



TPC Benchmark™ H
Full Disclosure Report



INSPUR®K1-810

using

Action Vector 3.0.0

and

Inspur K-UX 2.2

First Edition

May 2014

First Edition – May, 2014

Inspur Inc., 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 in this document is believed to accurately reflect the current prices as of the date of publication. However, the sponsor provides no warranty of the pricing information in this document. Vector is a trademark of Actian. Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. TPC Benchmark, TPC-H, QphH, and \$/QphH are trademarks of Transaction Processing Performance Council. All other trademarks and copyrights are property of their respective owners.

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, 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.

© 2014 Inspur Inc. All rights Reserved.

Table of Contents

Table of Contents	3
Abstract	6
Inspur K1-810	6
Preface	11
TPC Benchmark© H Overview	11
General Items	13
1 Test Sponsor	13
2 Parameter Settings	13
3 Configuration Diagrams	13
Clause 1: Logical Database Design	15
1.1 Database Definition Statements	15
1.2 Physical Organization	15
1.3 Horizontal Partitioning	15
1.4 Replication	15
Clause 2: Queries and Refresh Functions	16
2.1 Query Language	16
2.2 Verifying Method of Random Number Generation	16
2.3 Generating Values for Substitution Parameters	16
2.4 Query Text and Output Data from Qualification Database	16
2.5 Query Substitution Parameters and Seeds Used	16
2.6 Isolation Level	17
2.7 Source Code of Refresh Functions	17
Clause 3: Database System Properties	18
3.1 ACID Properties	18
3.2 Atomicity Requirements	18
3.2.1 Atomicity of the Completed Transactions	18
3.2.2 Atomicity of Aborted Transactions	18
3.3 Consistency Requirements	19
3.4 Isolation Requirements	19

3.5 Durability Requirements	22
Clause 4: Scaling and Database Population	24
4.1 Initial Cardinality of Tables	24
4.2 Distribution of Tables and Logs Across Media	24
4.3 Mapping of Database Partitions/Replications	24
4.4 Implementation of RAID	25
4.5 DBGEN Modifications	25
4.6 Database Load time	25
4.7 Data Storage Ratio	25
4.8 Database Load Mechanism Details and Illustration	25
Clause 5: Performance Metrics and Execution Rules	27
5.1 System Activity Between Load and Performance Tests	27
5.2 Steps in the Power Test	27
5.3 Timing Intervals for Each Query and Refresh Function	27
5.4 Number of Streams for The Throughput Test	27
5.5 Start and End Date/Times for Each Query Stream	27
5.6 Total Elapsed Time for the Measurement Interval	28
5.7 Refresh Function Start Date/Time and Finish Date/Time	28
5.8 Timing Intervals for Each Query and Each Refresh Function for Each Stream	28
5.9 Performance Metrics	28
5.10 The Performance Metric and Numerical Quantities from Both Runs	28
5.11 System Activity Between Tests	29
5.12 Dataset Verification	29
5.13 Referential Integrity	29
Clause 6: SUT and Driver Implementation	30
6.1 Driver	30
6.2 Implementation Specific Layer	30
6.3 Profile-Directed Optimization	30
Clause 7: Pricing	31


7.1 Hardware and Software Pricing	31
7.2 Three Year Price	31
7.3 Availability Dates	31
Supporting File Index	32
Auditors' Information and Attestation Letter	32

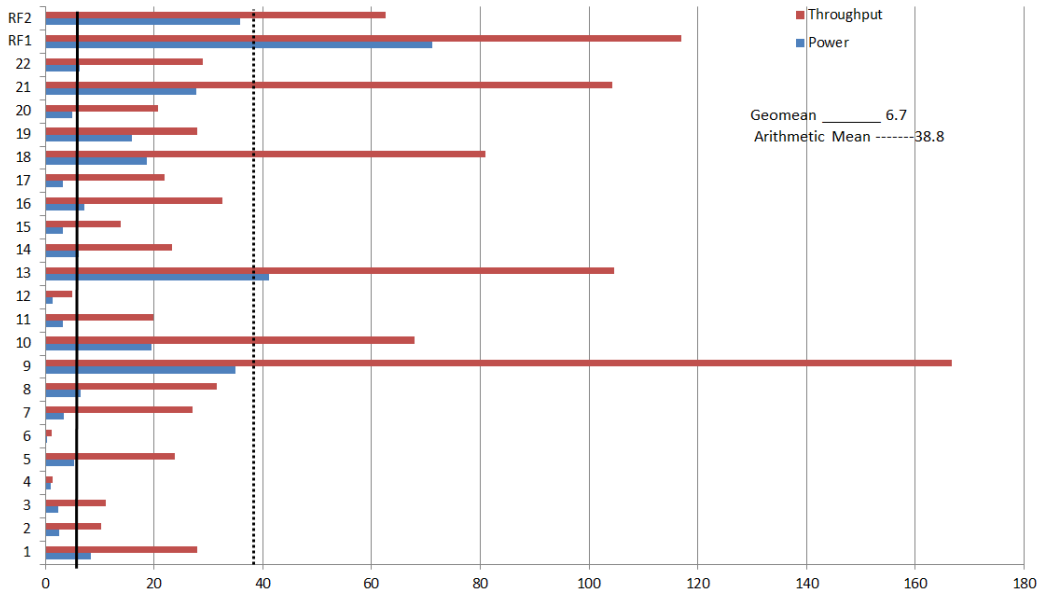
Abstract

This report documents the methodology and results of the TPC Benchmark© H test conducted on the Inspur K1-810 using Inspur K-UX2.2 and Actian Vector.

Inspur K1-810

Company Name	Processor/Cores/ Threads/Type	Database Software	Operating System
Inspur, Inc.	4/40/80 Intel(R) CPU E7-8891 v2 37.5M cache with Intel Turbo Boost Technology up to 4.0 GHz	Actian Vector 3.0.0	Inspur K-UX2.2
TPC Benchmark© H Metrics			
Total System Cost	TPC-H Throughput	Price/Performance	Availability Date
¥1,955,931	485,242.7 QphH@1000 GB	¥4.03 CNY ¥/QphH@1000 GB	26/5/2014

		INSPUR® K1		TPC-H Rev. 2.16.0 TPC Pricing Rev. 1.7.0
				Report Date: May 26, 2014
Total System Cost		Composite Query per Hour Metric		Price/Performance
¥ 1,955,931CNY		485,242.7_{QphH@1000 GB}		¥4.03CNY Price/QphH@1000GB
Database Size	Database Manager	Operating System	Other Software	Availability Date
1,000 GB	Action Vector 3.0.0	K-UX2.2	n/a	21/5/2014



Database Load Time = 13:12:22		Storage Redundancy Level	
Load Includes Backup: N		Base Tables and Auxiliary Data Structures	1
Total Data Storage / Database Size = 4.32		DBMS Temporary Space	1
Percentage Memory / Database Size = 51.2%		OS and DBMS Software	1

System Configuration

Number of Nodes: 1
 Processor/ Type: Intel Xeon E7-8891 v2 @3.2GHz with 37.5MB L3
 Proc/Cores/Threads: 4/40/80
 Memory: 512GB
 Disk Controller: LSI MR9271-8i Adapter with 1024MB
 Disk Drives: 9 x 480GB SATA 6Gb SSD
 Total Disk Storage: 4,320GB
 LAN Controllers: Intel Ethernet i350 QP 1Gb Network Daughter Card



INSPUR® K1

TPC-H Rev. 2.16.0
TPC Pricing Rev. 1.7.0

Report Date: May
26, 2014

Description	Part Number	Price Source	Unit Price	Qty	Extended Price	3 yr. Maint. Price
Server Hardware						
Inspur K1-810 chassis	GTSK1C800002	1	¥192,815	1	¥192,815	
3.2 GHz E7-8891 V2 10c CPU /37.5M Cache	BOC8891	1	¥56,942	4	¥227,768	
512GB Memory (32*16G) 1600MHz	BOM006	1	¥4,243	32	¥135,776	
Memory Riser	BOR006	1	¥9,599	8	¥76,796	
LSI CONTROLLER 9271-8i	BRE028	1	¥8,923	1	¥8,923	
480 GB 2.5 Enterprise SATA SSD	BOS073	1	¥9,500	9	¥85,503	
1000W POWER SUPPLY UNIT	BOD1000	1	¥7,691	4	¥30,763	
Inspur LCD Monitor 19"	BOV005	1	¥6,853	1	¥6,853	
Inspur USB-EN Keyboard & Mouse	BOK001、BOM001	1	¥203	1	¥203	
Inspur 42U RACK	BOG016	1	¥21,153	1	¥21,153	
ServicePac for 3-year 24 x 7 suport	BOS009	1	include	1	¥0	¥0
40% discount					¥314,621	¥0
				Subtotal	¥471,931	¥0
Server Software						
K-UX 2.1,Factory Install,IA64	X0B1T01010101001	1	¥20,000	1	¥20,000	
Vector 3.0, 1-year license, single server, up to 1 TB	VECTOR-AAD-LTD-TRM1	2	¥400,000.00	3	¥1,200,000	
Vector 1-year maintenance bug fixes	VECTOR-AAD-LTD-TRM -MNT	2	¥88,000.00	3	¥264,000	
				Subtotal	¥1,484,000	¥0
				Total	¥1,955,931	¥0

Price Source:
1=Inspur
Order method: 800-860-6708 or 86 0531-85105430
2=Actian
Order method: Jason.Leonidas@actian.com

Three-Year Cost of Ownership : ¥1,955,931
QpH@1000G: 485,242.7
¥/ QpH@1000G: ¥4.03

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

Prices used in TPC benchmarks reflect the actual prices a customer would pay for a one-time purchase of the stated components. Individually negotiated discounts are permitted. Special prices based on assumptions about past or future purchase are not permitted. All discounts reflect standard pricing policies for the list 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 items, please inform the TPC at pricing@tpc.org



INSPUR® K1

TPC-H Rev. 2.16.0
TPC Pricing Rev. 1.7.0
Report Date: May 26,2014

Measurement Results

Total Data Storage/Database Size	4.32
Percentage Memory/Database Size	51.2%
Start of Database Load Time	2014/4/11 20:00:44
End of Database Load Time	2014/4/12 09:13:06
Database Load Time	13:12:22
Query Streams for Throughput Test (S)	7
TPC-H Power @1000 GB	534,288.1
TPC-H Throughput @1000 GB	440,699.5
TPC-H Composite @1000 GB	485,242.7
Total System Price Over 3 Years	¥1,955,931
TPC-H Price/Performance Metrics (¥/QphH@1000 GB)	¥4.03

Measurement Interval in Throughput Test (Ts)

1,258

Power Run	Seed	Query Start Time	Duration	RF1 Start Time	RF2 Start Time
		Query End Time	(sec)	RF1 End Time	RF2 End Time
	412091306	2014-04-12 10:05:18	00 00:03:44	2014-04-10 06:34:52	2014-04-10 06:38:02
		2014-04-12 10:09:02	224	2014-04-10 06:35:28	2014-04-10 06:38:30

Throughput Stream	Seed	Query Start Time	Duration	RF1 Start Time	RF2 Start Time
		Query End Time	(sec)	RF1 End Time	RF2 End Time
1	412091307	2014-04-12 10:09:40	00 00:16:43	2014-04-12 10:09:39	2014-04-12 10:12:01
		2014-04-12 10:26:23	1,003	2014-04-12 10:12:01	2014-04-12 10:13:11
2	412091308	2014-04-12 10:09:39	00 00:12:39	2014-04-12 10:13:11	2014-04-12 10:15:00
		2014-04-12 10:22:18	759	2014-04-12 10:15:00	2014-04-12 10:15:45
3	412091309	2014-04-12 10:09:40	00 00:15:29	2014-04-12 10:15:45	2014-04-12 10:17:19
		2014-04-12 10:25:09	929	2014-04-12 10:17:19	2014-04-12 10:18:39
4	412091310	2014-04-12 10:09:39	00 00:18:06	2014-04-12 10:18:39	2014-04-12 10:20:51
		2014-04-12 10:27:45	1,086	2014-04-12 10:20:51	2014-04-12 10:22:18
5	412091311	2014-04-12 10:09:39	00 00:10:04	2014-04-12 10:22:18	2014-04-12 10:25:10
		2014-04-12 10:19:43	604	2014-04-12 10:25:10	2014-04-12 10:26:14
6	412091312	2014-04-12 10:09:39	00 00:12:39	2014-04-12 10:26:14	2014-04-12 10:27:47
		2014-04-12 10:22:18	759	2014-04-12 10:27:47	2014-04-12 10:28:31
7	412091313	2014-04-12 10:09:40	00 00:13:57	2014-04-12 10:28:31	2014-04-12 10:29:51
		2014-04-12 10:23:37	837	2014-04-12 10:29:51	2014-04-12 10:30:37



INSPUR® K1

TPC-H Rev. 2.16.0
TPC Pricing Rev. 1.7.0
Report Date: May 26,2014

TPC-H Timing Intervals (in seconds)

Duration of query execution:

Stream ID	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12
0	8.3	2.5	2.4	0.9	5.3	0.4	3.4	6.5	35.0	19.6	3.2	1.3
1	8.0	6.6	5.3	0.9	17.7	1.1	37.6	9.9	71.7	82.2	24.5	5.7
2	23.1	3.3	4.8	1.6	10.4	0.5	89.2	30.9	138.6	46.2	36.1	5.1
3	35.5	28.6	27.6	1.2	46.2	1.3	20.8	93.0	127.2	65.7	8.2	3.4
4	26.4	14.9	2.4	2.0	38.8	1.4	6.7	21.5	464.3	72.3	29.2	4.0
5	27.7	2.9	3.2	1.4	11.7	1.3	11.8	23.7	144.1	64.5	22.3	5.6
6	52.7	11.3	25.5	1.5	15.7	1.5	12.6	18.6	110.7	68.3	11.2	7.3
7	22.0	4.2	8.7	1.3	26.5	0.9	11.3	22.9	110.1	76.3	7.1	3.1
Min	8.0	2.5	2.4	0.9	5.3	0.4	3.4	6.5	35.0	19.6	3.2	1.3
Avg	25.5	9.3	10.0	1.4	21.5	1.1	24.2	28.4	150.2	61.9	17.7	4.4
Max	52.7	28.6	27.6	2.0	46.2	1.5	89.2	93.0	464.3	82.2	36.1	7.3
Stream ID	Q13	Q14	Q15	Q16	Q17	Q18	Q19	Q20	Q21	Q22	RF1	RF2
0	41.2	6.2	3.3	7.1	3.3	18.6	15.9	4.9	27.8	6.3	71.1	35.9
1	122.4	22.0	4.9	47.4	14.7	53.4	13.9	22.8	398.9	30.1	141.0	70.6
2	94.8	9.9	6.4	23.2	32.0	63.0	28.7	29.0	46.7	34.5	108.5	45.1
3	96.2	15.8	8.4	10.8	15.8	172.5	33.8	8.4	87.6	20.4	94.1	80.2
4	64.7	46.2	11.7	58.8	27.2	29.0	26.5	38.5	55.0	43.1	131.2	87.3
5	89.6	11.5	3.8	22.3	14.3	46.2	24.8	10.0	28.8	31.5	171.9	64.7
6	115.1	31.2	46.2	32.1	20.4	45.1	41.2	12.1	55.1	21.9	92.1	44.1
7	149.1	26.3	16.1	33.2	29.4	156.9	27.1	24.9	58.2	20.9	79.8	46.6
Min	41.2	6.2	3.3	7.1	3.3	18.6	13.9	4.9	27.8	6.3	71.1	35.9
Avg	96.6	21.1	12.6	29.4	19.6	73.1	26.5	18.8	94.8	26.1	111.2	59.3
Max	149.1	46.2	46.2	58.8	32.0	172.5	41.2	38.5	398.9	43.1	171.9	87.3

Preface

The Transaction Processing Performance Council (TPC) is a non-profit corporation founded to define transaction processing and database benchmarks and to disseminate objective, verifiable TPC performance data to the industry. The TPC Benchmark© H (TPC-H) is a decision support benchmark.

TPC Benchmark© H Overview

The TPC Benchmark© H (TPC-H) 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 data modifications against all tables, except possibly during infrequent (e.g., once a month) maintenance sessions;

- The TPC-H database tracks, possibly with some delay, the state of the OLTP database through on-going refresh functions 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 refresh functions may be executed against the database at any time, especially in relation to each other. In addition, this mix of queries and refresh functions is subject to specific ACIDity requirements, since queries and refresh functions may execute concurrently;

- To achieve the optimal compromise between performance and operational requirements, the database administrator can set, once and for all, the locking levels and the concurrent scheduling rules for queries and refresh functions.

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

Further information is available at www.tpc.org

General Items

1 Test Sponsor

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

This benchmark was sponsored by Inspur Inc. and developed and engineered in partnership with Actian Corporation.

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 File Archive contains the Operating System and DBMS parameters used in this benchmark.

3 Configuration Diagrams

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

- *Total number of nodes used, total number and type of processors used/total number of cores*
- *used/total number of threads used (including sizes of L2 and L3 caches);*
- *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, middleware components, software drivers, etc.).*

The Inspur K1-810 features:

Intel(R) Xeon(R) CPU E7-8891 v2 series processors

8-rack unit (RU) rack-mount chassis

Up to 12TB (192 DIMM slots): 4GB/8GB/16GB/32GB/64GB DDR3 up to 1600MHz

Drives are installed in configurable (1 or 2) drive bay modules that provide hot-pluggable front-panel access Each drive bay module can hold up to eight 2.5 x 0.55 in. (63.5 x 14 mm) SAS or SATA hard disk drives (HDDs) or solid state drives (SSDs), for a total of 16 drives 22 PCIe slots

Integrated quad-port Gigabit Ethernet

Baseboard management controller (BMC)

Availability: High-efficiency, hot-plug, redundant power supplies; hot-plug drive bays; TPM; dual internal SD support; hot-plug redundant fans; optional bezel; luggage-tag; ECC memory, interactive LCD screen; extended thermal support; ENERGY STAR(r) compliant, extended power range; Switch Agnostic Partitioning (SWAP)



Inspur K1-810

Both the measured and priced configurations are the same and consist of a single Inspur k1-810 server node with:

4*3.2GHz/37.5MB Intel(R) Xeon(R) CPU E7-8891 v2

16*32GB DDR(2R*4 PC3L-12800R)

1*480GB SSD (SATA6.0gbps)(OS)

8*480GB SSD (SATA6.0gbps)(RDBMS)

1 LSI CONTROLLER 9271-8i

The tested and priced configuration were identical.

Clause 1: Logical Database Design

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 File Archive contains the table definitions and all other statements used to set up the test and qualification databases.

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. No column reordering was used.

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.

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 the queries.

2.2 Verifying Method of 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 DBGEN version 2.16.1 and QGEN version 2.16.1 were used.

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.

TPC supplied QGEN version 2.16.1 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 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.

Supporting Files Archive contains the actual query text and query output. Following are the modifications to the query.

- *Supporting Files Archive contains the actual query text and query output. Following are the modifications to the query.*
- *“select first 100” is used to limit row count*
- *substr(str, start, length) is used for sub-string function*
- *Variant A is used for Q15*

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.

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

Supporting Files Archive contains the Source Code of refresh functions.

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.

All ACID tests were conducted according to specification. The Supporting Files Archive contains the source code of the ACID test scripts.

3.2 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.

3.2.1 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 was 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.

3.2.2 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.

3.3 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:

*$O_TOTALPRICE = SUM(L_EXTENDEDPRICE - L_DISCOUNT) * (1 + L_TAX)$ For each ORDER and LINEITEM defined by $(O_ORDERKEY = L_ORDERKEY)$*

3.3.1 Consistency Test

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 eight execution streams.
3. The consistency of the ORDER and LINEITEM tables was re-verified.

3.4 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.

3.4.1 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.

3.4.2 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 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 Rollback.
3. An ACID query was started for the same O_KEY used in step 1. The ACID query did not see any uncommitted changes made by the ACID Transaction.
4. The ACID Transaction was ROLLED BACK.
5. The ACID query completed.

3.4.3 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 with a transaction T2 that COMMITS and the second with a transaction T2 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. The ACID transaction T1 was suspended prior to Commit.
3. Another ACID Transaction T2 was started using the same O_KEY and L_KEY and a randomly selected DELTA.
4. T2 committed and completed normally.
5. 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".
6. The ACID Transaction T1 was repeated and completed successfully.
7. 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. An ACID 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 and completed.
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.

3.4.4 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.

3.4.5 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 to verify isolation of concurrent read and write transactions on different tables:

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

3.4.6 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 to verify isolation of update transaction during continuous read-only query:

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 verified that T1 completed before Q1.
4. Q1 completed.
5. It was verified that O_TOTALPRICE was updated by T1.

3.5 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.

3.5.1 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 8 disk drives in a RAID5 configuration. The tests were conducted on the qualification database. These steps were performed to demonstrate that committed updates are preserved across a permanent irrecoverable failure of a disk drive:

1. The consistency of the ORDERS and LINEITEM tables was verified using 800 randomly chosen values for O_ORDERKEY.
2. Exactly 8 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 disk drive was removed from the SUT. Because of data redundancy the SUT continued to process transactions without interruption.
4. After processing transactions from all 8 streams for a few more minutes the power source of the SUT was switched off
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.

3.5.2 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.

See durability test in section 3.5.1.

3.5.3 Loss of System Power

Guarantee the database and committed updates are preserved despite a Loss of all external power to the SUT for an indefinite time period.

See durability test in section 3.5.1.

3.5.4 Memory Failure

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

See durability test in section 3.5.1.

Clause 4: Scaling and Database Population

4.1 Initial 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 4.1 lists the TPC Benchmark H defined tables and the row count for each table as they existed upon completion of the build.

Table 4. 1: Initial Number of Rows

Table Name	Row Count
customer	150,000,000
Region	5
Nation	25
Supplier	10,000,000
Part	200,000,000
Partsupp	800,000,000
Orders	1,500,000,000
Lineitem	5,999,989,709

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.

The storage system consisted of:

- 1 x LSI 9271-8I Adapter RAID controller.
- 8 x 480 GB 6Gb SAMSUNG SSD (SATA6.0gbps).

The database tables and the temporary files were distributed on a RAID 5 volume created across eight 480 GB SAMSUNG SSD (SATA6.0gbps) disk drives.

LSI 9271-8I Adapter	1 x 480GB	RAID 0	480GB	OS, Vector
	8 x 480GB	RAID 5	3,840GB	DB Files, temp

4.3 Mapping of Database Partitions/Replications

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

Database partitioning/replication were not used.

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.

The database tables and the temporary files were distributed on a RAID 5 volume created across nine 480 GB SAMSUNG SSD drives.

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 2.16.1 was used, no modifications were made.

4.6 Database Load time

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

The database load time was 13 hours 12 minutes and 22 seconds

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.

The database storage ratio can be found in Table 4.7

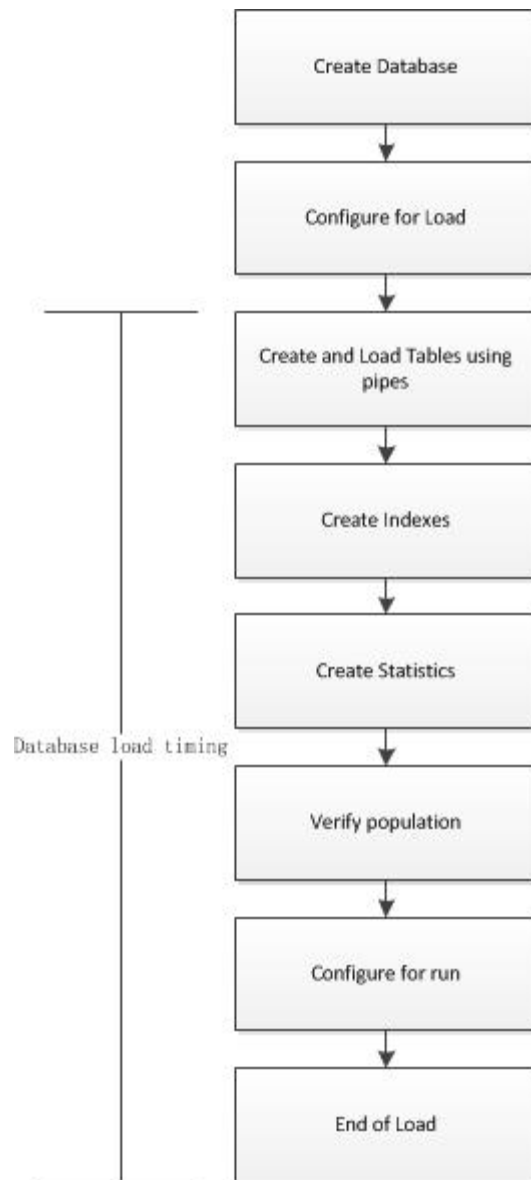
Table 4.7: Data Storage Ratio

Storage Devices	Total Storage Capacity	Scale factor	Data Storage Ratio
9X480G SAMSUNG SSD drivers	4320G	1000	4.32

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.

DBGEN data was passed to the load script using pipes.



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 must be disclosed.

The memory to database size percentage is 51.2%

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 indices that had been created on the database verify referential integrity of the data and the row counts at the end of the load. 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 Function
2. Stream 00 Execution
3. RF2 Refresh Function.

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 the executive summary.

5.4 Number of Streams for The Throughput Test

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

7 query streams were used for the Throughput Test.

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 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 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.

Start and finish time for each update function in the update stream are included in the Numerical Quantities Summary earlier in the Executive Summary.

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 included in the Numerical Quantities Summary earlier in the Executive Summary.

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 Metrics and Numerical Quantities from Both Runs

A description of the method used to determine the reproducibility of the measurement results must be reported. This must include the performance metrics (QppH and QthH) from the reproducibility runs.

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

Table 5.10: Performance Metric

Run	QppH @ 1000GB	QthH @ 1000GB	QpH @ 1000GB
Run 1	508,628.9	502,173.9	505,391.1
Run 2	534,288.1	440,699.5	485,242.7

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.

There was no activity between Run1 and Run2.

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

The supporting files contains the result of this verification

5.13 Referential Integrity

Verify referential integrity in the database after the initial load.

The supporting files contains the result of this verification

.

Clause 6: SUT and Driver Implementation

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 starts the stream 0 SQL script along with the refresh functions 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 initiates all of the SQL streams and the refresh stream.

The scripts used for the driver are included in the supporting files.

6.2 Implementation Specific Layer

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

There was no Implementation Specific Layer, only native scripts and SQL.

6.3 Profile-Directed Optimization

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

Profile-directed optimization was not used.

Clause 7: Pricing

7.1 Hardware and Software Pricing

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 all hardware and software, including the 3-year support, is provided in the Executive Summary in the Abstract section of this report. The price quotations are included in Appendix A.

7.2 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.

The pricing details for this disclosure are contained in the executive summary pages.

7.3 Availability Dates

The committed delivery date for general availability of products used in the price calculations must be reported. When the priced system includes products with different availability dates, the availability date reported on the executive summary must be the date by which all components are committed to being available. The full disclosure report must report availability dates individually for at least each of the categories for which a pricing subtotal must be provided.

All components of the SUT will be available on the date of publication.

Supporting File Index

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

Table 8.0: Supporting File Index

Clause	Description	Archive File
Clause 1	Operating System and Database settings	Clause1.zip
Clause 2	Qualification Queries and Output	Clause2.zip
Clause 3	ACID scripts and output	Clause3.zip
Clause 4	Database load scripts	Clause4.zip
Clause 5	Queries and output	Clause5.zip
Clause 6	Implementation code for measured runs	Clause6.zip
Clause 8	RFs source and parameters	Clause8.zip

Auditors' Information and Attestation Letter

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.

The auditor's letter is included in the following section.

This benchmark was audited by:

Francois Raab
InfoSizing, Inc. (sizing.com)
531 Crystal Hills Blvd.
Manitou Springs, CO 80829
USA
+1 (719) 473-7555

Appendix A

Prepared by: Contact Name: Albert Cheng Contact Phone: +86 21 32533981 Contact Email: albert.cheng@actian.com	Actian Corporation 500 Arguello Street Suite 200 Redwood City, CA 94063 United States Phone: (650) 587-5500 Fax: (650) 649-2358	Order Number: 10-16942 Order Date: 31 March, 2014 Account Number: 911985 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. Payment Terms: Net 90 Validity Period: 30 Days Message: SMP - INSPUR - AAS
---	--	--

PREPARED FOR (BILL TO)	SHIP TO
Attn: Albert Cheng Advanced Analytic Service Limited 100 Anxi Road Shanghai Changning District 200050 China	Advanced Analytic Service Limited 100 Anxi Road Shanghai Changning District 200050 China

Product	Contract Start	Contract End	Version	Platform	Unit	Qty	Price	Extended Price
VECTOR-AAD-LTD-TRM Includes during the term: software license rights per purchased node for one year per contract, no support services. This license only allows Actian Analytics Database - Vector on a single server (with up to 1 TB of uncompressed source system extracted data).	31-Mar-2014	30-Mar-2017	VW 3.0	Linux X86 64-bit	Tera Byte	3.0	CNY 400,000.00	CNY 1,200,000.00

Product	Contract Start	Contract End	Version	Platform	Unit	Qty	Price	Extended Price
VECTOR-AAD-LTD-TRM -MNT Includes during the 1-year term: Enterprise Support Services for Vector on a single server (with up to 1 TB of uncompressed source system extracted data).	31-Mar-2014	30-Mar-2015	VW 3.0	Linux X86 64-bit	Tera Byte	3.0	CNY 88,000.00	CNY 264,000.00

Grand Total: CNY 1,464,000.00

End User
Inspur (Beijing) Electronic Information Industry Co.,Ltd

**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.*

ACCEPTANCE	Actian
SIGNATURE:	
NAME:	
TITLE:	
DATE:	

ACCEPTANCE	Customer
SIGNATURE:	
NAME:	
TITLE:	
DATE:	



The Right Metric For Sizing IT

TPC Transaction Processing Performance Council

Certified Auditors

Benchmark sponsor: Inspur Beijing Electronic Information Industry Co.,Ltd Building C.NO.2 Xinxu Road Shangdi, Haidian District,Beijing China

May 20, 2014

I verified the TPC Benchmark H (TPC-HTM v2.16.0) performance of the following configuration:

Platform: INSPUR K1
Operating System: Inspur K-UX2.2
Database Manager: Actian Vector 3.0.0
Other Software: n/a

The results were:

Performance Metric 485,242.7 QphH@1000GB

TPC-H Power 534,288.1
TPC-H Throughput 440,699.5
Database Load Time 13h 12m 22s

Server

INSPUR K1

CPUs 4 x Intel Xeon Processor E7-4891 v2 (3.2 GHz, 37.5MB Cache)
Memory 512 GB
Disks Qty Size Type
9 480 GB SATA 6Gbps SSD

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:

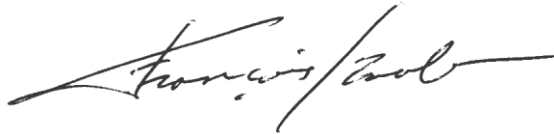
- The database records were defined with the proper layout and size
The database population was generated using DBGen
The database was properly scaled to 1,000GB and populated accordingly
The compliance of the database auxiliary data structures was verified
The database load time was correctly measured 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 7 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