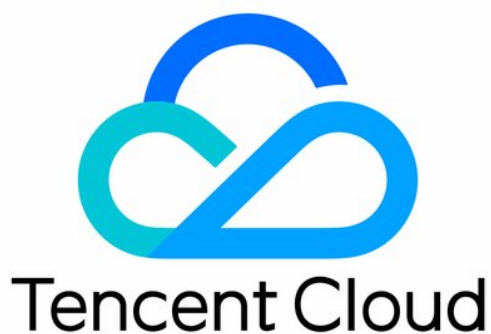


Tencent Cloud TCHouse-D

Performance Testing

Product Documentation



Copyright Notice

©2013-2025 Tencent Cloud. All rights reserved.

Copyright in this document is exclusively owned by Tencent Cloud. You must not reproduce, modify, copy or distribute in any way, in whole or in part, the contents of this document without Tencent Cloud's the prior written consent.

Trademark Notice

 Tencent Cloud

All trademarks associated with Tencent Cloud and its services are owned by Tencent Cloud Computing (Beijing) Company Limited and its affiliated companies. Trademarks of third parties referred to in this document are owned by their respective proprietors.

Service Statement

This document is intended to provide users with general information about Tencent Cloud's products and services only and does not form part of Tencent Cloud's terms and conditions. Tencent Cloud's products or services are subject to change. Specific products and services and the standards applicable to them are exclusively provided for in Tencent Cloud's applicable terms and conditions.

Contents

Performance Testing

TPC-H Performance Testing

Test Scheme Introduction

Test Results for Reference

SSB Performance Testing

Test Scheme Introduction

Test Results for Reference

TPC-DS 性能测试

Testing Scheme Introduction

Test Results for Reference

Performance Testing

TPC-H Performance Testing

Test Scheme Introduction

Last updated : 2025-01-07 11:19:57

This document will introduce how to use TPC-H (Business Intelligence Computing Test) to perform performance testing on Tencent Cloud TCHouse-D. Taking the TPC-H query performance of a 16-core cluster under a 100 GB data set as an example, a reference test scheme is given.

About TPC-H Performance Test

TPC-H is a decision support benchmark that consists of a set of business-oriented ad hoc queries and concurrent data modifications. The data it queries and populates in the database is extensively industry-related. This benchmark test demonstrates the ability of a decision support system to examine large amounts of data, perform highly complex queries, and answer critical business questions. The performance metric reported by TPC-H is called TPC-H Composite Query-per-Hour Performance Metric (QphH@Size), which reflects the system's ability to process multiple queries.

Note:

TPC-H simulates a data warehouse of a sales system. The benchmark test includes 22 queries in total, and the main evaluation metric is the response time of each query, that is, the time required from submitting the query to returning the result. TPC-H test results can comprehensively reflect the system's ability to process queries.

Test Scheme Introduction

Test Environment Preparation

Hardware Environment

In the reference scheme given in this document, the tested cluster includes 1 FE and 3 BEs. The FE/BE node processes are deployed separately. The specific specifications are as follows. It should be noted that in actual testing, such a large amount of hardware resources will not be consumed.

Node Type	Node Specifications
1 FE, standard	CPU:4 cores Memory: 16 GB Hard disk: Enhanced SSD Cloud Disk 200 GB

3 BEs, standard	CPU:16 cores Memory: 64 GB Hard disk: Enhanced SSD Cloud Disk 1000 GB
-----------------	---

Software Version

Tencent Cloud TChouse-D 1.2.7

Test Script Preparation

Download the TPC-H toolkit from [Toolkit Address](#) and compile it.

TPC-H 100 G Data Test

Generating a 100 G Data Set

```
sh gen-tpch-data.sh -s 100 -c 10
```

The data generated is shown in the following table:

TPC-H Table Name	Number of rows	Remarks
REGION	5	Region Table
NATION	25	Country Table
SUPPLIER	1 million	Supplier Table
PART	20 million	Parts List
PARTSUPP	80 million	Parts Supply List
CUSTOMER	15 million	Customer Table
ORDERS	150 million	Order Table
LINEITEM	600 million	Order Details Table

Creating a Table

Modify the doris-cluster.conf configuration file.

Modify configuration: FE_HOST, PASSWORD, DB.

```
# cat doris-cluster.conf

# Any of FE host
export FE_HOST='127.0.0.1'
```

```
# http_port in fe.conf
export FE_HTTP_PORT=8030
# query_port in fe.conf
export FE_QUERY_PORT=9030
# Doris username
export USER='root'
# Doris password
export PASSWORD=''
# The database where TPC-H tables located
export DB='tpch_100g_decimalv3'
# The scale of testing data
export SCALE='100g' # only support '100g' or '1t'
```

Create a table.

```
sh create-tpch-tables.sh
```

Importing the Data

```
sh load-tpch-data.sh
```

```
MySQL [tpch100g]> show data;
```

```
+-----+-----+-----+
| TableName | Size          | ReplicaCount |
+-----+-----+-----+
| customer  | 1.317 GB      | 24            |
| lineitem  | 20.880 GB     | 96            |
| nation    | 2.571 KB      | 1             |
| orders    | 6.302 GB      | 96            |
| part      | 752.470 MB    | 24            |
| partsupp  | 4.375 GB      | 24            |
| region    | 1.090 KB      | 1             |
| supplier  | 85.528 MB     | 12            |
| Total     | 33.693 GB     | 278           |
| Quota     | 1024.000 TB   | 1073741824    |
| Left      | 1023.967 TB   | 1073741546    |
+-----+-----+-----+
11 rows in set (0.00 sec)
```

Querying

```
[root@9 tpch-tools]# sh bin/run-tpch-queries.sh
q1: 2103
q2: 305
```

```
q3: 792
q4: 516
q5: 1036
q6: 60
q7: 493
q8: 954
q9: 4411
q10: 870
q11: 183
q12: 1847
q13: 2886
q14: 165
q15: 255
q16: 398
q17: 520
q18: 1665
q19: 468
q20: 347
q21: 1741
q22: 412
total time: 22427 ms
```

Till then, TCP-H data generation, table creation, import, and query under the 100 GB data set scene is completed.

Test Results for Reference

Last updated : 2025-01-07 11:19:57

This document will provide a reference for the results of the performance test of Tencent Cloud TCHouse-D using TPC-H (Business Intelligence Computing Test).

About TPC-H Performance Test

TPC-H is a decision support benchmark that consists of a set of business-oriented ad hoc queries and concurrent data modifications. The data it queries and populates in the database is extensively industry-related. This benchmark test demonstrates the ability of a decision support system to examine large amounts of data, perform highly complex queries, and answer critical business questions. The performance metric reported by TPC-H is called TPC-H Composite Query-per-Hour Performance Metric (QphH@Size), which reflects the system's ability to process multiple queries.

Test Environment

Hardware Environment

In this document, two model clusters are tested, each containing 1 FE and 3 BEs. The FE/BE node processes are deployed separately. It should be noted that the principle for selecting a cluster of models is to be close to common user configurations, and in actual testing, such a large amount of hardware resources will not be consumed.

Cluster Specifications	Node Type	Specification
Specification 1 (small and medium-sized data scenes)	1 FE	CPU:4 cores Memory: 16 GB Hard disk: Enhanced SSD Cloud Disk 200 GB
	3 BEs	CPU:16 cores Memory: 64 GB Hard disk: Enhanced SSD Cloud Disk 1500 GB
Specification 2 (large-scale data scenes)	1 FE	CPU:16 cores Memory: 64 GB Hard disk: Enhanced SSD Cloud Disk 200 GB
	3 BEs	CPU:64 cores Memory: 256 GB Hard disk: Enhanced SSD Cloud Disk 1500 GB

Software Version

Tencent Cloud TCHouse-D 1.2.7

Test Results for Reference

Test Data

The test is conducted using two data sets, Scale 100 and Scale 1000. The descriptions and data volume of the created tables are as follows:

TPC-H table name	Number of rows - Scale 100	Number of rows - Scale 1000	Remarks
REGION	5	5	Region Table
NATION	25	25	Country Table
SUPPLIER	1 million	10 million	Supplier Table
PART	20 million	200 million	Parts List
PARTSUPP	80 million	800 million	Parts Supply List
CUSTOMER	15 million	150 million	Customer Table
ORDERS	150 million	1.5 billion	Order Table
LINEITEM	600 million	6 billion	Order Details Table

Performance Test Results

Note:

The test result is the average time of three queries of the corresponding SQL file, in seconds.

Query Number	Specification 1-Scale 100 Data Set	Specification 2-Scale 100 Data Set	Specification 2-Scale 1000 Data Set
SQL-1	2.05	1.00	9.67
SQL-2	0.23	0.24	2.05
SQL-3	0.71	0.62	30.46
SQL-4	0.5	0.38	9.74
SQL-5	1.01	0.72	11.10

SQL-6	0.06	0.05	0.58
SQL-7	0.48	0.40	32.13
SQL-8	0.86	0.61	16.00
SQL-9	4.2	3.19	76.98
SQL-10	0.84	0.64	11.36
SQL-11	0.18	0.16	2.14
SQL-12	1.76	1.47	17.03
SQL-13	2.85	1.60	19.02
SQL-14	0.16	0.15	1.73
SQL-15	0.25	0.20	1.66
SQL-16	0.39	0.35	3.56
SQL-17	0.51	0.42	12.38
SQL-18	1.72	1.07	19.64
SQL-19	0.48	0.28	6.75
SQL-20	0.35	0.34	12.90
SQL-21	1.74	0.82	14.61
SQL-22	0.42	0.39	9.60
Total Time	21.74	15.09	321.12

SSB Performance Testing

Test Scheme Introduction

Last updated : 2024-07-31 09:19:12

This document introduces how to use the Star Schema data set to perform performance testing on Tencent Cloud TCHouse-D, and provides a reference scheme for data import and performance testing.

About SSB Performance Test

SSB (Star Schema Benchmark) is a lightweight performance test set for data warehouse scenes. SSB provides a simplified star model data set based on TPC-H, which is mainly used to test the performance of multi-table join queries under the star model. In addition, the industry practice is to flatten SSB into a wide table model (hereinafter referred to as: SSB FLAT) to test the performance of the query engine.

Test Scheme Introduction

Test Environment Preparation

Hardware Environment

In the reference scheme given in this document, the tested cluster includes 1 FE and 3 BEs. The FE/BE node processes are deployed separately. The specific specifications are as follows. It should be noted that in actual testing, such a large amount of hardware resources will not be consumed.

Node Type	Node Specifications
1 FE, standard	CPU:4 cores Memory: 16 GB Hard disk: Enhanced SSD Cloud Disk 200 GB
3 BEs, standard	CPU:16 cores Memory: 64 GB Hard disk: Enhanced SSD Cloud Disk 1000 GB

Software Version

Tencent Cloud TChouse-D 1.2.7

Test Script Preparation

Download the SSB-H toolkit from [Toolkit Address](#) and compile it.

Test SSB data set.

Generate data set.

```
sh bin/gen-ssb-data.sh -s 100 -c 100

277M    /data/ssb-tools/bin/ssb-data//customer.tbl
228K    /data/ssb-tools/bin/ssb-data//date.tbl
5.9G    /data/ssb-tools/bin/ssb-data//lineorder.tbl.1
7.7G    /data/ssb-tools/bin/ssb-data//lineorder.tbl.10
7.6G    /data/ssb-tools/bin/ssb-data//lineorder.tbl.2
7.2G    /data/ssb-tools/bin/ssb-data//lineorder.tbl.3
6.0G    /data/ssb-tools/bin/ssb-data//lineorder.tbl.4
6.0G    /data/ssb-tools/bin/ssb-data//lineorder.tbl.5
6.0G    /data/ssb-tools/bin/ssb-data//lineorder.tbl.6
6.0G    /data/ssb-tools/bin/ssb-data//lineorder.tbl.7
7.4G    /data/ssb-tools/bin/ssb-data//lineorder.tbl.8
7.5G    /data/ssb-tools/bin/ssb-data//lineorder.tbl.9
116M    /data/ssb-tools/bin/ssb-data//part.tbl
17M     /data/ssb-tools/bin/ssb-data//supplier.tbl
```

Under the -s 100 parameter, the size of the data set generated is:

SSB Table Name	Number of rows	Remarks
LINEORDER	600,037,902	Product Order Details Table
CUSTOMER	3,000,000	Customer Information Table
PART	1,400,000	Parts Information Table
SUPPLIER	200,000	Supplier Information Table
DATE	2,556	Date Table

Create a table

Modify the doris-cluster.conf configuration file vim conf/doris-cluster.conf.

Modify configuration: FE_HOST, PASSWORD, DB.

```
# cat doris-cluster.conf

# Any of FE host
export FE_HOST='127.0.0.1'
# http_port in fe.conf
```

```
export FE_HTTP_PORT=8030
# query_port in fe.conf
export FE_QUERY_PORT=9030
# Doris username
export USER='root'
# Doris password
export PASSWORD=''
# The database where SSB tables located
export DB='ssb_100g'
# The scale of testing data
export SCALE='100g' # only support '100g' or '1t'
```

Create a Table:

```
sh bin/create-ssb-tables.sh
```

Import Data

```
sh bin/load-ssb-data.sh
```

Check Imported Data

```
selectcount(*)from part;
selectcount(*)from customer;
selectcount(*)from supplier;
selectcount(*)fromdate;
selectcount(*)from lineorder;
selectcount(*)from lineorder_flat;
```

The amount of data shall be consistent with the number of rows of generated data.

Query

Query SSB tables.

```
# sh bin/run-ssb-queries.sh
q1.1: 46
q1.2: 29
q1.3: 26
q2.1: 340
q2.2: 273
q2.3: 257
q3.1: 542
q3.2: 237
q3.3: 297
q3.4: 57
```

```
q4.1: 732
q4.2: 372
q4.3: 483
total time: 3691 ms
```

Query FLAT tables.

```
# sh bin/run-ssb-flat-queries.sh
q1.1: 26
q1.2: 10
q1.3: 35
q2.1: 85
q2.2: 83
q2.3: 60
q3.1: 157
q3.2: 78
q3.3: 75
q3.4: 13
q4.1: 131
q4.2: 49
q4.3: 33
total time: 835 ms
```

Thus, SSB data generation, table creation, import, and query are completed.

Test Results for Reference

Last updated : 2025-01-07 11:19:57

This document will provide a reference for the results of performance testing of Tencent Cloud TCHouse-D using SSB.

About SSB Performance Test

SSB (Star Schema Benchmark) is a lightweight performance test set for data warehouse scenes. SSB provides a simplified star model data set based on TPC-H, which is mainly used to test the performance of multi-table join queries under the star model. In addition, the industry practice is to flatten SSB into a wide table model (hereinafter referred to as: SSB FLAT) to test the performance of the query engine.

Test Environment

Hardware Environment

In this document, two model clusters are tested, each containing 1 FE and 3 BEs. The FE/BE node processes are deployed separately. It should be noted that the principle for selecting a cluster of models is to be close to common user configurations, and in actual testing, such a large amount of hardware resources will not be consumed.

Cluster Specifications	Node Type	Specification
Specification 1 (small and medium-sized data scenes)	1 FE	CPU:4 cores Memory: 16 GB Hard disk: Enhanced SSD Cloud Disk 200 GB
	3 BEs	CPU:16 cores Memory: 64 GB Hard disk: Enhanced SSD Cloud Disk 1500 GB
Specification 2 (large-scale data scenes)	1 FE	CPU:16 cores Memory: 64 GB Hard disk: Enhanced SSD Cloud Disk 200 GB
	3 BEs	CPU:64 cores Memory: 256 GB Hard disk: Enhanced SSD Cloud Disk 1500 GB

Software Version

Tencent Cloud TCHouse-D 1.2.7

Test Results for Reference

Test Data

The test is conducted using two data sets, Scale 100 and Scale 1000. The descriptions and data volume of the created tables are as follows:

SSB Table Name	Number of Rows - Scale 100	Number of Rows - Scale 1000	Remarks
LINEORDER	600,037,902	5,999,989,709	Product Order Details Table
CUSTOMER	3,000,000	30,000,000	Customer Information Table
PART	1,400,000	2,000,000	Parts Information Table
SUPPLIER	200,000	2,000,000	Supplier Information Table
DATE	2,556	2,556	Date Table

Performance Test Results

Note:

The test result is the average time of three queries of the corresponding SQL file, in seconds (s).

SSB Test Results

Query Number	Specification 1- Scale 100 Data Set	Specification 2- Scale 100 Data Set	Specification 1- Scale 1000 Data Set	Specification 2-Scale 1000 Data Set
Q1.1	0.05	0.04	0.27	0.26
Q1.2	0.03	0.03	0.14	0.13
Q1.3	0.03	0.03	0.13	0.13
Q2.1	0.33	0.15	5.32	1.29
Q2.2	0.27	0.12	4.89	1.19
Q2.3	0.26	0.12	4.67	1.14
Q3.1	0.53	0.39	9.52	7.27
Q3.2	0.23	0.12	4.33	1.06

Q3.3	0.30	0.12	3.58	0.71
Q3.4	0.06	0.05	0.34	0.24
Q4.1	0.73	0.41	10.84	5.59
Q4.2	0.38	0.34	4.15	2.56
Q4.3	0.47	0.54	4.37	4.22
Total time	3.66	2.43	52.54	25.79

SSB FLAT Test Results

Query Number	Specification 1- Scale 100 Data Set	Specification 2- Scale 100 Data Set	Specification 1- Scale 1000 Data Set	Specification 2-Scale 1000 Data Set
Q1.1	0.03	0.02	0.13	0.10
Q1.2	0.01	0.01	0.02	0.02
Q1.3	0.03	0.04	0.23	0.22
Q2.1	0.08	0.05	3.61	1.04
Q2.2	0.08	0.05	2.86	0.37
Q2.3	0.06	0.04	2.54	0.28
Q3.1	0.15	0.09	4.30	1.50
Q3.2	0.08	0.05	3.45	1.09
Q3.3	0.07	0.04	1.93	0.26
Q3.4	0.01	0.02	0.03	0.02
Q4.1	0.13	0.08	4.96	2.22
Q4.2	0.05	0.04	1.59	0.23
Q4.3	0.03	0.03	1.23	0.14
Total time	0.82	0.54	26.87	7.49

TPC-DS 性能测试

Testing Scheme Introduction

Last updated : 2024-12-25 17:55:17

This document introduces how to use TPC-DS to perform performance testing on TCHouse-D. The following section provides a reference testing scheme using a 100 GB data set as an example to evaluate the TPC-DS inquiry performance.

About TPC-DS Performance Testing

TPC-DS is a benchmark test with a focus on decision-making support, designed to evaluate the performance of data warehouses and analytical systems. Developed by the TPC (Transaction Processing Performance Council), it is used to compare different systems' capabilities in handling complex inquiries and large-scale data analysis.

The design goal of TPC-DS is to simulate complex decision-making support workloads found in real-world scenarios. It tests system performance through a series of complex inquiries and data operations, including joins, aggregations, sorts, filters, sub-inquiries, and more. These inquiry patterns cover a range of simple and complex scenarios, such as report generation, data mining, and OLAP (online analytical processing).

Testing Scheme Introduction

Test Environment Preparation

Hardware Environment

In the reference scheme provided in this document, the tested cluster consists of 3 FEs and 3 BEs, with the FE and BE node processes deployed separately. The specific specifications are shown below. Please note that in actual testing, such an extensive amount of hardware resources would not be consumed.

Node Type	Node Specifications
3 FEs, standard	CPU: 16 cores Memory: 64 GB Hard disk: Enhanced SSD 200 GB
3 BEs, standard	CPU: 16 cores Memory: 64 GB Hard disk: Enhanced SSD 1000 GB

Software Version

Tencent Cloud TCHouse-D 2.1.7

Test Script Preparation

Download the TPC-DS toolkit from [Toolkit Address](#) and compile it.

TPC-DS 100 G Data Testing

Generate a 100 G data set.

```
sh bin/gen-tpcds-data.sh -s 100
```

The data generated after execution is as follows:

```
# du -sh bin/tpcds-data/
96G      bin/tpcds-data/
```

Table Name	Original Text File Size	Size After Importing 100 G	Number of Buckets	Number of Rows
call_center	9.2 KiB	13.784 KB	1	30
catalog_page	2.8 MiB	1.216 MB	3	20400
catalog_returns	2.2 GiB	736.137 MB	32	14404374
catalog_sales	29 GiB	9.225 GB	960	143997065
customer	256 MiB	111.185 MB	12	2000000
customer_address	106 MiB	21.386 MB	12	1000000
customer_demographics	76 MiB	6.468 MB	12	1920800
date_dim	9.8 MiB	1.823 MB	12	73049
dbgen_version	111 B	1.184 KB	1	1
household_demographics	142 KiB	20.372 KB	3	7200
income_band	308 B	724.000 B	1	20
inventory	7.7 GiB	871.378 MB	32	399330000
item	56 MiB	25.314 MB	12	204000

promotion	122 KiB	73.989 KB	1	1000
reason	1.9 KiB	7.748 KB	1	55
ship_mode	1.1 KiB	3.251 KB	1	20
store	104 KiB	54.449 KB	1	402
store_returns	3.3 GiB	1.090 GB	32	28795080
store_sales	38 GiB	12.529 GB	960	287997024
time_dim	4.8 MiB	1.087 MB	12	86400
warehouse	1.8 KiB	4.999 KB	1	15
web_page	193 KiB	38.753 KB	1	2040
web_returns	998 MiB	350.227 MB	32	7197670
web_sales	15 GiB	4.645 GB	960	72001237
web_site	6.7 KiB	11.185 KB	1	24
Total	96 G	29.566 GB	3096	959037906

Creating a Table

Modify the doris-cluster.conf configuration file.

Modify the configuration: FE_HOST, PASSWORD, and DB.

```
# cat doris-cluster.conf

# Any of FE host
export FE_HOST='127.0.0.1'
# http_port in fe.conf
export FE_HTTP_PORT=8030
# query_port in fe.conf
export FE_QUERY_PORT=9030
# Doris username
export USER='root'
# Doris password
export PASSWORD=''
# The database where TPC-DS tables located
export DB='tpch_100g'
```

Creating a Table

```
sh bin/create-tpcds-tables.sh -s 100
```

Importing the Data

```
sh load-tpcds-data.sh
```

```
Start time: Thu Oct 31 21:03:55 CST 2024
```

```
End time: Thu Oct 31 21:14:44 CST 2024
```

```
Finish load tpcds data, Time taken: 649 seconds
```

```
=====
analyze database tpcds_100g
```

```
analyze database tpcds_100g with full with sync;
```

```
analyze database tpcds_100g with full with sync total time: 67 s
```

```
MySQL [tpcds_100g]> show data;
```

TableName	Size	ReplicaCount	RemoteSize
call_center	13.784 KB	1	0.000
catalog_page	1.216 MB	3	0.000
catalog_returns	736.137 MB	32	0.000
catalog_sales	9.225 GB	960	0.000
customer	111.185 MB	12	0.000
customer_address	21.386 MB	12	0.000
customer_demographics	6.468 MB	12	0.000
date_dim	1.823 MB	12	0.000
dbgen_version	1.184 KB	1	0.000
household_demographics	20.372 KB	3	0.000
income_band	724.000 B	1	0.000
inventory	871.378 MB	32	0.000
item	25.314 MB	12	0.000
promotion	73.989 KB	1	0.000
reason	7.748 KB	1	0.000
ship_mode	3.251 KB	1	0.000
store	54.449 KB	1	0.000
store_returns	1.090 GB	32	0.000
store_sales	11.713 GB	960	0.000
time_dim	1.087 MB	12	0.000
warehouse	4.999 KB	1	0.000
web_page	38.753 KB	1	0.000
web_returns	350.227 MB	32	0.000
web_sales	4.645 GB	960	0.000
web_site	11.185 KB	1	0.000
Total	28.750 GB	3096	0.000
Quota	1024.000 TB	1073741824	
Left	1023.972 TB	1073738728	

```
| Transaction Quota | 1000 | 1000 | |
+-----+-----+-----+
29 rows in set (0.02 sec)
```

Querying

```
# bash bin/run-tpcds-queries.sh -s 100
```

Thus, the process of TPC-DS data generation, table creation, data import, and inquiry for a 100 GB data set scenario is now complete.

Test Results for Reference

Last updated : 2024-12-25 17:55:16

This document provides a reference for the results of performance testing for TCHouse-D using TPC-DS.

About TPC-DS Performance Testing

TPC-DS is a benchmark test with a focus on decision-making support, designed to evaluate the performance of data warehouses and analytical systems. Developed by the TPC (Transaction Processing Performance Council), it is used to compare different systems' capabilities in handling complex inquiries and large-scale data analysis.

The design goal of TPC-DS is to simulate complex decision-making support workloads found in real-world scenarios. It tests system performance through a series of complex inquiries and data operations, including joins, aggregations, sorts, filters, sub-inquiries, and more. These inquiry patterns cover a range of simple and complex scenarios, such as report generation, data mining, and OLAP (online analytical processing).

Test Environment

Hardware Environment

In the reference scheme provided in this document, the tested cluster consists of 3 FEs and 3 BEs, with the FE and BE node processes deployed separately. The specific specifications are shown below. Please note that in actual testing, such an extensive amount of hardware resources would not be consumed.

Node Type	Node Specifications
3 FEs, standard	CPU: 16 cores Memory: 64 GB Hard disk: Enhanced SSD 200 GB
3 BEs, standard	CPU: 16 cores Memory: 64 GB Hard disk: Enhanced SSD 1000 GB

Software Version

Tencent Cloud TCHouse-D 2.1.7

Test Results for Reference

Test Data

Table Name	Original Text File Size	Size After Importing 100 G	Number of Buckets	Number of Rows
call_center	9.2 KiB	13.784 KB	1	30
catalog_page	2.8 MiB	1.216 MB	3	20400
catalog_returns	2.2 GiB	736.137 MB	32	14404374
catalog_sales	29 GiB	9.225 GB	960	143997065
customer	256 MiB	111.185 MB	12	2000000
customer_address	106 MiB	21.386 MB	12	1000000
customer_demographics	76 MiB	6.468 MB	12	1920800
date_dim	9.8 MiB	1.823 MB	12	73049
dbgen_version	111 B	1.184 KB	1	1
household_demographics	142 KiB	20.372 KB	3	7200
income_band	308 B	724.000 B	1	20
inventory	7.7 GiB	871.378 MB	32	399330000
item	56 MiB	25.314 MB	12	204000
promotion	122 KiB	73.989 KB	1	1000
reason	1.9 KiB	7.748 KB	1	55
ship_mode	1.1 KiB	3.251 KB	1	20
store	104 KiB	54.449 KB	1	402
store_returns	3.3 GiB	1.090 GB	32	28795080
store_sales	38 GiB	12.529 GB	960	287997024
time_dim	4.8 MiB	1.087 MB	12	86400
warehouse	1.8 KiB	4.999 KB	1	15
web_page	193 KiB	38.753 KB	1	2040

web_returns	998 MiB	350.227 MB	32	7197670
web_sales	15 GiB	4.645 GB	960	72001237
web_site	6.7 KiB	11.185 KB	1	24
Total	96 G	29.566 GB	3096	959037906

Performance Test Results

Note:

The test result is the average time of three queries of the corresponding SQL file, in seconds.

Query Number	100 GB data set
query1	0.265
query2	1.644
query3	0.156
query4	3.147
query5	0.313
query6	0.171
query7	0.271
query8	0.169
query9	1.893
query10	0.202
query11	1.974
query12	0.087
query13	0.323
query14	4.288
query15	0.16
query16	0.809
query17	0.44

query18	0.343
query19	0.157
query20	0.095
query21	0.093
query22	3.586
query23	8.414
query24	2.067
query25	0.33
query26	0.142
query27	0.255
query28	1.688
query29	0.363
query30	0.138
query31	0.576
query32	0.051
query33	0.248
query34	0.295
query35	0.583
query36	0.287
query37	0.065
query38	1.494
query39	0.85
query40	0.093
query41	0.054
query42	0.088

query43	0.445
query44	0.64
query45	0.177
query46	0.517
query47	1.214
query48	0.264
query49	0.313
query50	0.294
query51	4.325
query52	0.086
query53	0.157
query54	0.4
query55	0.09
query56	0.219
query57	0.863
query58	0.217
query59	2.468
query60	0.255
query61	0.117
query62	0.273
query63	0.14
query64	1.314
query65	1.608
query66	0.239
query67	11.163

query68	0.476
query69	0.226
query70	1.164
query71	0.329
query72	1.853
query73	0.271
query74	1.8
query75	1.347
query76	0.453
query77	0.26
query78	3.484
query79	1.114
query80	0.597
query81	0.188
query82	0.148
query83	0.187
query84	0.078
query85	0.367
query86	0.306
query87	1.738
query88	1.747
query89	0.207
query90	0.109
query91	0.141
query92	0.043

query93	0.034
query94	0.533
query95	0.205
query96	0.242
query97	1.667
query98	0.3
query99	0.493
Total Time	88.572