#### **操作系统环境**

[admin@oceanbase-single ~]$ cat /etc/system-release

CentOS Linux release 7.6.1810 (Core)

[admin@oceanbase-single ~]$ free -g

 total used free shared buff/cache available

Mem: 15 0 15 0 0 12

Swap: 7 0 7

[admin@oceanbase-single ~]$ cat /proc/cpuinfo | grep processor

processor : 0

processor : 1

processor : 2

processor : 3

processor : 4

processor : 5

processor : 6

processor : 7

#### **数据库环境**

[admin@oceanbase-single ~]$ obclient -h127.0.0.1 -P2883 -uroot@sys -Doceanbase -A

obclient [oceanbase]> select version();

+------------------------------+

| version() |

+------------------------------+

| 5.7.25-OceanBase\_CE-v4.2.2.0 |

+------------------------------+

obclient [oceanbase]> show variables like '%chara%';

+--------------------------+---------+

| Variable\_name | Value |

+--------------------------+---------+

| character\_set\_client | utf8mb4 |

| character\_set\_connection | utf8mb4 |

| character\_set\_database | utf8mb4 |

| character\_set\_filesystem | binary |

| character\_set\_results | utf8mb4 |

| character\_set\_server | utf8mb4 |

| character\_set\_system | utf8mb4 |

+--------------------------+---------+

#### **下载BenmarkSQL并解压**

# 复制文件

[admin@oceanbase-single run]$ cp props.ob my\_ob.ob

# 编辑文件，根据自己配置修改用户和密码

[admin@oceanbase-single run]$ vim my\_ob.ob

db=oracle

driver=com.alipay.oceanbase.obproxy.mysql.jdbc.Driver

conn=jdbc:oceanbase://127.1:2883/tpcc?useUnicode=true&characterEncoding=utf-8

user=root@test\_tenant

password=

warehouses=2

loadWorkers=10 #10个并发

terminals=10 #10个客户端

//To run specified transactions per terminal- runMins must equal zero

runTxnsPerTerminal=0

//To run for specified minutes- runTxnsPerTerminal must equal zero

runMins=1

//Number of total transactions per minute

limitTxnsPerMin=0

//Set to true to run in 4.x compatible mode. Set to false to use the

//entire configured database evenly.

terminalWarehouseFixed=true

//The following five values must add up to 100

newOrderWeight=45

paymentWeight=43

orderStatusWeight=4

deliveryWeight=4

stockLevelWeight=4

// Directory name to create for collecting detailed result data.

// Comment this out to suppress.

resultDirectory=my\_result\_%tY-%tm-%td\_%tH%tM%tS

osCollectorScript=./misc/os\_collector\_linux.py

osCollectorInterval=1

#### **创建测试数据库**

obclient [(none)]> create database tpcc;

obclient [(none)]> show databases;

+--------------------+

| Database |

+--------------------+

| information\_schema |

| mysql |

| oceanbase |

| test |

| tpcc |

+--------------------+

#### **修改创建表的sql文件**

# 修改数据类型varchar2-->varchar

# 社区版只能创建mysql租户，不识别varchar2，因此需要修改下

[admin@oceanbase-single run]$ sed -i 's/varchar2/varchar/g' tableCreates.sql

# 文件中的创建语句给tablegroup加了双引号，我的4.0版本使用会报错，所以就去掉引号了

[admin@oceanbase-single run]$ sed -i 's/"tpcc\_group"/tpcc\_group/g' tableCreates.sql

obclient [tpcc]> create tablegroup "tpcc\_group" ;

ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your OceanBase version for the right syntax to use near '"tpcc\_group"' at line 1

[admin@oceanbase-single run]$ ./runSQL.sh my\_ob.ob sql.oceanbase/tableCreates.sql

# ------------------------------------------------------------

# Loading SQL file sql.oceanbase/tableCreates.sql

# ------------------------------------------------------------

create table bmsql\_config (

cfg\_name varchar(30) primary key,

cfg\_value varchar(50)

);

create tablegroup tpcc\_group ;

create table bmsql\_warehouse (

w\_id integer not null,

w\_ytd decimal(12,2),

w\_tax decimal(4,4),

w\_name varchar(10),

w\_street\_1 varchar(20),

w\_street\_2 varchar(20),

w\_city varchar(20),

w\_state char(2),

w\_zip char(9),

primary key(w\_id)

)tablegroup='tpcc\_group' ;

create table bmsql\_district (

d\_w\_id integer not null,

d\_id integer not null,

d\_ytd decimal(12,2),

d\_tax decimal(4,4),

d\_next\_o\_id integer,

d\_name varchar(10),

d\_street\_1 varchar(20),

d\_street\_2 varchar(20),

d\_city varchar(20),

d\_state char(2),

d\_zip char(9),

PRIMARY KEY (d\_w\_id, d\_id)

)tablegroup='tpcc\_group' ;

create table bmsql\_customer (

c\_w\_id integer not null,

c\_d\_id integer not null,

c\_id integer not null,

c\_discount decimal(4,4),

c\_credit char(2),

c\_last varchar(16),

c\_first varchar(16),

c\_credit\_lim decimal(12,2),

c\_balance decimal(12,2),

c\_ytd\_payment decimal(12,2),

c\_payment\_cnt integer,

c\_delivery\_cnt integer,

c\_street\_1 varchar(20),

c\_street\_2 varchar(20),

c\_city varchar(20),

c\_state char(2),

c\_zip char(9),

c\_phone char(16),

c\_since timestamp,

c\_middle char(2),

c\_data varchar(500),

PRIMARY KEY (c\_w\_id, c\_d\_id, c\_id)

)tablegroup='tpcc\_group' ;

create sequence bmsql\_hist\_id\_seq;

create table bmsql\_history (

hist\_id integer,

h\_c\_id integer,

h\_c\_d\_id integer,

h\_c\_w\_id integer,

h\_d\_id integer,

h\_w\_id integer,

h\_date timestamp,

h\_amount decimal(6,2),

h\_data varchar(24)

)tablegroup='tpcc\_group' ;

create table bmsql\_new\_order (

no\_w\_id integer not null ,

no\_d\_id integer not null,

no\_o\_id integer not null,

PRIMARY KEY (no\_w\_id, no\_d\_id, no\_o\_id)

)tablegroup='tpcc\_group' ;

create table bmsql\_oorder (

o\_w\_id integer not null,

o\_d\_id integer not null,

o\_id integer not null,

o\_c\_id integer,

o\_carrier\_id integer,

o\_ol\_cnt integer,

o\_all\_local integer,

o\_entry\_d timestamp,

PRIMARY KEY (o\_w\_id, o\_d\_id, o\_id)

)tablegroup='tpcc\_group' ;

create table bmsql\_order\_line (

ol\_w\_id integer not null,

ol\_d\_id integer not null,

ol\_o\_id integer not null,

ol\_number integer not null,

ol\_i\_id integer not null,

ol\_delivery\_d timestamp,

ol\_amount decimal(6,2),

ol\_supply\_w\_id integer,

ol\_quantity integer,

ol\_dist\_info char(24),

PRIMARY KEY (ol\_w\_id, ol\_d\_id, ol\_o\_id, ol\_number)

)tablegroup='tpcc\_group' ;

create table bmsql\_item (

i\_id integer not null,

i\_name varchar(24),

i\_price decimal(5,2),

i\_data varchar(50),

i\_im\_id integer,

PRIMARY KEY (i\_id)

);

create table bmsql\_stock (

s\_w\_id integer not null,

s\_i\_id integer not null,

s\_quantity integer,

s\_ytd integer,

s\_order\_cnt integer,

s\_remote\_cnt integer,

s\_data varchar(50),

s\_dist\_01 char(24),

s\_dist\_02 char(24),

s\_dist\_03 char(24),

s\_dist\_04 char(24),

s\_dist\_05 char(24),

s\_dist\_06 char(24),

s\_dist\_07 char(24),

s\_dist\_08 char(24),

s\_dist\_09 char(24),

s\_dist\_10 char(24),

PRIMARY KEY (s\_w\_id, s\_i\_id)

)tablegroup='tpcc\_group' ;

#### **检查创建的表**

obclient [tpcc]> show tables;

+------------------+

| Tables\_in\_tpcc |

+------------------+

| bmsql\_config |

| bmsql\_customer |

| bmsql\_district |

| bmsql\_history |

| bmsql\_item |

| bmsql\_new\_order |

| bmsql\_oorder |

| bmsql\_order\_line |

| bmsql\_stock |

| bmsql\_warehouse |

+------------------+

#### **创建索引**

[admin@oceanbase-single run]$ ./runSQL.sh my\_ob.ob sql.oceanbase/indexCreates.sql

# ------------------------------------------------------------

# Loading SQL file sql.oceanbase/indexCreates.sql

# ------------------------------------------------------------

create index bmsql\_customer\_idx1

on bmsql\_customer (c\_w\_id, c\_d\_id, c\_last, c\_first) local;

create index bmsql\_oorder\_idx1

on bmsql\_oorder (o\_w\_id, o\_d\_id, o\_carrier\_id, o\_id) local;

#### **加载数据**

[admin@oceanbase-single run]$ ./runLoader.sh my\_ob.ob

Starting BenchmarkSQL LoadData

driver=com.alipay.oceanbase.obproxy.mysql.jdbc.Driver

conn=jdbc:oceanbase://127.1:2883/tpcc?useUnicode=true&characterEncoding=utf-8

user=root@test\_tenant

password=\*\*\*\*\*\*\*\*\*\*\*

warehouses=2

loadWorkers=10

fileLocation (not defined)

csvNullValue (not defined - using default 'NULL')

Worker 000: Loading ITEM

Worker 001: Loading Warehouse 1

Worker 002: Loading Warehouse 2

Worker 000: Loading ITEM done

Worker 002: Loading Warehouse 2 done

Worker 001: Loading Warehouse 1 done

obclient [tpcc]> select count(1) , 'bmsql\_config' from bmsql\_config

 -> union

 -> select count(1) , 'bmsql\_customer' from bmsql\_customer

 -> union

 -> select count(1) , 'bmsql\_district' from bmsql\_district

 -> union

 -> select count(1) , 'bmsql\_history' from bmsql\_history

 -> union

 -> select count(1) , 'bmsql\_item' from bmsql\_item

 -> union

 -> select count(1) , 'bmsql\_new\_order' from bmsql\_new\_order

 -> union

 -> select count(1) , 'bmsql\_oorder' from bmsql\_oorder

 -> union

 -> select count(1) , 'bmsql\_order\_line' from bmsql\_order\_line

 -> union

 -> select count(1) , 'bmsql\_stock' from bmsql\_stock

 -> union

 -> select count(1) , 'bmsql\_warehouse' from bmsql\_warehouse;

+----------+------------------+

| count(1) | bmsql\_config |

+----------+------------------+

| 4 | bmsql\_config |

| 60000 | bmsql\_customer |

| 20 | bmsql\_district |

| 60000 | bmsql\_history |

| 100000 | bmsql\_item |

| 18000 | bmsql\_new\_order |

| 60000 | bmsql\_oorder |

| 599653 | bmsql\_order\_line |

| 200000 | bmsql\_stock |

| 2 | bmsql\_warehouse |

+----------+------------------+

#### **开始压测**

[admin@oceanbase-single run]$ ./runBenchmark.sh my\_ob.ob

21:15:36,140 [main] INFO jTPCC : Term-00,

21:15:36,141 [main] INFO jTPCC : Term-00, +-------------------------------------------------------------+

21:15:36,141 [main] INFO jTPCC : Term-00, BenchmarkSQL v5.0

21:15:36,141 [main] INFO jTPCC : Term-00, +-------------------------------------------------------------+

21:15:36,141 [main] INFO jTPCC : Term-00, (c) 2003, Raul Barbosa

21:15:36,141 [main] INFO jTPCC : Term-00, (c) 2004-2016, Denis Lussier

21:15:36,142 [main] INFO jTPCC : Term-00, (c) 2016, Jan Wieck

21:15:36,142 [main] INFO jTPCC : Term-00, +-------------------------------------------------------------+

21:15:36,142 [main] INFO jTPCC : Term-00,

21:15:36,142 [main] INFO jTPCC : Term-00, db=oracle

21:15:36,142 [main] INFO jTPCC : Term-00, driver=com.alipay.oceanbase.obproxy.mysql.jdbc.Driver

21:15:36,142 [main] INFO jTPCC : Term-00, conn=jdbc:oceanbase://127.1:2883/tpcc?useUnicode=true&characterEncoding=utf-8

21:15:36,143 [main] INFO jTPCC : Term-00, user=root@test\_tenant

21:15:36,143 [main] INFO jTPCC : Term-00,

21:15:36,143 [main] INFO jTPCC : Term-00, warehouses=2

21:15:36,143 [main] INFO jTPCC : Term-00, terminals=10

21:15:36,143 [main] INFO jTPCC : Term-00, runMins=1

21:15:36,143 [main] INFO jTPCC : Term-00, limitTxnsPerMin=0

21:15:36,143 [main] INFO jTPCC : Term-00, terminalWarehouseFixed=true

21:15:36,143 [main] INFO jTPCC : Term-00,

21:15:36,143 [main] INFO jTPCC : Term-00, newOrderWeight=45

21:15:36,143 [main] INFO jTPCC : Term-00, paymentWeight=43

21:15:36,143 [main] INFO jTPCC : Term-00, orderStatusWeight=4

21:15:36,143 [main] INFO jTPCC : Term-00, deliveryWeight=4

21:15:36,143 [main] INFO jTPCC : Term-00, stockLevelWeight=4

21:15:36,143 [main] INFO jTPCC : Term-00,

21:15:36,143 [main] INFO jTPCC : Term-00, resultDirectory=my\_result\_%tY-%tm-%td\_%tH%tM%tS

21:15:36,144 [main] INFO jTPCC : Term-00, osCollectorScript=./misc/os\_collector\_linux.py

21:15:36,144 [main] INFO jTPCC : Term-00,

21:15:36,149 [main] INFO jTPCC : Term-00, copied my\_ob.ob to my\_result\_2024-03-12\_211536/run.properties

21:15:36,149 [main] INFO jTPCC : Term-00, created my\_result\_2024-03-12\_211536/data/runInfo.csv for runID 6

21:15:36,149 [main] INFO jTPCC : Term-00, writing per transaction results to my\_result\_2024-03-12\_211536/data/result.csv

21:15:36,149 [main] INFO jTPCC : Term-00, osCollectorScript=./misc/os\_collector\_linux.py

21:15:36,149 [main] INFO jTPCC : Term-00, osCollectorInterval=1

21:15:36,149 [main] INFO jTPCC : Term-00, osCollectorSSHAddr=null

21:15:36,150 [main] INFO jTPCC : Term-00, osCollectorDevices=null

21:15:36,178 [main] INFO jTPCC : Term-00,

21:15:36,277 [main] INFO jTPCC : Term-00, C value for C\_LAST during load: 44

21:15:36,277 [main] INFO jTPCC : Term-00, C value for C\_LAST this run: 119

21:15:36,277 [main] INFO jTPCC : Term-00,

Term-00, Running Average tpmTOTAL: 26320.68 Current tpmTOTAL: 175320 Memory Usage: 14MB / 212MB

21:16:36,353 [Thread-8] INFO jTPCC : Term-00,

21:16:36,354 [Thread-8] INFO jTPCC : Term-00,

21:16:36,354 [Thread-8] INFO jTPCC : Term-00, Measured tpmC (NewOrders) = 11923.23

21:16:36,354 [Thread-8] INFO jTPCC : Term-00, Measured tpmTOTAL = 26318.27

21:16:36,354 [Thread-8] INFO jTPCC : Term-00, Session Start = 2024-03-12 21:15:36

21:16:36,354 [Thread-8] INFO jTPCC : Term-00, Session End = 2024-03-12 21:16:36

21:16:36,354 [Thread-8] INFO jTPCC : Term-00, Transaction Count = 26330

#### **分析 TPC-C TOP SQL**

# 展示最近3条top sql

obclient [tpcc]> SELECT avg\_exe\_usec,tenant\_id, svr\_ip, svr\_port, sql\_id, plan\_id FROM oceanbase.GV$OB\_PLAN\_CACHE\_PLAN\_STAT ORDER BY avg\_exe\_usec DESC LIMIT 3 ;

+--------------+-----------+-----------+----------+----------------------------------+---------+

| avg\_exe\_usec | tenant\_id | svr\_ip | svr\_port | sql\_id | plan\_id |

+--------------+-----------+-----------+----------+----------------------------------+---------+

| 260435 | 1002 | 127.0.0.1 | 2882 | A54144E8291C19C7DD29E95C68CA0960 | 498 |

| 58634 | 1002 | 127.0.0.1 | 2882 | F59A700FA168324279B0DBC25E19760F | 500 |

| 39079 | 1002 | 127.0.0.1 | 2882 | 6A94DDD2CA964DE15108D7C710A6931C | 440 |

+--------------+-----------+-----------+----------+----------------------------------+---------+

#### **第1条语句的执行计划**

# 查看第1条sql的内容

obclient [tpcc]> select query\_sql from oceanbase.GV$OB\_PLAN\_CACHE\_PLAN\_STAT where sql\_id='FC3FED8CCB2946DE54F1C5BA3656023C'\G

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 1. row \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

query\_sql: SELECT d\_tax, d\_next\_o\_id FROM bmsql\_district WHERE d\_w\_id = 2 AND d\_id = 9 FOR UPDATE

# 查看第1条sql解析执行计划

obclient [tpcc]> explain SELECT d\_tax, d\_next\_o\_id FROM bmsql\_district WHERE d\_w\_id = 2 AND d\_id = 8 FOR UPDATE;

+--------------------------------------------------------------------------------------------------------------------------------------------+

| Query Plan |

+--------------------------------------------------------------------------------------------------------------------------------------------+

| ======================================================= |

| |ID|OPERATOR |NAME |EST.ROWS|EST.TIME(us)| |

| ------------------------------------------------------- |

| |0 |MATERIAL | |1 |3 | |

| |1 |└─FOR UPDATE | |1 |3 | |

| |2 | └─TABLE GET|bmsql\_district|1 |3 | |

| ======================================================= |

| Outputs & filters: |

| ------------------------------------- |

| 0 - output([bmsql\_district.d\_tax], [bmsql\_district.d\_next\_o\_id]), filter(nil), rowset=16 |

| 1 - output([bmsql\_district.d\_tax], [bmsql\_district.d\_next\_o\_id]), filter(nil), rowset=16 |

| lock tables(bmsql\_district) |

| 2 - output([bmsql\_district.d\_w\_id], [bmsql\_district.d\_id], [bmsql\_district.d\_tax], [bmsql\_district.d\_next\_o\_id]), filter(nil), rowset=16 |

| access([bmsql\_district.d\_w\_id], [bmsql\_district.d\_id], [bmsql\_district.d\_tax], [bmsql\_district.d\_next\_o\_id]), partitions(p0) |

| is\_index\_back=false, is\_global\_index=false, |

| range\_key([bmsql\_district.d\_w\_id], [bmsql\_district.d\_id]), range[2,8 ; 2,8], |

| range\_cond([bmsql\_district.d\_w\_id = 2], [bmsql\_district.d\_id = 8]) |

+--------------------------------------------------------------------------------------------------------------------------------------------+

# 第1条语句的实际执行计划

obclient [tpcc]> SELECT SVR\_IP, plan\_depth, plan\_line\_id,operator,name,rows,cost,property from oceanbase.`GV$OB\_PLAN\_CACHE\_PLAN\_EXPLAIN` where tenant\_id=1002 AND SVR\_IP = '127.0.0.1' AND SVR\_PORT=2882 AND plan\_id=498;

+-----------+------------+--------------+------------------------+-------------------------------------+--------+-------+-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------+

| SVR\_IP | plan\_depth | plan\_line\_id | operator | name | rows | cost | property |

+-----------+------------+--------------+------------------------+-------------------------------------+--------+-------+-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------+

| 127.0.0.1 | 0 | 0 | PHY\_HASH\_DISTINCT | NULL | 10 | 31761 | NULL |

| 127.0.0.1 | 1 | 1 | PHY\_MERGE\_UNION | NULL | 10 | 31758 | NULL |

| 127.0.0.1 | 2 | 2 | PHY\_SCALAR\_AGGREGATE | NULL | 1 | 3 | NULL |

| 127.0.0.1 | 3 | 3 | PHY\_TABLE\_SCAN | bmsql\_config | 4 | 3 | table\_rows:4, physical\_range\_rows:4, logical\_range\_rows:4, index\_back\_rows:0, output\_rows:4, avaiable\_index\_name[bmsql\_config], estimation info[table\_id:500003, (table\_type:12, version:-1--1--1, logical\_rc:4, physical\_rc:4)] |

| 127.0.0.1 | 2 | 4 | PHY\_SCALAR\_AGGREGATE | NULL | 1 | 1698 | NULL |

| 127.0.0.1 | 3 | 5 | PHY\_TABLE\_SCAN | bmsql\_customer(bmsql\_customer\_idx1) | 58563 | 637 | table\_rows:60001, physical\_range\_rows:58563, logical\_range\_rows:58563, index\_back\_rows:0, output\_rows:58563, avaiable\_index\_name[bmsql\_customer\_idx1,bmsql\_customer], estimation info[table\_id:500015, (table\_type:11, version:-1--1--1, logical\_rc:36000, physical\_rc:36000), (table\_type:12, version:-1--1--1, logical\_rc:15000, physical\_rc:15000), (table\_type:0, version:-1--1--1, logical\_rc:7563, physical\_rc:7563)] |

| 127.0.0.1 | 2 | 6 | PHY\_SCALAR\_AGGREGATE | NULL | 1 | 4 | NULL |

| 127.0.0.1 | 3 | 7 | PHY\_TABLE\_SCAN | bmsql\_district | 20 | 3 | table\_rows:20, physical\_range\_rows:20, logical\_range\_rows:20, index\_back\_rows:0, output\_rows:20, avaiable\_index\_name[bmsql\_district], estimation info[table\_id:500006, (table\_type:11, version:-1--1--1, logical\_rc:12, physical\_rc:12), (table\_type:12, version:-1--1--1, logical\_rc:6, physical\_rc:6), (table\_type:0, version:-1--1--1, logical\_rc:2, physical\_rc:2)] |

| 127.0.0.1 | 2 | 8 | PHY\_SCALAR\_AGGREGATE | NULL | 1 | 1740 | NULL |

| 127.0.0.1 | 3 | 9 | PHY\_TABLE\_SCAN | bmsql\_history | 60009 | 652 | table\_rows:60009, physical\_range\_rows:60009, logical\_range\_rows:60009, index\_back\_rows:0, output\_rows:60009, avaiable\_index\_name[bmsql\_history], estimation info[table\_id:500009, (table\_type:11, version:-1--1--1, logical\_rc:36000, physical\_rc:36000), (table\_type:12, version:-1--1--1, logical\_rc:15000, physical\_rc:15000), (table\_type:0, version:-1--1--1, logical\_rc:9009, physical\_rc:9009)] |

| 127.0.0.1 | 2 | 10 | PHY\_SCALAR\_AGGREGATE | NULL | 1 | 2897 | NULL |

| 127.0.0.1 | 3 | 11 | PHY\_TABLE\_SCAN | bmsql\_item | 100000 | 1085 | table\_rows:100000, physical\_range\_rows:100000, logical\_range\_rows:100000, index\_back\_rows:0, output\_rows:100000, avaiable\_index\_name[bmsql\_item], estimation info[table\_id:500013, (table\_type:12, version:-1--1--1, logical\_rc:64431, physical\_rc:64431), (table\_type:12, version:-1--1--1, logical\_rc:35569, physical\_rc:35569)] |

| 127.0.0.1 | 2 | 12 | PHY\_SCALAR\_AGGREGATE | NULL | 1 | 524 | NULL |

| 127.0.0.1 | 3 | 13 | PHY\_TABLE\_SCAN | bmsql\_new\_order | 18000 | 198 | table\_rows:18000, physical\_range\_rows:18000, logical\_range\_rows:18000, index\_back\_rows:0, output\_rows:18000, avaiable\_index\_name[bmsql\_new\_order], estimation info[table\_id:500010, (table\_type:11, version:-1--1--1, logical\_rc:14400, physical\_rc:14400), (table\_type:0, version:-1--1--1, logical\_rc:3600, physical\_rc:3600)] |

| 127.0.0.1 | 2 | 14 | PHY\_SCALAR\_AGGREGATE | NULL | 1 | 1735 | NULL |

| 127.0.0.1 | 3 | 15 | PHY\_TABLE\_SCAN | bmsql\_oorder(bmsql\_oorder\_idx1) | 59850 | 650 | table\_rows:60002, physical\_range\_rows:59850, logical\_range\_rows:59850, index\_back\_rows:0, output\_rows:59850, avaiable\_index\_name[bmsql\_oorder\_idx1,bmsql\_oorder], estimation info[table\_id:500016, (table\_type:11, version:-1--1--1, logical\_rc:36000, physical\_rc:36000), (table\_type:12, version:-1--1--1, logical\_rc:15000, physical\_rc:15000), (table\_type:0, version:-1--1--1, logical\_rc:8850, physical\_rc:8850)] |

| 127.0.0.1 | 2 | 16 | PHY\_SCALAR\_AGGREGATE | NULL | 1 | 17358 | NULL |

| 127.0.0.1 | 3 | 17 | PHY\_TABLE\_SCAN | bmsql\_order\_line | 599729 | 6489 | table\_rows:599729, physical\_range\_rows:599729, logical\_range\_rows:599729, index\_back\_rows:0, output\_rows:599729, avaiable\_index\_name[bmsql\_order\_line], estimation info[table\_id:500012, (table\_type:11, version:-1--1--1, logical\_rc:325369, physical\_rc:325369), (table\_type:12, version:-1--1--1, logical\_rc:157217, physical\_rc:157217), (table\_type:0, version:-1--1--1, logical\_rc:117143, physical\_rc:117143)] |

| 127.0.0.1 | 2 | 18 | PHY\_SCALAR\_AGGREGATE | NULL | 1 | 5791 | NULL |

| 127.0.0.1 | 3 | 19 | PHY\_TABLE\_SCAN | bmsql\_stock | 200000 | 2166 | table\_rows:200000, physical\_range\_rows:200000, logical\_range\_rows:200000, index\_back\_rows:0, output\_rows:200000, avaiable\_index\_name[bmsql\_stock], estimation info[table\_id:500014, (table\_type:12, version:-1--1--1, logical\_rc:106690, physical\_rc:106690), (table\_type:12, version:-1--1--1, logical\_rc:93310, physical\_rc:93310)] |

| 127.0.0.1 | 2 | 20 | PHY\_SCALAR\_AGGREGATE | NULL | 1 | 3 | NULL |

| 127.0.0.1 | 3 | 21 | PHY\_TABLE\_SCAN | bmsql\_warehouse | 2 | 3 | table\_rows:2, physical\_range\_rows:2, logical\_range\_rows:2, index\_back\_rows:0, output\_rows:2, avaiable\_index\_name[bmsql\_warehouse], estimation info[table\_id:500005, (table\_type:12, version:-1--1--1, logical\_rc:2, physical\_rc:2)] |

+-----------+------------+--------------+------------------------+-------------------------------------+--------+-------+-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------+

#### **第2条语句的执行计划**

obclient [tpcc]> select query\_sql from oceanbase.GV$OB\_PLAN\_CACHE\_PLAN\_STAT where sql\_id='F59A700FA168324279B0DBC25E19760F'\G

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 1. row \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

query\_sql: SELECT count(\*) AS low\_stock FROM ( SELECT s\_w\_id, s\_i\_id, s\_quantity FROM bmsql\_stock WHERE s\_w\_id = 1 AND s\_quantity < 13 AND s\_i\_id IN ( SELECT ol\_i\_id FROM bmsql\_district JOIN bmsql\_order\_line ON ol\_w\_id = d\_w\_id AND ol\_d\_id = d\_id AND ol\_o\_id >= d\_next\_o\_id - 20 AND ol\_o\_id < d\_next\_o\_id WHERE d\_w\_id = 1 AND d\_id = 4 ) )

obclient [tpcc]> explain SELECT count(\*) AS low\_stock FROM ( SELECT s\_w\_id, s\_i\_id, s\_quantity FROM bmsql\_stock

 -> WHERE s\_w\_id = 1 AND s\_quantity < 14 AND s\_i\_id IN (

 -> SELECT ol\_i\_id FROM bmsql\_district JOIN bmsql\_order\_line ON ol\_w\_id = d\_w\_id

 -> AND ol\_d\_id = d\_id AND ol\_o\_id >= d\_next\_o\_id - 20 AND ol\_o\_id < d\_next\_o\_id

 -> WHERE d\_w\_id = 1 AND d\_id = 5 ) );

+----------------------------------------------------------------------------------------------------------------------------------------------------------+

| Query Plan |

+----------------------------------------------------------------------------------------------------------------------------------------------------------+

| ================================================================================ |

| |ID|OPERATOR |NAME |EST.ROWS|EST.TIME(us)| |

| -------------------------------------------------------------------------------- |

| |0 |SCALAR GROUP BY | |1 |3600 | |

| |1 |└─HASH SEMI JOIN | |1206 |3578 | |

| |2 | ├─TABLE RANGE SCAN |bmsql\_stock |1232 |2830 | |

| |3 | └─SUBPLAN SCAN |VIEW1 |4675 |36 | |

| |4 | └─NESTED-LOOP JOIN | |4675 |24 | |

| |5 | ├─TABLE GET |bmsql\_district |1 |3 | |

| |6 | └─DISTRIBUTED TABLE RANGE SCAN|bmsql\_order\_line|1 |21 | |

| ================================================================================ |

| Outputs & filters: |

| ------------------------------------- |

| 0 - output([T\_FUN\_COUNT(\*)]), filter(nil), rowset=256 |

| group(nil), agg\_func([T\_FUN\_COUNT(\*)]) |

| 1 - output(nil), filter(nil), rowset=256 |

| equal\_conds([bmsql\_stock.s\_i\_id = VIEW1.ol\_i\_id]), other\_conds(nil) |

| 2 - output([bmsql\_stock.s\_i\_id]), filter([bmsql\_stock.s\_quantity < 14]), rowset=256 |

| access([bmsql\_stock.s\_i\_id], [bmsql\_stock.s\_quantity]), partitions(p0) |

| is\_index\_back=false, is\_global\_index=false, filter\_before\_indexback[false], |

| range\_key([bmsql\_stock.s\_w\_id], [bmsql\_stock.s\_i\_id]), range(1,MIN ; 1,MAX), |

| range\_cond([bmsql\_stock.s\_w\_id = 1]) |

| 3 - output([VIEW1.ol\_i\_id]), filter(nil), rowset=256 |

| access([VIEW1.ol\_i\_id]) |

| 4 - output([bmsql\_order\_line.ol\_i\_id]), filter(nil), rowset=256 |

| conds(nil), nl\_params\_([bmsql\_district.d\_next\_o\_id(:0)]), use\_batch=false |

| 5 - output([bmsql\_district.d\_next\_o\_id]), filter([bmsql\_district.d\_next\_o\_id > bmsql\_district.d\_next\_o\_id - 20]), rowset=256 |

| access([bmsql\_district.d\_next\_o\_id]), partitions(p0) |

| is\_index\_back=false, is\_global\_index=false, filter\_before\_indexback[false], |

| range\_key([bmsql\_district.d\_w\_id], [bmsql\_district.d\_id]), range[1,5 ; 1,5], |

| range\_cond([bmsql\_district.d\_w\_id = 1], [bmsql\_district.d\_id = 5]) |

| 6 - output([bmsql\_order\_line.ol\_i\_id]), filter(nil), rowset=256 |

| access([bmsql\_order\_line.ol\_i\_id]), partitions(p0) |

| is\_index\_back=false, is\_global\_index=false, |

| range\_key([bmsql\_order\_line.ol\_w\_id], [bmsql\_order\_line.ol\_d\_id], [bmsql\_order\_line.ol\_o\_id], [bmsql\_order\_line.ol\_number]), range(MIN ; MAX), |

| range\_cond([bmsql\_order\_line.ol\_w\_id = 1], [bmsql\_order\_line.ol\_d\_id = 5], [bmsql\_order\_line.ol\_o\_id >= :0 - 20], [bmsql\_order\_line.ol\_o\_id < :0]) |

+----------------------------------------------------------------------------------------------------------------------------------------------------------+

# 第2条语句的实际执行计划

obclient [tpcc]> SELECT SVR\_IP, plan\_depth, plan\_line\_id,operator,name,rows,cost,property from oceanbase.`GV$OB\_PLAN\_CACHE\_PLAN\_EXPLAIN` where tenant\_id=1002 AND SVR\_IP = '127.0.0.1' AND SVR\_PORT=2882 AND plan\_id=500;

+-----------+------------+--------------+-------------------------+------------------+------+------+-----------------------------------------------------------------------------------------------------------------------------------------------+

| SVR\_IP | plan\_depth | plan\_line\_id | operator | name | rows | cost | property |

+-----------+------------+--------------+-------------------------+------------------+------+------+-----------------------------------------------------------------------------------------------------------------------------------------------+

| 127.0.0.1 | 0 | 0 | PHY\_SCALAR\_AGGREGATE | NULL | 1 | 7414 | NULL |

| 127.0.0.1 | 1 | 1 | PHY\_HASH\_JOIN | NULL | 2663 | 7366 | NULL |

| 127.0.0.1 | 2 | 2 | PHY\_SUBPLAN\_SCAN | NULL | 2988 | 31 | NULL |

| 127.0.0.1 | 3 | 3 | PHY\_NESTED\_LOOP\_JOIN | NULL | 2988 | 23 | NULL |

| 127.0.0.1 | 4 | 4 | PHY\_TABLE\_SCAN | bmsql\_district | 1 | 2 | table\_rows:20, physical\_range\_rows:1, logical\_range\_rows:1, index\_back\_rows:0, output\_rows:1, avaiable\_index\_name[bmsql\_district] |

| 127.0.0.1 | 4 | 5 | PHY\_TABLE\_SCAN | bmsql\_order\_line | 1 | 20 | table\_rows:361152, physical\_range\_rows:1, logical\_range\_rows:1, index\_back\_rows:0, output\_rows:1, avaiable\_index\_name[bmsql\_order\_line] |

| 127.0.0.1 | 2 | 6 | PHY\_TABLE\_SCAN | bmsql\_stock | 2706 | 6528 | table\_rows:124984, physical\_range\_rows:75084, logical\_range\_rows:75084, index\_back\_rows:0, output\_rows:2706, avaiable\_index\_name[bmsql\_stock] |

+-----------+------------+--------------+-------------------------+------------------+------+------+-----------------------------------------------------------------------------------------------------------------------------------------------+

#### **第3条语句的执行计划**

obclient [tpcc]> select query\_sql from oceanbase.GV$OB\_PLAN\_CACHE\_PLAN\_STAT where sql\_id='F95CD7A05C7064A028D76E0088B147F7'\G

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 1. row \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

query\_sql: DELETE FROM bmsql\_new\_order WHERE no\_w\_id = 2 AND no\_d\_id = 1 AND no\_o\_id = 2101

obclient [tpcc]> explain DELETE FROM bmsql\_new\_order WHERE no\_w\_id = 2 AND no\_d\_id = 1 AND no\_o\_id = 2313;

+--------------------------------------------------------------------------------------------------------------------------------------------+

| Query Plan |

+--------------------------------------------------------------------------------------------------------------------------------------------+

| ====================================================== |

| |ID|OPERATOR |NAME |EST.ROWS|EST.TIME(us)| |

| ------------------------------------------------------ |

| |0 |DELETE | |1 |15 | |

| |1 |└─TABLE GET|bmsql\_new\_order|1 |3 | |

| ====================================================== |

| Outputs & filters: |

| ------------------------------------- |

| 0 - output(nil), filter(nil) |

| table\_columns([{bmsql\_new\_order: ({bmsql\_new\_order: (bmsql\_new\_order.no\_w\_id, bmsql\_new\_order.no\_d\_id, bmsql\_new\_order.no\_o\_id)})}]) |

| 1 - output([bmsql\_new\_order.no\_w\_id], [bmsql\_new\_order.no\_d\_id], [bmsql\_new\_order.no\_o\_id]), filter(nil), rowset=16 |

| access([bmsql\_new\_order.no\_w\_id], [bmsql\_new\_order.no\_d\_id], [bmsql\_new\_order.no\_o\_id]), partitions(p0) |

| is\_index\_back=false, is\_global\_index=false, |

| range\_key([bmsql\_new\_order.no\_w\_id], [bmsql\_new\_order.no\_d\_id], [bmsql\_new\_order.no\_o\_id]), range[2,1,2313 ; 2,1,2313], |

| range\_cond([bmsql\_new\_order.no\_w\_id = 2], [bmsql\_new\_order.no\_d\_id = 1], [bmsql\_new\_order.no\_o\_id = 2313]) |

+--------------------------------------------------------------------------------------------------------------------------------------------+

15 rows in set (0.003 sec)

# 实际执行计划

obclient [tpcc]> SELECT SVR\_IP, plan\_depth, plan\_line\_id,operator,name,rows,cost,property

 -> from oceanbase.`GV$OB\_PLAN\_CACHE\_PLAN\_EXPLAIN`

 -> where tenant\_id=1002 AND SVR\_IP = '127.0.0.1' AND SVR\_PORT=2882 AND plan\_id=540;

+-----------+------------+--------------+------------------------+------+------+------+----------+

| SVR\_IP | plan\_depth | plan\_line\_id | operator | name | rows | cost | property |

+-----------+------------+--------------+------------------------+------+------+------+----------+

| 127.0.0.1 | 0 | 0 | PHY\_SCALAR\_AGGREGATE | NULL | 1 | 3 | NULL |

| 127.0.0.1 | 1 | 1 | PHY\_BLOCK\_SAMPLE\_SCAN | NULL | 1 | 3 | NULL |

+-----------+------------+--------------+------------------------+------+------+------+----------+