 344 KB 3/9/2003 8:00:00 AM C:\WINDOWS\system32 Microsoft Corporation
iecont.dll <File Missing> Not Available Not Available Not Available Not Available
iecontlc.dll<File Missing> Not Available Not Available Not Available Not Available
iedkcs32.dll 16.0.3785.0 300 KB 3/9/2003 8:00:00 AM C:\WINDOWS\system32 Microsoft Corporation
iepeers.dll 6.0.3785.0 230 KB 3/9/2003 8:00:00 AM C:\WINDOWS\system32 Microsoft Corporation
iesetup.dll 6.0.3785.0 59 KB 3/9/2003 8:00:00 AM C:\WINDOWS\system32 Microsoft Corporation
ieuinit.inf Not Available 20 KB 3/9/2003 8:00:00 AM C:\WINDOWS\system32 Not Available
iexplore.exe 6.0.3785.0 90 KB 3/9/2003 8:00:00 AM C:\Program Files\Internet Explorer Microsoft Corporation
imgutil.dll 5.2.3785.0 35 KB 3/9/2003 8:00:00 AM C:\WINDOWS\system32 Microsoft Corporation

inetcpl.cpl 6.0.3785.0 303 KB 3/9/2003 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation
inetcplc.dll6.0.3785.0 109 KB 3/9/2003 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation
inseng.dll 6.0.3785.0 72 KB 3/9/2003 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation
mlang.dll 6.0.3785.0 570 KB 3/9/2003 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation
msencode.dll 2002.10.4.0 112 KB 3/9/2003
8:00:00 AM C:\WINDOWS\system32 Not Available
mshta.exe 6.0.3785.0 26 KB 3/9/2003 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation
mshtml.dll 6.0.3785.0 2,847 KB 3/9/2003 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation
mshtml.tlb 6.0.3785.0 1,319 KB 3/9/2003 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation
mshtmled.dll 6.0.3785.0 443 KB 3/9/2003 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation
mshtmler.dll 6.0.3785.0 55 KB 3/9/2003 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation
msident.dll6.0.3785.0 47 KB 3/9/2003 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation
msidntld.dll 6.0.3785.0 15 KB 3/9/2003 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation
msieftp.dll 6.0.3785.0 230 KB 3/9/2003 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation
msrating.dll 6.0.3785.0 132 KB 3/9/2003 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation
mstime.dll 6.0.3785.0 491 KB 3/9/2003 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation
occache.dll 6.0.3785.0 89 KB 3/9/2003 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation
proctexe.ocx 6.3.3785.0 78 KB 3/9/2003 8:00:00 AM
C:\WINDOWS\system32 Intel Corporation
sendmail.dll 6.0.3785.0 52 KB 3/9/2003 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation
shdoclc.dll 6.0.3785.0 589 KB 3/9/2003 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation
shdocvw.dll 6.0.3785.0 1,361 KB 3/9/2003 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation
shfolder.dll 6.0.3785.0 23 KB 3/9/2003 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation
shlwapi.dll6.0.3785.0 281 KB 3/9/2003 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation
tdc.ocx 1.3.0.3130 58 KB 3/9/2003 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation
url.dll 6.0.3785.0 36 KB 3/9/2003 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation
urlmon.dll 6.0.3785.0 502 KB 3/9/2003 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation
webcheck.dll 6.0.3785.0 262 KB 3/9/2003 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation
wininet.dll 6.0.3785.0 609 KB 3/9/2003 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

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/sqllib/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"};
$ddlpath=             $ENV{"TPCD_DDLPATH"};
$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"};
$logfilsiz=          $ENV{"TPCD_LOGFILSIZ"};
$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"};
      $tbspddl=      $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"};
      $tbspddl=      $ENV{"TPCD_TBSP_DDL"};
}


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";
}

#-------------------------------------------------------------------------#
#  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
      $rc=&dodb_noconn("db2set DB2OPTIONS=\"  -t -v +c\";db2set
DB2NTNOCACHE=ON",$all_ln);
   }
   else{
      $rc=&dodb_noconn("db2set DB2OPTIONS=\\\"  -t -v
+c\\\";db2set DB2NTNOCACHE=ON",$all_ln);
   }
}
else{
   if (($mode eq "uni") || ($mode eq "smp")){
      $rc=&dodb_noconn("db2set DB2OPTIONS=\"-t -v +c\"",$all_ln);
   }
   else{
      $rc=&dodb_noconn("db2set DB2OPTIONS=\\\"-t -v
+c\\\"",$all_ln);
   }
}
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 \" call ${ddlpath}${delim}$loadsetScript\" ";
         $rc=system(" db2_all \" call
${ddlpath}${delim}$loadsetScript\" ");
      }
   }
   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 specifed 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 ( $addRI ne "NULL" ){
   &outtime("*** Adding RI contraints started");
   &dodb2file($dbname,"$ddlpath${delim}$addRI",$once);
   &outtime("*** Adding RI contraints 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 buildptpcdbatch 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 tablesdb $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_ln);
      }
      $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 logbufsz
128";
      $rc = &dodb_noconn("$setLogString",$all_ln);

}

#-------------------------------------------------------------------------#
# 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_ln);
      &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 $rc;
    &outtime("*** Creating the bufferpools");
    if ( $buffpooldef ne "NULL" ){
        #run the create bufferpool ddl
        $rc =
&dodb2file($dbname,"$ddlpath${delim}$buffpooldef",$once);
    }
}

#-------------------------------------------------------------------------#
# Function: createTablespaces                                             #
#-------------------------------------------------------------------------#
sub createTablespaces{
    &outtime("*** Ready to start creating the tablespaces");
    # setup required variables
    my $rc;
    $rc = &dodb2file($dbname,"$ddlpath${delim}$tbspddl",$once);
    ($rc == 0) || return $rc;
    # 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
        $rc =
&dodb2file($dbname,"$ddlpath${delim}$stagingTbl",$once);
        ($rc == 0) || return $rc;
        if ( $qual ne "QUAL" ){
            if ( $preloadSampleUF ne "NULL" ){
                # preload the sample UF data for statistics
gathering
                $rc = system ("perl
$ddlpath${delim}$preloadSampleUF");
                #($rc == 0) || return $rc;
            }
            if ( $deleteSampleUF ne "NULL" ){
                # delete the sample rows now that stats have been
gathered
                $rc =
&dodb2file($dbname,"$ddlpath${delim}$deleteSampleUF",$once);
                #($rc == 0) || return $rc;
            }
        }
    }
}

#-------------------------------------------------------------------------#
# Function: createTables                                                  #
#-------------------------------------------------------------------------#
sub createTables{
    my $rc;
    $rc = &dodb2file($dbname,"$ddlpath${delim}$ddl",$once);
    ($rc == 0) || return $rc;
    # update the locksize on the non-updated tables to be table level
locking
    # update the tables for appendmode
    if ($appendOn eq "yes"){
        $rc = &dodb_conn($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{
        $rc = &dodb_conn($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 $rc;
    &outtime("*** Starting to create $indexType indexes");
    if( $indexType eq "extra"){
        $rc =
&dodb2file($dbname,"$ddlpath${delim}$extraindex",$once);
    }elsif ($indexType eq "early" || $indexType eq "normal"){
        $rc =
&dodb2file($dbname,"$ddlpath${delim}$indexddl",$once);
    }
    &outtime("*** Create $indexType index completed");
#    $rc = &dodb_conn($dbname,
        "db2 alter table tpcd.orders add primary key (o_orderkey) $sep
\
        db2 alter table tpcd.lineitem add primary key (l_orderkey,
l_linenumber)",
#        $once);

    return $rc;
}



#-------------------------------------------------------------------------#
# Function: setLoadConfig                                                 #
#-------------------------------------------------------------------------#
sub setLoadConfig{
```

```
	&outtime("*** Setting LOAD configuration.");
	my $rc;
	my $buffpage;
	my $sortheap;
	my $sheapthres;
	my $util_heap_sz;
	my $ioservers;
	my $ioclnrs=		1;
	my $chngpgs=		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_ln);
	}
	else{

		print "$ddlpath${delim}$loadconfigfile\n";
		$rc =
&dodb2file_noconn("$ddlpath${delim}$loadconfigfile",$all_ln);
		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_ln);
    &dodb_noconn("db2 update database manager configuration
using max_querydegree $smpdegree",$once);
    &dodb2file_noconn("${ddlpath}${delim}$configfile",$all_ln);
    &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";
    }
}

#-------------------------------------------------------------------------#
# Function: createBackup                                                  #
#-------------------------------------------------------------------------#
sub createBackup{
    my $rc;
    &dodb_noconn("db2 update database configuration for $dbname
using LOGRETAIN yes",$all_ln);
    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_ln);
    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_ln);
        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 ( $nodegroupdef ne "NULL" ){
          print "   Nodegroup ddl in $ddlpath${delim}$nodegroupdef \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}$configfile\n";
        }
        else{
          print "   Database Configuration paramters taken from
$ddlpath${delim}dss.dbconfig${sfReal}GB\n";
          $configfile="dss.dbconfig${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";
          $configfile="dss.dbmconfig${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 = $sortheap \n";
      print "     SHEAPTHRES = $sheapthres\n";
      print "     NUM_IOSERVERS = $ioservers\n";
      print "     NUM_IOCLEANERS = $ioclnrs\n";
      print "     CHNGPGS_THRESH = $chngpgs\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;
```

affinity_8mln.bat

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

bkuptestdb.bat

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

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

db2start
set db2node=1
db2 backup database tpcd to o:\backup, p:\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, p:\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, p:\backup WITH 16
BUFFERS PARALLELISM 8 without prompting
time < xxx
rem  End of backup node 3
db2stop
time < xxx

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

db2start
set db2node=5
db2 backup database tpcd to m:\backup, n:\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 m:\backup, n:\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 m:\backup, n:\backup WITH 16
BUFFERS PARALLELISM 8 without prompting
time < xxx
rem  End of backup node 7
time < xxx
```

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_PKSK" 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 6 ;
commit work;


values(current timestamp);
create unique index tpcd.l_ok_ln on tpcd.lineitem (L_orderkey,
l_linenumber) 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_linenumber);
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_ tablespaces_8mln_mp

```
create regular tablespace small_tables
  in nodegroup catalog_node
  managed by system
  using ('d:\MP\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_D2'       5900M,
          device 'd:\MP\li_data_D26'      5900M,
          device 'd:\MP\li_data_D9'       5900M) on node(4)
  using ( device 'd:\MP\li_data_D3'       5900M,
          device 'd:\MP\li_data_D27'      5900M,
```

```
        device 'd:\MP\li_data_D10'      5900M) on node(5)
  using ( device 'd:\MP\li_data_D5'       5900M,
        device 'd:\MP\li_data_D29'      5900M,
        device 'd:\MP\li_data_D12'      5900M) on node(6)
  using ( device 'd:\MP\li_data_D6'       5900M,
        device 'd:\MP\li_data_D30'      5900M,
        device 'd:\MP\li_data_D13'      5900M) on node(7)
  using ( device 'd:\MP\li_data_D15'      5900M,
        device 'd:\MP\li_data_D21'      5900M) on node(0)
  using ( device 'd:\MP\li_data_D16'      5900M,
        device 'd:\MP\li_data_D22'      5900M) on node(1)
  using ( device 'd:\MP\li_data_D18'      5900M,
        device 'd:\MP\li_data_D24'      5900M) on node(2)
  using ( device 'd:\MP\li_data_D19'      5900M,
        device 'd:\MP\li_data_D25'      5900M) on node(3)
  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_D2'       995M,
        device 'd:\MP\LI_INDEX_D26'      995M,
        device 'd:\MP\LI_INDEX_D9'       995M) on node(4)
  using ( device 'd:\MP\LI_INDEX_D3'       995M,
        device 'd:\MP\LI_INDEX_D27'      995M,
        device 'd:\MP\LI_INDEX_D10'      995M) on node(5)
  using ( device 'd:\MP\LI_INDEX_D5'       995M,
        device 'd:\MP\LI_INDEX_D29'      995M,
        device 'd:\MP\LI_INDEX_D12'      995M) on node(6)
  using ( device 'd:\MP\LI_INDEX_D6'       995M,
        device 'd:\MP\LI_INDEX_D30'      995M,
        device 'd:\MP\LI_INDEX_D13'      995M) on node(7)
  using ( device 'd:\MP\LI_INDEX_D15'      995M,
        device 'd:\MP\LI_INDEX_D21'      995M) on node(0)
  using ( device 'd:\MP\LI_INDEX_D16'      995M,
        device 'd:\MP\LI_INDEX_D22'      995M) on node(1)
  using ( device 'd:\MP\LI_INDEX_D18'      995M,
        device 'd:\MP\LI_INDEX_D24'      995M) on node(2)
  using ( device 'd:\MP\LI_INDEX_D19'      995M,
        device 'd:\MP\LI_INDEX_D25'      995M) on node(3)
  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_D2'    2750M,
        device 'd:\MP\OTHER_TABLES_D26'   2750M,
        device 'd:\MP\OTHER_TABLES_D9'    2750M) on node(4)
  using ( device 'd:\MP\OTHER_TABLES_D3'    2750M,
        device 'd:\MP\OTHER_TABLES_D27'   2750M,
        device 'd:\MP\OTHER_TABLES_D10'   2750M) on node(5)
  using ( device 'd:\MP\OTHER_TABLES_D5'    2750M,
        device 'd:\MP\OTHER_TABLES_D29'   2750M,
        device 'd:\MP\OTHER_TABLES_D12'   2750M) on node(6)
  using ( device 'd:\MP\OTHER_TABLES_D6'    2750M,
        device 'd:\MP\OTHER_TABLES_D30'   2750M,
        device 'd:\MP\OTHER_TABLES_D13'   2750M) on node(7)
  using ( device 'd:\MP\OTHER_TABLES_D15'   2750M,
        device 'd:\MP\OTHER_TABLES_D21'   2750M) on node(0)
  using ( device 'd:\MP\OTHER_TABLES_D16'   2750M,
        device 'd:\MP\OTHER_TABLES_D22'   2750M) on node(1)
  using ( device 'd:\MP\OTHER_TABLES_D18'   2750M,
        device 'd:\MP\OTHER_TABLES_D24'   2750M) on node(2)
  using ( device 'd:\MP\OTHER_TABLES_D19'   2750M,
        device 'd:\MP\OTHER_TABLES_D25'   2750M) on node(3)
  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_D2'     790M,
        device 'd:\MP\OTHER_INDEX_D26'    790M,
        device 'd:\MP\OTHER_INDEX_D9'     790M) on node(4)
  using ( device 'd:\MP\OTHER_INDEX_D3'     790M,
        device 'd:\MP\OTHER_INDEX_D27'    790M,
        device 'd:\MP\OTHER_INDEX_D10'    790M) on node(5)
  using ( device 'd:\MP\OTHER_INDEX_D5'     790M,
        device 'd:\MP\OTHER_INDEX_D29'    790M,
        device 'd:\MP\OTHER_INDEX_D12'    790M) on node(6)
  using ( device 'd:\MP\OTHER_INDEX_D6'     790M,
        device 'd:\MP\OTHER_INDEX_D30'    790M,
        device 'd:\MP\OTHER_INDEX_D13'    790M) on node(7)
  using ( device 'd:\MP\OTHER_INDEX_D15'    790M,
        device 'd:\MP\OTHER_INDEX_D21'    790M) on node(0)
  using ( device 'd:\MP\OTHER_INDEX_D16'    790M,
        device 'd:\MP\OTHER_INDEX_D22'    790M) on node(1)
  using ( device 'd:\MP\OTHER_INDEX_D18'    790M,
        device 'd:\MP\OTHER_INDEX_D24'    790M) on node(2)
  using ( device 'd:\MP\OTHER_INDEX_D19'    790M,
        device 'd:\MP\OTHER_INDEX_D25'    790M) on node(3)
  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_D2'        11500M,
        device 'd:\MP\temp_D26'       11500M,
        device 'd:\MP\temp_D9'        11500M) on node(4)
  using ( device 'd:\MP\temp_D3'        11500M,
        device 'd:\MP\temp_D27'       11500M,
        device 'd:\MP\temp_D10'       11500M) on node(5)
  using ( device 'd:\MP\temp_D5'        11500M,
        device 'd:\MP\temp_D29'       11500M,
        device 'd:\MP\temp_D12'       11500M) on node(6)
  using ( device 'd:\MP\temp_D6'        11500M,
        device 'd:\MP\temp_D30'       11500M,
        device 'd:\MP\temp_D13'       11500M) on node(7)
  using ( device 'd:\MP\temp_D15'       11500M,
        device 'd:\MP\temp_D21'       11500M) on node(0)
  using ( device 'd:\MP\temp_D16'       11500M,
        device 'd:\MP\temp_D22'       11500M) on node(1)
  using ( device 'd:\MP\temp_D18'       11500M,
        device 'd:\MP\temp_D24'       11500M) on node(2)
  using ( device 'd:\MP\temp_D19'       11500M,
        device 'd:\MP\temp_D25'       11500M) on node(3)
  bufferpool   BP16K
  extentsize    24
  prefetchsize   48
  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);


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);



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);

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;
```

db2nodes.cfg

```
0 ruthless1 RUTHLESS1 0
1 ruthless1 RUTHLESS1 1
2 ruthless1 RUTHLESS1 2
3 ruthless1 RUTHLESS1 3
4 ruthless1 RUTHLESS1 4
5 ruthless1 RUTHLESS1 5
6 ruthless1 RUTHLESS1 6
7 ruthless1 RUTHLESS1 7
```

load.db2set_8mln.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.db2set_8mln.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
```

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
```

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_8mln.bat

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

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 MESSAGES d:\tmp0\supplier.msg
   REPLACE INTO TPCD.SUPPLIER NONRECOVERABLE
   CPU_PARALLELISM 8 partitioned db config mode load_only
output_dbpartnums (0,1,2,3,4,5,6,7)
   part_file_location w:\100GB_8mln_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 of del
  MODIFIED BY COLDEL| FASTPARSE MESSAGES
d:\tmp0\orders.msg
  REPLACE INTO TPCD.orders NONRECOVERABLE
CPU_PARALLELISM 8
  partitioned db config mode load_only output_dbpartnums
(0,1,2,3,4,5,6,7)
  part_file_location w:\100GB_8mln_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 MESSAGES
d:\tmp0\lineitem.msg
    REPLACE INTO TPCD.lineitem NONRECOVERABLE
CPU_PARALLELISM 8
    partitioned db config mode load_only output_dbpartnums
(0,1,2,3,4,5,6,7)
    part_file_location w:\100GB_8mln_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 MESSAGES d:\tmp0\customer.msg
   REPLACE INTO TPCD.customer NONRECOVERABLE
CPU_PARALLELISM 8
   partitioned db config mode load_only output_dbpartnums
(0,1,2,3,4,5,6,7)
   part_file_location w:\100GB_8mln_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 MESSAGES
d:\tmp0\part.msg
   REPLACE INTO TPCD.part NONRECOVERABLE
CPU_PARALLELISM 8
   partitioned db config mode load_only output_dbpartnums
(0,1,2,3,4,5,6,7)
   part_file_location w:\100GB_8mln_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 MESSAGES
d:\tmp0\partsupp.msg
        REPLACE INTO TPCD.partsupp NONRECOVERABLE
CPU_PARALLELISM 8
        partitioned db config mode load_only output_dbpartnums
(0,1,2,3,4,5,6,7)
        part_file_location w:\100GB_8mln_flatfiles;
commit work;
values(current timestamp, 'TS*** Load done lineitem ');
LOAD FROM w:\100GB_8mln_flatfiles\region.tbl OF DEL
    MODIFIED BY COLDEL| FASTPARSE MESSAGES
d:\tmp\region.msg
    REPLACE INTO TPCD.REGION STATISTICS NO
NONRECOVERABLE;
commit work;

values(current timestamp, 'TS*** Load done region ');
LOAD FROM w:\100GB_8mln_flatfiles\nation.tbl OF DEL
    MODIFIED BY COLDEL| FASTPARSE 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
chngpgs_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
stmtheap 4096
util_heap_sz 32000
applheapsz 768
app_ctl_heap_sz 1024
dft_degree  2;
get database configuration for tpcd;

--connect reset;
```

load_dbmcfg_8mln

```
update database manager configuration using
cpuspeed -1
comm_bandwidth 1
sheapthres 430000
agent_stack_sz 16
aslheapsz 15
```

```
rqrioblk 32767
intra_parallel yes
max_querydegree -1
maxagents 200
num_poolagents 4
num_initagents 4
diaglevel 3
svcename db2_db2_end;

get database manager configuration;

--connect reset;
```

run.dbcfg_8mln

```
update database configuration for tpcd using
buffpage  60000
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   8000
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 80;

get database configuration for tpcd;

--connect reset;
```

run.dbmcfg_8mln

```
update database manager configuration using
cpuspeed -1
comm_bandwidth 1
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
svcename DB2_TPCH_END
diaglevel 3;
get database manager configuration;
```

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 e:\logs
set db2node=1
db2 update db cfg for %1 using newlogpath f:\logs
set db2node=2
db2 update db cfg for %1 using newlogpath g:\logs
set db2node=3
db2 update db cfg for %1 using newlogpath h:\logs
set db2node=4
db2 update db cfg for %1 using newlogpath i:\logs
set db2node=5
db2 update db cfg for %1 using newlogpath j:\logs
set db2node=6
db2 update db cfg for %1 using newlogpath k:\logs
set db2node=7
db2 update db cfg for %1 using newlogpath l:\logs
db2stop
set db2node=0
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 TPCH)
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=100                       # 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\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_8mln_mp # 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=100GB_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>/sqllib/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:\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=5                    # number of streams for the
throughput test
TPCD_FLATFILES=M:\100GB_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=LoadSampleUFDat
a  # 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=DELETEDUMMYUFD
ATA.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=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=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
                                     # 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=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:\sqllib\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
```

Appendix C: Qualification Query Output

Qualification Queries

Query 1

Start timestamp 06/11/03 23:39:17.437

-- 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       56586554400.728
53758257134.869      55909065222.829              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      111701729697.737
106118230307.604     110367043872.494              25.502
38249.118             0.050     2920374
R            F                37719753.000       56568041380.900
53741292684.604      55889619119.832              25.506
38250.855             0.050     1478870


Number of rows retrieved is:      4
------------------------------------------------

Stop timestamp 06/11/03 23:39:28.609
Query Time =           11.2 secs
```

Query 2

Start timestamp 06/11/03 23:37:25.093

-- 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,vZGiwu2FWEJoLDx04               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


...Rows deleted

        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             2Z0JGkiv01Y00oCFwUGfviIbhzCdy
        16-464-517-8943  carefully bold accounts doub



Number of rows retrieved is:    100
------------------------------------------------


Stop timestamp 06/11/03 23:37:25.890
Query Time =            0.8 secs
```

Query 3

```
Start timestamp 06/11/03 23:38:50.156

------------------------------------------------

-- 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 06/11/03 23:39:02.296
Query Time =            12.1 secs
```

Query 4

```
Start timestamp 06/11/03 23:39:05.843

------------------------------------------------

-- 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 06/11/03 23:39:15.921
Query Time =           10.1 secs
```

Query 5

```
Start timestamp 06/11/03 23:39:45.812

------------------------------------------------

-- 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 06/11/03 23:39:57.921
Query Time =           12.1 secs
```

Query 6

```
Start timestamp 06/11/03 23:37:41.203

------------------------------------------------

-- 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 06/11/03 23:37:41.875
Query Time =           0.7 secs
```

Query 7

```
Start timestamp 06/11/03 23:39:57.921

------------------------------------------------

-- 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 06/11/03 23:40:05.109
Query Time =             7.2 secs
```

Query 8

```
Start timestamp 06/11/03 23:38:04.796

------------------------------------------------

-- 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 06/11/03 23:38:16.671
Query Time =            11.9 secs
```

Query 9

```
Start timestamp 06/11/03 23:37:25.890

------------------------------------------------

-- 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      31342867.235
ALGERIA                     1997      57138193.023
ALGERIA                     1996      56140140.133
ALGERIA                     1995      53051469.653
ALGERIA                     1994      53867582.129

... Rows deleted

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 06/11/03 23:37:38.703
Query Time =          12.8 secs
```

Query 10

```
Start timestamp 06/11/03 23:39:28.609

------------------------------------------------

-- 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                734235.245
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
64EaJ5vMAHWJlBOxJklpNc2RJiWE              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

... Rows Deleted

    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
ChqEoK43OysjdHbtKCp6dKqjNyvvi9            19-955-562-2398
regular, unusual dependencies boost slyly; ironic attainments nag
fluffily into the unusual packages?
    146149  Customer#000146149                557254.986
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                   HgiV0phqhaIa9aydNoIlb
  29-915-458-2654  instructions nag quickly. furiously bold accounts
cajol


Number of rows retrieved is:    20
------------------------------------------------


Stop timestamp 06/11/03 23:39:37.296
```

Query Time = 8.7 secs

Query 11

Start timestamp 06/11/03 23:39:15.921

-- 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

... Rows Deleted

154731	7888301.330
101674	7879324.600
51968	7879102.210
72073	7877736.110
5182	7874521.730

Number of rows retrieved is: 1048

Stop timestamp 06/11/03 23:39:16.750
Query Time = 0.8 secs

Query 12

Start timestamp 06/11/03 23:40:05.109

-- 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 06/11/03 23:40:14.656
Query Time = 9.5 secs

Query 13

Start timestamp 06/11/03 23:38:46.078

-- 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

... Rows Deleted

        37            7
        40            4
        38            4
        39            2
        41            1


Number of rows retrieved is:     42
-----------------------------------------------


Stop timestamp 06/11/03 23:38:50.156
Query Time =              4.1 secs
```

Query 14

```
Start timestamp 06/11/03 23:37:23.640

-----------------------------------------------
--#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 06/11/03 23:37:25.093
Query Time =              1.5 secs
```

Query 15a

```
Start timestamp 06/11/03 23:39:16.750

-----------------------------------------------

-- 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 06/11/03 23:39:17.437
Query Time =              0.7 secs
```

Query 16

```
Start timestamp 06/11/03 23:39:04.718

-----------------------------------------------

-- 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

Rows Deleted

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 06/11/03 23:39:05.843
Query Time =            1.1 secs
```

Query 17

```
Start timestamp 06/11/03 23:37:41.875

-----------------------------------------------

-- 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 06/11/03 23:37:49.406
Query Time =            7.5 secs
```

Query 18

```
Start timestamp 06/11/03 23:37:49.406

-----------------------------------------------

-- 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

...Rows Deleted

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 06/11/03 23:38:04.796
Query Time =             15.4 secs
```

Query 19

```
Start timestamp 06/11/03 23:39:37.296

-------------------------------------------------

-- 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 06/11/03 23:39:45.812
Query Time =             8.5 secs
```

Query 20

```
Start timestamp 06/11/03 23:37:38.703

-------------------------------------------------

-- 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        Br7e1nnt1yxrw6ImgpJ7YdhFDjuBf

... Rows Deleted

Supplier#000009862        rJzweWeN58
Supplier#000009868        ROjGgx5gvtkmnUUoeyy7v
Supplier#000009869
ucLqxzrpBTRMewGSM29t0rNTM30g1Tu3Xgg3mKag
Supplier#000009899        7XdpAHrzr1t,UQFZE
Supplier#000009974        7wJ,J5DKcxSU4Kp1cQLpbcAvB5AsvKT



Number of rows retrieved is:    204
-------------------------------------------------


Stop timestamp 06/11/03 23:37:41.203
Query Time =            2.5 secs
```

Query 21

```
Start timestamp 06/11/03 23:38:16.671

-------------------------------------------------

-- 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

... Rows Deleted

Supplier#000001916                12
Supplier#000001925                12
Supplier#000002039                12
Supplier#000002357                12
Supplier#000002483                12


Number of rows retrieved is:    100
-------------------------------------------------


Stop timestamp 06/11/03 23:38:46.078
Query Time =            29.4 secs
```

Query 22

```
Start timestamp 06/11/03 23:39:02.296

-----------------------------------------------
```

```
-- 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 06/11/03 23:39:04.718
Query Time =             2.4 secs
```

First 10 Rows of the Database

```
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_MFGR
         P_BRAND   P_TYPE                 P_SIZE      P_CONTAINER
P_RETAILPRICE          P_COMMENT
----------- ----------------------------------------------------
------------------------- ---------- ------------------------- ----------- -----------
------------------------ ----------------------
        11 chocolate turquoise sandy snow misty
Manufacturer#2           Brand#25   STANDARD BURNISHED
NICKEL         43 WRAP BOX     +9.11010000000000E+002
furiousl
        37 turquoise ivory orange sandy maroon
Manufacturer#4           Brand#45   LARGE POLISHED TIN
48 JUMBO BOX     +9.37030000000000E+002 blithely regular
        43 medium khaki chocolate rosy blush
Manufacturer#4           Brand#44   PROMO POLISHED STEEL
    5 WRAP CASE     +9.43040000000000E+002 carefully iro
        47 sky firebrick red linen dim                    Manufacturer#4
       Brand#45   LARGE BURNISHED BRASS              14 JUMBO
PACK    +9.47040000000000E+002 bold, unusual a
        50 yellow cornflower royal blush almond
Manufacturer#3           Brand#33   LARGE ANODIZED TIN
 25 WRAP PKG      +9.50050000000000E+002 regular dinos ar
        55 antique cream pale tomato rose
Manufacturer#2           Brand#23   ECONOMY BRUSHED COPPER
          9 MED BAG       +9.55050000000000E+002 furiously
        68 bisque frosted pale puff sandy                 Manufacturer#1
         Brand#11   PROMO ANODIZED STEEL              10 WRAP
BOX      +9.68060000000000E+002 carefully
        76 bisque light yellow puff salmon
Manufacturer#3           Brand#34   MEDIUM BRUSHED COPPER
        9 SM PKG        +9.76070000000000E+002 final
        87 pale khaki sandy antique black
Manufacturer#4           Brand#41   LARGE PLATED STEEL
 41 WRAP PACK     +9.87080000000000E+002 slyly even instruction

        89 ghost khaki lawn pale dim                      Manufacturer#5
         Brand#53   STANDARD BURNISHED STEEL           7 MED
JAR       +9.89080000000000E+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

----------- ------------------------- ----------------------------------------
----------- --------------- -------------------------
---------------------------------------------------------------------------------------------
---------------
      3367 Supplier#000003367
P76gRphGSXxNx411Sm5cOU5Ris                  17
27-129-619-8835   +9.58224000000000E+003 bold packages nag
blithely. furiously regu
      3516 Supplier#000003516        ,SQUAz8ZNT g
      17 27-431-841-3625   +3.49580000000000E+003 express
platelets affix. quickly regu
      3518 Supplier#000003518        Vmz9fkGm
rRd9,IFekdsCJnWAwRikASxDykp9            17 27-302-463-9407
+5.81900000000000E+001 regular, bold warhorses cajole toward the
quickly unusual patterns.
      3552 Supplier#000003552        PMWgNdKj2YRTgdEaZZdqizMj
                  17 27-230-716-8375   +5.01977000000000E+003
furiously unusual accounts integrate quickly after the bl

      3956 Supplier#000003956        r0QOyhOZ ov 7Dn
       17 27-326-227-8739   +9.53682000000000E+003 ironic ideas
haggle fluffily according
      4523 Supplier#000004523        SoBu9nw6O2tVpM
        17 27-301-120-3294   +2.65035000000000E+003 slyly bold
packages integrate along the platelets. boldly express p

      4526 Supplier#000004526
idM5u6EjYXrwcM76vkb89JKLkvxV3d5DGYZWREbT          17
27-660-230-2130   +3.23631000000000E+003 platelets wake
furiously regular pinto
      4631 Supplier#000004631        ,Se21e0egM,Vodq
ABbNjMf25kTmWgkDqQjA1f           17 27-251-691-3991
+6.58473000000000E+003 quickly special deposits cajole slyly. slyly
even foxes are about the bold packages. final as
      4719 Supplier#000004719
yR99LcVeVOxcix1XIiMb7UpFYblyQaEe aMg             17
27-773-971-6110   +8.48912000000000E+003 express requests boost
evenly after the unusual theodolites-- quickly regular depos
      4917 Supplier#000004917        Rhe84NYGtbpBE4xv
        17 27-143-767-1631   +6.77345000000000E+003 unusual,
even deposits are quickly packages! furiously regular instructi


  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   +7.09870000000000E+002 final packages
mold after the carefully unusual requests. quickly fi

       11      250012        4729   +8.94900000000000E+002 regular
packages sleep carefully fluffily ironic ac

       11      500012        3708   +8.18740000000000E+002 slyly
pending theodolites wake quickly unusual, express accounts. fluffily
regular requests cajole furiously quickly even dugouts. slyly bold
platelets
       11      750012        3213   +4.71980000000000E+002 ideas nag
regular instructions. regular, thin pinto beans unwind furiously ironic
accounts. quickly express platele

       37          38        7171   +8.24960000000000E+002 pending
packages wake quietly. special, even accounts haggle unusual

       37      250038        5542   +1.26590000000000E+002 ironic,
ironic packages are slyly. furiously regular excuses cajole blithely
quickly ironic dependencies. regular packages use carefully furiously
pending depths
       37      500038        7113   +1.57200000000000E+001 regular,
final requests nag across the regular, even requests. carefully regular
accounts cajole carefully carefully ruthless dep

       37      750038        1449   +7.45640000000000E+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   +8.05780000000000E+002 final, express
dependencies sleep according to the express requests. bold, regular
accounts detect outside the slyly

      43     250044       6770   +4.93190000000000E+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_MKTSEGMENT C_COMMENT

----------- ------------------------- ----------------------------------------
----------- --------------- ----------------------- ------------
--------------------------------------------------------------------------------
--------------------------------
      1282 Customer#000001282        qeYHABkf21,5C5OC5it6q
             14 24-750-627-7414   +8.99882000000000E+003
FURNITURE    blithely ironic forges cajole

      1375 Customer#000001375        lpKhW7g
QK7Y13sxKlRvRYI7SItbTbcBxae               14 24-620-497-1489
+2.01111000000000E+003 BUILDING     quietly special pinto beans
nag. never furious warthogs use blithely ar
      1497 Customer#000001497        D8e2U3gYd57H4grcOr,02
             14 24-506-574-8552   +2.44957000000000E+003
AUTOMOBILE   slyly regular instructions wake furiously about the
carefully unusual deposits. final deposits boost ir
      1516 Customer#000001516
VFbEMU7LSQZPCZ3m73dNP2WH0ywr5loATV4r             14
24-797-943-8908   +9.26327000000000E+003 FURNITURE    closely
ironic deposits was. blithely ironic asymptotes affix furiously: blithe

      1944 Customer#000001944        LbrsZ4CFBqJZ
       14 24-492-922-6990   +7.27993000000000E+003 HOUSEHOLD
  silently regular accounts are furiously; blithely

      2016 Customer#000002016        g3bzz5x,FEgV6epU
          14 24-980-669-6118   +6.91667000000000E+003
MACHINERY    slyly pending accounts cajole after the special
packages.
      2295 Customer#000002295        5UB1DSumcZPU1
           14 24-681-434-4597   +7.81202000000000E+003
FURNITURE    regular, even requests sleep above the slyly regular
deposits.
      2363 Customer#000002363        Wcmdwslo1hp8yt1Dp5JN9Xbc
                 14 24-657-526-1700   +4.79711000000000E+003
MACHINERY    theodolites cajole furiously final foxes. furiously
ironic
      2503 Customer#000002503        h0km52TfvBUw22
          14 24-273-904-5521   +3.16559000000000E+003
AUTOMOBILE   slyly even instructions haggle ironic somas

      2550 Customer#000002550        cH9pc9mK,wAisF2JPgGn061
               14 24-571-317-8138   +2.34058000000000E+003
FURNITURE    ironic accounts integrate blithely according to the final
theodolites: blithely bold packages across t

  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_SHIPPRIORITY O_COMMENT

----------- ----------- ------------- ------------------------ -----------
--------------- --------------- --------------
--------------------------------------------------------------------------------
      59718     2474884 F             +9.11137700000000E+004
01/01/1992  5-LOW          Clerk#000089682              0 carefully
regular pinto beans across the even grouches dete
     334181     2866669 F             +9.08873600000000E+004
01/01/1992  2-HIGH         Clerk#000020657              0 fluffily
pending foxes haggle carefully furiously final pinto beans. s
     360261     9374104 F             +1.56040000000000E+004
01/01/1992  5-LOW          Clerk#000012667              0 express tithes
doze stealthily around the final requ
     368004    11531287 F             +1.00257930000000E+005
01/01/1992  2-HIGH         Clerk#000064446              0 slyly unusual
theodolites snooze pending instructions! q
     414725     3497104 F             +3.95006600000000E+004
01/01/1992  5-LOW          Clerk#000085868              0 sly accounts
detect along the slyly express
     466659     7788136 F             +1.39924110000000E+005
01/01/1992  4-NOT SPECIFIED Clerk#000044277              0 blithely
pending packages are. carefully s
     470693    12298895 F             +2.38349540000000E+005
01/01/1992  3-MEDIUM       Clerk#000013340              0 slyly final
packages sleep fl
     560930     2504651 F             +2.46194520000000E+005
01/01/1992  3-MEDIUM       Clerk#000001761              0 pinto beans
use carefully quickly ironic foxes! carefully ironic
     646854    13450189 F             +2.50926940000000E+005
01/01/1992  2-HIGH         Clerk#000074674              0 furiously
express requests boost never across the slyly express pl
     682656     8385209 F             +1.90867350000000E+005
01/01/1992  5-LOW          Clerk#000079535              0 silent ideas
doubt along the careful

  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

----------- ----------- ----------- ------------ -----------------------
------------------------ ------------------------ ------------------------ ------------
------------ ---------- ------------ ------------ ------------------------ ----------
--------------------------------------------
  477701990    10433806     183837           2
+2.30000000000000E+001   +4.00034400000000E+004
+3.00000000000000E-002   +4.00000000000000E-002 A          F
    01/02/1992 02/07/1992   01/16/1992    DELIVER IN PERSON
 FOB        furiously regular pinto beans boost; slyly
  481134787     2773896     773897           2
+4.50000000000000E+001   +8.86392000000000E+004
+7.00000000000000E-002   +6.00000000000000E-002 A          F
    01/02/1992 02/18/1992   01/13/1992    DELIVER IN PERSON
 REG AIR    bold excuses haggle blithely fur
  485460674     2760811     260816           1
+4.20000000000000E+001   +7.86105600000000E+004
+6.00000000000000E-002   +6.00000000000000E-002 A          F
```

```
    01/02/1992 03/26/1992   01/22/1992    DELIVER IN PERSON
 TRUCK      blithely even courts
  490303523      777103      527104           4
+1.60000000000000E+001   +1.88811200000000E+004
+1.00000000000000E-001   +3.000000000000000E-002 R          F
    01/02/1992 03/13/1992   01/03/1992    TAKE BACK RETURN
  FOB        packages across the final, final depen
  497609184    12237834      737859           1
+4.60000000000000E+001   +8.14761200000000E+004
+4.00000000000000E-002   +5.000000000000000E-002 A          F
    01/02/1992 02/22/1992   01/12/1992    DELIVER IN PERSON
 AIR        patterns wake after t
  505643460     3846985      846986           5
+6.00000000000000E+000   +1.15907400000000E+004
+8.00000000000000E-002   +4.000000000000000E-002 R          F
    01/02/1992 02/10/1992   01/27/1992    TAKE BACK RETURN
  REG AIR    slyly regular accounts after the sl
  507468900    12805663      805664           3
+2.20000000000000E+001   +3.44964400000000E+004
+7.00000000000000E-002   +3.000000000000000E-002 R          F
    01/02/1992 03/29/1992   01/28/1992    DELIVER IN PERSON
 REG AIR    unusual excuses c
  511350948    13144409      644436           2
+4.40000000000000E+001   +6.39210000000000E+004
+8.00000000000000E-002   +8.000000000000000E-002 A          F
    01/02/1992 02/27/1992   01/17/1992    COLLECT COD
RAIL       furiously regular dolphins use
  512770531    18361939      111994           2
+1.90000000000000E+001   +3.80003800000000E+004
+1.00000000000000E-001   +3.000000000000000E-002 R          F
    01/02/1992 03/12/1992   02/01/1992    DELIVER IN PERSON
 REG AIR    furiously regular fo
  512989187     5078719       78720           1
+4.00000000000000E+001   +6.78984000000000E+004
+3.00000000000000E-002   +8.000000000000000E-002 A          F
    01/02/1992 03/03/1992   01/30/1992    DELIVER IN PERSON
 RAIL       slyly sly deposits boost blithely. bli

  10 record(s) selected.
```

Query Substitution Parameters

```
"Power stream    Seed = 608164853"
-- TPC TPC-H Parameter Substitution (Version 1.3.0)
-- using 608164853 as a seed to the RNG
Q1  DELTA       95
Q2  SIZE        35
   TYPE        STEEL
   REGION      EUROPE
Q3  SEGMENT     AUTOMOBILE
   DATE        1995-03-20
Q4  DATE        1996-02-01
Q5  REGION      AMERICA
   DATE        1993-01-01
Q6  DATE        1993-01-01
   DISCOUNT    0.07
   QUANTITY    25
Q7  NATION1     CHINA
   NATION2     ROMANIA
Q8  NATION      ROMANIA
   REGION      EUROPE
   TYPE        LARGE BRUSHED STEEL
Q9  COLOR       pale
Q10 DATE        1994-03-01
Q11 NATION      KENYA
   FRACTION    0.0000010000
Q12 SHIPMODE1   TRUCK
   SHIPMODE2   RAIL
   DATE        1994-01-01
Q13 WORD1       unusual
   WORD2       requests
Q14 DATE        1995-03-01
Q15 DATE        1994-02-01
Q16 BRAND       Brand#32
   TYPE        PROMO BURNISHED
   SIZE1       47
   SIZE2       15
   SIZE3       12
   SIZE4       37
   SIZE5       33
   SIZE6       48
   SIZE7       35
   SIZE8       14
Q17 BRAND       Brand#25
   CONTAINER   MED CAN
Q18 QUANTITY    315
Q19 BRAND1      Brand#32
   BRAND2      Brand#43
   BRAND3      Brand#13
   QUANTITY1   6
   QUANTITY2   12
   QUANTITY3   26
Q20 COLOUR      firebrick
   DATE        1996-01-01
   NATION      IRAN
Q21 NATION      MOZAMBIQUE
Q22 I1          21
   I2          18
   I3          24
   I4          22
   I5          16
   I6          13
   I7          15

"Throughput Stream = 1    Seed = 608164854"
-- TPC TPC-H Parameter Substitution (Version 1.3.0)
-- using 608164854 as a seed to the RNG
Q1  DELTA       104
Q2  SIZE        23
   TYPE        BRASS
   REGION      AFRICA
Q3  SEGMENT     FURNITURE
   DATE        1995-03-05
Q4  DATE        1993-11-01
Q5  REGION      ASIA
   DATE        1993-01-01
Q6  DATE        1993-01-01
   DISCOUNT    0.05
   QUANTITY    24
Q7  NATION1     IRAN
   NATION2     IRAQ
Q8  NATION      IRAQ
   REGION      MIDDLE EAST
   TYPE        LARGE PLATED COPPER
Q9  COLOR       moccasin
Q10 DATE        1995-01-01
Q11 NATION      BRAZIL
   FRACTION    0.0000010000
Q12 SHIPMODE1   RAIL
   SHIPMODE2   TRUCK
   DATE        1995-01-01
Q13 WORD1       unusual
   WORD2       requests
Q14 DATE        1995-06-01
Q15 DATE        1996-09-01
```

```
Q16 BRAND       Brand#12
   TYPE         SMALL PLATED
   SIZE1        46
   SIZE2        36
   SIZE3        15
   SIZE4        32
   SIZE5        26
   SIZE6        33
   SIZE7        30
   SIZE8        19
Q17 BRAND       Brand#22
   CONTAINER    JUMBO CASE
Q18 QUANTITY    313
Q19 BRAND1      Brand#44
   BRAND2       Brand#21
   BRAND3       Brand#53
   QUANTITY1    1
   QUANTITY2    13
   QUANTITY3    22
Q20 COLOUR      pink
   DATE         1994-01-01
   NATION       ALGERIA
Q21 NATION      INDIA
Q22 I1          10
   I2           21
   I3           18
   I4           30
   I5           19
   I6           14
   I7           12

"Throughput Stream = 2    Seed = 608164855"
-- TPC TPC-H Parameter Substitution (Version 1.3.0)
-- using 608164855 as a seed to the RNG
Q1  DELTA       112
Q2  SIZE        11
   TYPE         NICKEL
   REGION       EUROPE
Q3  SEGMENT     MACHINERY
   DATE         1995-03-22
Q4  DATE        1996-05-01
Q5  REGION      MIDDLE EAST
   DATE         1993-01-01
Q6  DATE        1993-01-01
   DISCOUNT     0.02
   QUANTITY     24
Q7  NATION1     BRAZIL
   NATION2      CANADA
Q8  NATION      CANADA
   REGION       AMERICA
   TYPE         LARGE ANODIZED COPPER
Q9  COLOR       maroon
Q10 DATE        1993-10-01
Q11 NATION      MOROCCO
   FRACTION     0.0000010000
Q12 SHIPMODE1   AIR
   SHIPMODE2    MAIL
   DATE         1995-01-01
Q13 WORD1       unusual
   WORD2        accounts
Q14 DATE        1995-09-01
Q15 DATE        1994-05-01
Q16 BRAND       Brand#52
   TYPE         LARGE BRUSHED
   SIZE1        35
   SIZE2        3
   SIZE3        21
   SIZE4        31
   SIZE5        13
   SIZE6        14
   SIZE7        24
   SIZE8        25
Q17 BRAND       Brand#24
   CONTAINER    JUMBO JAR
Q18 QUANTITY    314
Q19 BRAND1      Brand#41
   BRAND2       Brand#14
   BRAND3       Brand#52
   QUANTITY1    7
   QUANTITY2    14
   QUANTITY3    29
Q20 COLOUR      burlywood
   DATE         1993-01-01
   NATION       MOROCCO
Q21 NATION      ALGERIA
Q22 I1          25
   I2           28
   I3           14
   I4           20
   I5           17
   I6           11
   I7           23

"Throughput Stream = 3    Seed = 608164856"
-- TPC TPC-H Parameter Substitution (Version 1.3.0)
-- using 608164856 as a seed to the RNG
Q1  DELTA       120
Q2  SIZE        49
   TYPE         TIN
   REGION       AMERICA
Q3  SEGMENT     FURNITURE
   DATE         1995-03-07
Q4  DATE        1994-02-01
Q5  REGION      AFRICA
   DATE         1994-01-01
Q6  DATE        1994-01-01
   DISCOUNT     0.08
   QUANTITY     25
Q7  NATION1     ROMANIA
   NATION2      SAUDI ARABIA
Q8  NATION      SAUDI ARABIA
   REGION       MIDDLE EAST
   TYPE         MEDIUM POLISHED COPPER
Q9  COLOR       lawn
Q10 DATE        1994-07-01
Q11 NATION      CANADA
   FRACTION     0.0000010000
Q12 SHIPMODE1   REG AIR
   SHIPMODE2    MAIL
   DATE         1995-01-01
Q13 WORD1       unusual
   WORD2        accounts
Q14 DATE        1995-12-01
Q15 DATE        1996-12-01
Q16 BRAND       Brand#32
   TYPE         STANDARD ANODIZED
   SIZE1        38
   SIZE2        27
   SIZE3        13
   SIZE4        33
   SIZE5        14
   SIZE6        6
   SIZE7        8
   SIZE8        15
Q17 BRAND       Brand#21
   CONTAINER    JUMBO CAN
```

```
Q18 QUANTITY    312
Q19 BRAND1      Brand#43
   BRAND2      Brand#41
   BRAND3      Brand#51
   QUANTITY1   2
   QUANTITY2   15
   QUANTITY3   25
Q20 COLOUR      medium
   DATE        1996-01-01
   NATION      EGYPT
Q21 NATION      PERU
Q22 I1          14
   I2          27
   I3          20
   I4          21
   I5          26
   I6          10
   I7          25

"Throughput Stream = 4    Seed = 608164857"
-- TPC TPC-H Parameter Substitution (Version 1.3.0)
-- using 608164857 as a seed to the RNG
Q1  DELTA       67
Q2  SIZE        37
   TYPE        COPPER
   REGION      EUROPE
Q3  SEGMENT     MACHINERY
   DATE        1995-03-24
Q4  DATE        1996-09-01
Q5  REGION      AMERICA
   DATE        1994-01-01
Q6  DATE        1994-01-01
   DISCOUNT    0.05
   QUANTITY    25
Q7  NATION1     IRAQ
   NATION2     JAPAN
Q8  NATION      JAPAN
   REGION      ASIA
   TYPE        MEDIUM BURNISHED COPPER
Q9  COLOR       hot
Q10 DATE        1993-04-01
Q11 NATION      MOZAMBIQUE
   FRACTION    0.0000010000
Q12 SHIPMODE1   SHIP
   SHIPMODE2   MAIL
   DATE        1996-01-01
Q13 WORD1       unusual
   WORD2       accounts
Q14 DATE        1996-03-01
Q15 DATE        1994-09-01
Q16 BRAND       Brand#12
   TYPE        MEDIUM PLATED
   SIZE1       28
   SIZE2       46
   SIZE3       45
   SIZE4       16
   SIZE5       4
   SIZE6       40
   SIZE7       26
   SIZE8       36
Q17 BRAND       Brand#23
   CONTAINER   WRAP CASE
Q18 QUANTITY    313
Q19 BRAND1      Brand#55
   BRAND2      Brand#34
   BRAND3      Brand#45
   QUANTITY1   7
   QUANTITY2   17
   QUANTITY3   22
Q20 COLOUR      turquoise
   DATE        1995-01-01
   NATION      ROMANIA
Q21 NATION      IRAN
Q22 I1          12
   I2          22
   I3          34
   I4          26
   I5          24
   I6          25
   I7          19

"Throughput Stream = 5    Seed = 608164858"
-- TPC TPC-H Parameter Substitution (Version 1.3.0)
-- using 608164858 as a seed to the RNG
Q1  DELTA       75
Q2  SIZE        24
   TYPE        BRASS
   REGION      AMERICA
Q3  SEGMENT     BUILDING
   DATE        1995-03-09
Q4  DATE        1994-06-01
Q5  REGION      ASIA
   DATE        1994-01-01
Q6  DATE        1994-01-01
   DISCOUNT    0.03
   QUANTITY    24
Q7  NATION1     CANADA
   NATION2     EGYPT
Q8  NATION      EGYPT
   REGION      MIDDLE EAST
   TYPE        MEDIUM ANODIZED TIN
Q9  COLOR       gainsboro
Q10 DATE        1994-01-01
Q11 NATION      EGYPT
   FRACTION    0.0000010000
Q12 SHIPMODE1   FOB
   SHIPMODE2   MAIL
   DATE        1996-01-01
Q13 WORD1       unusual
   WORD2       accounts
Q14 DATE        1996-07-01
Q15 DATE        1997-04-01
Q16 BRAND       Brand#52
   TYPE        ECONOMY POLISHED
   SIZE1       24
   SIZE2       22
   SIZE3       4
   SIZE4       2
   SIZE5       35
   SIZE6       41
   SIZE7       49
   SIZE8       14
Q17 BRAND       Brand#24
   CONTAINER   WRAP JAR
Q18 QUANTITY    315
Q19 BRAND1      Brand#52
   BRAND2      Brand#12
   BRAND3      Brand#45
   QUANTITY1   2
   QUANTITY2   18
   QUANTITY3   29
Q20 COLOUR      grey
   DATE        1993-01-01
   NATION      INDONESIA
Q21 NATION      BRAZIL
Q22 I1          11
```

I2	27
I3	30
I4	14
I5	28
I6	21
I7	29

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"};
$platform=$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];


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: loadata [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_uf\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"};
$inlistmax=$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 ($inlistmax eq "default")
{
 $inlistmax = 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
runthroughtput tests
 system("echo \"The following disks are assigned to the indicated
volume groups\" > $runDir/sysunused$runNum") && die "cannot
create $runDir/sysunused$runNum";

 system("lspv >> $runDir/sysunused$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")
     {
       $rc= `perl5 zap "-f" "sar"`;
       $rc= `perl5 zap "-f" "sadc"`;
     }
     else
     {
       $rc= `perl5 zap "-f" "vmstat"`;
       $rc= `perl5 zap "-f" "iostat"`;
     }
     if ( $pn > 1 )
     {
       $rc= `perl5 zap "-f" "netstat"`;
     }
     $rc= `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 dependancy 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 'calcmetricsp.pl';
if ( $runUF eq "UF")
{
  system("perl calcmetricsp.pl UF");
}
else
```

```
{
  system("perl calcmetricsp.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${delim}mpinter$runNu
m.metrics");

if ($runUF eq "UF") {

&cat("$runDir${delim}mpufinter*","$runDir${delim}mpinter$runNu
m.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";
  $dbtunefile="$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");
  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";
    $ostunefile="$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}schedtu
ne >> $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");
}
```

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_ln="once";
 $all_pn="once";
 $once="once";
}
else
{
 $all_ln="all_ln";
 $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%'\" >> $runDir${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")
      {
         $rc= `perl5 zap "-f" "sar"`;
         $rc= `perl5 zap "-f" "sadc"`;
      }
      else
      {
         $rc= `perl5 zap "-f" "vmstat"`;
         $rc= `perl5 zap "-f" "iostat"`;
      }
      if ( $pn > 1 )
      {
         $rc= `perl5 zap "-f" "netstat"`;
      }
      $rc= `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 $TPCD_RUN_DIR
&cat("$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");
$rc=&dodb_noconn("db2 get db cfg for $dbname | grep -i log >>
$runDir${delim}endLog.Info",$all_ln);
if ( $logDir ne "NULL" )
{
   $rc=&dodb_noconn("$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 -ltra /node??vg.log/NODE00* >>
$runDir${delim}endLog.Info");

#Create Catalog info
$rc = system("perl catinfo.pl p");


if ( $rc != 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}schedtu
ne >> $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 tjg */
#define TPCDBATCH_SELECT      20
#define TPCDBATCH_NONSELECT   30
#define TPCDBATCH_EOBLOCK     40                /*
@d30369 tjg */
#define TPCDBATCH_INSERT      50
#define TPCDBATCH_DELETE      60

#define TPCDBATCH_MAX_COLS    100               /*
@d30369 tjg */

#define TPCDBATCH_CHAR char

#define TPCDBATCH_PRINT_FLOAT_WIDTH  20
  /* kmw - allow 15 whole digit for %#.3f format */
  /*     - note: use > 18, size of long indentifier 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"
#ifdef 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-paritioned
#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_1LEN 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      "\0"
#define READMODE    "r\0"
#define WRITEMODE   "w\0"
#define APPENDMODE  "a\0"
#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 tjg */
/** 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 countig 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 'tpcdaudit' 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 uf1
* 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 realligned UF code for readablility
*             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 functiosn 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 gettimer 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 djd 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_uf1 and load_uf2 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 calcmetricp.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 tjg */
  unsigned int      stmt_num;                       /* @d24993 tjg */
  double            elapse_time;                    /* @d24993 tjg */
  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 tjg */

};


/*****************************************************
**************/
/* Structure containing command line options                         */
/*****************************************************
**************/
struct comm_line_opt
{                                                   /* @d22275 tjg */
/* 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 tjg */
  int eo_infile;
  int time_stamp;
  int eo_block;                                     /* @d30369 tjg */
```

```
  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 tjg */
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 tjg */
void dumpCa(struct sqlca*);       /*kmw*/

void display_usage(void);
char *uppercase(char *string);
char *lowercase(char *string);
void comm_line_parse(int agrc, char *argv[], struct global_struct
*g_struct);
int sqlrxd2a(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
tjg */
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.eo_infile) {  /* Check to see if there's no more input
*/

   c_flags.eo_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 tjg */


   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)
{
#ifndef 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;
   }
#if 0
   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 tjg */
  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")) { /* 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
tjg */
        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 tjg */
        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 tjg */
      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
tjg */
    }

  }  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 tjg */
    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 tjg */
        ((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_NSMALL:

        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].sqllen).m;
          n=(*(struct declen *)&sqlda->sqlvar[col].sqllen).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].sqllen+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;

#if defined (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;
#ifndef SQLWINT
  long int            result_sec;
  long int            result_usec;
#endif


#if defined(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
    {
#if 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   : %.8s\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->sqlerrml,
ca->sqlerrmc);
    fprintf(outstream,"SQLERRP   : %.8s\n", ca->sqlerrp);

    for (i = 0; i < 6; i++)
    {
    fprintf(outstream,"SQLERRD[%d]: %d\n", i, ca->sqlerrd[i] );
    }
    fprintf(outstream,"SQLWARN   : %.11s\n", ca->sqlwarn);
    fprintf(outstream,"SQLSTATE  : %.5s\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
tjg */

    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= %.8s, errd 1-6= %d %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= %.8s, errd 1-6= %d %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 tjg */

      /** Determine if the error is critical or a connection can be made
**/

      EXEC SQL CONNECT ;                                  /* @d28763 tjg */

      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\n");

        /*Print out contents of SQLCA variables*/
        fprintf(outstream, "SQLCABC  = %ld\n", temp_sqlca.sqlcabc);
        fprintf(outstream, "SQLCODE  = %ld\n", temp_sqlca.sqlcode);
        fprintf(outstream, "SQLERRMC = %0.70s\n",
temp_sqlca.sqlerrmc);
        fprintf(outstream, "SQLERRP  =  %0.8s\n", temp_sqlca.sqlerrp);

        for (i = 0; i < 6; i++)
```

```
        {
          fprintf(outstream, "sqlerrd[%d] = %lu \n", i,
temp_sqlca.sqlerrd[i]);
        }

        fprintf(outstream, "SQLWARN  = %0.11s\n",
temp_sqlca.sqlwarn);
        fprintf(outstream, "SQLSTATE = %0.5s\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("tpcdbatch [-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 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("\ncan 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\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--#COMMENT   comment         (user specified comments
for output)");
    printf("\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 authent_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 tjg */
       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 tjg */
       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 tjg */
       case 'D' :
                 strcpy(dbname,argv[++loopvar]);
                 break;

       case 'v' :                              /* @d26350 tjg */
       case 'V' :
         if (!strcmp(uppercase(argv[++loopvar]),"ON"))
           verbose=1;
         else
           verbose=0;
         break;

       case 'u' :                              /* @d26350 tjg */
       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 tjg */
       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 tjg */
       case 'N' :
         g_struct->c_l_opt->intStreamNum = atoi(argv[++loopvar]);
         break;

       case 's' :                              /* @d26350 tjg */
       case 'S' :    g_struct->scale_factor=atof(argv[++loopvar]); break;

       case 'h':
       case 'H' :                              /* @d26350 tjg */
         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]);
#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 tpcdbatch 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 tpcdbatch\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\nnor in 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;
#ifndef 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 */
      fprintf(stdout,"Make sure that the filename is correct.\n");
      fprintf(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%smseedme",
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");
    sqlestar ();
  }
  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");
        sqlestar ();
        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%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%spstrcntuf.%s",g_struct->run_dir,
               env_tpcd_path_delim, g_struct->file_time_stamp);
          break;
     case (1):
          /* power query stream */
          sprintf(file_name, "%s%spstrcnt%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;
  }

  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*/
  }

#ifndef 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 sqlda *sqlda, 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->sqld; col++) {
    col_name_length=sqlda->sqlvar[col].sqlname.length;
    col_type = sqlda->sqlvar[col].sqltype;
    col_width = sqlda->sqlvar[col].sqllen;
    strncpy(col_name,(char
*)sqlda->sqlvar[col].sqlname.data,col_name_length) ;

    switch (col_type)
    {
     case SQL_TYP_SMALL:
     case SQL_TYP_NSMALL:                          /* @d30369 tjg
*/
       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 *)&sqlda->sqlvar[col].sqllen).m;
       n=(*(struct declen *)&sqlda->sqlvar[col].sqllen).n;

       col_lengths[col] = TPCDBATCH_MAX ((int)(m+n),
col_name_length);
       /* Special handling for DECIMAL */    /* @d26350 tjg */
       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_l_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_l_opt->intStreamNum >=0) &&
       (g_struct->c_l_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_l_opt->intStreamNum > 0) &&
        (g_struct->c_l_opt->update != 2)  )
    {
      fprintf( outstream, "Stream number =
%d\n",g_struct->c_l_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\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->elapse_time > max->elapse_time)
        max = s_info_ptr;
      else
        if ((s_info_ptr->elapse_time < min->elapse_time)
           && (s_info_ptr->elapse_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->elapse_time != -1)  {
      s_info_ptr->adjusted_time = s_info_ptr->elapse_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->elapse_time/min->elapse_time > 1000)
        {
          /* jmc fix geo_mean calculation...round adjusted time
properly ROUNDING*/
          adjusted_max_time = max->elapse_time/1000;
          if (s_info_ptr->elapse_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->elapse_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->elapse_time;
       adjusted_a_mean += s_info_ptr->adjusted_time;
      }

      if (s_info_ptr->elapse_time > 0) {  /* don't bother finding log of
                                    numbers < 0 */
        geo_mean += log(s_info_ptr->elapse_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->elapse_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->elapse_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 (whcih can handle
the */
    /* os differences on its own */

#if defined (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 calcmetricsp.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
tjg */
{
  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*/

#ifndef 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*/
  {
#ifndef 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 */
#ifndef SQLWINT
  if (sem_op (su_semid, 0, concurrent_inserts * -1) != 0)
/*jenCI*/
  {                                                 /*jenSEM*/
    fprintf(stderr,
        "Failure to wait on insert semaphone with %d of children\n",
        concurrent_inserts);
    exit(1);
  }                                                 /*jenSEM*/
  semctl (su_semid, 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_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,"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 */
#ifndef SQLWINT
  /*    we will use 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, "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_semid = 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++)
  {
#ifndef 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 */
#ifndef 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 semaphone 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,"\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,"UF2 for update pair %d complete\n",updatePair);

}


/* runUF2_fn() moved to another SQC file                    aph 981205
*/


/*--------------------------------------------------------------*/
/*       General semaphore function.                            */
/*--------------------------------------------------------------*/
#ifndef 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(strncmp(tag, "UF", 2) == 0)
            qnum = atoi(tag+2)*(-1);
       else if(strncmp(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: /* qualifiying */
      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;
#ifndef 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*/
#ifndef 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 tjg */


  /* 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 :*sqlda 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(sqlda,col_lengths);              /* @d22817 tjg */

        allocate_sqlda(sqlda);      /* 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.                                    */

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

          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
:*sqlda;
            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_sqlda(sqlda,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 tjg */

      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 PostSQLprocess\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 tjg */
   }
  }

  fflush(outstream);

  s_info_ptr->elapse_time = get_elapsed_time(start_time);

  if (g_struct->c_flags->time_stamp == TRUE)            /* @d25594
tjg */
    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->elapse_time);


  /** Allocate space for a new stmt_info structure **/   /* @d24993
tjg */
  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->elapse_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 (sqlda_allocated)
    free_sqlda(sqlda,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)
{
#ifndef 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%spstrcntuf.%s",g_struct->run_dir,
            env_tpcd_path_delim, g_struct->file_time_stamp);
        break;
    case (1):
        /* power query stream */
        sprintf(file_name, "%s%spstrcnt%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;
  }

#ifndef LINUX

  if( (g_struct->stream_report_file = fopen(file_name,
APPENDMODE)) == 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);
  }

#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\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\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;
#ifndef 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 tjg */
  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)
{

#ifndef 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)
{
#ifndef 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 semaphone 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 semaphone 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)
{
#ifndef 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 {                          /* jen LONG */
             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;

#ifndef 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*/
  {                                                    /*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 (myoutstream, "in loop before orders exec sql\n");
  fflush(myoutstream);
#endif
      EXEC SQL INSERT INTO TPCD.ORDERS
           SELECT

O_ORDERKEY,O_CUSTKEY,O_ORDERSTATUS,O_TOTALPRIC
E,


O_ORDERDATE,O_ORDERPRIORITY,O_CLERK,O_SHIPPRIORI
TY,O_COMMENT
           FROM TPCDTEMP.ORDERS_NEW
           WHERE APP_ID = :UF_chunk;
           /*AND

12*(YEAR(O_ORDERDATE)-1992)+MONTH(O_ORDERDATE)-0
1 = :month;*/

      if (sqlca.sqlcode < 0)
        sqlcode = error_check();

      if (sqlcode == SQL_RC_E911)
      {       /* we've hit a deadlock */
        fprintf (myoutstream,
              "\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(myoutstream,
             "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 (myoutstream, "in lineitem for update pair number %d
set %d chunk %d\n",
             updatePair, i,UF_chunk);
        fflush(myoutstream);
#endif

        EXEC SQL INSERT INTO TPCD.LINEITEM
          SELECT

L_ORDERKEY,L_PARTKEY,L_SUPPKEY,L_LINENUMBER,L_Q
UANTITY,
```

```
        L_EXTENDEDPRICE,L_DISCOUNT,L_TAX,

L_RETURNFLAG,L_LINESTATUS,L_SHIPDATE,L_COMMITDA
TE,L_RECEIPTDATE,
        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)-0
1 = :month);*/

      if (sqlca.sqlcode < 0)
        sqlcode = error_check();

      if (sqlcode == SQL_RC_E911)
      {       /* we've hit a deadlock */
        fprintf (myoutstream,
              "\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(myoutstream,
              "Insert into lineitem pair %d chunk %d failed
sqlcode=%d\n",
              updatePair,UF_chunk,sqlcode);
        dumpCa(&sqlca);
        rc = -1;
      }
      else
      {
#ifdef UF1DEBUG
        fprintf (myoutstream, "lineitem insert succeeded\n");
        fflush(myoutstream);
#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(myoutstream, "    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(myoutstream,
              "    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( myoutstream,
              "    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(myoutstream, "%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(myoutstream, "%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 (myoutstream,"# of deadlocks exceeds %i\n",
MAXWAIT);
    }
    rc=-1;
    EXEC SQL ROLLBACK WORK;
    error_check();              /* @d22275 tjg */

    goto UF1_exit;
  }

/* UF1_conn_reset:  */
  EXEC SQL CONNECT RESET;
  error_check();              /* @d22275 tjg */


UF1_exit:
  fclose (myoutstream);
  /* exiting, increment the semaphore */

  /* we used the first flat file to generate the semaphore key */

#ifndef 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_semid = 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_semid, 0, 1) != 0)                    /*jen SEM*/
  {
    fprintf(stderr,"Failure to increment semaphore UF1 set %d\n",i);
    fprintf(stderr,"  semaphore sourcefile = %s  su_semid =
su_semid\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*/

#ifndef SQLWINT
  int               su_semid; /* semaphore for controlling split updates*/
  key_t              su_semkey; /* key to generate semid */
#else
  HANDLE              su_hSem;
  char               UF2_semfile[256];
#endif

  char myoutstreamfile[256];
  FILE *myoutstream, *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 (myoutstreamfile, UF2OUTSTREAMPATTERN,
env_tpcd_tmp_dir, PATH_DELIM,
        updatePair, thisConcurrentDelete);
  if ( (myoutstream = fopen (myoutstreamfile, WRITEMODE)) ==
NULL)
  {
    fprintf (stderr,
          "\nThe output file '%s' for update pair %d set %d could not
be opened runUF2_fn.\n",
          myoutstreamfile,updatePair,thisConcurrentDelete);
    rc=-1;
    goto UF2_exit;
  }

  outstream=myoutstream;    /* initialize outstream for error_check
dxxxxhar*/

#ifdef UF2DEBUG
  fprintf (myoutstream, "RunUF2 Called %d   %d   %d\n",
                updatePair, thisConcurrentDelete, numChunks );
  fflush(myoutstream);
#endif
  fprintf( myoutstream,
        "\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 (myoutstream, "before connect\n");
  fflush(myoutstream);
#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 (myoutstream, "after connect  startchunk= %d, EndChunk =
%d\n",
                        startChunk, startChunk+numChunks);
  fflush(myoutstream);
#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 (myoutstream, "Chunk = %d\n", UF_chunk);
  fflush(myoutstream);
#endif
    while (sqlcode == SQL_RC_E911 && maxwait <= MAXWAIT
&& rc == 0)
    {

#ifdef UF2DEBUG
  fprintf (myoutstream, "in loop before orders exec sql\n");
  fflush(myoutstream);
#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)-0
1 = :month));*/
      if (sqlca.sqlcode < 0)
        sqlcode = error_check();

      if (sqlcode == SQL_RC_E911)
      {       /* we've hit a deadlock */
        fprintf (myoutstream,
          "\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 (myoutstream,"\n%s\n", stmt_str.data);
        fprintf (myoutstream,"\nsqlcode %d occurred deleting from
TPCD.LINEITEM\n", sqlca.sqlcode);
        dumpCa(&sqlca);
        fprintf (myoutstream,
              "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 (myoutstream, "in loop for update pair number %d set
%d chunk %d\n",
              updatePair, thisConcurrentDelete,UF_chunk);
        fflush(myoutstream);
#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)-0
1 = :month;*/

        if (sqlca.sqlcode < 0)
          sqlcode = error_check();

        if (sqlcode == SQL_RC_E911)
        {       /* we've hit a deadlock */
#ifdef UF2DEBUG
  fprintf (myoutstream, "orders deadlocked\n");
  fflush(myoutstream);
#endif
          fprintf (myoutstream,
          "\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 (myoutstream, "orders failed\n");
  fflush(myoutstream);
#endif
          fprintf (myoutstream,"\nAn error %d occurred deleting from
TPCD.ORDERS\n",sqlca.sqlcode);
          dumpCa(&sqlca);
          fprintf (myoutstream,"for update pair number %d set %d
chunk %d..Exiting\n",
                updatePair, thisConcurrentDelete,UF_chunk);
          rc=-1;
        }
        else
        {
#ifdef UF2DEBUG
  fprintf (myoutstream, "orders succeeded\n");
  fflush(myoutstream);
#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(myoutstream, "    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(myoutstream, "    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( myoutstream,
                 "    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 (myoutstream, "after loop\n");
  fflush(myoutstream);
#endif
  /* report the number of row deleted */
  fprintf(myoutstream, "%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(myoutstream, "%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 (myoutstream,"# 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 (myoutstream);

  /* exiting, increment the semaphore */
#ifndef 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 semaphone 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, LastError: %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: (%us %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: (%us %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(DECI
MAL
        (INTEGER(100*DECIMAL(L_EXTENDEDPRICE,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: (%us %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: (%us
%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: (%us
%06uu) %s",
        mypid, tv.tv_sec, tv.tv_usec, ctime(&timeT));

  /*
  ** use the same query acidQ except don't select for specfic 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_EXTENDEDPRICE*(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_lEprice;  /* use ast_eprice  for printout to be
consistent*/

  gettimeofday(&tv, &tz);
  time(&timeT);
  fprintf(out,"AST_ACID tag: %d, after read of LINEITEM: (%us
%06uu) %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\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_linenumber = :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%cacidT.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\n",mypid);
```

```
    fprintf(out, "acidT tag: %d, begin transaction time: (%us %06uu)
%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: (%us
%06uu) %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: (%us %06uu)
%s",
            mypid, tv.tv_sec, tv.tv_usec, ctime(&timeT));
    fprintf(out, "l_partkey: %d  l_suppkey: %d  l_quantity:
%0.3f\nl_tax: %0.3f  l_discount: %0.3f  l_extendedprice: %0.3f\n",
            l_partkey, l_suppkey, 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_totalprince= %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_suppkey, :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_suppkey = l_suppkey;
  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);
          }
          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
}

#if defined(SQLPTX) || defined(SQLWINT) || defined(SQLSUN) ||
defined(Linux)
/* added fot 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

Microsoft Corporation
One Microsoft Way
Redmond, WA 98052-6399

Tel 425 882 8080
Fax 425 936 7329
http://www.microsoft.com/

Microsoft

June 20, 2003

IBM Corporation
Chris King
3039 Cornwallis Road
Research Triangle Park,
NC 27709

Ms. King:

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 Server** *Server license only - No CALs* *Discount Schedule: Open Program - No Level* *Unit Price reflects a 40% discount from the retail unit price of $3,999.*	$2,399	1	$2,399
659-00844	**Visual Studio Professional** *No discounts applied*	$1,079	1	$1,079
PRO-PRORS-16U-01	**Database Server Support Package** *1 Year Term*	$1,950	3	$5,850

Some products may not be currently orderable but will be available through Microsoft's normal distribution channels by April 2, 2003.

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: PHchki0320069389
Please include this Reference ID in any correspondence regarding this price quote.