TPC Benchmark™ H
Full Disclosure Report

DELL PowerEdge R910
8-core Intel® Xeon™ E7-8837 2.66GHz 24MB L3 Cache
using
VectorWise™ 1.6
and
RedHat™ RHEL 6.0

Submitted for Review
Report Date May 3, 2011
TPC Benchmark H™ Full Disclosure Report
First Printing

First Edition – May 3, 2011

Overview

This report documents the methodology and results of the TPC Benchmark™ H test conducted on a Dell PowerEdge R910 server, in conformance with the requirements of the TPC Benchmark™ H Standard Specification, Revision 2.14.0. The operating system used for the benchmark was Red Hat Enterprise Linux Server release 6.0; the DBMS was VectorWise release 1.6.

Standard and Executive Summary Statements

The pages following this preface contain the Executive Summary and Numerical Quantities Summary of the benchmark results.

Auditor

The benchmark configuration, environment and methodology used to produce and validate the test results and the pricing model used to calculate the cost per QphH was audited by Steve Barrish, Performance Metrics, to verify compliance with the relevant TPC specifications.

TPC Benchmark H Overview

The TPC Benchmark ™ H (TPC-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 industry-wide 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 sets 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 to 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 updates against all tables, except possibly during infrequent (e.g., once a month) maintenance sessions;

The TPC-H database tracks, possibly with some delay, the state of the OLTP database through on-going updates which batch together a number of modifications impacting some part of the decision support database;

Due to the world-wide nature of the business data stored in the TPC-H database, the queries and the updates may be executed against the database at any time, especially in relation to each other. In addition, this mix of queries and updates is subject to specific ACIDity requirements, since queries and updates 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 updates.

The minimum database required to run the benchmark holds business data from 10,000 suppliers. It contains almost ten million rows representing a raw storage capacity of about 1 GB. Compliant benchmark implementations may also use one of the larger permissible database populations (e.g. 1000 GB), as defined in Clause 4.1.3.

The performance metrics reported by TPC-H measure multiple aspects of the capability of the system to process queries. The TPC-H metric at the selected size (QphH@Size) is the performance metric. 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 5.4.7). 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. TPC-H uses terminology and metrics that are similar to other benchmarks, originated by the TPC and others. Such similarity in terminology does not in any way imply that TPC-H results are comparable to other benchmarks. The only benchmark results comparable to TPC-H are other TPC-H results compliant with the same revision.

Despite the fact that this benchmark offers a rich environment representative of many decision support systems, this benchmark does not reflect the entire range of decision support requirements. In addition, the extent to which a customer can achieve the results reported by a vendor is highly dependent on how closely TPC-H approximates the customer application. The relative performance of systems derived from this benchmark does not necessarily hold for other workloads or environments. Extrapolations to any other environment are not recommended.

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

Benchmark sponsors are permitted several possible system designs, provided that they adhere to the model described in Clause 6. A full disclosure report (FDR) of the implementation details, as specified in Clause 8, must be made available along with the reported results.

General Implementation Guidelines

The purpose of TPC benchmarks is to provide relevant, objective performance data to industry users. To achieve that purpose, TPC benchmark specifications require that benchmark tests be implemented with systems, products, technologies and pricing that:

Are generally available to users;

Are relevant to the market segment that the individual TPC benchmark models or represents (e.g. TPC-H models and represents complex, high data volume, decision support environments);

Would plausibly be implemented by a significant number of users in the market segment the benchmark models or represents.

Dell, Inc. does not warrant or represent that a user can or will achieve performance similar to the benchmark results contained in this report. No warranty of system performance or price/performance is expressed or implied by this report.
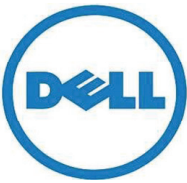
	Dell PowerEdge R910 using VectorWise 1.6	TPC-H Rev 2.14.0 TPC Pricing Rev 1.6.0
		Report Date: May 3, 2011

Total System Cost	Composite Query per Hour Metric	Price/Performance
$384,935 USD	**436,788.9** QphH@1000GB	**$0.88 USD** Price/QphH@1000GB

Database Size	Database Manager	Operating System	Other Software	Availability Date
1000 GB	**VectorWise 1.6**	**RedHat Enterprise Linux 6.0**	**None**	**6/30/2011**

Geometric mean 8.5   Arithmetic Mean 54.2
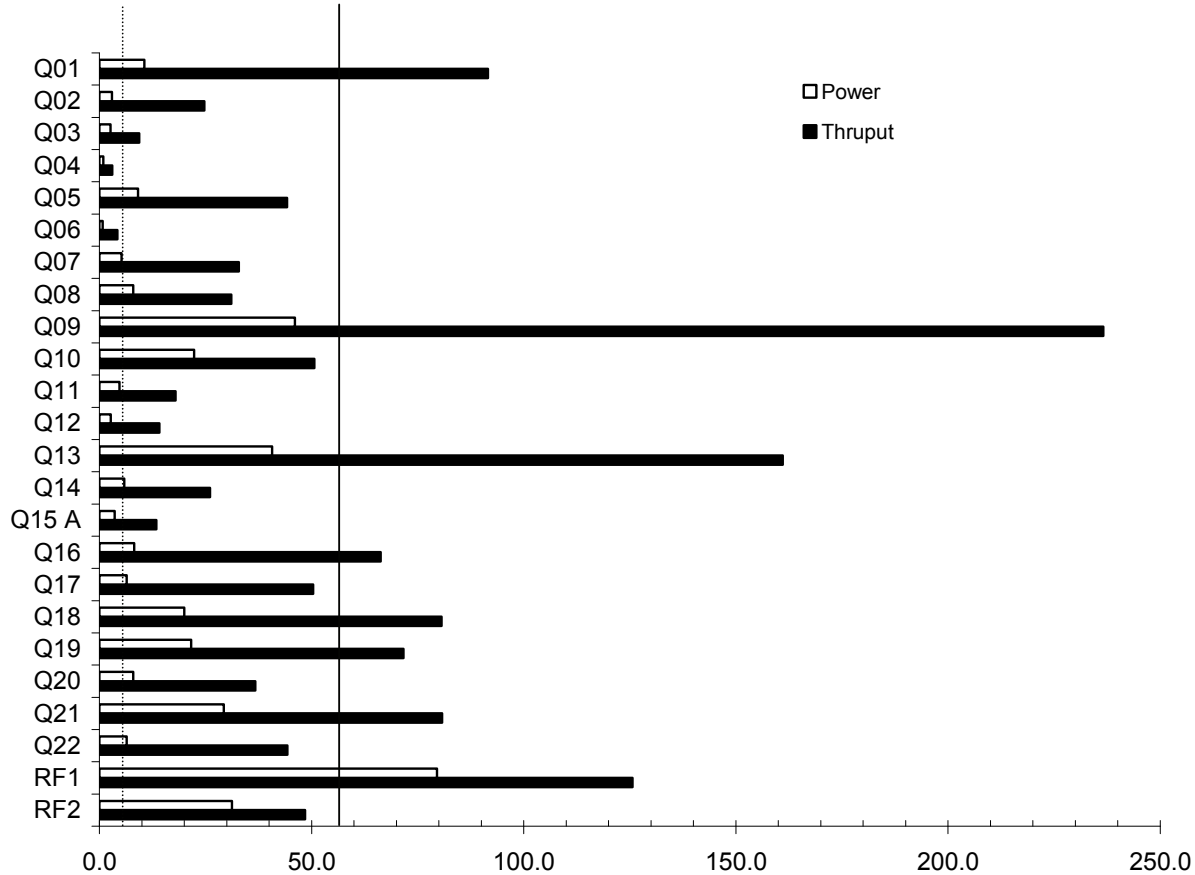
Database Load Time = 15:39:54	Load Includes Backup: N
Total Data Storage/Database Size = 2.34	Memory/Database Size Percentage = 102.4%

Storage Redundancy Level 1 for Base Tables, Auxiliary Data Structures, DBMS temporary space, and OS and DBMS Software

System Configuration

Number of Nodes:	1
Processors/Cores/Threads/Type:	4/32/32/Intel Xeon E7-8837 2.67 Ghz, 24MB L3 cache
Memory:	1024 GB
Disk Controller:	Dell PERC H700 RAID Controller with 1GB NVRAM Cache on-board
Disk Drives:	16 x 146GB 2.5in SAS 6Gb/s 15k rpm internal
Total Disk Storage	2336GB
LAN Controllers	4 x 1Gb Ethernet on-board
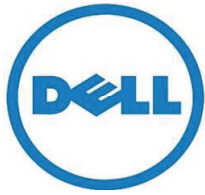
Dell PowerEdge R910 using VectorWise 1.6			TPC-H Rev 2.14.0 TPC Pricing Rev 1.6.0			
			Report Date: May 3, 2011			

Description	Part Number	Source	Reference Price	Qty	Extended Price	3 yr Maint Price
Server Hardware						
R910 Chassis for 16 2.5" HDD & Intel E7 Processors	225-1158	1	$2,954.00	1	$2,954.00	included
2x Intel® Xeon® E7-8837 2.67Gz 24MB Cache	317-7086	1	$6,078.00	1	$6,078.00	
Upgrade to 4 Intel® Xeon® E7-8837 2.67GHz 24MB Cache	317-7097	1	$6,078.00	1	$6,078.00	
1TB Memory (64x16GB) 1066MHz, Perf Optimized	317-7171	1	$51,834.00	1	$51,834.00	
Memory Riser PER910, Qty 8	331-2767	1	$1,596.00	1	$1,596.00	
Red Hat Enterprise Linux 6.0 ,Factory Install, x64	421-5717	1	$0.00	1	$0.00	
Red Hat Enterprise Linux,4SKT,3yr Subscription	421-5730	1	$4,679.00	1	$4,679.00	
Red Hat Enterprise Linux 6,Media Only X86_64	421-5737	1	$0.00	1	$0.00	
PERC H700 Integrated RAID Controller, 1GB NV Cache	342-1571	1	$799.00	1	$799.00	
RAID 5 for PERC H700 Controller	342-0392	1	$0.00	1	$0.00	
High Output Power Supply (1100W), Redundant	330-7498	1	$1,196.00	1	$1,196.00	
146GB 15K RPM SAS 6Gbps 2.5in Hard Drive	342-0427	1	$359.00	16	$5,744.00	
Dual Two-Port Embedded Broadcom NetXtreme 5709	330-7444	1	$299.00	1	$299.00	
Sliding Rails Without Cable Management Arm	330-9420	1	$169.00	1	$169.00	
DVD-ROM, SATA	313-9126	1	$30.00	1	$30.00	
Dell Keyboard, USB, Black, English	331-2254	1	$10.00	1	$10.00	
Optical Mouse, Two Buttons, USB, Black	331-0846	1	$9.00	1	$9.00	
Dell E170S 17in LCD Monitor, VGA Only	320-8083	1	$149.00	1	$149.00	
3 Year ProSupport and Mission Critical 4HR 7x24	923-6692	1	$1,067.00	1	$1,067.00	
PSMC TECH SPT,PER910,3 YR	923-6882	1	$1,575.00	1	$1,575.00	
PSMC 4H7X24 OS,PER910,UNY,INIT	926-9840	1	$333.00	1	$333.00	
HW WRTY + SVC,PER910,UNY,INIT	909-3537	1	$280.00	1	$280.00	
HW WRTY + SVC,PER910,UNY,EXT	909-3558	1	$419.00	1	$419.00	
PSMC MISSION CRIT SVC,PER910,3YR	909-3608	1	$324.00	1	$324.00	
Dell Discount 30%					-$25,686.60	
				Subtotal	**$59,935.40**	**$0.00**
No External Storage Required						
Server Software						
VectorWise 1.6 3-year license Dell server**	ING-VW-3Y-GB-DELL	2	$250.00	1,000	$250,000.00	
VectorWise 1-year maintenance bug fixes**	ING-VW-3Y-GB-DELL-MNT	2	$25.00	3,000		$75,000.00
				Subtotal	**$250,000.00**	**$75,000.00**
				Total	**$309,935.40**	**$75,000.00**

*All discounts are based on US list prices and for similar quantities and configurations.	**3-yr Cost of Ownership:** **$384,935.40**
** These components are not immediately orderable. See FDR for more information.	**QphH@1000GB:** **436,788.9**
Source 1=Dell: Pricing may be verified by calling 1-800-BUY-DELL, contact:	**$/QphH@1000GB:** **$0.88**

Source 1=Dell: Pricing may be verified by calling 1-800-BUY-DELL, contact:
   Gregory Olsen, Greg_Olsen@Dell.com, referencing quote #582792206.

   2=Ingres: Pricing may be verified by calling 1-650-587-5538, contact:
   Mark Van de Wiel, mark.vandwiel@ingres.com.

Audited By: Steve Barrish for Performance Metrics, Inc.

Prices used in TPC benchmarks reflect 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 refeflect standard pricing policies for the listed components. For complete details, see the pricing sections of the TPC benchmark specifications. If you find the stated prices are not available according to these terms, please inform the TPC at pricing@tpc.org. Thank you.

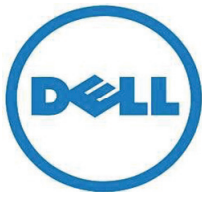|	**Dell PowerEdge R910 using VectorWise 1.6**	TPC-H Rev 2.14.0 TPC Pricing Rev 1.6.0
		Report Date: May 3, 2011

Measurement Results

|	
Database Scaling (SF/size)	1000
Total Data Storage/Database Size	2.34
Memory/Database Size Percentage	102.40%
Start of Database Load Time	04/25/11 21:06:54
End of Database Load Time	04/26/11 12:46:48
Database Load Time	15:39:54
Query Streams for Throughput Test (S)	7
TPC-H Power	422,485.9
TPC-H Throughput	451,576.1
TPC-H Composite Query-per-Hour Metric (QphH@1000GB)	436,788.9
Total System Price Over 3 Years	384,935
TPC-H Price/Performance Metric ($/QphH@1000GB)	0.88

Measurement Intervals

Measurement Interval in Throughput Test (Ts)	1,228

Duration of Stream Execution:

Power Run	Seed	Query Start Time Query End Time	Duration (sec)	RF1 Start Time RF1 End Time	RF2 Start Time RF2 End Time
	426124648	04/26/11 15:06:36 04/26/11 15:11:02	266	04/26/11 15:05:16 04/26/11 15:06:36	04/26/11 15:11:02 04/26/11 15:11:33

Throuput Stream	Seed	Query Start Time Query End Time	Duration (sec)	RF1 Start Time RF1 End Time	RF2 Start Time RF2 End Time
1	426124649	04/26/11 15:11:35 04/26/11 15:31:14	1,179	04/26/11 15:11:35 04/26/11 15:14:18	04/26/11 15:14:18 04/26/11 15:15:16
2	426124650	04/26/11 15:11:35 04/26/11 15:31:42	1,207	04/26/11 15:15:16 04/26/11 15:16:56	04/26/11 15:16:56 04/26/11 15:17:39
3	426124651	04/26/11 15:11:35 04/26/11 15:31:29	1,195	04/26/11 15:17:39 04/26/11 15:19:24	04/26/11 15:19:24 04/26/11 15:20:05
4	426124652	04/26/11 15:11:35 04/26/11 15:31:34	1,199	04/26/11 15:20:05 04/26/11 15:21:49	04/26/11 15:21:49 04/26/11 15:22:32
5	426124653	04/26/11 15:11:35 04/26/11 15:30:10	1,116	04/26/11 15:22:32 04/26/11 15:24:22	04/26/11 15:24:22 04/26/11 15:25:02
6	426124654	04/26/11 15:11:35 04/26/11 15:32:02	1,228	04/26/11 15:25:03 04/26/11 15:28:20	04/26/11 15:28:20 04/26/11 15:29:36
7	426124655	04/26/11 15:11:35 04/26/11 15:31:53	1,218	04/26/11 15:29:36 04/26/11 15:31:15	04/26/11 15:31:15 04/26/11 15:31:54
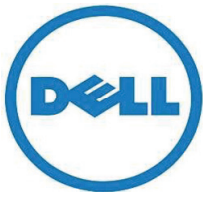
TPC-H Timing Intervals (in seconds)

Duration of stream execution:

Stream ID	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12
Stream 00	10.6	3.0	2.6	0.9	9.1	0.8	5.2	8.0	46.1	22.3	4.7	2.7
Stream 01	80.8	9.7	7.3	1.7	37.1	5.4	20.4	33.6	297.2	41.0	22.9	20.3
Stream 02	77.6	6.9	4.7	2.8	26.2	0.9	22.4	29.2	307.2	42.1	12.7	16.8
Stream 03	69.7	7.2	6.2	3.0	40.8	6.1	28.1	30.7	242.3	51.2	12.9	9.0
Stream 04	181.8	8.8	23.1	3.5	33.3	3.0	21.4	19.6	242.5	48.7	15.3	17.1
Stream 05	45.3	23.2	6.3	4.1	26.2	5.0	73.2	34.1	243.1	46.5	13.7	8.2
Stream 06	97.4	110.3	11.1	2.9	23.7	7.9	36.6	39.0	160.8	53.6	19.2	14.0
Stream 07	88.6	7.0	6.9	3.5	122.1	1.5	28.0	31.6	162.9	71.4	28.7	13.7
Minimum	10.6	3.0	2.6	0.9	9.1	0.8	5.2	8.0	46.1	22.3	4.7	2.7
Maximum	181.8	110.3	23.1	4.1	122.1	7.9	73.2	39.0	307.2	71.4	28.7	20.3
Average	73.6	19.9	7.9	2.6	36.4	3.5	26.7	26.0	194.2	44.3	15.0	11.6

Stream ID	Q13	Q14	Q15a	Q16	Q17	Q18	Q19	Q20	Q21	Q22	RF1	RF2
Stream 00	40.7	5.9	3.6	8.2	6.4	20.0	21.6	8.0	29.3	6.4	79.6	31.2
Stream 01	179.8	24.8	12.2	43.8	48.1	87.0	40.1	32.6	68.5	64.9	163.0	58.2
Stream 02	171.6	31.3	10.0	90.5	40.8	115.8	57.2	33.6	72.1	35.1	100.2	43.0
Stream 03	106.8	26.8	10.8	69.4	81.9	61.4	152.5	36.5	119.9	21.7	105.3	40.4
Stream 04	106.6	30.8	25.8	64.1	27.2	52.2	101.3	21.6	128.2	23.6	104.6	42.6
Stream 05	181.0	15.5	10.9	58.6	51.9	95.2	34.8	48.6	68.4	21.9	110.1	40.5
Stream 06	192.2	36.7	15.6	62.7	52.0	82.6	61.5	24.6	49.4	73.9	197.3	75.8
Stream 07	189.5	16.4	8.8	75.0	50.8	70.3	54.1	59.6	58.9	68.8	98.9	38.9
Minimum	40.7	5.9	3.6	8.2	6.4	20.0	21.6	8.0	29.3	6.4	79.6	31.2
Maximum	192.2	36.7	25.8	90.5	81.9	115.8	152.5	59.6	128.2	73.9	197.3	75.8
Average	146.0	23.5	12.2	59.0	44.9	73.1	65.4	33.1	74.3	39.5	119.9	46.3

0 General Items

0.1 Benchmark Sponsor

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

Dell, Inc. is the test sponsor of this TPC Benchmark H benchmark.

0.2 Parameter Settings

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

> *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.*

The Supporting Files Archive contains the Operating System and DBMS parameters used in this benchmark.

0.3   Configuration Diagrams

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

Both the priced and measured configrations are the same (Dell PowerEdge R910):

4 x Intel Xeon E7-8837 CPU's @2.67 GHz

1024 GB Memory

16 x 146 GB 15K RPM SAS Drives

4 x 1GB Ethernet Connections

1 Clause 1 Logical Database Design Related Items

1.1 Database Definition Statements

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

The Supporting Files Archive contains the scripts that define, create, and analyze the tables and indices for the TPC-H database.

1.2 Physical Organization

The physical organization of tables and indices, 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.

No record clustering or index clustering was used. Columns were not reordered in the tables.

1.3 Horizontal Partitioning

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

No horizontal partitioning was used

1.4 Replication

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

No replication was used.

2 Clause 2 Queries and Refresh Functions

2.1 Query Language

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

SQL was the query language used to implement all queries.

2.2 Verifying Method for Random Number Generation

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

TPC supplied versions 2.14.0 of DBGEN and QGEN were used for this TPC-H benchmark.

2.3 Generating Values for Substitution Parameters

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.

QGEN version 2.14.0 was used to generate the substitution parameters.

2.4 Query Text and Output Data from Qualification 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 definition 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.

The Supporting Files Archive contains the actual query text and query output.

2.5 Query Substitution Parameters and Seeds Used

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

The Supporting Files Archive contains the seed and query substitution parameters.

2.6 Query Isolation Level

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

The queries and transactions were run with "Snapshot Isolation".

2.7 Source Code of Refresh Functions

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

The source code for the refresh functions is included in the Supporting Files Archive.

3   Clause 3 Database System Properties

3.1   ACID Properties

The ACID (Atomicity, Consistency, Isolation, and Durability) properties of transaction processing systems must be supported by the system under test during the timed portion of this benchmark. Since TPC-H is not a transaction processing benchmark, the ACID properties must be evaluated outside the timed portion of the test.

Source code for ACID test is included in the Supporting Files Archive.

3.2   Atomicity

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.

Completed Transaction

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

1.   The total price from the ORDERS table and the extended price from the LINEITEM table were retrieved for a randomly selected order key.

2.   The ACID Transaction was performed using the order key from step 1.

3.   The ACID Transaction committed.

4.   The total price from the ORDERS table and the extended price from the LINEITEM table were retrieved for the same order key. It was verified that the appropriate rows had been changed.

Aborted Transaction

Perform the ACID Transaction for a randomly selected set of input data, substituting a ROLLBACK of the transaction for the COMMIT of the transaction. Verify that the appropriate rows have not been changed in the ORDERS, LINEITEM, and HISTORY tables.

5.   The total price from the ORDERS table and the extended price from the LINEITEM table were retrieved for a randomly selected order key.

6.   The ACID Transaction was performed using the order key from step 1. The transaction was stopped prior to the commit.

7.   The ACID Transaction was ROLLED BACK.

8.   The total price from the ORDERS table and the extended price from the LINEITEM table were retrieved for the same order key. It was verified that the appropriate rows had not been changed.

3.3   Consistency

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

Consistency Test

Verify that ORDERS and LINEITEM tables are initially consistent, submit the prescribed number of ACID Transactions with randomly selected input parameters, and re-verify the consistency of the ORDERS and LINEITEM.

9.   The consistency of the ORDERS and LINEITEM tables was verified based on a sample of order keys.

10.   100 ACID Transactions were submitted from each of 8 execution streams.

11.   The consistency of the ORDERS and LINEITEM tables was re-verified.

3.4   Isolation

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

Read-Write Conflict with Commit

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

12. An ACID query was run with randomly selected values for O_KEY, L_KEY and DELTA to get the initial value for O_TOTALPRICE

13. An ACID Transaction was started using the randomly dselected values from step 1. The ACID Transaction was suspended prior to COMMIT.

14. An ACID Query was started for the same O_KEY used in step 1. The ACID Query ran to completion but did not see any uncommitted changes made by the ACID Transaction.

15. The ACID Transaction was resumed, and COMMITTED.

16. The ACID Query was run again to verify that the transaction updated O_TOTALPRICE.

Read-Write Conflict with Rollback

Demonstrate isolation for the read-write conflict of a read-write transaction and a read-only transaction when the read-write transaction is rolled back.

17. *An ACID Query was run for* a randomly selected O_KEY, L_KEY and DELTA to get the initial value for O_TOTALPRICE.

18. An ACID Transaction was started using the values selected in step 1.. The ACID Transaction was suspended prior to ROLLBACK.

19. An ACID Query was started for the same O_KEY used in step 1. The ACID Query ran to completion but did not see the uncommitted changes made by the ACID Transaction.

20. The ACID Transaction was ROLLED BACK.

21. The ACID Query completed was run again to verify that O_TOTALPRICE was unchanged from step 1..

Write-Write Conflict with Commit

Demonstrate isolation for the write-write conflict of two update transactions when the first transaction is committed.

Two tests were run, the first with a transaction that COMMITS and the second with a transaction that ROLLS BACK

Results from the first test were as follows:

22. An ACID Query was run for a randomly selected O_KEY, L_KEY and DELTA to get the initial value for O_TOTALPRICE.

23. An ACID Transaction, T1, was started with the values used in stp 1. The ACID transaction T1 was suspended prior to COMMIT.

24. Another ACID Transaction, T2, was started using the same O_KEY and L_KEY used in step 1 and a randomly selected DELTA.

25. T2 COMMITTED and completed normally.

26. T1 was allowed to commitand reveviced an error, this was expected due to the "Snapshot Isolation" in use by the DBMS. This is also known as "First Committer Wins" .

27. The ACID Query was run to verify that O_TOTALPRICE was the value from T2.

Results from the second test were as follows:

28. An ACID Query was run for a randomly selected O_KEY, L_KEY and DELTA to get the initial value for O_TOTALPRICE.

29. An ACID Transaction, T1, was started with the values used in stp 1. The ACID transaction T1 was suspended prior to COMMIT.

30. A Second ACID transaction, T2, was started with the same O_KEY and L_KEY as step 1 and a different value for DELTA.

31. T2 ROLLED BACK and completed.

32. T1 resumed and completed normally.

33. The ACID Query was run to verify the database was updated with the values from T1 and not T2.

Write-Write Conflict with Rollback

Demonstrate isolation for the write-write conflict of two update transactions when the first transaction is rolled back.

Two tests were run, the first with a transaction that COMMITS and the second with a transaction that ROLLS BACK

The results from the first test were as follows:

34. An ACID Query was run for a randomly selected O_KEY, L_KEY and DELTA to get the initial value for O_TOTALPRICE

35. An ACID Transaction, T1, was started for a randomly using the values from step 1. The ACID transaction T1 was suspended prior to ROLLBACK.

36. Another ACID Transaction, T2, was started using the same O_KEY and L_KEY and a randomly selected DELTA.

37. T2 completed normally.

38. T1 was allowed to ROLLBACK.

39. It was verified that O_TOTALPRICE was from T2..

The results from the second test were as follows:

40. An ACID Query was run for a randomly selected O_KEY, L_KEY and DELTA to get the initial value for O_TOTALPRICE.

41. An ACID Transaction, T1, was started with the same values as from step 1. T1 suspended prior to COMMIT.

42. Another ACID Transaction, T2, was started.and it ROLLED BACK its updates and completed normally.

43. T1 resumed and COMMITED its updates.

44. An ACID Query was run to verify thaqt O_TOTALPRICE was the value from T1 and not T2.

Concurrent Progress of Read and Write on Different Tables

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

45. An ACID Query was run for a randomly selected O_KEY, L_KEY and DELTA to get the initial value for O_TOTALPRICE.

46. An ACID Transaction, T1, was started with the values from step 1. T1 was suspended prior to COMMIT.

47. A query was started using random values for PS_PARTKEY and PS_SUPPKEY, all columns of the PARTSUPP table for which PS_PARTKEY and PS_SUPPKEy are equal are returned. The query completed normally.

48. T1 was allowed to COMMIT.

49. It was verified that O_TOTALPRICE had been changed by T1.

Read-Only Query Conflict with Update Transactions

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

50. A Stream was submitted that executed Q1 10 times in a row with a delta of 0 to ensure that each query ran as long a possible.

51. An ACID Transaction, T1, was started for a randomly selected O_KEY, L_KEY and DELTA.

52. T1 completed and it was verified that O_TOTALPRICE was updated correctly.

53. The stream submitting Q1 finished..

3.5 Durability

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

Failure of a Durable Medium

54. The consistency of the ORDERS and LINEITEM tables was verified using 120 randomly chosen values for O_ORDERKEY.

55. At least 100 ACID transactions were submitted from 12 streams.

56. A randomly selected disk drive was removed from the SUT and the SUT comtimued to process work until each stream had submitted 300 transactions.

57. An analysis of the transaction start and end times from each stream showed that there was at least 1 transactioon in-flight at all times.

58. An analysis of the HISTORY table showed that all of the values used for O_ORDERKEY in step 1 were used by some transaction in step 2.

59. An analysis of the success file and the HISTORY table showed that all entries in the HISTORY table had a corresponding entry in the success file and that every entry in the success file had a corresponding entry in the HISTORY table.

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.

The system crash and memory failure tests were combined. First the consistency of the ORDER and LINEITEM tables was verified. Then transactions were submitted from 8 streams, once the driver script indicated that 100 transactions had been submitted from each stream power to the SUT was removed by turning off the switch to the power strip. When power was restored to the SUT, the system rebooted and the database was restarted. The HISTORY table and success files were compared to verify that every record in the HISTORY table had a corresponding record in the success file and that each record in the success file had a corresponding entry in the HISTORY table. The consistency of the ORDERS and LINEITEM tables was then verified again.

Memory Failure

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

See "System Crash"

Disk Durability

First the consistency of the ORDER and LINEITEM tables was verified. Then 8 streams were used to submit 300 transactions to the SUT. Once the driver script indicated that at least 100 transactions had been submitted from each stream a randomly selected disk drive was removed. The SUT continued to process work until all 300 transactions had completed from all 8 streams. The the start and end time stamps for every transaction in each stream were analyzed to verify that there was always at least 1 in-flight transaction. Then the HISTORY table and success files were compared to verify that every record in the HISTORY table had a corresponding record in the success file and that each record in the success file had a correcponding entry in the HISTORY table. The consistency of the ORDERS and LINEITEM tables was then verified again.

4 Clause 4 Scaling and Database Population

4.1 Ending 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	Cardinality
Region	5
Nation	25
Supplier	10,000,000
Partsupp	800,000,000
Customer	150,000,000
Orders	1,500,000,000
LineItem	5,999,989,709
Part	200,000,000

4.2 Distribution of Tables and Logs Across Media

Distribution of tables and logs across media.

The SUT has 16 physical disk drives which appear to the OS as a single RAID-5 logical drive. This drive is partitioned into a 1GB and another partition occupying all remaining disk space which is used to store the OS, DBMS and user software and the DBMS data store. There is no swap partition.

Database (/ivw): executable files, database files, and database transaction logs.

Home (/home): all user files including benchmark scripts.

OS: Red Hat Enterprise Linux release 6.0 Installation

4.3 Database Partition/Replication Mapping

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

No database partitioning or replication was used

4.4 RAID Feature

Implementation 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 5 storage was used, the RAID configuration is described in 4.2

4.5 DBGEN Modification

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 supplied DBGEN version 2.14.0 was modified (changes made to a header file) to generate the database population for this benchmark. This header file is included in the supporting files archive.

4.6   Database Load Time

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

The database load time is disclosed in the Executive Summary at the beginning of this Full Disclosure Report.

4.7   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 data storage ratio is computed from the following information:

Type	Number	Size
6Gb SAS 15k RPM	16	146G
TOTAL		2336GB
Scale Factor		1000
Size ratio		2.33

4.8   Database Load Mechanism Details and Illustration

The details of the database load must be described, including a block diagram illustrating the overall process.

The database was loaded using flat files stored on on an NFS server not included in the priced configuration

Disk Init and RAID array creation → Create Database and Tables → Create Indicies → Load all tables from flat files → Optimize all database tables → Ready to Run
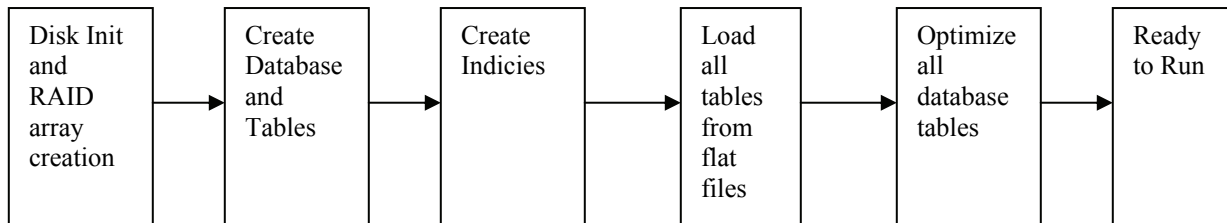
4.9   Qualification Database Configuration

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

The qualification database used identical scripts to create and load the data with changes to adjust for the database scale factor.

4.10   Memory to Database Size Percentage

The memory to database size percentage, as defined in clause 8.3.5.10, must be disclosed.

The memory to database size percentage is disclosed in the Executive Summary at the beginning of this Full Disclosure Report.

5   Clause 5 Performance Metrics and Execution-Rules

5.1   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.

Auditor requested script was run to display the indicies that had been created on the database.

All scripts and queries used are included in the Supporting Files Archive.

5.2   Steps in the Power Test

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

The following steps were used to implement the power test:
1. RF1 Refresh Transaction
2. Stream 0 Execution
3. RF2 Refresh Transaction

5.3   Timing Intervals for Each Query and Refresh Functions

The timing intervals for each query for both refresh functions must be reported for the power test.

The timing intervals for each query and both update functions are given in the Executive Summary earlier in this document.

5.4   Number of Streams for the Throughput Test

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

11 streams were used for the throughput test.

5.5   Start and End Date/Time of Each Query Stream

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

The throughput test start time and finish time for each stream are given in the Executive Summary earlier in this document.

5.6   Total Elapsed Time of the Measurement Interval

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

The total elapsed time of the throughput test is given in the Executive Summary earlier in this document.

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

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

Start and finish time for each update function in the update stream are given in the Executive Summary earlier in this document.

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

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

The timing intervals for each query and each update function are given in the Executive Summary earlier in this document.

5.9   Performance Metrics

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

The performance metrics, and the numbers, on which they are based, is given in the Executive Summary earlier in this document.

5.10   The Performance Metric and Numerical Quantities from Both Runs

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

Performance results from the first two executions of the TPC-H benchmark indicated the following percent difference for the metric points:

	Qpph@1000GB	QthH@1000GB	QphH@1000GB
Reported Run	422,485.9	451,576.1	436,788.9
Reproducibility Run	431,275.0	477,889.8	453,984.5
% Difference	2.08%	5.83%	3.94%

5.11   System Activity Between Performance Tests

> *Any activity on the SUT that takes place between the conclusion of the Reported Run and the beginning of Reproducibility Run must be disclosed.*

There was no activity on the SUT between the reported run and reproducibility run.

5.12   Dataset Verification

> *Verify that the rows in the loaded database after the performance test are correct by comparing some small number of rows extracted at random from any two files of the corresponding Base, Insert and Delete reference data set files for each table and the corresponding rows of the database.*

Verified according to the specification.

5.13   Referential Integrity

> *Verify referential integrity in the database after the initial load.*

An auditor supplied script was to verify referential integrity.

6   Clause 6 SUT and Driver Implementation Related Items

6.1   Driver

A detailed description of how the driver performs its functions must be supplied, including any related source code or scripts. This description should allow an independent reconstruction of the driver.

The supporting files archive contains the scripts that were used to implement the driver.

The power test is invoked through the script power_test.sh. It start the stream 0 SQL script along with the refresh functns such that:
• The SQL for RF1 is submitted and executed by the database
• Then the queries as generated by QGEN are submitted in the order defined by Clause 5.3.5.4
• The SQL for RF2 is then submitted from the same connection used for RF1 and executed by database

The Throughput test is invoked through the script throughput_test.sh. This script then ititiates all of the SQL streams and the refresh stream.

6.2   Implementation-Specific Layer (ISL)

If an implementation specific layer is used, then a detailed description of how it performs its functions must be provided. All related source code, scripts and configuration files must be disclosed. The information provided should be sufficient for an independent reconstruction of the implementation specific layer.

There was no Implementation Specific Layer.

6.3   Profile-Directed Optimization

If profile-directed optimization as described in Clause 5.2. is used, such use must be disclosed.

Profile-directed optimization was not used.

7 Clause 7 Pricing

7.1 Hardware and Software Used in the Priced System

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

A detailed list of hardware and software used in the priced system is included in the pricing sheet in the executive summary. All prices are currently effective.

7.2 Total Three Year Price

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

A detailed pricing sheet of all the hardware and software used in this configuration and the 3-year maintenance costs, demonstrating the computation of the total 3-year price of the configuration, is included in the executive summary at the beginning of this document.

7.3 Availability Date

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

Server Hardware	Currently Available
Server Software	Currently Available
Storage	Currently Available
Ingres VectorWise 1.6	Available June 30, 2011

8 Clause 8 Full Disclosure

8.1 Supporting Files Index Table

An index for all files included in the supporting files archive as required by Clauses 8.3.2 must be provided in the report.

Clause	Description	Archive File	Pathname
Clause 1	Device setup	benchmark_scripts.zip	scripts/ingres_vectorwise/sysinfo/disk
	Installation and configuration	benchmark_scripts.zip	scripts/ingres_vectorwise/sysinfo/install_*.txt
	OS Tunable Parameters	benchmark_scripts.zip	scripts/ingres_vectorwise/sysinfo/sysctl.conf
	DB creation scripts	benchmark_scripts.zip	scripts/ingres_vectorwise/ddl/create_*.sql scripts/ingres_vectorwise/create_db.sh
Clause 2	QGen Modifications	benchmark_scripts.zip	tpch_tools/tpcd.h
Clause 3	ACID Test scripts	benchmark_scripts.zip	scripts/ingres_vectorwise/acid/*.sh scripts/ingres_vectorwise/acid/{atom cons iso dur}/*.sh
	ACID Test Results	benchmark_scripts.zip	scripts/ingres_vectorwise/acid/{atom cons iso dur}/*output
Clause 4	Qualification db load results	benchmark_scripts.zip	scripts/ingres_vectorwise/output/7
	Qualification db validation results	benchmark_scripts.zip	scripts/ingres_vectorwise/output/8
	DBGEN Modifications	benchmark_scripts.zip	tpch_tools/tpcd.h
	Database Load Scripts	benchmark_scripts.zip	scripts/ingres_vectorwise/load_test.sh
	Test db Load results	benchmark_scripts.zip	scripts/ingres_vectorwise/output/9
Clause 5	Run 1 (10 performance run, 11 power, 12 throughput)	run1results.zip	scripts/ingres_vectorwise/output/10 scripts/ingres_vectorwise/output/11 scripts/ingres_vectorwise/output/12
	Run 2 (10 performance run, 13 power, 14 throughput)	run1results.zip	scripts/ingres_vectorwise/output/10 scripts/ingres_vectorwise/output/13 scripts/ingres_vectorwise/output/14
Clause 6	implementation scripts	benchmark_scripts.zip	scripts/ingres_vectorwise/run_perf.sh scripts/ingres_vectorwise/performance_test.sh scripts/ingres_vectorwise/power_test.sh scripts/ingres_vectorwise/throughput_test.sh
Clause 7	n/a	n/a	n/a
Clause 8	Executable query test	benchmark_scripts.zip	scripts/ingres_vectorwise/output/*/queries/stream*/*.sql
	Query substitution parameters and seeds	benchmark_scripts.zip	scripts/ingres_vectorwise/output/*/queries/stream*/*_param scripts/ingres_vectorwise/output/*/*test_report.txt
	RF function source code	benchmark_scripts.zip	scripts/ingres_vectorwise/*rf*

9   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 Steve Barrish for Performance Metrics. Further information regarding the audit process may be obtained from:

Performance Metrics

Box 984

Klamath, CA 95548

707-482-0523

April 30, 2011

Mr. David J Morse
Director, Systems Performance Analysis
Dell Enterprise Product Group
200 Dell Way
Round Rock TX, 78682

I have verified on-site and remote the TPC Benchmark™ H for the following configuration:

Platform: PowerEdge R910
Database Manager: VectorWise R1.6
Operating System: Red Hat Enterprise Linux 6.0

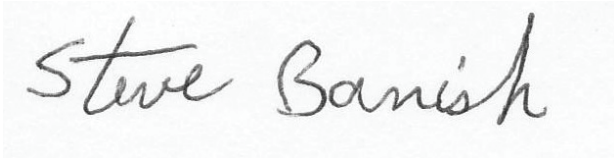| CPU's	Memory	Total Disks	Qpph@ 100GB	QthH @100GB	QphH@100GB
4 x Intel Xeon E7-8837 @ 2.67 Ghz	1024 GB	16 @ 146 GB	**422,485.9**	**451,576.1**	**436,788.9**

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

- The database tables were defined with the proper columns, layout and sizes.

- The tested database was correctly scaled and populated for 1000GB using DBGEN. The version of DBGEN was 2.14.0.

- The data generated by DBGEN was successfully compared to reference data.

- The qualification database layout was identical to the tested database except for the size of the files.

- The query text was verified to use only compliant variants and minor modifications.

- The executable query text was generated by QGEN and submitted through a standard interactive interface. The version of QGEN was 2.14.0.

- The validation of the query text against the qualification database produced compliant results.

- The refresh functions were properly implemented and executed the correct number of inserts and deletes.

- The load timing was properly measured and reported.

- The execution times were correctly measured and reported.

- The performance metrics were correctly computed and reported.

- The repeatability of the measurement was verified.

- The ACID properties were successfully demonstrated and verified.

- The system pricing was checked for major components and maintenance.

- The executive summary pages of the FDR were verified for accuracy.

Auditor's Notes:

1. This database uses a modified MVCC locking scheme as permitted by clause 3.4.2. Consequently isolation tests #3 and #4 did not complete as described in the spec, but did correctly demonstrated the isolation level required.

Sincerely,

Steve Barrish
Auditor

Appendix A   Price Quotes

INGRES
Order Confirmation

Ingres Corporation
500 Arguello Street Suite 200
Redwood City, CA 94063
United States

Phone: (650) 587-5500
Fax: (650) 649-2358

Order Number: 04-12801
Account Number: 123456
Agreement ID:
Payment Terms: Net 30
Partner:

Prepared by:
Mark Van de Wiel
(650) 587-5538
mark.vandewiel@ingres.com

Message:

PREPARED FOR (BILL TO)	SHIP TO
David J. Morse **Director, Systems Performance Analysis** **Dell | Enterprise Product Group** **office + 1 512 723 2758**	**Dell | Enterprise Product Group** **Attn. David J. Morse** **Mail Stop: RR5-18** **200 Dell Way** **Round Rock, TX, 78682**

Product item	Memo	Contract Start	Contract End	Unit	Qty	Price	Discount %	Extended Price
ING-VW-3Y-GB-DELL	3 year license, 1 GB source data, Dell server	30-June-2011	29-June-2014	GB	1000	USD 250	0	USD 250,000
ING-VW-3Y-GB-DELL-MNT	1 year professional support, bug fixes only, 1 GB source data, Dell server	30-June-2011	29-June-2014	GB	3000	USD 25	0	USD 75,000

Grand Total:	USD 325,000

*Fees herein are exclusive of taxes, withholding, levies, imposts, and duties ("Taxes"), and Customer, not Ingres, is responsible for any such Taxes.