



Ant Group

TPC Benchmark™ H Full Disclosure Report

# Alipay Cloud Explorer

(with 354 Compute Containers)

using **Explorer V1.0**

and **UCS Storage System V1.15**

**First Edition**

**May 23, 2024**

Alipay Cloud Explorer  
using  
Explorer V1.0 and UCS V1.15

In this report, the results of benchmark tests are greatly influenced by the specific workload. Differences in this and other factors will cause variations in the relative performance of different systems. Therefore, TPC Benchmark<sup>TM</sup> H should be only considered as a reference.

**Copyright © 2024 Ant Group All rights reserved.**

All the information contained in this report is available for direct citation, but it is not permitted to publish altered versions of the results publicly.

Alipay Cloud Explorer, Explorer and UCS are trademarks owned by Ant Group in China and other countries.

We would like to highlight that all the systems and infrastructure referred to in this document are now fully accessible through the Realtime Analysis product offered by Alipay Mini Program Cloud. It is worth mentioning that while we do provide commercially available entrances to all users, specific commercial scenarios may necessitate human assistance for their activation. Moreover, the costing associated with these services is determined by the pricing information disclosed on the platform. For further details and a more extensive understanding, we encourage you to visit the following website: <https://cloud.alipay.com/>.

# **Abstract**

## **Overview**


This report provides an in-depth exploration of the methodology, processes, and outcomes achieved through the utilization of Explorer V1.0 as the computing engine, UCS V1.15 as the storage engine, and the deployment of 354 computing node containers to conduct TPC Benchmark™ H testing on the Alipay Mini Program Cloud. The chosen operating system for this endeavor is Alibaba Group Enterprise Linux Server release 7.2 (Paladin).

TPC Benchmark™ H, which stands for Transaction Processing Performance Council - TPC-H, represents a revered benchmark utilized to evaluate the performance of decision support systems and data warehouses. It effectively measures the query and reporting capabilities of a system by simulating real-world business intelligence workloads. Through a meticulously crafted set of SQL queries, TPC-H assesses various aspects of a system's performance, encompassing elements such as response time and throughput. The industry widely adopts TPC-H to facilitate comparisons between database systems and hardware configurations, thus empowering organizations to make informed decisions when selecting a database solution or optimizing their existing systems to enhance decision support capabilities.

The testing conducted in this report diligently adheres to the requirements set forth in the TPC Benchmark™ H Standard Specification, Revision 3.0.1 version.

## **Executive Summary**

The first section started from next page is an overview of a TPC-H implementation, which contains the main metrics, configuration and pricing information.

 <b>ANT GROUP</b>	<b>Alipay Cloud Explorer</b>		<b>TPC-H Rev. 3.0.1</b> <b>TPC Pricing Rev. 2.9.0</b>	
			<b>Report Date:</b> <b>23-May-24</b>	
Total System Cost	Composite Query per Hour Metric		Price/Performance	
<b>334,100,842 CNY</b>	<b>54,803,403.2</b> <b>QphH@100,000GB</b>		<b>6096.36 CNY</b> <b>¥/kQphH@100,000GB</b>	
Database Size	Database Manager	Operating System	Other Software	Availability Date
<b>100,000 GB</b>	<b>Explorer V1.0</b>	<b>Alibaba Group</b> <b>Enterprise Linux</b> <b>Server 7.2 (Paladin)</b>	<b>UCS Storage System V1.15</b>	<b>07/01/2024</b>



Database Load Time: 07:32:27 Load Includes Backup: N Total Data Storage/Database Size: 10.97 Memory to Database Size Percentage: 182.8%	<b>Storage Redundancy Levels:</b> Base Tables: Level Two Auxiliary Data Structures: Level Two DBMS Temporary Space: Level Two OS and DBMS Software: Level Two
--	---

**System Configuration:**

	ExplorerProxy	ExplorerComputing	UCS Master	System Total
#Container	1	354	2	357
vCPUs	104	104	104	37,128
Memory	512 GB	512 GB	512 GB	182,784 GB
Storage	3072 GB	3,072 GB	3,072 GB	1,096,704 GB

**Total Storage: 1,096,704.00 GB**



## Alipay Cloud Explorer

**TPC-H Rev. 3.0.1  
TPC Pricing Rev. 2.9.0**

**Report Date:  
23-May-24**

Description	Part Number	Source	Unit Price	Qty	Extended Price	3yr Maint. Price	
<b>License Computer and Software Service</b>							
Alipay Cloud							
Explorer Computing Resources							
37,128 vCPUs, 1,096,704 GB memory, prepay and occupied for 3 years							
1 ExplorerProxy, each with							
104 vCPUs,							
512 GB RAM,							
354 ExplorerComputing, each with							
104 vCPUs,							
512 GB RAM,							
China							
(Shanghai)							
1							
282,209,357							
1							
282,209,357							
Included							
2 UCSMaster, each with							
104 vCPUs,							
512 GB RAM,							
Data Storage Resources							
1,096,704 GB							
China							
(Shanghai)							
1							
51,878,486							
1							
51,878,486							
Included							
					<b>Subtotal</b>	<b>334,087,843</b>	<b>0.00</b>
<b>Other Components</b>							
MacBook Pro M3							
2							
12,999							
					<b>Subtotal</b>	<b>12,999</b>	
Source: 1=Alipay Cloud, 2=Tmall.com					<b>3-year Cost of Ownership:</b>		<b>334,100,842.0</b>
					<b>QphH@100000GB:</b>		<b>54,803,403.2</b>
					<b>¥/kQphH@100000GB:</b>		<b>6096.36</b>

**Audited by: Doug Johnson**

*Prices used in TPC benchmarks reflect the actual prices a customer would pay for a one-time purchase of the stated Line Items. 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 Line Items. For complete details, see the pricing section of the TPC Benchmark Standard. If you find that the stated prices are not available according to these terms, please inform the TPC at [pricing@tpc.org](mailto:pricing@tpc.org). Thank you.*



# Alipay Cloud Explorer

TPC-H Rev. 3.0.1  
TPC Pricing Rev. 2.9.0

Report Date:  
23-May-24

## Measurement Results

Database Scaling (SF/Size)	100,000 GB
Total Data Storage/Database Size	10.97
Percentage Memory/Database Size	182.8%
Start of Database Load Time	2024/04/10 03:25:45
End of Database Load Time	2024/04/10 10:58:12
Database Load Time	07:32:27
Query Streams for Throughput Test (S)	11
TPC-H Power	47,911,194.3
TPC-H Throughput	62,687,082.7
TPC-H Composite	54,803,403.2
Total System Price Over 3 Years	334,100,842.0 (CNY)
TPC-H Price/Performance Metric (¥/kQpH@100,000GB)	6096.36 (CNY)

## Measurement Interval

Measurement Interval in Throughput Test (Ts)	1,389.76
--	----------

## Duration of stream execution:

Power Run	Seed	Query Start Time	Total Time	RF1 Start Time	RF2 Start Time
		Query End Time	(hh:mm:ss)	RF1 End Time	RF2 End Time
	410105812	2024-04-10 11:02:19	00:03:17	2024-04-10 11:01:40	2024-04-10 11:05:36
		2024-04-10 11:05:36		2024-04-10 11:02:18	2024-04-10 11:05:57
Throughput Stream	Seed	Query Start Time	Total Time	RF1 Start Time	RF2 Start Time
		Query End Time	(hh:mm:ss)	RF1 End Time	RF2 End Time
1	410105813	2024-04-10 11:05:57	00:20:01	2024-04-10 11:05:56	2024-04-10 11:06:38
		2024-04-10 11:25:58		2024-04-10 11:06:38	2024-04-10 11:07:02
2	410105814	2024-04-10 11:05:57	00:23:09	2024-04-10 11:07:02	2024-04-10 11:07:37
		2024-04-10 11:29:06		2024-04-10 11:07:37	2024-04-10 11:07:55
3	410105815	2024-04-10 11:05:57	00:15:46	2024-04-10 11:07:55	2024-04-10 11:08:38
		2024-04-10 11:21:43		2024-04-10 11:08:38	2024-04-10 11:09:01
4	410105816	2024-04-10 11:05:57	00:21:56	2024-04-10 11:09:01	2024-04-10 11:09:38
		2024-04-10 11:27:53		2024-04-10 11:09:38	2024-04-10 11:10:04
5	410105817	2024-04-10 11:05:57	00:19:01	2024-04-10 11:10:04	2024-04-10 11:10:40
		2024-04-10 11:24:58		2024-04-10 11:10:40	2024-04-10 11:11:01
6	410105818	2024-04-10 11:05:57	00:19:34	2024-04-10 11:11:02	2024-04-10 11:11:39
		2024-04-10 11:25:31		2024-04-10 11:11:39	2024-04-10 11:12:02
7	410105819	2024-04-10 11:05:57	00:18:33	2024-04-10 11:12:02	2024-04-10 11:12:38
		2024-04-10 11:24:30		2024-04-10 11:12:38	2024-04-10 11:13:03
8	410105820	2024-04-10 11:05:57	00:15:14	2024-04-10 11:13:03	2024-04-10 11:13:42
		2024-04-10 11:21:11		2024-04-10 11:13:42	2024-04-10 11:14:09
9	410105821	2024-04-10 11:05:57	00:19:35	2024-04-10 11:14:09	2024-04-10 11:14:46
		2024-04-10 11:25:32		2024-04-10 11:14:46	2024-04-10 11:15:06
10	410105822	2024-04-10 11:05:57	00:21:16	2024-04-10 11:15:06	2024-04-10 11:15:42
		2024-04-10 11:27:13		2024-04-10 11:15:42	2024-04-10 11:16:04
11	410105823	2024-04-10 11:05:57	00:21:40	2024-04-10 11:16:04	2024-04-10 11:16:37
		2024-04-10 11:27:37		2024-04-10 11:16:37	2024-04-10 11:16:59



# Alipay Cloud Explorer

TPC-H Rev. 3.0.1  
TPC Pricing Rev. 2.9.0

Report Date:  
23-May-24

## 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	2.97	8.12	5.41	2.35	7.12	2.28	7.31	7.61	16.41	5.13	38.05	3.29
1	4.56	18.66	9.66	3.97	33.82	63.74	12.22	14.11	41.79	11.81	47.75	5.56
2	3.23	56.51	5.74	3.10	128.10	87.00	11.08	233.08	87.60	48.76	37.15	18.12
3	3.63	48.59	8.99	96.79	47.81	8.48	28.00	34.74	17.42	29.69	44.45	5.87
4	20.23	93.76	12.82	46.05	16.28	10.24	12.56	58.85	76.11	162.16	39.27	20.83
5	350.06	11.36	16.58	11.83	21.32	2.80	49.81	14.91	27.71	8.84	59.64	5.24
6	182.60	14.65	69.12	12.19	11.33	19.24	27.91	154.30	84.60	144.83	53.88	118.81
7	3.27	99.87	14.20	4.81	154.94	10.24	72.78	16.28	72.36	17.12	59.04	7.35
8	20.77	7.38	95.91	21.99	86.47	79.15	88.11	15.84	114.06	28.53	44.60	7.91
9	58.41	52.00	54.55	4.51	10.90	104.73	19.56	17.17	107.64	28.83	101.65	53.41
10	5.56	208.55	8.78	152.27	15.78	4.01	19.21	18.36	111.18	12.70	71.75	4.04
11	33.70	19.04	8.90	239.23	104.67	16.76	13.38	53.07	21.50	37.84	153.24	54.89
Minimum	2.97	7.38	5.41	2.35	7.12	2.28	7.31	7.61	16.41	5.13	37.15	3.29
Average	57.42	53.21	25.89	49.92	53.21	34.06	30.16	53.19	64.87	44.69	62.54	25.44
Maximum	350.06	208.55	95.91	239.23	154.94	104.73	88.11	233.08	114.06	162.16	153.24	118.81

Stream ID	Q13	Q14	Q15	Q16	Q17	Q18	Q19	Q20	Q21	Q22	RF1	RF2
0	4.54	8.92	6.34	6.44	12.01	4.24	4.60	20.02	19.34	3.02	37.80	21.00
1	6.27	46.83	117.44	91.37	96.88	38.50	39.39	252.60	238.73	3.49	41.73	23.84
2	9.25	4.22	124.11	159.85	165.96	18.27	123.33	16.37	19.26	28.15	34.98	17.81
3	6.17	2.44	162.34	58.57	93.19	57.45	25.18	53.42	28.45	83.25	43.12	22.72
4	8.01	266.58	35.98	183.57	18.37	12.08	68.62	22.35	97.99	31.72	36.97	26.33
5	8.41	194.16	13.57	7.29	57.99	14.63	35.88	31.27	48.51	147.96	35.46	21.76
6	11.18	104.18	7.74	8.71	54.50	4.28	8.76	27.17	40.98	12.02	37.09	23.07
7	12.01	3.00	8.59	111.48	82.90	54.49	159.91	68.08	26.08	52.75	35.92	25.08
8	8.39	65.39	28.88	12.35	22.75	7.13	22.17	42.07	78.30	14.39	39.58	26.07
9	43.89	3.04	10.93	13.93	27.55	107.09	56.75	63.20	76.82	156.75	37.92	19.90
10	8.28	15.15	136.11	16.77	161.68	63.24	17.01	20.43	39.69	164.72	35.33	22.09
11	6.06	42.62	54.00	168.19	24.92	168.57	6.71	21.87	27.55	21.46	33.09	21.73
Minimum	4.54	2.44	6.34	6.44	12.01	4.24	4.60	16.37	19.26	3.02	33.09	17.81
Average	11.04	63.04	58.84	69.88	68.23	45.83	47.36	53.24	61.81	59.97	37.42	22.62
Maximum	43.89	266.58	162.34	183.57	165.96	168.57	159.91	252.60	238.73	164.72	43.12	26.33

# Table of Contents

<b>ABSTRACT .....</b>	<b>3</b>
OVERVIEW.....	3
EXECUTIVE SUMMARY .....	3
<b>TPC BENCHMARK H OVERVIEW .....</b>	<b>10</b>
<b>0 GENERAL ITEMS.....</b>	<b>1</b>
0.1 BENCHMARK SPONSOR.....	1
0.2 PARAMETER SETTINGS.....	1
0.3 CONFIGURATION DIAGRAM.....	2
<b>1 CLAUSE 1: LOGICAL DATABASE DESIGN RELATED ITEMS .....</b>	<b>3</b>
1.1 DATABASE DEFINITION STATEMENTS.....	3
1.2 PHYSICAL ORGANIZATION .....	3
1.3 HORIZONTAL PARTITIONING .....	3
1.4 REPLICATION .....	4
<b>2 CLAUSE 2: QUERIES AND REFRESH FUNCTIONS RELATED ITEMS.....</b>	<b>5</b>
2.1 QUERY LANGUAGE .....	5
2.2 VERIFYING METHOD OF RANDOM NUMBER GENERATION.....	5
2.3 QUERY TEXT AND OUTPUT DATA FROM QUALIFICATION DATABASE.....	5
2.4 QUERY SUBSTITUTION PARAMETERS AND SEEDS USED.....	5
2.5 ISOLATION LEVEL .....	6
2.6 SOURCE CODE OF REFRESH FUNCTIONS .....	6
<b>3 CLAUSE 3: DATABASE SYSTEM PROPERTIES .....</b>	<b>7</b>
3.1 ACID PROPERTIES .....	7
3.2 ATOMICITY REQUIREMENTS .....	7
3.2.1 <i>Atomicity of Completed Transactions</i> .....	7
3.2.2 <i>Atomicity of Aborted Transactions</i> .....	8
3.3 CONSISTENCY REQUIREMENTS .....	8
3.3.1 CONSISTENCY TEST.....	8
3.4 ISOLATION REQUIREMENTS.....	9
3.4.1 ISOLATION TEST 1 - READ-WRITE CONFLICT WITH COMMIT .....	9
3.4.2 ISOLATION TEST 2 – READ-WRITE CONFLICT WITH ROLLBACK .....	10
3.4.3 ISOLATION TEST 3 – WRITE-WRITE CONFLICT WITH COMMIT.....	10
3.4.4 ISOLATION TEST 4 – WRITE-WRITE CONFLICT WITH ROLLBACK.....	10
3.4.5 ISOLATION TEST 5 – CONCURRENT READ AND WRITE TRANSACTIONS ON DIFFERENT TABLES .....	11
3.4.6 ISOLATION TEST 6 – UPDATE TRANSACTIONS DURING CONTINUOUS READ-ONLY QUERY STREAM .....	11
3.5 DURABILITY REQUIREMENTS.....	12
3.5.1 PERMANENT UNRECOVERABLE FAILURE OF ANY DURABLE MEDIUM .....	12
3.5.2 SYSTEM CRASH.....	12
3.5.3 MEMORY FAILURE .....	13
3.5.4 LOSS OF EXTERNAL POWER .....	13



3.5.5	NODE OR CONTROLLER FAILURE .....	13
<b>4</b>	<b>CLAUSE 4: SCALING AND DATABASE POPULATION .....</b>	<b>14</b>
4.1	INITIAL CARDINALITY OF TABLES .....	14
4.2	DISTRIBUTION OF TABLES AND LOGS ACROSS MEDIA.....	14
4.3	MAPPING OF DATABASE PARTITIONS/REPLICATION .....	15
4.4	IMPLEMENTATION OF DATA REDUNDANCY .....	15
4.5	DBGEN MODIFICATIONS .....	15
4.6	DATABASE LOAD TIME .....	16
4.7	DATA STORAGE RATIO .....	16
4.8	DATABASE LOAD MECHANISM DETAILS AND ILLUSTRATION.....	16
4.9	QUALIFICATION DATABASE CONFIGURATION .....	17
4.10	MEMORY TO DATABASE SIZE PERCENTAGE .....	17
<b>5</b>	<b>CLAUSE 5: PERFORMANCE METRICS AND EXECUTION RULES RELATED ITEMS.....</b>	<b>18</b>
5.1	SYSTEM ACTIVITY BETWEEN LOAD AND PERFORMANCE TESTS.....	18
5.2	STEPS IN THE POWER TEST.....	18
5.3	TIMING INTERVALS FOR EACH QUERY AND REFRESH FUNCTIONS .....	18
5.4	NUMBER OF STREAMS FOR THE THROUGHPUT TEST .....	18
5.5	START AND END DATE/TIME OF EACH QUERY STREAM.....	19
5.6	TOTAL ELAPSED TIME OF THE MEASUREMENT INTERVAL .....	19
5.7	REFRESH FUNCTION START DATE/TIME AND FINISH DATE/TIME.....	19
5.8	TIMING METRIC PRECISION.....	19
5.9	PERFORMANCE METRICS.....	19
5.10	THE PERFORMANCE METRIC AND NUMERICAL QUANTITIES FROM BOTH RUNS.....	20
5.11	SYSTEM ACTIVITY BETWEEN PERFORMANCE TESTS .....	20
5.12	DOCUMENTATION TO SATISFY CLAUSE 5.2.7 .....	20
5.13	QUERY OUTPUT VALIDATION .....	20
<b>6.</b>	<b>CLAUSE6: SUT AND DRIVER IMPLEMENTATION RELATED ITEMS .....</b>	<b>21</b>
6.1	DRIVER.....	21
6.2	IMPLEMENTATION SPECIFIC LAYER (ISL) .....	21
<b>7.</b>	<b>CLAUSE 7: PRICING.....</b>	<b>23</b>
7.1	PRICED CONFIGURATION .....	23
7.2	AVAILABILITY DATE AND ORDERABILITY DATE .....	23
7.3	COUNTRY-SPECIFIC PRICING .....	24
<b>8.</b>	<b>CLAUSE 8: FULL DISCLOSURE.....</b>	<b>25</b>
8.1	SUPPORTING FILES INDEX TABLE .....	25
<b>9.</b>	<b>CLAUSE 9: AUDIT RELATED ITEMS.....</b>	<b>26</b>
	<b>APPENDIX A: PRICING INFORMATION .....</b>	<b>27</b>

# TPC Benchmark H Overview

The TPC Benchmark<sup>TM</sup> H (TPC-H) is a Decision Support benchmark. It is a suite of business-oriented adhoc queries and concurrent 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
- 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

## **0 General Items**

### **0.1 Benchmark Sponsor**

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

This TPC-H benchmark is sponsored by Ant Group.

### **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*

*In the event that some parameters and options are set multiple times, it must be easily discernible by an interested reader when the parameter or option was modified and what new value it received each time.*

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

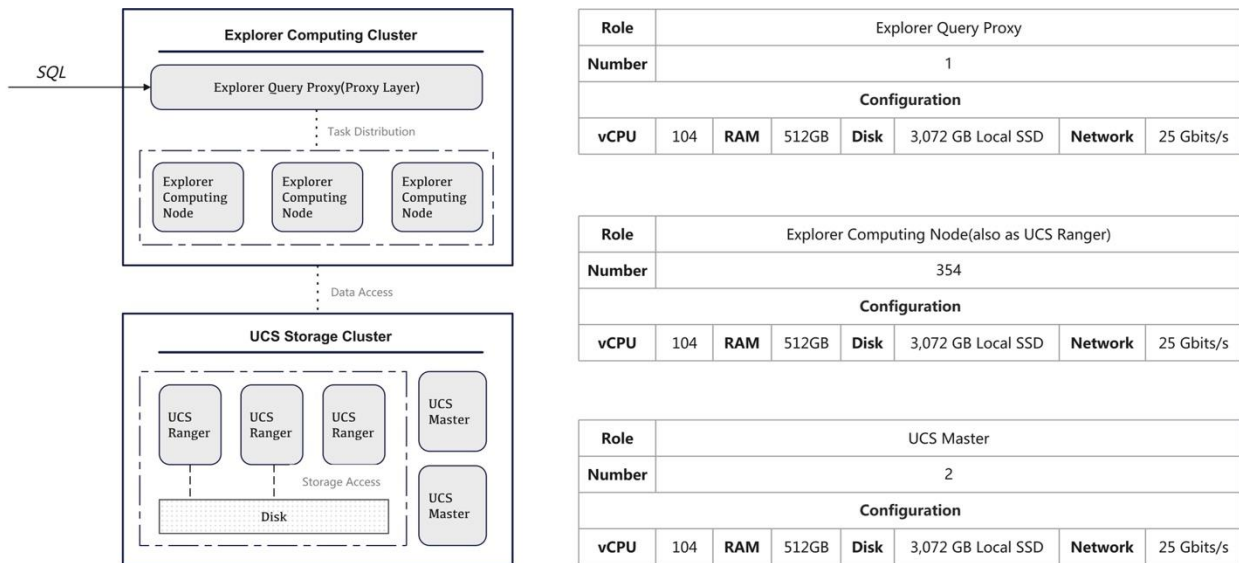
The Supporting File Archive contains the Operating System and DBMS parameters used in this benchmark.

### 0.3 Configuration Diagram

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.).

During this benchmark test, we meticulously ensured that the design, setup and configuration of the Explorer system being measured were identical to those of the priced system available on the Alipay Cloud platform. Our measured system as shown in Figure 1 consisted of 1 Explorer proxy node, 354 Explorer computing nodes which also contains UCS ranger , 2 UCS master nodes, and 1,096,704 GB storage capacity in total.



**Figure 1. Benchmarked and priced system configuration**

# 1 Clause 1: Logical Database Design Related Items

## 1.1 Database Definition Statements

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

The Supporting File Archive contains all table definition statements 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. The physical organization of tables must be reported in the supporting files archive.*

*The concept of physical organization includes, but is not limited to: record clustering (i.e., rows from different logical tables are co-located on the same physical data page), index clustering (i.e., rows and leaf nodes of an index to these rows are co-located on the same physical data page), and partial fill-factors (i.e., physical data pages are left partially empty even though additional rows are available to fill them).*

Physical organization requires no user input. The layout of all other tables except the "LINEITEM" table is consistent with the definition of Clause 1.4. We advance the "L\_LINENUMBER" column in the "LINEITEM" table to after "L\_ORDERKEY".

## 1.3 Horizontal Partitioning

*Horizontal partitioning of tables and rows in the test and qualification databases (see Clause 1.5.4) must be disclosed. Scripts to perform horizontal partitioning must be reported in the supporting files archive.*

The technique of horizontal partitioning is utilized to distribute the data across the cluster nodes through an automated process involving a hash algorithm. The specific columns utilized for the

hashing process are determined and regulated by the DDL statements (refer to the Supporting Files Archive for further details).

## **1.4 Replication**

*Any replication of physical objects must be disclosed and must conform to the requirements of Clause 1.5.7. Scripts to perform any replication must be reported in the supporting files archive.*

No replication was used.

## **2 Clause 2: Queries and Refresh Functions Related Items**

### **2.1 Query Language**

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

Throughout this benchmark, SQL was the exclusive query language utilized.

### **2.2 Verifying Method of Random Number Generation**

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

*Any modifications to the QGen source code must be reported in the supporting files archive.*

TPC-supplied DBGen version 3.0.0 and QGen version 3.0.0 were used.

### **2.3 Query Text and Output Data from Qualification Database**

*The executable query text used for query validation must be reported in the supporting files archive 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 Supporting Files Archive includes the actual query text and query output. The standard queries were used for the most part, but there were some modifications made:

- LIMIT syntax used to restrict the number of output rows (Q2, Q3, Q10, Q18, Q21).

### **2.4 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.

## **2.5 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 the isolation level 3.

## **2.6 Source Code of Refresh Functions**

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

The Supporting Files Archive contains the source code of the refresh functions.



### **3 Clause 3: Database System Properties**

*The results of the ACID tests must be disclosed along with a description of how the ACID requirements were met. All code (including queries, stored procedures etc.) used to test the ACID requirements and their entire output must be reported in the supporting files archive.*

The results of the ACID tests are disclosed as required.

#### **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 portions of the test.*

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

#### **3.2 Atomicity Requirements**

*The system under test must guarantee that transactions are atomic; the system will either perform all individual operations on the data, or will assure that no partially completed operations leave any effects on the data.*

##### **3.2.1 Atomicity of 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 ORDERS, LINEITEM, and HISTORY tables.*

The following operations were performed to verify the atomicity of the completed transactions:

1. Randomly selected one order key, and then retrieved the total price from table ORDERS and the extended price from table LINEITEM.
2. One transaction was performed using the selected order key.
3. The transaction was committed.
4. Retrieved the total price from the table ORDERS and the extended price from the table LINEITEM for the order key.

5. It was verified that the rows had been changed.

### 3.2.2 Atomicity of Aborted Transactions

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

The following operations were performed to verify the atomicity of the aborted transactions:

1. Randomly selected one order key, and then retrieved the total price from table ORDERS and the extended price from table LINEITEM.
2. One transaction was performed using the selected order key and the transaction was stopped before commit.
3. The transaction was rolled back.
4. Retrieved the total price from the table ORDERS and the extended price from the table LINEITEM for the order key.
5. It was verified that the rows had not been changed.

## 3.3 Consistency Requirements

*Consistency is the property of the application that requires any execution of transaction to take the database from one consistent state to another. A consistent state for the TPC-H database is defined to exist when:*

*$O\_TOTALPRICE = SUM(trunc(trunc(L\_EXTENDEDPRICE*(1-L\_DISCOUNT), 2) * (1 + L\_TAX), 2))$  For each ORDER and LINEITEM defined by  $(O\_ORDERKEY = L\_ORDERKEY)$ .*

### 3.3.1 Consistency Test

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

The following query was executed before and after the consistency tests to demonstrate that the database kept staying in a consistent state both initially and after submitting transactions:

```

select *
from (
    select o_orderkey
        , o_totalprice - sum(truncate(truncate(l_extendedprice * (1 -
l_discount), 2) * (1 + l_tax), 2)) AS part_res
    from orders, lineitem
    where o_orderkey = l_orderkey
    group by o_orderkey, o_totalprice
) temp
where not part_res = 0

```

The following operations were performed to verify the consistency of ACID transactions:

1. The consistency of the ORDERS and LINEITEM tables was verified.
2. For each of the 12 execution streams, 100 transactions were prepared.
3. For all 12 execution streams, the prepared ACID transactions were executed.
4. The consistency of the ORDERS and LINEITEM tables was verified again.

### 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.*

The steps of the isolation tests were adapted to the Explorer isolation environment.

#### 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 operations were performed to verify the isolation for a read-only and a read-write committed transaction:

1. Started a query and verified that the row was retrieved.
2. Started an update transaction, read and updated the same row. Stalled before commit.
3. Started the same query and verified that the row has not changed.
4. Committed the update transaction.
5. Started a query and verified that the new row was retrieved.

### **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 operations were performed to verify the isolation for a read-only and a rolled back read-write transaction:

1. Started a query and verified that the row was retrieved.
2. Started an update transaction, read and updated the same row. Stalled before commit.
3. Started the same query and verified that the row has not changed.
4. Rolled back the update transaction.
5. Started a query and verified that the old row was retrieved.

### **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.*

The following operations were performed to verify the isolation of two update transactions:

1. Started an update transaction T1, and stopped T1 immediately prior to COMMIT.
2. Started an update transaction T2. T2 was blocked.
3. Committed T1. T2 should now complete and commit too.
4. Compared extendedprice after update in T1 and extendedprice before update in T2. They should be same.

### **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.*

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

1. Started an update transaction T1, and stopped T1 immediately prior to COMMIT.
2. Started an update transaction T2. T2 was blocked.

3. Rolled back T1. T2 should now complete and commit, too.
4. Compared extendedprice before update in T1 and extendedprice before update in T2. They should be same.

### 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 operations were performed to demonstrate the ability of read and write transactions involving different tables to make progress concurrently:

1. Started a query and verified that the row was retrieved.
2. Started an update transaction, read and updated the same row. Stalled before commit.
3. Started another transaction that performed the following operation: Select random values of PS\_PARTKEY AND PS\_SUPPKEY. Return all columns of the PARTSUPP table for which PS\_PARTKEY and PS\_SUPPKEY are equal to the selected values.
4. Verified that the read transaction completed.
5. Committed the update transaction.
6. Started the same query and verified that the new row was retrieved.

### 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 query was used to ensure sufficient execution to perform the test:

```
SELECT l1.l_orderkey
FROM lineitem l1, lineitem l2, lineitem l3, lineitem l4, lineitem l5
WHERE l1.l_shipdate <= date '1998-12-01'
AND l1.l_orderkey = l2.l_orderkey
AND l1.l_linenumber = l2.l_linenumber
AND l1.l_extendedprice = l3.l_extendedprice
AND l3.l_quantity < 30
AND l4.l_quantity = l1.l_quantity
AND l4.l_orderkey < 20000
AND l5.l_receiptdate = l1.l_receiptdate
AND l5.l_partkey < 140
GROUP BY l1.l_orderkey
```

```
ORDER BY l1.l_orderkey  
LIMIT 2;
```

1. A transaction, T1, which executed the above query on the qualification database, was started using a random DELTA.
2. A transaction, T2, was started for a randomly selected O KEY, L KEY, and DELTA.
3. T2 completed and appropriate rows in the tables ORDERS, LINEITEM, and HISTORY had been changed.
4. T1 was still executing.
5. Transaction T1 completed executing the query.

### **3.5 Durability Requirements**

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

The following steps were performed for the durability test:

1. The consistency of the ORDERS and LINEITEM tables was verified.
2. 500 transactions for each of the 12 executions streams were prepared.
3. After at least 100 ACID transactions were completed by each of the 12 execution streams.
4. A durability failure was induced (see details for each failure shown below).
5. The consistency of the ORDERS and LINEITEM tables was verified again.
6. The durability success files were compared with the HISTORY table.

All durability tests were performed on the cluster shown in section 0.3.

#### **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.*

Disk, node, and controller failure tests were performed together as shown in section 3.5.5.

#### **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.*

The system crash, memory failure and node or controller failure tests were performed together as shown in section 3.5.5.

### **3.5.3 Memory Failure**

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

The system crash, memory failure and node or controller failure tests were performed together as shown in section 3.5.5.

### **3.5.4 Loss of External Power**

*Loss of External Power: Guarantee the database and the effects of committed updates are preserved during the loss of all external power to the SUT for an indefinite time period.*

Various roles of single node crash were performed as described in section 3.5.5

### **3.5.5 Node or Controller Failure**

*Guarantee the database and committed updates are preserved across failure of the controller or the whole node.*

The tested cluster consists of 1 Explorer Query Proxy, 2 UCS Master and 354 Explorer Computing Node. Three tests were performed to test the durability:

1. All power was instantaneously removed from the host machine of the Explorer-Query-Proxy node causing an immediate crash. The full cluster was restored automatically after the Explorer-Query-Proxy machine was started automatically.
2. All power was instantaneously removed from the host machine of the UCS-Master node causing an immediate crash. The full cluster was restored automatically after the UCS-Master machine was restarted automatically.
3. All power was instantaneously removed from the host machine of the Explorer-Computing-Node node causing an immediate crash. The full cluster was restored automatically after the Explorer-Computing-Node machine was restarted automatically.

## 4 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.*

The following table lists the TPC Benchmark H defined tables and the row count for each table (Scale factor 100,000) as they existed upon completion of the test database build.

**Table 1. Table Cardinalities**

<b>Table</b>	<b>#Rows</b>
Lineitem	599,999,969,200
Orders	150,000,000,000
Partsupp	80,000,000,000
Part	20,000,000,000
Customer	15,000,000,000
Supplier	1,000,000,000
Nation	25
Region	5

### 4.2 Distribution of Tables and Logs Across Media

*The distribution of tables and logs across all media must be explicitly described using a format similar to that shown in the following example for both the measured and priced configurations.*

UCS is the storage system, and the computing nodes are deployed together with ucs rangers.

The following table shows where the base tables are distributed.

**Table 2. Distribution of Tables and Logs**

<b>ServerNode</b>	<b>DiskType</b>	<b>Disk Drive</b>	<b>Content</b>
Ucs Ranger	Local SSD Disk	/dev/data	Base tables



All the base tables and transaction logs (WAL) were stored on the HDFS storage, the sizes of tables are shown in following table.

**Table 3. Table Size on HDFS Storage**

Table	Lineitem	Orders	Partsupp	Part	Customer	Supplier	Nation	Region
Size	41.80TB	10.20TB	5.06TB	1.34TB	1.51TB	95.70GB	7.74KB	5.62KB

### 4.3 Mapping of Database Partitions/Replication

*The mapping of database partitions/replications must be explicitly described. The intent is to provide sufficient detail about partitioning and replication to allow independent reconstruction of the test database.*

Tables are horizontally sharded using a hash algorithm. The columns used for the hashing are controlled by DDL statements (see Supporting Files Archive).

### 4.4 Implementation of Data Redundancy

*Implementations may use data redundancy mechanism(s). The type of data redundancy mechanism(s) and any configuration parameters (e.g., RAID level used must be disclosed for each device). If data redundancy mechanism(s) are used in an implementation, the logical intent of their use must be disclosed.*

RAID is not used. The Data Redundancy is internally managed by UCS, a distributed file system, which uses multiple replicas and erasure code to ensure high availability for data access service.

### 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 (see Appendix D) must be reported in the supporting files archive.*

The supplied DBGen version 3.0.0 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.

See Numerical Quantities Summary in the Executive Summary.

## 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 Clause 4.1.3.1. Let  $r$  be the ratio. The reported value for  $r$  must be rounded to the nearest 0.01. That is, reported value =  $\text{round}(r, 2)$ . For example, a system configured with 96 disks of 2.1 GB capacity for a 100GB test database has a data storage ratio of 2.02

**Table 4. Data Storage Capacity**

Server Node	Total Storage Capacity (GB)
Explorer Computing Node (UCS Ranger)	$354 * 3072 \text{ GB} = 1,087,488 \text{ GB}$
UCS Master	$2 * 3072 \text{ GB} = 6,144 \text{ GB}$
Explorer Proxy	$1 * 3072 = 3,072 \text{ GB}$

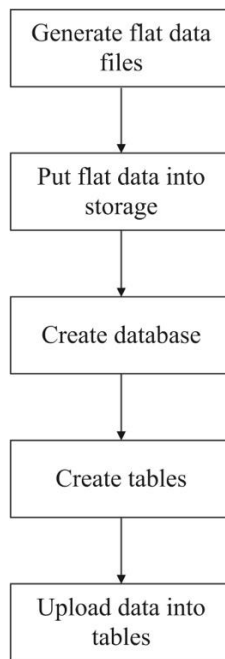
Total disk capacity: 1,096,704 GB Scale factor: 100,000.

The database storage ratio is 10.97.

## 4.8 Database Load Mechanism Details and Illustration

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

The following figure shows the database build and data upload procedures. The raw data were created using DBGen.



**Figure 2. Block Diagram of Database Loading Procedure**

## 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. It is computed by multiplying by 100 the total memory size priced on the SUT (see clause 6.2.1 ) and dividing this number by the size chosen for the test database as defined in Clause 4.1.3.1. Let  $r$  be this ratio. The reported ratio must be rounded to the nearest 0.1. That is, reported value= $\text{round}(r,1)$ . For example, a system configured with 256GB of memory for a 1000GB test database has a memory/database size percentage of 25.6.*

Available Memory: 182,784 GB Scale Factor: 100,000.

The memory to database size percentage is 182.8%.

## **5 Clause 5: Performance Metrics and Execution Rules**

### **Related Items**

#### **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 reported in the supporting files archive including listings of scripts, command logs and system activity.*

There is no activity on the SUT between the conclusion of the load test and the beginning of the performance test.

#### **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 reported in the supporting files archive.*

The following steps were used to implement the power test:

- RF1 Refresh Function from the refresh stream.
- Query Execution from query stream 0.
- RF2 Refresh Function from the refresh stream.

#### **5.3 Timing Intervals for Each Query and Refresh Functions**

*The timing intervals (see Clause 5.3.7) for each query and for both refresh functions must be reported for the power test. The output for each query and for both refresh functions must be reported in the supporting files archive.*

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

#### **5.4 Number of Streams for the Throughput Test**

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

11 query streams and one refresh stream were used in the throughput test.

## **5.5 Start and End Date/Time of Each Query Stream**

*The start time and finish time for each query stream for the throughput test must be disclosed. The output for each query stream for the throughput test must be reported in the supporting files archive.*

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

## **5.6 Total Elapsed Time of the Measurement Interval**

*The total elapsed time of the measurement interval (see Clause 5.3.6) must be disclosed for the throughput test.*

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

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

*The start time and, finish time for each refresh function in the refresh stream for the throughput test must be disclosed. The output of each refresh function in the refresh stream for the throughput test must be reported in the supporting files archive.*

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

## **5.8 Timing Metric Precision**

*The start time and finish time for each query and refresh stream shall be reported to the hundredth of a second. If times are measured with the precision greater than one hundredth of a second, the reported times shall be truncated to the hundredth of a second.*

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

## **5.9 Performance Metrics**

*The computed performance metric, related numerical quantities and the price/performance metric must be disclosed.*

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

## 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 Through-put@Size) from both of the runs must be disclosed.*

**Table 5. Performance Metric**

Run ID	QphH@100,000GB	QppH@100,000GB	QthH@100,000GB
Run 1	54,803,403.2	47,911,194.3	62,687,082.7
Run 2	55,156,246.5	49,807,483.9	61,079,406.0

## 5.11 System Activity between Performance Tests

*Any activity on the SUT that takes place between the conclusion of Run1 and the beginning of Run2 must be fully disclosed including system activity, listings of scripts or command logs along with any system reboots or database restarts.*

There was no activity 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.*

Explorer document can be obtained from <https://opendocs.alipay.com/cloud/09bogy>.

## 5.13 Query Output Validation

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

The output of the validation test is available in the Supporting Files Archive.

## **6. Clause6: SUT and Driver Implementation Related Items**

### **6.1 Driver**

*A detailed textual description of how the driver performs its functions, how its various components interact and any product functionalities or environmental settings on which it relies and all related source code, scripts and configuration files must be reported in the supporting files archive. The information provided should be sufficient for an independent reconstruction of the driver.*

All stream executions are performed by a script. QGEN is used to produce query text. For each power-test run:

4. A shell script is started, executes RF1 and then waits for the query stream to complete.
5. A shell script is started, executes the 22 queries in the required order for stream 0 and then signals to the shell script started in step 1.
6. The shell script started in step 1 is released and executes RF2.

For each throughput-test run:

1. The queries as generated by QGEN are submitted in the order defined by Clause 5.3 5.4 from the driver in several streams (the number of streams is listed in the Numerical Quantities) and runs concurrently.
2. Then throughput RF function pairs are scheduled to run sequentially in one update stream.

The source code of the used scripts are disclosed in the Supporting Files Archive.

### **6.2 Implementation Specific Layer (ISL)**

*If an implementation specific layer is used, then a detailed description of how it performs its functions, how its various components interact and any product functionalities or environmental setting on which it relies must be disclosed. All related source code, scripts and configuration files must be reported in the supporting files archive. The information provided should be sufficient for an independent reconstruction of the implementation specific layer.*

The scripts used to implement the ISL are available in the Supporting Files Archive.

## **6.3 Profile-Directed Optimization**

*If profile-directed optimization as described in Clause 5.2.9 is used, such use must be disclosed. In particular, the procedure and any scripts used to perform the optimization must be reported in the supporting files archive.*

Profile-directed optimization was not used.



## **7. Clause 7: Pricing**

### **7.1 Priced Configuration**

*The pricing methodology used for pricing the Priced Configuration is the “Default 3-Year Pricing Methodology”, as defined in the current revision of the TPC Pricing specification.*

*The system to be priced shall include the hardware, Licensed Compute Services and software components present in the System Under Test (SUT), a communication interface that can support user interface devices, additional operational components configured on the test system, and maintenance on all of the above:*

- *System Under Test.*
- *User Interface Devices and Communications.*
- *Database Storage and Recovery Log*
- *Additional Operational Components.*
- *Software.*

A 3-Year Pricing Methodology is used for this benchmark. A detailed list of hardware and software used in the priced system is included in the pricing sheet in the Executive Summary at the beginning of this report. The price quotations are included in Appendix A.

### **7.2 Availability Date and Orderability Date**

*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 priced configuration will be available for purchase on 07/01/2024.

*For each of the components that are not orderable on the report date of the FDR, the following information must be included in the FDR:*

- *Name and part number of the item that is not orderable*

- *The date when the component can be ordered (on or before the Availability Date)*
- *The method to be used to order the component (at or below the quoted price) when that date arrives*
- *The method for verifying the price*

All priced components will be orderable on 07/01/2024.

### **7.3 Country-Specific Pricing**

*The Priced Locale and Priced Currency of the Priced Configuration must be disclosed.*

The configuration is priced in CNY for the China market.

## 8. Clause 8: Full Disclosure

### 8.1 Supporting Files Index Table

An index for all files and/or directories included in the Supporting Files Archive as required by Clauses 8.3.2 through 8.3.8 must be provided in the report.

**Table 6. Supporting File Index Table**

Clause	Description	Archive Files	Pathname
1	Parameter Settings	benchmark_scripts.zip	RUN/params.log
	DB Creation Scripts		KIT/sql/check_database.sql
			KIT/sql/create_schema.sql
	System Verification		CONFIG/hwinfo.sh
			CONFIG/podinfo.sh
	Toolkit Common Scripts		KIT/scripts
2	Minor query modifications	benchmark_scripts.zip	KIT/tpch_archives/TPC-H_Tools_v3.0.0.zip.patch
3	<i>ACID Test Scripts</i>	benchmark_scripts.zip	KIT/scripts/acid
	<i>ACID Test Results</i>		ACID/
4	Database Load Scripts	benchmark_scripts.zip	KIT/scripts/prepare/load_init.sh
	Qualification Test Results		VLD/
5	Query Output Results	run1results.zip	RUN//run1/
		run2results.zip	RUN//run2/
6	Source Codes and Scripts of Driver	benchmark_scripts.zip	KIT/scripts/test/driver/
7	There are no files to be included for Clause 7.	N/A	N/A
8	Query Parameters & Seeds	run1results.zip	RUN/run1/substitution_parameters.txt
	Executable Query Text		RUN/run1/stream*.sql
	RF function source code		KIT/scripts/tpc_h_full.sh

## 9. Clause 9: Audit Related Items

*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 Doug Johnson of InfoSizing, a certified TPC-H auditor. Further information regarding the audit process may be obtained from:

**Doug Johnson**

InfoSizing ([www.sizing.com](http://www.sizing.com))

63 Lourdes Dr.

Leominster, MA 01453

(978) 343-6562

TPC Benchmark™ H Full Disclosure Report and other information can be downloaded from the Transaction Processing Performance Council website at [www.tpc.org](http://www.tpc.org).

# Appendix A: Pricing Information

Table 7. Product Pricing Table

Product	Part Item	Unit Price	Qty	Ext. Price
Alipay Cloud Explorer(Realtime Analysis)	Computing Resource	0.235CNY / CU·hour <sup>[1]</sup>	1,200,890,880 <sup>[2]</sup>	282,209,357CNY
	Storage Resource	0.0018CNY / GB·hour <sup>[1]</sup>	28,821,381,120 <sup>[3]</sup>	51,878,486CNY
<b>Total</b>				334,087,843CNY

[1] According to the product pricing description(<https://opendocs.alipay.com/cloud/09bdue>), the unit computing power is 0.235CNY, and the unit storage resource is 0.0018CNY.

[2] According to the Executive Summary, the memory used in this benchmark test is 182,784GB, which translates to  $182,784/4=45,696$ CU, and the three-year total is  $45,696*24*365*3$ CU·hour.

[3] According to the Executive Summary, the stroage used in this benchmark test is 1,096,704GB, and the three-year total is  $1,096,704*24*365*3$ CU·hour.

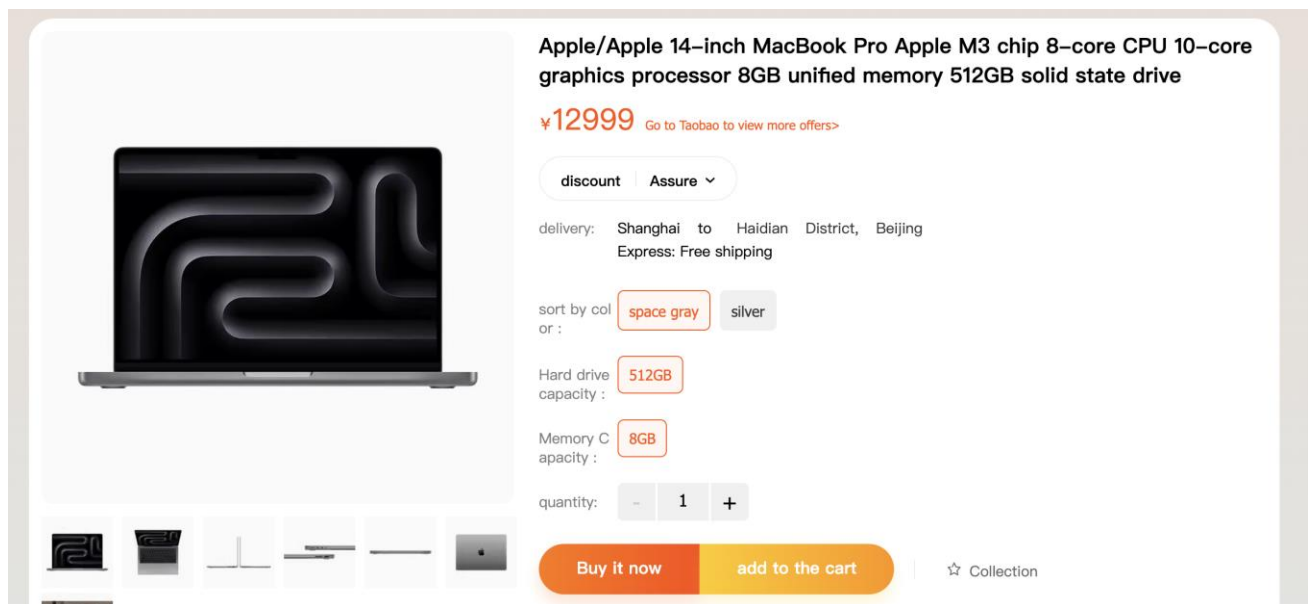


Figure 3. MacBook Pro Apple M3 Purchase Page (Google translated English version)