

ALIBABA

# RDS 性能白皮书

---

MySQL

ApsaraDB 团队

2016/5/23

## 目录

RDS 介绍.....	2
测试方法.....	3
测试环境.....	3
测试工具.....	4
SysBench 简介.....	4
安装方法.....	4
测试命令.....	5
准备数据.....	5
压测性能.....	5
清理环境.....	5
测试模型.....	6
库表结构.....	6
数据格式.....	6
SQL 样式.....	7
测试指标.....	8
TPS.....	8
QPS.....	8
MySQL 5.6.....	9
规格列表.....	9
测试结果.....	10

# RDS 介绍

云数据库 RDS ( Relational Database Service ) 是一种稳定可靠、可弹性伸缩的在线数据库服务。基于飞天分布式系统和全 SSD 盘高性能存储，支持 MySQL、SQL Server、PostgreSQL 和 PPAS ( 高度兼容 Oracle)引擎，默认部署主备架构且提供了容灾、备份、恢复、监控、迁移等方面的全套解决方案，彻底解决数据库运维的烦恼！

云数据库 RDS 提供了按量付费和包年包月两种付费方式，用户可以根据业务压力配置 RDS 实例的规格。其中：

- 1、按量付费实例支持随时随地进行规格的升降级
- 2、包年包月实例支持随时升级和续费升降级

在面对极限的业务压力时，用户还可以随时升级到 RDS 独占主机规格来度过意料外的状况。

# 测试方法

## 测试环境

所有测试均在华东 2（上海）地域的可用区 B 完成。测试用的 ECS 为系列 II 实例，配置为 8 核 16GB，网络类型为经典网络。压测用的镜像为 CentOS 7.0 64 位。

包年包月 按量付费 购买云盘 [购买须知](#)

① 若 ECS 用于网站 Web 访问，请及时备案。若 ECS 用于 SLB，请前往 SLB 新购页面购买带宽，ECS 仅需保留少量带宽以便您管理。

**华东弗吉尼亚节点盛大开通**

地域：

华北 1 (青岛)	华北 2 (北京)	华东 1 (杭州)	<b>华东 2 (上海)</b>	华南 1 (深圳)
香港	亚太 (新加坡)	美西 (硅谷)	美东 (弗吉尼亚)	

  
不同地域之间的产品内网不互通；订购后不支持更换地域，请谨慎选择 [教我选择>>](#) [查看我的产品地域](#)

可用区：**华东 2 可用区 B** [查看实例分布详情>>](#)

网络类型：**经典网络** 专有网络 [教我选择>>](#)  
经典网络与专有网络不能互通，购买后不能更换网络类型，请谨慎选择

实例系列：**系列 II**  
系列 II 采用 Intel Haswell CPU、DDR4 内存，拥有更好的内存计算能力；默认为 I/O 优化实例，搭配 SSD 云盘可获得更好的存储性能。

I/O 优化：**I/O 优化实例**

实例规格：**8 核 16GB (通用型 n1, ecs.n1.xlarge)**  
[请选择实例规格](#)

公网带宽：**按使用流量计费**  
带宽峰值：**25M 50M 100M 100 Mbps**  
按使用流量计费：是先使用后付费产品，每小时扣费。为了您的服务正常运行请保证您账户余额充足。阿里云免费提供最高 5Gbps 的恶意流量攻击防护，[了解更多>>](#) [提升防护能力>>](#)

镜像类型：**公共镜像** 自定义镜像 共享镜像 镜像市场  
公共镜像：**CentOS 7.0 64位**  
[教我选择>>](#)

系统盘：**高效云盘 40 GB 1240 IOPS** 系统盘设备名：/dev/xvda  
如何选择 SSD 云盘 / 高效云盘 / 普通云盘，请看 [详细说明>>](#)

数据盘：**+** 增加一块 您还可选配 4 块

设置密码：**立即设置** 创建后设置  
请牢记您所设置的密码，如遗忘可登录 ECS 控制台重置密码。

登录名：**root**

登录密码： 8 - 30 个字符，且同时包含三项（大、小写字母，数字和特殊符号）

确认密码：

实例名称： 长度为 2-128 个字符，以大写字母或中文开头，可包含数字，"\_"、"." 或 "-"

购买清单 0 台

当前配置

地域：**华东 2 (上海) (华东 2 可用区 B)**

规格：**8 核 16GB (通用型 n1)**

镜像：**CentOS 7.0 64位**

存储：**-**

网络：**带宽100Mbps (经典网络)**

购买量：**1个月 X 1台**

**¥864.00**

配置费用：**¥0.80 /GB**

[立即购买](#) [加入清单](#)

实际扣费以账单为准 [购买和计费说明>>](#)

## 测试工具

### SysBench 简介

SysBench 是一个跨平台且支持多线程的模块化基准测试工具，用于评估系统在运行高负载的数据库时相关核心参数的性能表现。它目的是为了绕过复杂的数据库基准设置，甚至在没有安装数据库的前提下，快速了解数据库系统的性能。

### 安装方法

本文用的 SysBench 版本为 0.5，下载地址为 <https://github.com/akopytov/sysbench>

```
# yum install gcc gcc-c++ autoconf automake make libtool bzip2-devel
```

```
# unzip sysbench-0.5.zip
```

```
# cd sysbench-0.5
```

```
# ./autogen.sh
```

```
# ./configure --prefix=/usr --mandir=/usr/share/man
```

```
# make
```

```
# make install
```

## 测试命令

### 准备数据

```
sysbench --num-threads=32 --max-time=3600 --max-requests=999999999  
--test= oltp.lua --oltp-table-size=10000000 --oltp-tables-count=64  
--db-driver=mysql --mysql-table-engine=innodb --mysql-host= XXXX  
--mysql-port=3306 --mysql-user= XXXX --mysql-password= XXXX prepare
```

### 压测性能

```
sysbench --num-threads=32 --max-time=3600 --max-requests=999999999  
--test= oltp.lua --oltp-table-size=10000000 --oltp-tables-count=64  
--db-driver=mysql --mysql-table-engine=innodb --mysql-host= XXXX  
--mysql-port=3306 --mysql-user= XXXX --mysql-password= XXXX run
```

### 清理环境

```
sysbench --num-threads=32 --max-time=3600 --max-requests=999999999  
--test= oltp.lua --oltp-table-size=10000000 --oltp-tables-count=64  
--db-driver=mysql --mysql-table-engine=innodb --mysql-host= XXXX  
--mysql-port=3306 --mysql-user= XXXX --mysql-password= XXXX cleanup
```

## 测试模型

### 库表结构

```
CREATE TABLE `sbtest` (  
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,  
  `k` int(10) unsigned NOT NULL DEFAULT '0',  
  `c` char(120) NOT NULL DEFAULT "",  
  `pad` char(60) NOT NULL DEFAULT "",  
  PRIMARY KEY (`id`),  
  KEY `k_1` (`k`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8
```

### 数据格式

id: 1

k: 3718516

c:

08566691963-88624912351-16662227201-46648573979-64646226163-77505759

394-75470094713-41097360717-15161106334-50535565977

pad: 63188288836-92351140030-06390587585-66802097351-49282961843

# SQL 样式

查询：

```
SELECT c FROM sbtest64 WHERE id=4957216
SELECT c FROM sbtest43 WHERE id BETWEEN 4573346 AND 4573346+99
SELECT SUM(K) FROM sbtest57 WHERE id BETWEEN 5034894 AND 5034894+99
SELECT DISTINCT c FROM sbtest50 WHERE id BETWEEN 4959831 AND 4959831+99 ORDER BY c
```

写入：

```
INSERT INTO sbtest3 (id, k, c, pad) VALUES (4974042, 4963580,
'33958272865-80411528812-36334179010-84793024318-25708692091-43736213170-3785379
7624-40480626242-32131452190-24509204411',
'07716658989-39745043214-17284860193-80004426880-14154945098')
```

更新：

```
UPDATE sbtest11 SET k=k+1 WHERE id=5013989
UPDATE sbtest14 SET
c='10695174948-02130015518-68664370682-70336600207-55943744221-72419172189-36252
607855-75106351226-86920614936-86254476316' WHERE id=5299388
```

删除：

```
DELETE FROM sbtest33 WHERE id=5002332
```



## 测试指标

### TPS

Transaction Per Second , 数据库每秒执行的事务数, 以 commit 为准。

### QPS

Query Per Second , 数据库每秒执行的 SQL 数( 含 insert、select、update、delete 等 )。

# MySQL 5.6

## 规格列表

规格编号	CPU/Core	内存/GB	连接数	IOPS	TPS	QPS
rds.mysql.t1.small	1	1	300	600	126	1464
rds.mysql.s1.small	1	2	600	1000	153	1950
rds.mysql.s2.large	2	4	1200	2000	261	4703
rds.mysql.s2.xlarge	2	8	2000	4000	314	5662
rds.mysql.s3.large	4	8	2000	5000	560	10896
rds.mysql.m1.medium	4	16	4000	7000	685	12337
rds.mysql.c1.large	8	16	4000	8000	982	20879
rds.mysql.c1.xlarge	8	32	8000	12000	1308	23555
rds.mysql.c2.xlarge	16	64	16000	14000	2040	41530
rds.mysql.c2.xlp2	16	96	24000	16000	2366	45000
rds.mysql.c2.2xlarge	16	128	32000	16000	2688	48384
rds.mysql.st.d13	30	220	64000	20000	4176	75163

# 测试结果

