Relational Database Service

Performance White paper

Issue 01

Date 2022-09-30

HUAWEI TECHNOLOGIES CO., LTD.

Trademarks and Permissions

HUAWEI and other Huawei trademarks are trademarks of Huawei Technologies Co., Ltd.
All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

The purchased products, services and features are stipulated by the contract made between Huawei and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Huawei Technologies Co., Ltd.

Address: Huawei Industrial Base
Bantian, Longgang
Shenzhen 518129
People's Republic of China

Website: https://e.huawei.com

Contents

1 RDS for MySQL

1.1 Test Method

MySQL is one of the world's most popular open-source relational databases. It works with the Linux, Apache, and PHP (LAMP) stack to provide efficient web solutions. It solves problems such as poor database performance, long data replication delay, and long fault recovery time in high concurrency scenarios.

RDS for MySQL is ready for immediate use, and provides backup and restoration, data migration, security protection, high availability, and elastic scalability. You can obtain a production database with high performance and scalability in a few minutes after simple configurations while your data integrity and service continuity are guaranteed.

Test Environment

- Elastic Cloud Server (ECS): general computing | c3.2xlarge.2 | 8 vCPUs | 16 GB, CentOS7.4 64 bit image. Bind an elastic IP (EIP) to the ECS because additional compilation tools need to be installed on stress testing tools.

  NOTE

  RDS for MySQL 8.0 test environment is as follows:

  - ECS: general computing-plus | c6.4xlarge.2 | 16 vCPUs | 32 GB, CentOS 7.6 (64 bit). Bind an EIP to the ECS because additional compilation tools need to be installed on stress testing tools.

Test Tool

Sysbench is a multi-threaded benchmark tool based on LuaJIT, allowing you to quickly get an impression of system performance by using a built-in database test model. For details, visit https://github.com/akopytov/sysbench.

Sysbench 1.0.18 is used in this test. Run the following commands to install it:

```
# wget -c https://github.com/akopytov/sysbench/archive/1.0.18.zip
```

```
# yum install autoconf libtool mysql mysql-devel vim unzip
```

```
# unzip 1.0.18.zip
# cd sysbench-1.0.18
# ./autogen.sh
# ./configure
# make
# make install
```

Test Procedure

Replace the database name, connection IP address, and user password based on the site requirements.

Step 1 Import data.

1. Run the following command to log in to a database and create the test database **loadtest**:

   ```
   mysql -u root -P 3306 -h <host> -p -e "create database loadtest"
   ```

2. Run the following command to import the test background data to the **loadtest** database:

   ```
   sysbench --test=/usr/local/share/sysbench/tests/include/oltp_legacy/oltp.lua --db-driver=mysql --mysql-db=loadtest --mysql-user=root --mysql-password=<password> --mysql-port=3306 --mysql-host=<host> --oltp-tables-count=64 --oltp-table-size=10000000 --num-threads=20 prepare
   ```

Step 2 Run the following command to perform a stress testing:

```
sysbench --test=/usr/local/share/sysbench/tests/include/oltp_legacy/oltp.lua --db-driver=mysql --mysql-db=loadtest --mysql-user=root --mysql-password=<password> --mysql-port=3306 --mysql-host=<host> --oltp-tables-count=64 --oltp-table-size=10000000 --max-time=3600 --max-requests=0 --num-threads=200 --report-interval=3 --forced-shutdown=1 run
```

Step 3 Run the following command to delete the test data:

```
sysbench --test=/usr/local/share/sysbench/tests/include/oltp_legacy/oltp.lua --db-driver=mysql --mysql-db=loadtest --mysql-user=root --mysql-password=<password> --mysql-port=3306 --mysql-host=<host> --oltp-tables-count=64 --oltp-table-size=10000000 --max-time=3600 --max-requests=0 --num-threads=200 cleanup
```

----End

Testing Model

1. Table structure:

   ```
   CREATE TABLE `sbtest` (
   `id` INTEGER UNSIGNED NOT NULL AUTO_INCREMENT,
   `k` INTEGER UNSIGNED DEFAULT '0' NOT NULL,
   `c` CHAR(120) DEFAULT '' NOT NULL,
   `pad` CHAR(60) DEFAULT '' NOT NULL,
   PRIMARY KEY (`id`)
   ```

) ENGINE=InnoDB

2. Read/write ratio:

   The default transaction submitted by sysbench contains 18 SQL statements. The details are as follows:

   - Ten primary key select statements:

     SELECT c FROM ${rand_table_name} where id=${rand_id};

   - Four range select statements:

     SELECT c FROM ${rand_table_name} WHERE id BETWEEN ${rand_id_start} AND ${rand_id_end};

     SELECT SUM(K) FROM ${rand_table_name} WHERE id BETWEEN ${rand_id_start} AND ${rand_id_end};

     SELECT c FROM ${rand_table_name} WHERE id BETWEEN ${rand_id_start} AND ${rand_id_end} ORDER BY c;

     SELECT DISTINCT c FROM ${rand_table_name} WHERE id BETWEEN ${rand_id_start} AND ${rand_id_end} ORDER BY c;

   - Two update statements:

     UPDATE ${rand_table_name} SET k=k+1 WHERE id=${rand_id}

     UPDATE ${rand_table_name} SET c=${rand_str} WHERE id=${rand_id}

   - One delete statement:

     DELETE FROM ${rand_table_name} WHERE id=${rand_id}

   - One insert statement:

     INSERT INTO ${rand_table_name} (id, k, c, pad) VALUES (${rand_id},${rand_k},${rand_str_c},${rand_str_pad})

Test Metrics

- Transaction per second (TPS) refers to the number of transactions executed per second by a database. Each transaction contains 18 SQL statements.
- Query per second (QPS) refers to the number of SQL statements, including INSERT, SELECT, UPDATE, and DELETE statements, executed per second.

1.2 RDS for MySQL 5.7 Test Data

1.2.1 General-Purpose DB Instances

About IOPS

The input/output operations per second (IOPS) supported by RDS for MySQL depends on the I/O performance of Elastic Volume Service (EVS) disks. For details, see **Disk Types and Performance** in the *Elastic Volume Service Service Overview*.

Test Data

NOTICE

The **Maximum Connections (Stress Testing)** columns in the following tables indicate the results of the RDS performance stress testing. For running services on the live network, set the parameter **max_connections**.

Table 1-1 vCPU:Memory = 1:2

vCPUs	Memory (GB)	Maximum Connections (Stress Testing)	TPS	QPS	IOPS
1	2	800	185	3,707	See **About IOPS**.
2	4	1,500	334	6,673	
4	8	2,500	756	15,122	
8	16	5,000	1,338	26,756	

Table 1-2 vCPU:Memory = 1:4

vCPUs	Memory (GB)	Maximum Connections (Stress Testing)	TPS	QPS	IOPS
2	8	2,500	552	11,039	See **About IOPS**.
4	16	5,000	1,062	21,249	
8	32	10,000	2,117	42,335	

Test Results

Figure 1-1 vCPU:Memory = 1:2

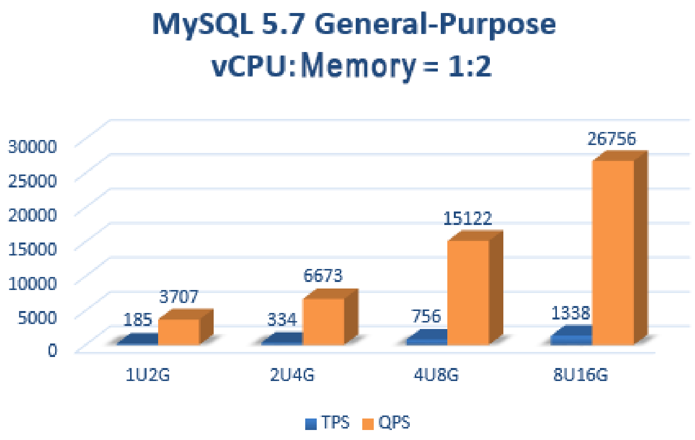
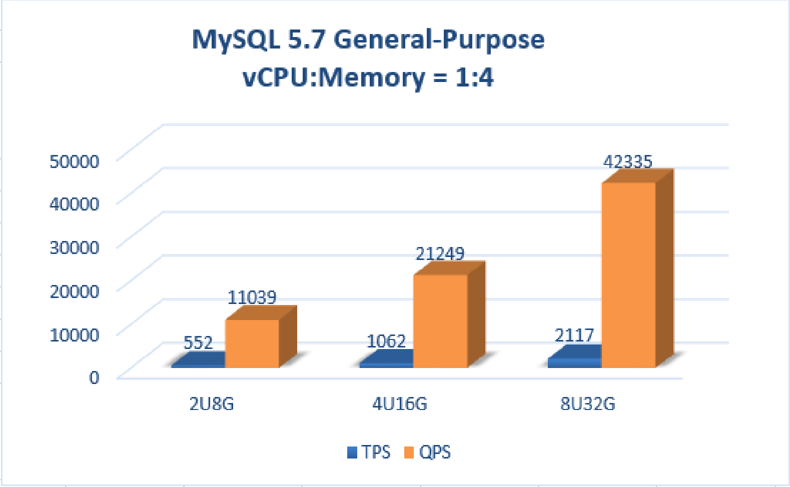


Figure 1-2 vCPU:Memory = 1:4



1.2.2 Dedicated DB Instances

About IOPS

The input/output operations per second (IOPS) supported by RDS for MySQL depends on the I/O performance of Elastic Volume Service (EVS) disks. For details, see **Disk Types and Performance** in the *Elastic Volume Service Service Overview*.

Test Data

NOTICE

The **Maximum Connections (Stress Testing)** columns in the following tables indicate the results of the RDS performance stress testing. For running services on the live network, set the parameter **max_connections**.

Table 1-3 vCPU:Memory = 1:4

vCPUs	Memory (GB)	Maximum Connections (Stress Testing)	TPS	QPS	IOPS
2	8	2,500	621	12,394	See **About IOPS**.
4	16	5,000	1,230	24,608	
8	32	10,000	2,514	50,290	
16	64	18,000	3,017	60,337	
32	128	30,000	4,368	87,354	

vCPUs	Memory (GB)	Maximum Connections (Stress Testing)	TPS	QPS	IOPS
64	256	60,000	4,536	90,729	

Table 1-4 vCPU:Memory = 1:8

vCPUs	Memory (GB)	Maximum Connections (Stress Testing)	TPS	QPS	IOPS
4	32	10,000	1,488	29,765	See **About IOPS**.
8	64	18,000	2,811	56,216	
16	128	30,000	4,095	81,910	
64	512	100,000	4,626	96,824	

Test Results

Figure 1-3 vCPU:Memory = 1:4

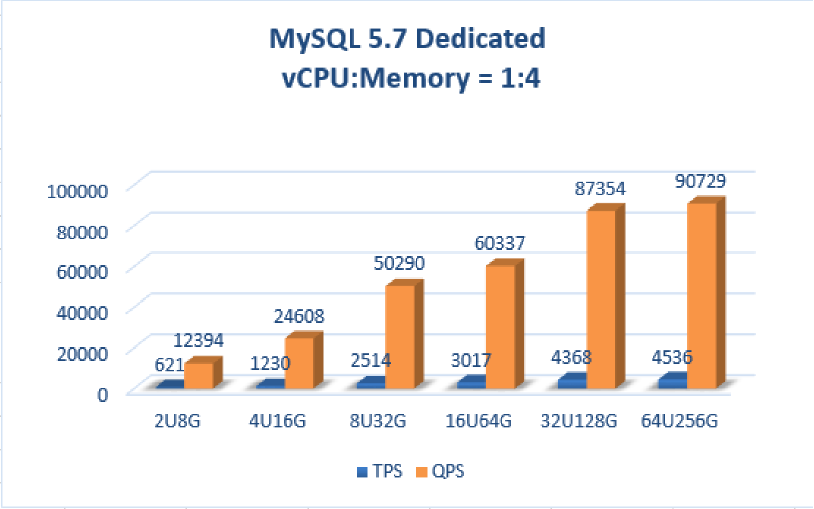
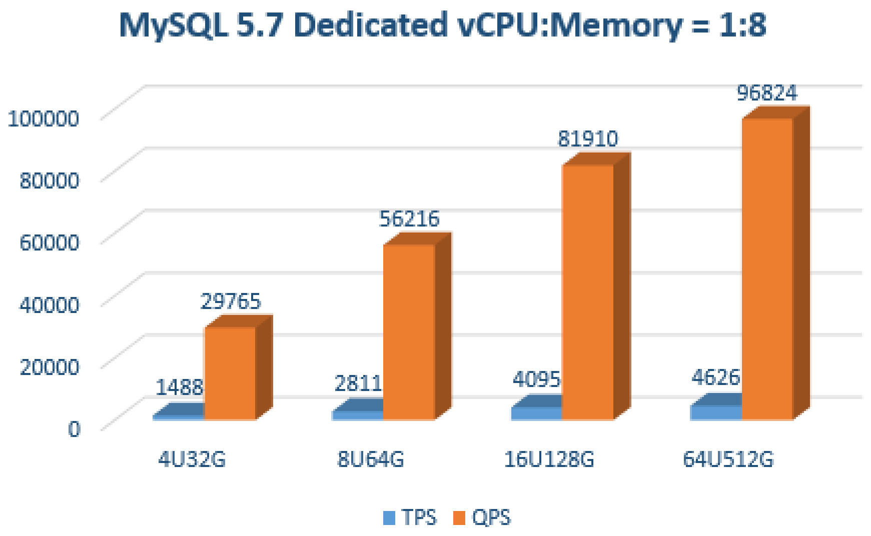


Figure 1-4 vCPU:Memory = 1:8



1.3 RDS for MySQL 8.0 Test Data

1.3.1 General-Purpose DB Instances

About IOPS

The input/output operations per second (IOPS) supported by RDS for MySQL depends on the I/O performance of Elastic Volume Service (EVS) disks. For details, see **Disk Types and Performance** in the *Elastic Volume Service Service Overview*.

NOTICE

The **Maximum Connections (Stress Testing)** columns in the following tables indicate the results of the RDS performance stress testing. For running services on the live network, set the parameter **max_connections**.

Test Data

Table 1-5 vCPU:Memory = 1:2

vCPUs	Memory (GB)	Maximum Connections (Stress Testing)	TPS	QPS	IOPS
2	4	1,500	395	7,914	See **About IOPS**.
4	8	2,500	1,013	20,276	

vCPUs	Memory (GB)	Maximum Connections (Stress Testing)	TPS	QPS	IOPS
8	16	5,000	1,591	31,829	

Table 1-6 vCPU:Memory = 1:4

vCPUs	Memory (GB)	Maximum Connections (Stress Testing)	TPS	QPS	IOPS
2	8	2,500	571	11,437	See **About IOPS**.
4	16	5,000	1,349	26,996	
8	32	10,000	2,308	46,176	

Test Results

Figure 1-5 vCPU:Memory = 1:2

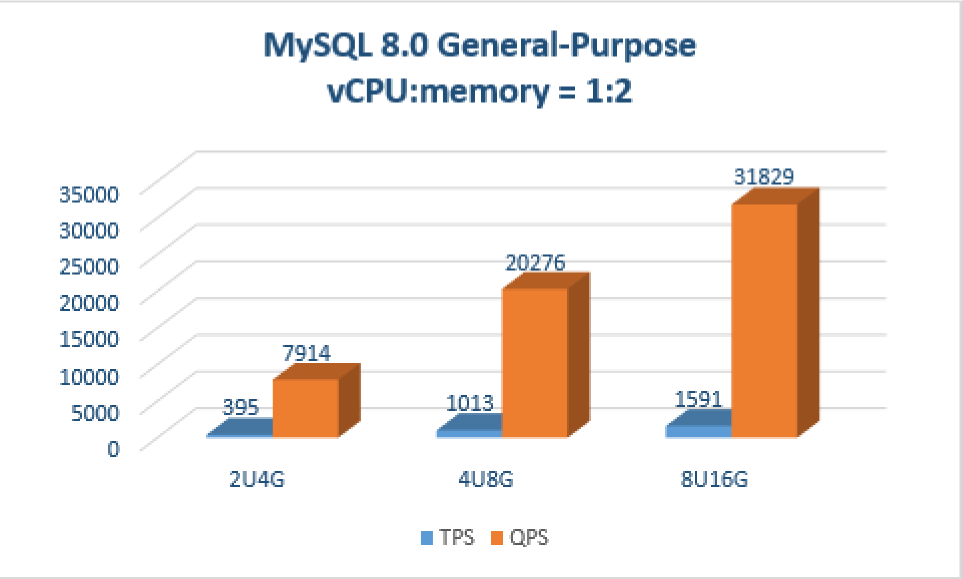
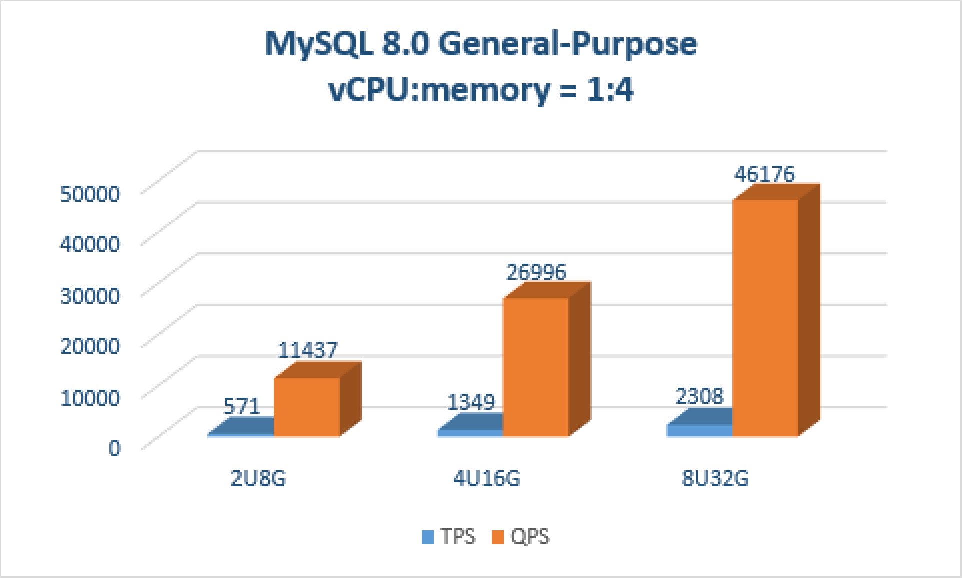


Figure 1-6 vCPU:Memory = 1:4



1.3.2 Dedicated DB Instances

About IOPS

The input/output operations per second (IOPS) supported by RDS for MySQL depends on the I/O performance of Elastic Volume Service (EVS) disks. For details, see **Disk Types and Performance** in the *Elastic Volume Service Service Overview*.

Test Data

NOTICE

The **Maximum Connections (Stress Testing)** columns in the following tables indicate the results of the RDS performance stress testing. For running services on the live network, set the parameter **max_connections**.

Table 1-7 vCPU:Memory = 1:4

vCPUs	Memory (GB)	Maximum Connections (Stress Testing)	TPS	QPS	IOPS
2	8	2,500	590	11,804	See **About IOPS**.
4	16	5,000	1,357	27,159	
8	32	10,000	2,364	47,302	
16	64	18,000	2,876	57,531	

vCPUs	Memory (GB)	Maximum Connections (Stress Testing)	TPS	QPS	IOPS
32	128	30,000	4,328	86,584	
64	256	60,000	4,646	90,754	

Table 1-8 vCPU:Memory = 1:8

vCPUs	Memory (GB)	Maximum Connections (Stress Testing)	TPS	QPS	IOPS
4	32	10,000	1,435	28,701	See **About IOPS**.
8	64	18,000	2,503	50,061	
16	128	30,000	4,060	81,210	
64	512	100,000	4,710	99,932	

Test Results

Figure 1-7 vCPU:Memory = 1:4

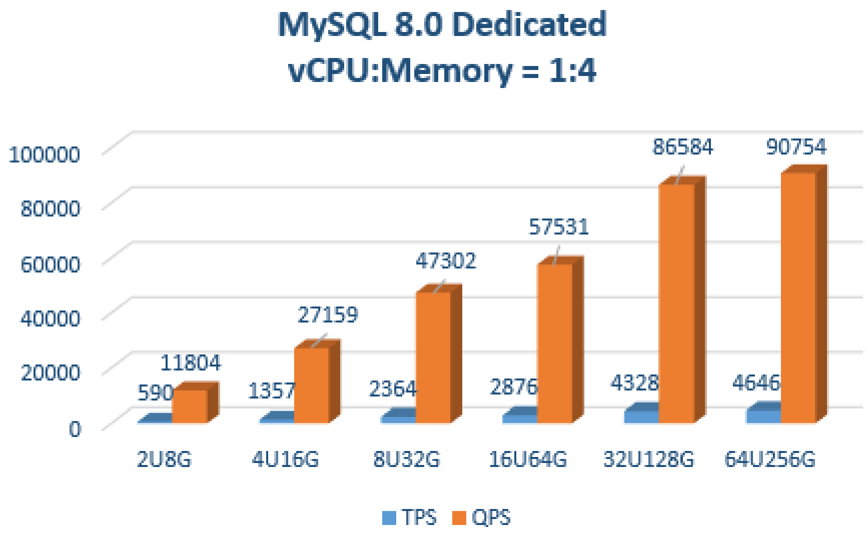
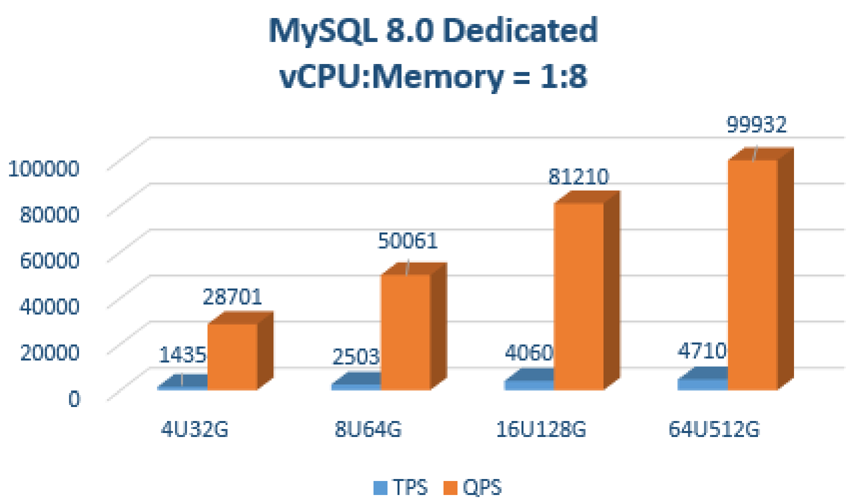


Figure 1-8 vCPU:Memory = 1:8



2 RDS for PostgreSQL

2.1 Test Method

PostgreSQL is an open-source object-relational database management system with an emphasis on extensibility and standards compliance. It is known as the most advanced open-source database. It excels in processing complex online transaction processing (OLTP) transactions and supports NoSQL (JSON, XML, or hstore) and geographic information system (GIS) data types. It has earned a reputation for reliability and data integrity, and is widely used for websites, location-based applications, and complex data object processing.

- RDS for PostgreSQL supports the postgis plugin and provides excellent spatial performance.
- RDS for PostgreSQL is suitable for various scenarios and is cost-effective. You can flexibly scale resources based on your service requirements and pay for only what you use.

Test Environment

- ECS: general computing | c3.2xlarge.2 | 8 vCPUs | 16 GB, CentOS7.4 64 bit image. Bind an EIP to the ECS because additional compilation tools need to be installed on stress testing tools.

  NOTE

  The test environment for RDS for PostgreSQL 12 is as follows:

  - ECS: general computing-plus | c6.4xlarge.2 | 16 vCPUs | 32 GB, CentOS 7.6 (64 bit). Bind an EIP to the ECS because additional compilation tools need to be installed on stress testing tools.

Test Tool

Sysbench is a multi-threaded benchmark tool based on LuaJIT. It is most frequently used for database benchmarks. With sysbench, you can quickly get an impression of database performance. For details, visit **https://github.com/akopytov/sysbench**.

Sysbench 1.0.12 is used as an example. Run the following commands to install it:

#wget -c https://github.com/akopytov/sysbench/archive/1.0.12.zip

#yum install make automake libtool pkgconfig libaio-devel postgresql-devel

#unzip 1.0.12.zip

#cd sysbench-1.0.12

#./autogen.sh

#./configure --with-pgsql --without-mysql

#make

#make install

NOTE

The test tool for RDS for PostgreSQL 12 and RDS for PostgreSQL 13 is as follows:

- **Sysbench 1.0.18**

Test Procedure

Replace the database name, connection IP address, and user password based on the site requirements.

Step 1 Import data.

1. Run the following commands to log in to a database and create the test database **loadtest**:

   psql -h*<host>* **-p5432 "dbname=postgres user=root password=<password>" <<TEST**

   create database loadtest;

   TEST

2. Run the following command to import the test background data to the **loadtest** database:

   sysbench --test=/usr/local/share/sysbench/tests/include/oltp_legacy/oltp.lua --db-driver=pgsql --pgsql-db=loadtest --pgsql-user=root --pgsql-password=*<password>* **--pgsql-port=5432 --pgsql-host=***<host>* **--oltp-tables-count=64 --oltp-table-size=10000000 --num-threads=20 prepare**

Step 2 Run the following command to perform a stress testing:

sysbench --test=/usr/local/share/sysbench/tests/include/oltp_legacy/oltp.lua --db-driver=pgsql --pgsql-db=loadtest --pgsql-user=root --pgsql-password=*<password>* **--pgsql-port=5432 --pgsql-host=***<host>* **--oltp-tables-count=64 --oltp-table-size=10000000 --max-time=3600 --max-requests=0 --num-threads=64 --report-interval=3 --forced-shutdown=1 run**

Step 3 Run the following command to delete the test data:

sysbench --test=/usr/local/share/sysbench/tests/include/oltp_legacy/oltp.lua --db-driver=pgsql --pgsql-db=loadtest --pgsql-user=root --pgsql-password=*<password>* **--pgsql-port=5432 --pgsql-host=***<host>* **--oltp-tables-count=64 --oltp-table-size=10000000 --max-time=3600 --max-requests=0 --num-threads=200 cleanup**

----End

Testing Model

1. Table structure:

   ```
   CREATE TABLE `sbtest` (
   `id` INTEGER IDENTITY(1,1) NOT NULL,
   `k` INTEGER DEFAULT '0' NOT NULL,
   `c` CHAR(120) DEFAULT '' NOT NULL,
   `pad` CHAR(60) DEFAULT '' NOT NULL,
   PRIMARY KEY (`id`)
   )
   ```

2. Read/write ratio:

   The default transaction submitted by sysbench contains 18 SQL statements. The details are as follows:

   - Ten primary key SELECT statements:

     ```
     SELECT c FROM ${rand_table_name} where id=${rand_id};
     ```

   - Four range SELECT statements:

     ```
     SELECT c FROM ${rand_table_name} WHERE id BETWEEN ${rand_id_start} AND ${rand_id_end};
     SELECT SUM(K) FROM ${rand_table_name} WHERE id BETWEEN ${rand_id_start} AND ${rand_id_end};
     SELECT c FROM ${rand_table_name} WHERE id BETWEEN ${rand_id_start} AND ${rand_id_end} ORDER BY c;
     SELECT DISTINCT c FROM ${rand_table_name} WHERE id BETWEEN ${rand_id_start} AND ${rand_id_end} ORDER BY c;
     ```

   - Two UPDATE statements:

     ```
     UPDATE ${rand_table_name} SET k=k+1 WHERE id=${rand_id}
     UPDATE ${rand_table_name} SET c=${rand_str} WHERE id=${rand_id}
     ```

   - One DELETE statement:

     ```
     DELETE FROM ${rand_table_name} WHERE id=${rand_id}
     ```

   - One INSERT statement:

     ```
     INSERT INTO ${rand_table_name} (id, k, c, pad) VALUES (${rand_id},${rand_k},${rand_str_c},${rand_str_pad})
     ```

Test Metrics

- TPS refers to the number of transactions executed per second by a database. Each transaction contains 18 SQL statements.
- QPS refers to the number of SQL statements, including INSERT, SELECT, UPDATE, and DELETE statements, executed per second.

2.2 RDS for PostgreSQL 12 Test Data

2.2.1 General-Purpose DB Instances

About IOPS

The IOPS supported by RDS for PostgreSQL depends on the I/O performance of Elastic Volume Service (EVS) disks. For details, see **Disk Types and Performance** in the *Elastic Volume Service Service Overview*.

Test Data

Table 2-1 vCPU:Memory = 1:2

vCPUs	Memory (GB)	TPS	QPS	IOPS
1	2	172	3,441	See **About IOPS**.
2	4	404	8,088	
4	8	894	17,887	
8	16	1,592	31,832	

Table 2-2 vCPU:Memory = 1:4

vCPUs	Memory (GB)	TPS	QPS	IOPS
2	8	419	8,378	See **About IOPS**.
4	16	961	19,227	
8	32	1,862	37,237	

Test Results

Figure 2-1 vCPU:Memory = 1:2

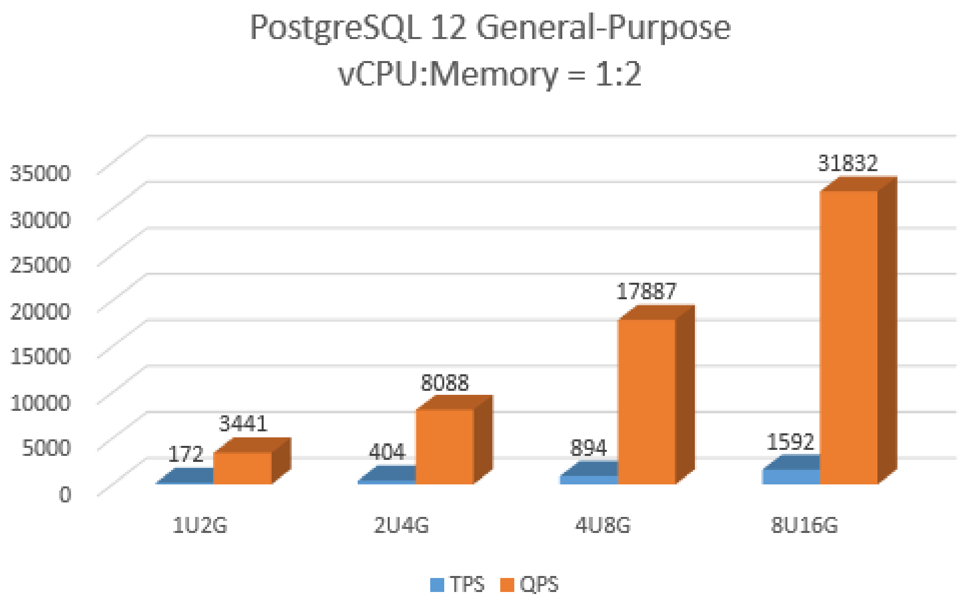
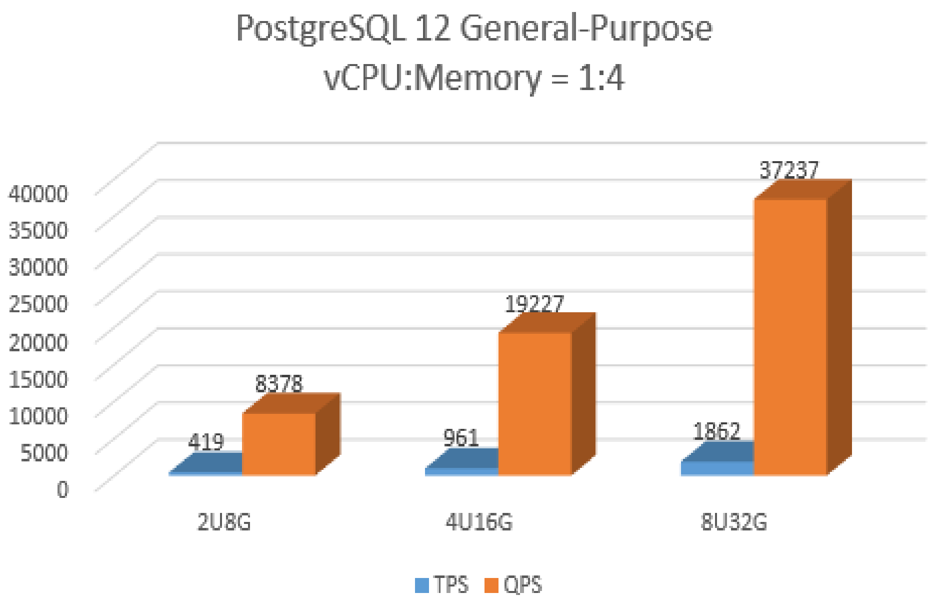


Figure 2-2 vCPU:Memory = 1:4



2.2.2 Dedicated DB Instances

About IOPS

The IOPS supported by RDS for PostgreSQL depends on the I/O performance of Elastic Volume Service (EVS) disks. For details, see **Disk Types and Performance** in the *Elastic Volume Service Service Overview.*

Test Data

Table 2-3 vCPU:Memory = 1:4

vCPUs	Memory (GB)	TPS	QPS	IOPS
2	8	458	9,168	See **About IOPS**.
4	16	978	19,560	
8	32	1,850	36,992	
16	64	2,555	51,101	
32	128	5,508	110,167	
64	256	10,510	210,245	

Table 2-4 vCPU:Memory = 1:8

vCPUs	Memory (GB)	TPS	QPS	IOPS
2	16	533	10,667	See **About IOPS**.
4	32	998	19,956	
8	64	2,061	41,216	
16	128	3,907	78,142	
64	512	10,526	210,567	

Test Results

Figure 2-3 vCPU:Memory = 1:4

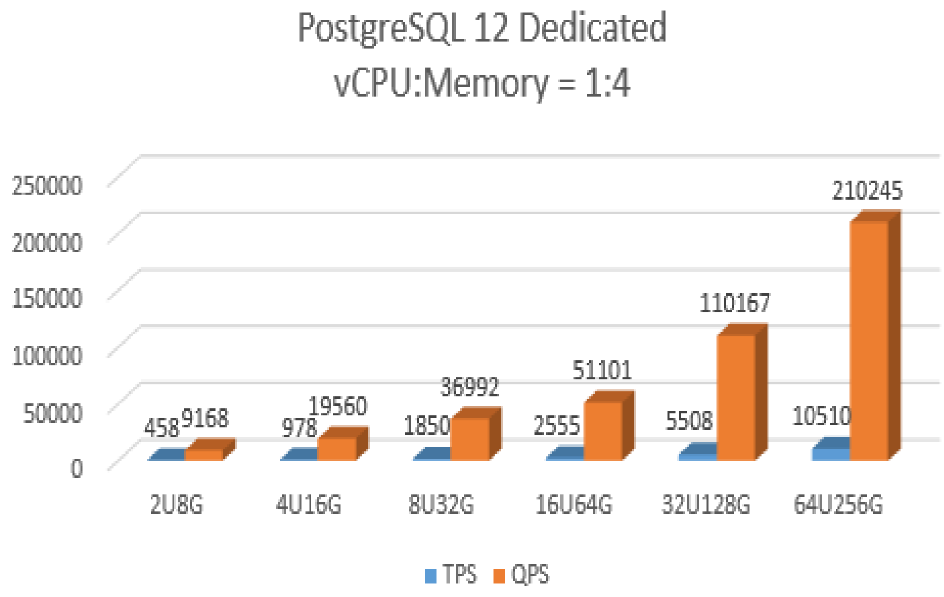
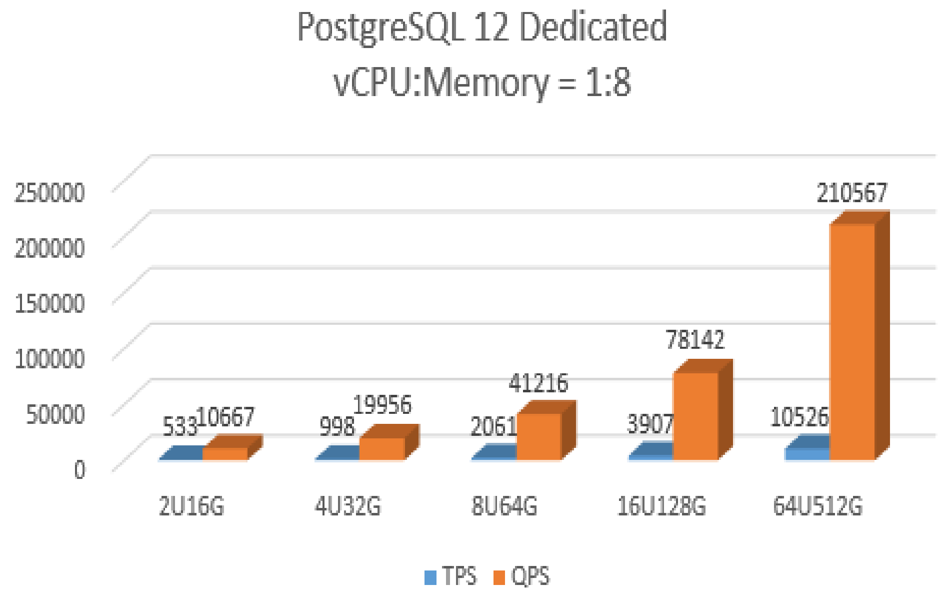


Figure 2-4 vCPU:Memory = 1:8


PostgreSQL 12 Dedicated
vCPU:Memory = 1:8
2U16G
4U32G
8U64G
16U128G
64U512G
TPS
QPS