Hewlett Packard Enterprise

Hewlett-Packard Enterprise Company

TPC Benchmark™ H
Full Disclosure Report
for
Hewlett Packard Enterprise ProLiant DL580 Gen9
using
Actian Vector 5.0
and
Red Hat Enterprise Linux Server 7.2 Edition

First Edition
June 6, 2016

Hewlett-Packard Enterprise Company (HPE), the Sponsor of this benchmark test, believes that the information in this document is accurate as of the publication date. The information in this document is subject to change without notice. The Sponsor assumes no responsibility for any errors that may appear in this document.

The pricing information in this document is believed to accurately reflect the current prices as of the publication date. However, the Sponsor provides no warranty of the pricing information in this document.

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

All performance data contained in this report was obtained in a rigorously controlled environment. Results obtained in other operating environments may vary significantly. No warranty of system performance or price/performance is expressed or implied in this report.

Printed in the United States, June 2016

Abstract

Overview

This report documents the methodology and results of the TPC Benchmark™ H test conducted on the HPE DL580 Gen9 using Actian Vector 5.0 database, in conformance with the requirements of the TPC Benchmark™ H Standard Specification, Revision 2.17.1. The operating system used for the benchmark was Red Hat Enterprise Linux 7.2

The TPC Benchmark™ H was developed by the Transaction Processing Performance Council (TPC). The TPC was founded to define transaction processing benchmarks and to disseminate objective, verifiable performance data to the industry.

TPC Benchmark™ H Full Disclosure Report and other information can be downloaded from the Transaction Processing Performance Council web site at www.tpc.org.

Standard and Executive Summary Statements

Pages iv – vii contain the Executive Summary and Numerical Quantities Summary of the benchmark results for Actian Vector 5.0 on HPE ProLiant DL580 Gen9.

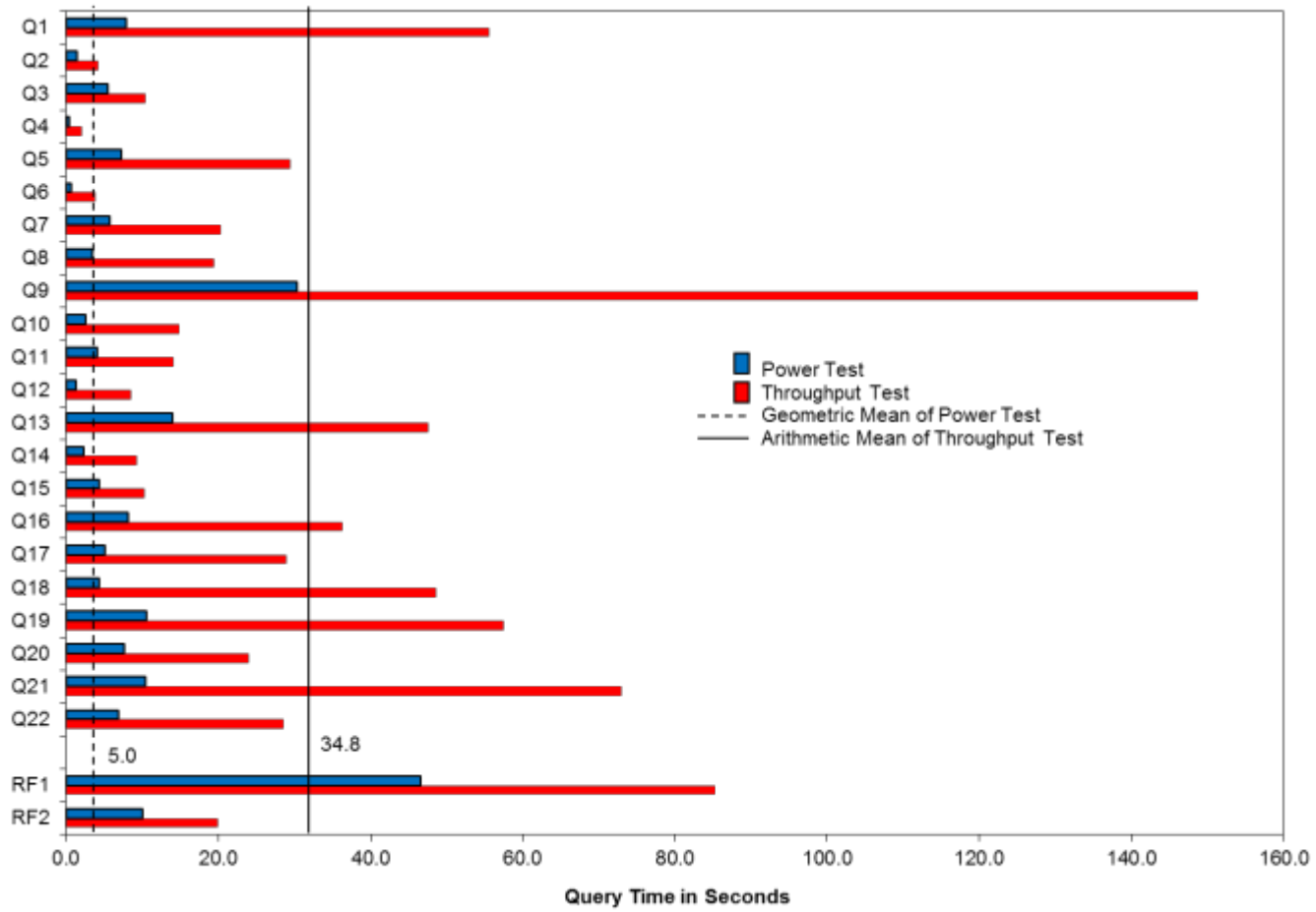
Auditor

The benchmark configuration, environment and methodology used to produce and validate the test results, and the pricing model used to calculate the QphH metric and the cost per QphH were audited by Francois Raab of InfoSizing to verify compliance with the relevant TPC specifications.

The auditor's letter of attestation is attached in Section 9.1 "Auditors' Report."

Hewlett Packard Enterprise	**HPE ProLiant DL580 Gen9** **Actian Vector 5.0**	TPC-H Rev. 2.17.1 TPC Pricing Rev. 1.7.0 Report Date: June 6, 2016
Total System Cost	Composite Query per Hour Rating	Price/Performance
$812,400 USD	**2,140,307.2** QphH @ 3000GB	**$0.38 USD** $ / QphH @3000GB

Database Size	Database Manager	Operating System	Other Software	Availability Date
3000GB	**Actian Vector 5.0**	**RHEL 7.2**	**None**	**July 31, 2016**



Database Load Time = 00d 01h 10m 34s	**Storage Redundancy Level**	
Load included backup: N	Base Tables and Auxiliary Data Structures	1
Total Data Storage / Database Size = 5.42	DBMS Temporary Space	1
Memory/Database Size = 102.4%	OS and DBMS Software	1

System Components	**System Total**
HPE ProLiant DL580 Gen9 :	1
Intel(R) Xeon(R) CPU E7-8890 v4 (24c/2.2Ghz / 60 MB L3):	4
Cores/Threads:	96/192
Memory:	3,072 GB
HPE 3.2 TB PCIe Workload Accelerator:	6
Operating System Drives (480GB SAS SSD):	2
Total Formatted Storage:	16,254 GB

Hewlett Packard Enterprise

HPE ProLiant DL580 Gen9 Actian Vector 5.0

TPC-H Rev. 2.17.1
TPC Pricing Rev. 1.7.0

Report Date: June 6, 2016

Description	Price Key	Part Number	Unit Price	Qty	Extended Price	3 Yr Maint Price
Server Hardware						
HPE ProLiant DL580 G9 CTO Server	1	793161-B21	$4,999	1	$4,999	
HPE DL580 Gen9 Intel® Xeon® E7-8890 v4 (2.20GHz/24-core) 1P Kit	1	816643-B21	$10,579	1	$10,579	
HPE DL580 Gen9 Intel® Xeon® E7-8890 v4 (2.20GHz/24-core) FIO 1P Kit	1	816643-L21	$10,579	3	$31,737	
HPE DL580 Gen8/Gen9 12 DIMM Memory Cartridge	1	788360-B21	$549	8	$4,392	
HPE 32GB 4Rx4 PC4-2400 Kit	1	728629-B21	$719	96	$69,024	
HPE Smart Array 2GB 24in FIO Flashback Write Cache	1	758836-B21	$549	1	$549	
HPE Universal Rack 11642 1075mm Shock Rack	1	H6J66A	$1,699	1	$1,699	
HPE 24A High Voltage Core Only Corded PDU	1	252663-D74	$259	1	$259	
HPE 480GB 6G SATA RI-2 SFF SC SSD	1	804593-B21	$1,039	2	$2,078	
HPE 3.2TB Read Intensive-2 HH/HL PCIe Workload Accelerator	1	831737-B21	$15,639	6	$93,834	
HPE 1200W CS Plat PL HtPlg Pwr Supply Kit	1	656364-B21	$429	4	$1,716	
HPE 3y 4h 24x7 ProLiant D580 Gen9 FC SVC HW Support	1	U8NF6E	$2,961	1		$2,961
HPE 17" FlatPanel Monitor	1	GV537A8	$130	1	$130	
HPE PS/2 Keyboard And Mouse Bundle	1	RC464AA	$39	1	$39	
			Subtotal		**$221,035**	$2,961
Server Software						
Actian Vector Capacity Based VECTOR-PPL VW 5.0	2	NA	$70,000	**5**	$350,000	
Actian Vector Gold Support VECTOR-PPL-GOLD-MNT	2	NA	$77,000	**3**		$231,000
RHEL Server 2 Sockets w/ 3Y Subscription 24x7 Support E-LTU	1	G3J30AAE	$3,702	**2**	$7,404	included
			Subtotal		**$357,404**	**$231,000**
			Total Extended Price		**$578,439**	**$233,961**

Price Key: 1 - HPE, 2 - Actian

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

3 year cost of ownership USD:	**$812,400**
QphH @ 3000GB:	**2,140,307.2**
$ USD/QphH @ 3000GB	**$0.38**

Sales contact: HPE WW Headquarters, 3000 Hanover St., Palo Alto, CA 94304-1185 (650) 857-1501 or HPE direct: 800-203-6748

Actian Corporation, 2300 Geng Rd., Suite 150, Palo Alto, CA (888) 446-4737

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

Numerical Quantities

Measurement Results

Database Scale Factor	3,000
Total Data Storage / Database Size	5.42
Percentage Memory / Database Size	102.4 %
Start of Database Load	2016-05-19 21:21:38
End of Database Load	2016-05-19 22:32:12
Database Load Time	00d 01h 10m 34s
Query Streams for Throughput Test	8
TPC-H Power	2,149,713.8
TPC-H Throughput	2,130,941.7
TPC-H Composite Query-per-Hour (QphH@3000GB)	2,140,307.2
Total System Price over 3 Years ($ USD)	$812,400
TPC-H Price/Performance Metric ($ USD / QphH@3000GB)	$0.38

Measurement Interval

Measurement Interval in Throughput Test	892 seconds

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
	519223212	2016-05-19 22:50:12	145	2016-05-19 22:49:25	2016-05-19 22:52:37
		2016-05-19 22:52:37		2016-05-19 22:50:12	2016-05-19 22:52:47

Throughput Stream	Seed	Query Start Time / Query End Time	Duration (sec)	RF1 Start Time / RF1 End Time	RF2 Start Time / RF2 End Time
1	519223213	2016-05-19 22:52:48	758	2016-05-19 22:52:47	2016-05-19 22:54:51
		2016-05-19 23:05:26		2016-05-19 22:54:51	2016-05-19 22:55:11
2	519223214	2016-05-19 22:52:49	795	2016-05-19 22:55:11	2016-05-19 22:56:12
		2016-05-19 23:06:04		2016-05-19 22:56:12	2016-05-19 22:56:31
3	519223215	2016-05-19 22:52:50	779	2016-05-19 22:56:31	2016-05-19 22:58:11
		2016-05-19 23:05:49		2016-05-19 22:58:11	2016-05-19 22:58:54
4	519223216	2016-05-19 22:52:51	799	2016-05-19 22:58:54	2016-05-19 22:59:45
		2016-05-19 23:06:10		2016-05-19 22:59:45	2016-05-19 23:00:08
5	519223217	2016-05-19 22:52:52	666	2016-05-19 23:00:08	2016-05-19 23:02:15
		2016-05-19 23:03:58		2016-05-19 23:02:15	2016-05-19 23:02:33
6	519223218	2016-05-19 22:52:53	703	2016-05-19 23:02:33	2016-05-19 23:03:22
		2016-05-19 23:04:36		2016-05-19 23:03:22	2016-05-19 23:03:45
7	519223219	2016-05-19 22:52:54	802	2016-05-19 23:03:45	2016-05-19 23:06:44
		2016-05-19 23:06:16		2016-05-19 23:06:44	2016-05-19 23:06:54
8	519223220	2016-05-19 22:52:55	823	2016-05-19 23:06:54	2016-05-19 23:07:26
		2016-05-19 23:06:38		2016-05-19 23:07:26	2016-05-19 23:07:39

TPC-H Timing Intervals (in seconds)

Stream ID	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12
0	7.9	1.4	5.5	0.4	7.3	0.7	5.7	3.4	30.4	2.5	4.1	1.3
1	35.7	4.0	15.6	1.7	26.6	3.4	18.9	15.9	219.8	9.0	26.7	13.0
2	91.4	3.6	10.8	3.9	31.8	2.1	25.2	36.8	205.3	23.8	10.0	15.0
3	88.2	4.2	9.1	1.7	55.5	3.6	25.5	23.0	111.1	23.5	13.6	4.1
4	87.6	4.6	8.1	3.3	36.3	7.7	17.9	16.0	212.3	13.7	13.3	7.7
5	46.8	4.2	10.2	1.9	28.3	3.6	15.5	33.7	112.5	14.9	11.3	8.4
6	46.1	4.3	11.8	2.1	28.1	2.8	38.2	17.5	112.3	29.0	11.0	7.5
7	47.9	4.3	11.8	1.6	24.0	1.8	16.3	15.8	115.4	6.6	32.8	2.3
8	48.8	7.1	11.4	2.1	27.6	8.7	20.2	12.5	219.3	10.7	4.3	17.6
Min	**7.9**	**1.4**	**5.5**	**0.4**	**7.3**	**0.7**	**5.7**	**3.4**	**30.4**	**2.5**	**4.1**	**1.3**
Avg	**55.6**	**4.2**	**10.5**	**2.1**	**29.5**	**3.8**	**20.4**	**19.4**	**148.7**	**14.9**	**14.1**	**8.5**
Max	**91.4**	**7.1**	**15.6**	**3.9**	**55.5**	**8.7**	**38.2**	**36.8**	**219.8**	**29.0**	**32.8**	**17.6**

Stream ID	Q13	Q14	Q15	Q16	Q17	Q18	Q19	Q20	Q21	Q22	RF1	RF2
0	14.0	2.3	4.4	8.1	5.1	4.3	10.6	7.6	10.4	6.9	46.6	10.1
1	53.7	18.8	9.7	28.7	24.1	85.1	49.2	21.4	22.1	54.5	122.2	20.7
2	43.6	7.5	8.7	30.0	40.1	43.5	43.9	38.6	52.9	25.5	60.9	18.5
3	82.1	6.6	10.5	23.1	27.5	90.8	43.9	23.7	74.7	31.8	100.4	42.4
4	73.8	7.9	17.0	80.9	27.0	30.2	50.3	12.2	56.1	14.0	51.1	23.5
5	47.2	8.0	11.0	32.9	34.8	39.1	87.0	22.9	65.1	26.1	127.3	17.7
6	35.5	10.1	8.8	79.6	21.0	43.6	50.4	39.4	78.1	24.7	48.6	23.1
7	37.6	4.9	5.0	13.8	59.5	92.2	87.6	25.3	161.7	32.7	178.9	10.1
8	40.6	17.4	17.7	29.7	21.4	9.0	95.2	24.7	135.3	40.9	31.8	13.1
Min	**14.0**	**2.3**	**4.4**	**8.1**	**5.1**	**4.3**	**10.6**	**7.6**	**10.4**	**6.9**	**31.8**	**10.1**
Avg	**47.6**	**9.3**	**10.3**	**36.3**	**28.9**	**48.6**	**57.6**	**24.0**	**72.9**	**28.6**	**85.3**	**19.9**
Max	**82.1**	**18.8**	**17.7**	**80.9**	**59.5**	**92.2**	**95.2**	**39.4**	**161.7**	**54.5**	**178.9**	**42.4**

0 General Items

0.1 Test Sponsor

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

This benchmark was sponsored by Hewlett-Packard Enterprise Company. The benchmark was developed and engineered in partnership with Actian Corporation. Testing took place at HPE facilities in Houston, Texas.

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*

This requirement can be satisfied by providing a full list of all parameters and options, as long as all those which have been modified from their default values have been clearly identified and these parameters and options are only set once.

The supporting files archive contains a list of all the database parameters and the operating system parameters used in this benchmark.

0.3 Configuration Items

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

- *Number and type of processors*
- *Size of allocated memory, and any specific mapping/partitioning of memory unique to the test.*
- *Number and type of disk units (and controllers, if applicable).*
- *Number of channels or bus connections to disk units, including 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 the run-time execution location of software components (e.g., DBMS, query processing tools /languages, middle-ware components, software drivers, etc.).*

The System Under Test (SUT), an HPE DL580 Gen9, depicted in Figure 0.1, consisted of :

HPE ProLiant DL580 G9 CTO Server

1x HPE DL580 Gen9 Intel® Xeon® E7-8890 v4 (2.20GHz/24-core) 1P Kit
3x HPE DL580 Gen9 Intel® Xeon® E7-8890 v4 (2.20GHz/24-core) FIO 1P Kit
96x HPE 32GB 4Rx4 PC4-2400 Kit
1x HPE Smart Array 2GB 24in FIO Flashback Write Cache

Storage Adapters

6x HPE 3.2TB Read Intensive-2 HH/HL PCIe Workload Accelerator

Solid-State Drives

2 X HPE 480GB 6G SATA Read Intensive-2 Small Form-Factor Smart Carrier SSD

Note: This system is the same system used for both the measured and priced configurations.

Figure 0.1 Benchmark and priced configuration for HPE DL580 Gen9

1.0 Clause 1: Logical Database Design

1.1 Table Definitions

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 table definitions and the program used to load the database.

1.2 Physical Organization of Database

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 column reordering was used.
A clustering index was used for each table as specified in the DDLs included in the supporting files archive

1.3 Horizontal Partitioning

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

Tables CUSTOMER, PART, PARTSUPP, LINEITEM and ORDERS were horizontally partitioned. The partitioning keys can be found in the DDLs included in the supporting files archive. Actian Vector uses a hash partitioning scheme to automatically distribute rows across all partitions.

1.4 Replication

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

No database-level replication was used.

2.0 Clause 2: Queries and Refresh Functions - Related Items

2.1 Query Language

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

Ingres/Vector SQL was the query language used.

2.2 Random Number Generation

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

DBGEN version 2.17.0 and QGEN version 2.17.0 were used to generate all database populations.

2.3 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 TPC source based QGEN version 2.17.0 was used to generate the substitution parameters

2.4 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.

The Supporting files archive contains the query text and query output. The following modifications were used:

- The “first” keyword is used to restrict the number of output rows in Q2, Q3, Q10, Q18 and Q21

2.5 Query Substitution Parameters and Seeds Used

All the query substitution parameters used during the performance test 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 used.

2.6 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 queries and transactions were run with “Snapshot Isolation”.

2.7 Refresh Functions

The details of how the refresh functions were implemented must be disclosed

The Supporting files archive contains the source code for the refresh functions.

3.0 Clause 3: Database System Properties

3.1 Atomicity Requirements

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 specification. The steps performed are outlined below.

3.1.1 Atomicity of the Completed Transactions

Perform the ACID Transaction 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.
2. The ACID Transaction was performed using the order key from step 1.
3. The ACID Transaction committed.
4. The total price from the ORDER 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.

3.1.2 Atomicity of Aborted Transactions

Perform the ACID transaction for a randomly selected set of input data, submitting a ROLLBACK of the transaction for the COMMIT of the transaction. Verify that the appropriate rows have not 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 total price from the ORDER 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. The transaction was stopped prior to the commit.
3. The ACID Transaction was ROLLED BACK. .
4. The total price from the ORDER table and the extended price from the LINEITEM table were retrieved for the same order key used in steps 1 and 2. It was verified that the appropriate rows had not been changed.

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

*O_TOTALPRICE = SUM(trunc(trunc((L_EXTENDEDPRICE – L_DISCOUNT) * (1 + L_TAX)))*

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

3.2.1 Consistency Tests

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

The following steps were performed to verify consistency:
1. The consistency of the ORDER and LINEITEM tables was verified.
2. One hundred ACID Transactions were submitted from each of 9 execution streams.
3. The consistency of the ORDER and LINEITEM tables was re-verified

3.3 Isolation Requirements

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.

3.3.1 Isolation Test 1 - 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)

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

1. An ACID query was run with randomly selected values for O_KEY, L_KEY and DELTA to get the initial value for O_TOTALPRICE.
2. An ACID Transaction was started using O_KEY, L_KEY and DELTA from step 1. The ACID Transaction was suspended prior to COMMIT.
3. An ACID query was started for the same O_KEY used in step 1. The ACID query ran to completion and did not see any uncommitted changes made by the ACID Transaction.
4. The ACID Transaction was resumed and committed.
5. The ACID query completed. It returned the data as committed by the ACID Transaction.

3.3.2 Isolation Test 2 - 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.

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

1. An ACID Query was run with a randomly selected O_KEY and verified that a row was returned.
2. An ACID Transaction was started for the same O_KEY as in Step 1 and randomly selected L_KEY and DELTA. The ACID transaction was suspected prior to COMMIT.
3. An ACID Query for the same O_KEY as in Step 1 was started and it was verified that did not see any uncommitted changes made by the ACID Transaction.
4. The suspended ACID transaction was resumed and made to perform a ROLLBACK
5. An ACID Query for same O_KEY as in Step 1 was started and it was verified that it returned the same row as in Step1 and that changed rolled back by the ACID transaction was not visible

3.3.3 Isolation Test 3 - 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 witha transactionT2that COMMITS and the second with a transactionT2 that ROLLS BACK.

Results from the first test were as follows:

1. An ACID query was run with randomly selected values for O_KEY, L_KEY and DELTA to get the initial value for O_TOTALPRICE.
2. An ACID Transaction T1 was started using the same O_KEY, L_KEY and DELTA as step 1.
3. The ACID transaction T1 was suspended prior to Commit.
4. Another ACID Transaction T2 was started using the same O_KEY and L_KEY and a randomly selected DELTA
5. T2 committed and completed normally
6. The ACID transaction T1 was allowed to Commit and received an error. This was expected due to the “Snapshot Isolation” in use by the DBMS. This is also known as “First Committer Wins”
7. The ACID Transaction T1 was repeated and completed successfully.
8. The ACID Query was run again to verify that the O_TOTALPRICE was the value from T2

Results from the second test were as follows:

1. An ACID Query was run for a randomly selected O_KEY, L_KEY and DELTA to get the initial value for O_TOTALPRICE.
2. AnACID Transaction, T1, was started with the values used in step 1. The ACID transaction T1 was suspended prior to COMMIT.
3. A Second ACID transaction, T2, was started with the same O_KEY and L_KEY as step 1 and a different value for DELTA.
4. T2 ROLLED BACK andcompleted.
5. T1 resumed and completed normally.
6. The ACID Query was run to verify the database was updated with the values from T1 and not T2.

3.3.4 Isolation Test 4 - 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:

1. An ACID Query was run for a randomly selected O_KEY, L_KEY and DELTA to get the initial value for O_TOTALPRICE
2. An ACID Transaction, T1, was started using the values from step 1. The ACID transaction T1 was suspended prior to ROLLBACK.
3. Another ACID Transaction, T2, was started using the same O_KEY and L_KEY and a randomly selected DELTA.
4. T2 completed normally.
5. T1 was allowed to ROLLBACK.
6. It was verified that O_TOTALPRICE was from T2.

The results from the second test were as follows:

1. An ACID Query was run for a randomly selected O_KEY, L_KEY and DELTA to get the initial value for O_TOTALPRICE.
2. An ACID Transaction, T1, was started with the same values as from step 1. T1 suspended prior to COMMIT.
3. Another ACID Transaction, T2, was started and it ROLLED BACK its updates and completed normally.
4. T1 was allowed to ROLLBACK.
5. An ACID Query was run to verify that O_TOTALPRICE was the value from STEP1.

3.3.5 Isolation Test 5 – Concurrent Read and Write Transactions on Different Tables

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

The following steps were performed:

1. An ACID Query was run for a randomly selected O_KEY, L_KEY and DELTA to get the initial value for O_TOTALPRICE.
2. An ACID Transaction, T1, was started with the values from step 1. T1 was suspended prior to COMMIT.
3. 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.
4. T1 was allowed to COMMIT.
5. It was verified that O_TOTALPRICE had been changed by T1

3.3.6 Isolation Test 6 – Update Transactions During Continuous Read-Only Query Stream

Demonstrate 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.

The following steps were performed:

1. A modified version of Q1 was started.
2. An ACID Transaction, T1, was started for a randomly selected O_KEY, L_KEY and DELTA.
3. T1 was verifiedthat T1 completed before Q1.
4. Q1 completed.
5. It was verifiedthat O_TOTALPRICE was updated by T1.

3.4 Durability Requirements

The tested system must guarantee durability: the ability to preserve the effects of committed transactions and insure database consistency after recovery from any one of the failures listed in Clause 3.5.2.

3.4.1 Permanent Unrecoverable Failure of Any 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 files including the logs were distributed across 6 PCIe Workload Accelerators that were configured using the RAID 10 option of the Linux software RAID driver (md) to mirror and stripe data. The tests were conducted on the qualification database. The following steps were performed to demonstrate that the committed updates are preserved across a permanent irrecoverable failure of a PCIe Workload Accelator card.

1. The consistency of the ORDERS and LINEITEM tables was verified using 800 randomly chosen values for O_ORDERKEY.
2. Exactly 9 streams of ACID transactions were started.
3. When the driver script indicated that at least 100 transactions had completed in each stream, a randomly selected PCIe Workload Accelator card was disabled from the SUT. Because of data redundancy the SUT continued to process transactions without interruption.
4. After processing transactions from all 9 streams for a few more minutes the power cords were removed from the SUT.
5. The system was restarted after restoring power to the SUT
6. An analysis of the transaction start and end times from each stream showed that there was at least 1 transaction in-flight at all times.
7. 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.
8. 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.
9. The consistency of the database was re-verified.

3.4.2 System Crash

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

This test was combined with the durable medium failure test.
See section 3.4.1

The counts in the history table and success files were compared and verified, and the consistency of the ORDERS and LINEITEM tables was verified

3.4.3 Memory Failure

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

This test was combined with the durable medium failure test.
See section 3.4.1

4.0 Clause 4: Scaling and Database Population

4.1 Initial Cardinality of Tables

The cardinality (i.e., 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 4.1 lists the TPC-H Benchmark defined tables and the row count for each table as they existed upon completion of the build.

TABLE	# of Rows
Lineitem	18000048306
Orders	4500000000
Partsupp	2400000000
Part	600000000
Customer	450000000
Supplier	30000000
Nation	25
Region	5

Table 4.1 Initial Number of Rows

4.2 Distribution of Tables and Logs Across Media

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

Actian Vector 5.0 was configured on an HPE ProLiant DL580 Gen9 with the following configuration:

6 HPE 3.2TB Read Intensive-2 HH/HL PCIe Workload Accelerators for database tables, log files and work area.

2 480 GB SSD disks were used for OS, Boot and kit files.

Controller/Channel	# of Disks	Array Fault Tolerance	Nominal Capacity	Partition Format	Content
Bay 1 and 2	2	RAID 1	480 GB	XFS	OS, Boot, kit files
Slot 2	1	RAID 10	3.2 TB	XFS	Database and Log Files
Slot 3	1		3.2 TB		
Slot 4	1		3.2 TB		
Slot 5	1		3.2 TB		
Slot 6	1		3.2 TB		
Slot 7	1		3.2 TB		

4.3 Mapping of Database Partitions/Replications

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

The partition key for each table is defined in the DDL. The database automatically distributed the rows evenly across al the partitions for a given table, based on the partitioning clause specidfeid in the DDL.

4.4 Implementation of RAID

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 used must be disclosed for each device.

RAID 10 was used for all the generated data, database files, recovery logs and temporary work storage.

4.5 DBGEN Modifications

The version number, release number, modification number, and patch level of DBGEN must be disclosed. 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.

DBGEN version 2.17.0 was used. Added an option to DBGEN in driver.c and print.c to write the generated data to standard out. The modified files are 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.

Load	
Load Start	05/19/2016 21:21:38
Load End	05/19/2016 22:32:12
Load Time	00d 01h 10m 34s

4.7 Data Storage Ratio

The data storage ratio must be disclosed. It is computed by dividing the total data storage of the priced configuration (expressed in GB) by the size chosen for the test database as defined in 4.1.3.1. The ratio must be reported to the nearest 1/100th, rounded up.

Data Storage Type	# of Units	Formatted Capacity	Total (GB)
HPE Read Intensive-2 HH/HL PCIe Workload Accelerator	6	2,560 GB	15,360
HPE 480GB 6G SATA Read Intensive-2 SFF SC SSD	2	447 GB	894
		Total	16,254

Size of test database: 3,000 GB Data Storage Ratio: 5.42

4.8 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 tables were loaded as depicted in Figure 4.8. All steps, scripts and configuration files are included in the Supporting Files.

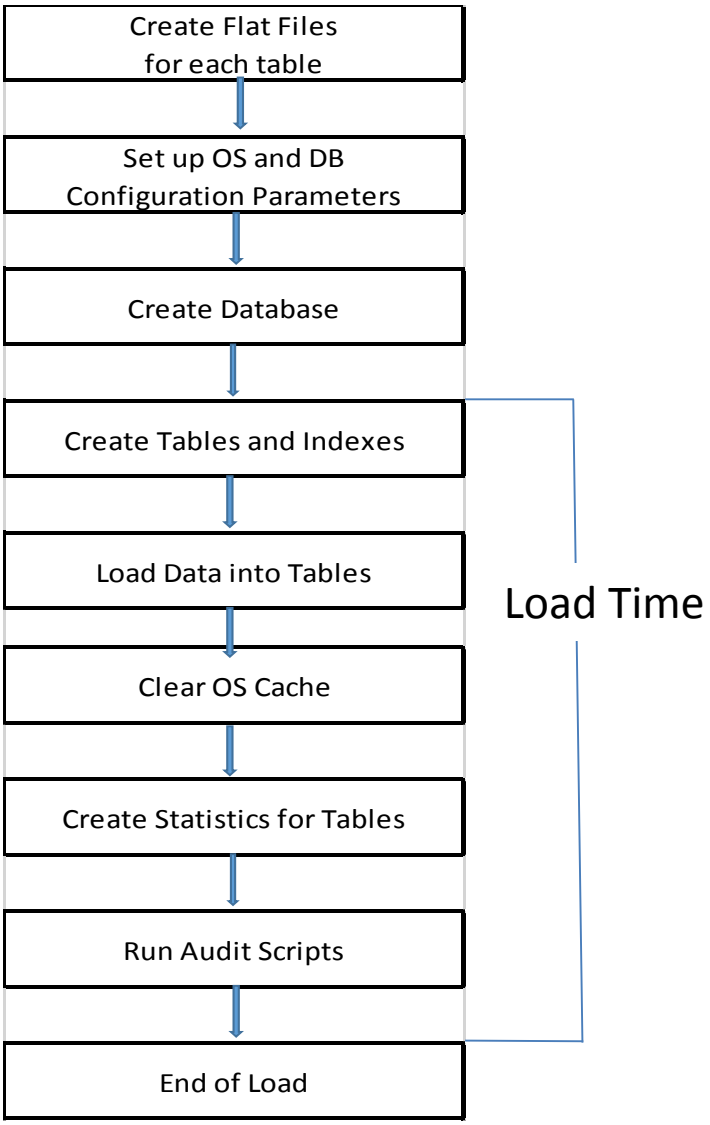


Figure 4.8: Block Diagram of Database Load Process

4.9 Qualification Database Configuration

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.

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 8.3.6.10, must be disclosed.

The main memory used in the SUT consisted of 96 registered modules of 32 GB each, for a total of 3072 Gibibytes.The memory to database size percentage is 102.4%.

5.0 Clause 5: Performance Metrics and Execution Rules Related Items

5.1 Steps after the Load Test

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 including listings of scripts or command logs.

The queries were generated using QGen with the seed collected at the end of the load test.
There was no database activity between the end of the load test and beginning of the first performance run.

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 sequentially during the power test:

1. Execute RF1 from the update stream
2. Execute Power queries from the query stream
3. Execute RF2 from the update stream

5.3 Timing Intervals for Each Query and Refresh Function

The timing intervals (see Clause 5.3.6) for each query of the measured set and for both refresh functions must be reported for the power test.

The timing intervals for each query and both refresh functions are given in the Numerical Quantities Summary earlier in this document on page vi.

5.4 Number of Streams for The Throughput Test

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

Eight query streams were used in the throughput test. Each stream ran all 22 queries. One stream was used for RF.

5.5 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.

The Numerical Quantities Summary on page vi contains the start and stop times for the query execution streams run on the system reported.

5.6 Total Elapsed Time for the Measurement Interval

The total elapsed time of the measurement interval(see Clause 5.3.5) must be reported for the throughput test.

The Numerical Quantities Summary on page vi contains the timing intervals for the throughput test run on the system reported.

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.

The Numerical Quantities Summary on page vi contains the start and finish times for the refresh functions of each stream.

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

The timing intervals (see Clause 5.3.6) for each query of each stream and for each update function must be reported for the throughput test.

The timing intervals for each query and each update function are given in the Numerical Quantities Summary earlier in this document on page vi.

5.9 Performance Metrics

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

The Numerical Quantities Summary contains the performance metrics, related numerical quantities, and the price/performance metric for the system reported.

5.10 The Performance Metric and Numerical Quantities from Both Runs

The performance metric (QphH@Size) and the numerical quantities (TPC-H Power@Size and TPC-H Throughput@ Size) from both of the runs must be disclosed

Run ID	Performance Metric QppH@3000G	TPC-H Power@3000G	TPC-H Throughput@3000G
Run 1	2,195,209.8	2,142,260.9	2,249,467.5
Run 2	2,140,307.2	2,149,713.8	2,130,941.7

5.11 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.

No database activity took place between Run 1 and Run 2.

5.12 Documentation to satisfy Clause 5.2.7

All documentation necessary to satisfy Clause 5.2.7 must be made available upon request

The supporting files archive contains the documentation

5.13 Query Validation Output

The output of the Query Output Validation Test must reported in the supporting files archive

The supporting files archive contains the documentation

6.0 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 TPC-H benchmark was implemented using Linux shell ("bash") scripts. The supporting files archive contains all the scripts.

The database build phase is performed by a script called load_test.sh. After creating the benchmark database and creates the tables and indexes using the DDLs provided in the ddl directory. It then sets up the flatfiles and calls Vector bulk loader vwload to load all the data in the tables. Once the data is loaded it sets up the server for the performance test by clearing OS, compacting memory, and by creating statistics for all tables. The load_test script timestamps the beginning and end of each sub-steps and also saves the end time in a file called seed_file for use during the performance test.

The power test is implemented by the script power_test.sh. It starts the stream 0 SQL script along with the refresh functions such that:
• First, the SQL queries for RF1 is submitted and executed by the database
• After RF1 is completed, the queries as generated by QGEN (using the seed produced by load_test.sh) are submitted to the database, one at a time in the order defined by Clause 5.3.5.4.
• After the last query completes, the SQL for RF2 is submitted from the same connection used for RF1 and executed by the database
The script records the begin and end time for each query, RF1, and RF2 in a log file for submission to the auditor.

The Throughput test is implemented by the script called tput_test.sh. This script initiates all of the query streams and the refresh stream in parallel and waits for them to finish. Each query stream and the refresh stream records the begin and end time of each query for later analysis.

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 supplied, including any related source code or scripts. This description should allow an independent reconstruction of the implementation-specific layer.

See Section 6.1 for details.

6.3 Profile-Directed Optimization

If profile-directed optimization as described in Clause 5.2.9 is used, such use must be disclosed.

Profile-directed optimization was not used.

7.0 Clause 7: Pricing Related Items

7.1 Hardware and Software Used

A detailed list of 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 indicate 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.

The pricing summary sheet is given on page *v* in the Executive Summary at the front of this report. The source for all prices is indicated.

Server and all storage components are available at publication date.

The pricing and availability of the Actian Vector 5.0 software are provided by Actian in a quote which is included in this report in Appendix A.

7.2 Three-Year Cost of System Configuration

The total 3-year price of the entire configuration must be reported, including: hardware, software, and maintenance charges. Separate component pricing is required.

The pricing summary sheet on page v in the front of this report contains all cost details.

7.3 Availability Dates

The committed delivery date for general availability (availability date) of products used in the priced calculations must be reported. When the priced system includes products with different availability dates, the single availability date reported on the first page of 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 7.3.1.4). All availability dates, whether for individual components or for the SUT as a whole, must be disclosed to a precision of 1 day, but the precise format is left to the test sponsor.

Category	Available
Server Hardware	Now (date of publication)
Storage	Now (date of publication)
Server Software	Now (date of publication)
Actian Vector 5.0	July 31, 2016

8.0 Clause 8: Support Files Index Table

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	Pathname
Clause 1	DB Creation/Load Scripts - QUAL	SupportingFiles/Clause1/ddl
	DB Creation/Load Scripts - TEST	SupportingFiles/Clause1/ddl
	OS and DB_Settings	SupportingFiles/Clause1/OS_DB_Settings
Clause 2	Queries	SupportingFiles/Clause2//queries
Clause 3	ACID Test Scripts	SupportingFiles/Clause3/AcidScripts
	ACID Test Results	SupportingFiles/Clause3/AcidResults
Clause 4	DGGen files	SupportingFiles/Clause6/scripts
	Qualification Test Results	SupportingFiles/Clause4/QualDBReport
	DB Load Report	SupportingFiles/Clause4/LoadReport
Clause 5	Queries and Output from Run 1	SupportingFiles/Clause5/Run1
	Queries and Output from Run 2	SupportingFiles/Clause5/Run2
Clause 6	Implementation Specific layer source code and Driver	SupportingFiles/Clause6/scripts

9.0 Clause 9: Audit Related Items

9.1 Auditors' Report

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

This implementation of the TPC Benchmark™ H was audited by Francois Raab of InfoSizing, a certified TPC-H auditor. Further information regarding the audit process may be obtained from:

Francois Raab
InfoSizing (www.sizing.com)
20 Kreg Ln.
Manitou Springs, CO 80829
(719) 473-7555

TPC Benchmark™ H Full Disclosure Report and other information can be downloaded from the Transaction Processing Performance Council web site at www.tpc.org.

Attestation Letter

The auditor's Attestation Letter is on the next two pages.

InfoSizing
The Right Metric For Sizing IT

TPC Transaction Processing Performance Council
Certified Auditors

Benchmark sponsor: Paul Cao
Hewlett-Packard Enterprise
11445 Compaq Center Dr. West
Houston, TX 77070

May 23, 2016

I verified the TPC Benchmark H (TPC-H™ v2.17.1) performance of the following configuration:

Platform:	Hewlett Packard Enterprise ProLiant DL580 Gen9
Operating System:	Red Hat Enterprise Linux Server 7.2 Edition
Database Manager:	Actian Vector 5.0
Other Software:	n/a

The results were:

Performance Metric	**2,140,307.2 QphH@3,000GB**
TPC-H Power	2,149,713.8
TPC-H Throughput	2,130,941.7
Database Load Time	1h 10m 34s

Server	**HPE ProLiant DL580 Gen9**		
CPUs	4 x Intel(R) Xeon(R) CPU E7-8890 v4 (2.20GHz, 24c, 60MB L3)		
Memory	3,072 GB		
Disks	***Qty***	***Size***	***Type***
	2	480 GB	SATA SSD
	6	3.2 TB	PCIe Workload Accelerator

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

The following verification items were given special attention:

- The database records were defined with the proper layout and size
- The database population was generated using DBGen
- The database was properly scaled to 3,000GB 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 no query variant
- The execution of the queries against the SF1 database produced compliant answers
- A compliant implementation specific layer was used to drive the tests
- The throughput tests involved 8 query streams
- The ratio between the longest and the shortest query was such that no query timings were adjusted
- The execution times for queries and refresh functions were correctly measured and reported
- The repeatability of the measured results was verified
- The system pricing was verified for major components and maintenance
- The major pages from the FDR were verified for accuracy

Additional Audit Notes:

None.

Respectfully Yours,

François Raab, President

20 Kreg Lane • Manitou Springs, CO 80829 • 719-473-7555 • www.sizing.com

Order Confirmation

Prepared by:
John Bard
+1 6505875584
john.bard@actian.com

Phone:

Message:

Order Number: 608715
Order Date: 17-May-2016
Account Number: 1410535
Agreement: This order is subject to your signed agreement with Actian (f/k/a Ingres) or (if none) the license agreement included with the product.
Billing Terms: Annual in Advance
Payment Terms: Due on receipt
Validity Period: 30 Days

BILL TO	SHIP TO
Hewlett Packard Enterprise **Paul Cao** **11445 Compaq Center W Dr** **Houston TX 77070** **USA**	**Hewlett Packard Enterprise** **Paul Cao** **11445 Compaq Center W Dr** **Houston TX 77070** **USA**

Product	Contract Start	Contract End	Version	Platform	Unit	Quantity	Price	Extended Price
VECTOR-PPL Actian Vector Capacity Based Actian Vector Capacity Based: Includes perpetual software license rights per purchased compute unit (as defined in Customer's contract) for production usage, no support services. This license only allows tables created 'with structure=vectorwise' (the default) or 'with structure=vectorwise_row' and use of other table storage types is permitted only to support the analytics functions of Vector. The software cannot be used as a transactional database.	18-May-2016		VW 5.0	Linux X86 64-bit	Compute Unit	5.0	USD 70,000.00	USD 350,000.00
VECTOR-PPL-GOLD-MNT Actian Vector Gold Support Actian Vector Gold Support: Includes during the 1-year term: Gold Support Services for a Capacity Based Perpetual License 3 year term	18-May-2016	17-May-2019	VW 5.0	Linux X86 64-bit	Each	3.0	USD 77,000.00	USD 231,000.00

Grand Total:	USD 581,000.00

AGREEMENT TERMS:
1) ACCEPTANCE OF THIS ORDER IS SUBJECT TO THE TERMS OF THE APPLICABLE SIGNED AGREEMENT WITH ACTIAN US (FORMERLY INGRES, OR ANY OF ITS SUBSIDIARIES, INCLUDING PARACCEL, PERVASIVE, AND VERSANT), OR (IF NONE), THE LICENSE AGREEMENT ASSOCIATED WITH THE ABOVE REFERENCED PRODUCT. **2) FEES HEREIN ARE EXCLUSIVE OF TAXES, WITHHOLDING, LEVIES, IMPOSTS, AND DUTIES (TAXES) AND CUSTOMER, NOT ACTIAN, IS RESPONSIBLE FOR ANY SUCH TAXES. ANY TERMS AND CONDITIONS IN ANY CUSTOMER PURCHASE ORDER OR SIMILAR DOCUMENT ARE EXPRESSLY REJECTED AND SHALL NOT APPLY TO THE PURCHASES AND LICENSES HEREIN.**

Any terms and conditions in any Customer Purchase Order or similar document are expressly rejected and shall not apply to the purchases and licenses herein.

Actian Support Policy - http://www.actian.com/support-services/support#policy

ACCEPTANCE	Actian
SIGNATURE:	
NAME:	
TITLE:	
DATE:	

ACCEPTANCE	Customer
SIGNATURE:	
NAME:	
TITLE:	
DATE:	