

# 团体标准

T/GCC 7010—2025

## 服务器应用场景性能测试方法 分布式关系型数据库

Performance test methods for server application scenario

Distributed Relational Database

2025-12-01 发布

2025-12-01 实施

全球计算联盟

发布



版权保护文件

版权所有归属于该标准的发布机构，除非有其他规定，否则未经许可，此发行物及其章节不应以其他形式或任何手段进行复制、再版或使用，包括电子版，影印件，或发布在互联网及内部网络等。使用许可可于发布机构获取。

# 目 录

前言 .....	II
引言 .....	3
1 范围 .....	1
2 规范性引用文件 .....	1
3 术语和定义 .....	1
4 缩略语 .....	2
5 测试对象 .....	2
6 测试工具 .....	3
7 测试指标 .....	3
8 测试方法 .....	3
8.1 总体要求 .....	4
8.2 测试内容 .....	4
9 测试报告 .....	10
参考文献 .....	10

## 前 言

本文件按照GB/T 1.1—2020《标准化工作导则 第1部分：标准化文件的结构和起草规则》的规定起草。

本文件由中国电子技术标准化研究院提出

本文件由全球计算联盟归口。

本文件起草单位：中国电子技术标准化研究院、北京万里开源软件有限公司、华为技术有限公司。

本文件主要起草人：陈颖、李雪莲、司同、李杨桅、徐爽、包振忠、杨海俊、鲁江华、李聪聪。

中国团体标准

## 引 言

绿色经济、低碳经济是全球共同追求的发展方向。在云计算、大数据快速兴起的背景下，市场对高效率、低成本、低能耗的绿色计算产品需求日趋强烈。为了融入国际主流，全球计算联盟积极参与和推动全球产业生态健康发展，协同构建绿色、开放、自主、共享生态体系为目标，进一步推动绿色计算产业发展，建设产业交流与合作平台，促进开放创新的绿色计算产业生态加速完善。当前已发布典型应用场景测评系列标准包括《服务器应用场景性能测试方法 大数据》、《服务器应用场 景性能测试方法 分布式存储》、《服务器应用场景性能测试方法 Web应用》、《服务器应用场 景性能测试方法 高性能计算》、《服务器应用场景性能测试方法 集中式关系型数据库》《服务器应用场景性能测试方法 Arm原生云手机》、《服务器应用场景性能测试方法 虚拟化》等七项标准，旨在推动底层服务器硬件和上层应用软件的联合调优，以体现Arm全栈解决方案的整体竞争力。《服务器应用场景性能测试方法 分布式关系型数据库》也作为系列标准之一，本文件可用于各行业分布式关系型数据库场景的服务器的性能评测，可指导服务器的设计、实现、验证，能推动基于绿色计算服务器的分布式关系型数据库解决方案的能力提升，帮助用户更好的匹配服务器产品和解决方案，软硬件协同更好的发挥产业价值。

# 服务器应用场景性能测试方法 分布式关系型数据库

## 1 范围

本文件规定了服务器在分布式关系型数据库管理系统场景的性能测试方法。

本文件适用于分布式关系型数据库管理系统的检测与评估，可供各行业组织参考，也可作为第三方评估机构衡量数据库服务能力的标准依据。

## 2 规范性引用文件

下列文件对于本文件的应用是必不可少的。凡是注日期的引用文件，仅注日期的版本适用于本文件。凡是不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

GB/T 5271.17-2010 信息技术 词汇 第17部分：数据库

GB/T 17532-2005 术语工作 计算机应用 词汇

GB/T 30994-2014 关系数据库管理系统检测规范

GB/T 32633-2016 分布式关系数据库服务 接口规范

## 3 术语和定义

下列术语和定义适用于本文件。

### 3.1

**数据库 database**

支持一个或多个应用领域，按概念结构组织的数据集合，其概念结构描述这些数据的特征及其对应实体间的联系。

### 3.2

**数据对象 data object**

看作一个单位离散数据，且该数据表示已知或假定已知的数据结构的实例。

### 3.3

**数据类型 data type**

规定数据结构中经定义的数据对象的集合和一组许可的运算，在这些运算中任何一个执行时，其中数据对象都被当作运算数。

### 3.4

**查询语言 query language**

一种供用户对数据库中的数据进行检索并可能加以修改的数据操纵语言。

### 3.5

**分布式数据库 distributed database**

一种为用户提供的逻辑上由集中式数据库视图管理，而物理上由某一个数据库管理系统处置的分散的数据库。

### 3.6

#### 数据库管理员 database administrator

负责数据库综合管理的个人或小组。

### 3.7

#### 架构 architecture

描述信息系统构成要素及相互关系，指导信息系统设计和完善的框架原则。

### 3.8

#### 业务系统 business system

由计算机硬件（物理和虚拟资源）、网络和通信设备、计算机软件、信息资源、信息用户和规章制度组成的以收集（或获取）、处理、存储、分配信息为目的的人机一体化系统。

### 3.9

#### 一致性 consistency

一致性就是数据保持一致，在分布式系统中，可以理解为多个节点中数据的值是一致的。

## 4 缩略语

下列缩略语适用于本文件。

CPU：中央处理器（Central Processing Unit）

I/O：输入/输出（Input/Output）

RTO：复原时间目标（Recovery Time Objective）

RPO：复原点目标（Recovery Point Objective）

SQL：结构化查询语言（Structural Query Language）

TPC-C 专门针对联机交易处理系统（OLTP系统）的规范（Transaction Processing Performance Council Benchmark C）

TPC-H：测试数据库系统复杂查询的响应时间（Transaction Processing Performance Council Benchmark H）

tpmC：每分钟内系统处理的新订单个数（Transaction Per Minute）

TPS：每秒钟处理的事务量（Transaction Per Second）

## 5 测试对象

本文件的测试对象为用于分布式关系型数据库的服务器，在测试过程中需要把数据压力放在数据库服务器上。测试场景对分布式关系型数据库针对事务处理能力、多用户并发操作的响应时间、资源利用率等性能测试及稳定性的极限能力进行测试。

分布式测试架构组网见下页如图1所示，功能评测需要的设备包含：3台用于安装被测数据库产品的分布式数据库服务器、1台用于安装加压客户端的负载服务器，1台用于远程连接操作的测试端服务器。

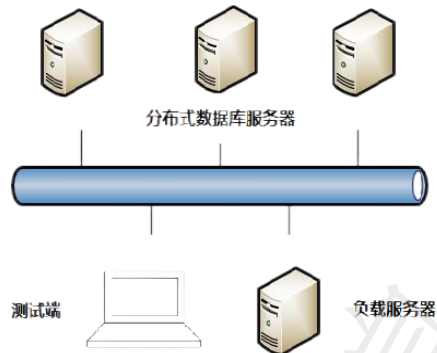


图1 分布式测试架构组网

## 6 测试工具

对分布式关系型数据库场景的服务器进行性能测试所使用的测试工具见表1。

表1 分布式关系型数据库功能测试工具

测试工具名称	用途
BenchmarkSQL V5.0	对数据库进行 TPC-C 标准测试
Sysbench V1.0	对数据库进行标准性能测试
TPC-H V3.0.1	对数据库进行 TPC-H 标准测试
TPC-C V1.0	对数据库进行标准性能测试

## 7 测试指标

### 7.1 每秒事务量

服务器系统每秒钟处理的事务量，单位为（TPS）。

### 7.2 每秒查询量

服务器系统每秒钟处理的查询数量，单位为（QPS）。

### 7.3 吞吐量

服务器系统每分钟处理新建订单的数量，单位为（tpmC）。

### 7.4 执行时长

服务器系统运行某一命令所需要的时间，单位为(ms)或(s)。

## 8 测试方法

### 8.1 总体要求

用例参数的取值应使得服务器正常运行的情况下相应资源达到瓶颈，在测试报告应按要求明确详细的软硬件配置及优化参数，以达到根据详细配置进行结果复测的目的。

### 8.2 测试内容

#### 8.2.1 读性能测试

读性能测试规范对数据量、工具、方法和指标等测试要求见表 2。

表2 读性能测试要求

编号	并发	数据量 (单位: 亿行)	测试工具	测试方法	测试指标
1	800	1	Sysbench	使用压测工具测试只读场景, 调用oltp_read_only.lua, 达到对目标数据库发起压力, 模拟在线业务运行的场景。 数据量=表数量*行数	每秒查询量 (单位: QPS)
2	1000	1			每秒查询量 (单位: QPS)
3	2000	1			每秒查询量 (单位: QPS)

#### 8.2.2 写性能测试

写性能测试规范对数据量、工具、方法和指标等测试要求见表 3。

表3 写性能测试要求

编号	并发	数据量 (单位: 亿行)	测试工具	测试方法	测试指标
1	800	1	sysbench	使用压测工具测试只写场景, 调用oltp_insert.lua, 达到对目标数据库发起压力, 模拟在线业务运行的场景。 数据量=表数量*行数	每秒处理的事务量 (单位: TPS)
2	1000	1			每秒处理的事务量 (单位: TPS)
3	2000	1			每秒处理的事务量 (单位: TPS)

#### 8.2.3 删除性能测试

删除性能测试规范对数据量、工具、方法和指标等测试要求见表 4。

表4 删除性能测试要求

编号	并发	数据量 (单位: 亿行)	测试工具	测试方法	测试指标
1	800	1	sysbench	使用压测工具测试删除场景, 调用oltp_delete.lua, 达到对目标数据库发	每秒处理的事务量 (单位: TPS)
2	1000	1			每秒处理的事务量 (单位: TPS)

3	2000	1		起压力，模拟在线业务运行的场景。 数据量=表数量*行数	每秒处理的事务量（单位：TPS）
---	------	---	--	--------------------------------	------------------

#### 8.2.4 更新性能测试

更新性能测试规范对数据量、工具、方法和指标等测试要求见表 5。

表5 更新性能测试要求

编号	并发	数据量 (单位：亿行)	测试工具	测试方法	测试指标
1	800	1	sysbench	使用压测工具测试工具索引更新场景，调用oltp_update_index.lua，达到对目标数据库发起压力，模拟在线业务运行的场景。 数据量=表数量*行数	每秒处理的事务量（单位：TPS）
2	1000	1			每秒处理的事务量（单位：TPS）
3	2000	1			每秒处理的事务量（单位：TPS）

#### 8.2.5 批量加载

批量加载测试规范对数据量、工具、方法和指标等测试要求见表 6。

表6 批量加载测试要求

编号	仓数	测试工具	测试方法	测试指标
1	1000	benchmarksql	用benchmarksql工具生成1000仓离线数据，使用数据库批量加载语句将数据加载到集群。	记录加载时长（单位：h和min）

#### 8.2.6 TPCC 性能测试

性能测试工具可以从 benchmarksql 或 TPC-C 中选择，其中，benchmarksql 用于测量每分钟交易次数，TPC-C 用于测试数据库各方面性能。

##### 8.2.6.1 benchmarksql 测量每分钟内 TPC-C 的新建订单数

收集有效 TPC-C 配置期间每分钟处理的平均交易次数测量，运行 1 小时。测试对数据量、工具、方法和指标等测试要求见表 7。

表7 benchmarksql 测试要求

编号	仓数	并发	测试工具	测试方法	测试指标
1	1000	500	benchmarksql	基础数据为1000 仓，测试机构统一提供，每次测试完一组数据重新清除并加载1000仓基础数据，在测试完成后进行数据一致性验证	每分钟处理新建订单的数量 (单位：tpmC)
2	1000	1000			每分钟处理新建订单的数量 (单位：tpmC)
3	1000	1500			每分钟处理新建订单的数量 (单位：tpmC)
4	1000	2000			每分钟处理新建订单的数量 (单位：tpmC)

## 8.2.6.2 TPC-C 性能测试

TPC-C 是专门针对联机交易处理系统（OLTP 系统）的规范。TPCC 测试关系型数据库的各方面性能。TPCC 的测试库能覆盖大多数的业务场景，测试的结果也能反映出真实业务中数据库的实际性能。测试对数据量、工具、方法和指标等测试要求见表 8。

表8 TPC-C 性能测试要求

测试项目	性能测试	分项目	TPCC测试工具
用例编号	7.2.6	版本	1.0
前置条件	1、下载地址 <a href="https://github.com/Percona-Lab/">https://github.com/Percona-Lab/</a> 2、编译： <code>unzip tpcc-mysql-master.zip</code> <code>cd /soft/tpcc-mysql-master/src</code> <code>make</code> <code>ls *.sql</code> <code>create_table.sql</code> 是创建表的sql文件 <code>add_fkey_idx.sql</code> 是索引等约束文件 <code>mysql -uroot -proot tpcc &lt; create_table.sql</code> <code>mysql -uroot -proot tpcc &lt; add_key_idx.sql</code>		
测试步骤	1、准备测试数据 <code>./tpcc_load -h 192.168.1.27 -d tpcc -u admin -p Abc_123456 -w 1</code> <code>-h 192.168.1.27</code> 数据库ip地址 <code>-d tpcc</code> 数据库名字 <code>-u admin</code> 用户名 <code>-p Abc_123456</code> 密码 <code>-w 1</code> 仓库数量，由于数量庞大，插入时间较长，所以这里使用1个仓库数量，如果使用多个仓库，耗时很长。 2、执行测试 <code>./tpcc_start -h 192.168.1.27 -d tpcc -u admin -p Abc_123456 -w 1 -c 5 -r 300 -l 600 - &gt;tpcc-outpit.log</code> <code>-h 192.168.1.27</code> 数据库ip地址 <code>-d tpcc</code> 数据库名字 <code>-u admin</code> 用户名 <code>-p Abc_123456</code> 密码 <code>-w 1</code> 仓库数量 <code>-c 5</code> 并发线程数 <code>-r 300</code> 数据库预热时间 单位秒 <code>-l 600</code> 测试时间 单位秒 <code>- &gt;tpcc-outpit.log</code> 测试结果输出到文件 为了真实性可以将-r 和-l时间设置长一些，比如预热1个小时，测试24小时 3、执行结果含义 <code>sc</code> : 成功执行的次数 <code>lt</code> : 超时执行的次数 <code>rt</code> : 重试执行的次数 <code>fl</code> : 失败执行的次数  第一行是新订单执行的测试结果 第二行是支付业务的测试结果 第三行是订单状态的测试结果 第四行是发货业务的测试结果 第五行是库存业务的测试结果		
预期结果	执行结果显示ok		
备注	记录测试结果		

## 8.2.7 资源占用测试

### 8.2.7.1 数据库压力测试

在数据库压力测试过程中，记录数据库服务器资源占用的情况：将每2分钟通过nmon记录一次，每次测试期间的CPU、内存、i/o资源占用情况，最后输出表。

### 8.2.7.2 资源占用记录

Nmon (Nigel's Monitor)是由IBM提供、免费监控AIX系统与Linux系统资源的工具。该工具可将服务器系统资源耗用情况收集起来并输出一个特定的文件,并可利用excel分析工具(nmon analyser)进行数据的统计分析。测试对数据量、工具、方法和指标等测试要求见表9。

表9 资源占用测试要求

测试项目	性能测试	分项目	资源占用测试
用例编号	7.2.7	版本	1.0
预置条件	yum 安装: yum install -y nmon 二进制安装: 下载地址 http://nmon.sourceforge.net/pmwiki.php?n=Site.Download 找到对应版本的文件下载到本地,可以上传到linux指定目录下,修改权限, chmod a+x nmon		
测试步骤	交互式查看监控信息,可以执行 ./nmon 进入监控窗口, 以后台运行的方式收集监控信息nmon -f -N -m /data/log -s 30 -c 120 其中各参数表示: -f按标准格式输出文件: <hostname>_YYYYMMDD_HHMM.nmon -N include NFS sections -m 切换到路径去保存日志文件 -s 每隔n秒抽样一次,这里为30 -c 取出多少个抽样数量,这里为120,即监控=120*(30/60/60)=1小时 根据小时计算这个数字的公式为: c=h*3600/s,比如要监控10小时,每隔30秒采样一次,则c=10*3600/30=1200 该命令启动后,会在nmon所在目录下生成监控文件,并持续写入资源数据,直至360个监控点收集完成——即监控1小时,这些操作均自动完成,无需手工干预,测试人员可以继续完成其他操作。如果想停止该监控,需要通过 **ps -ef grep nmon **查询进程号,然后杀掉该进程以停止监控,可以使用 nmon analyser 对文件进行分析。		
预期结果	正确显示资源占用情况		
备注	记录测试结果		

## 8.2.8 一致性验证

每次进行完TPCC性能测试后,进行数据一致性验证,并记录查询结果、执行时间,同时记录每个表单的数据量(针对查询,执行前后数据量没有发生变化;增删改查,删除后的一致)。数据量、工具、方法和指标等测试要求见表10,执行语句内容见表11。

表10 一致性验证测试要求

编号	测试项	测试方法	测试指标
1	TPCC测试后, 进行数据一致性验证	执行结果(预期结论: 结果都是空集, 满足一致性要求)	执行时间(单位: S)

表11 一致性验证执行语句

执行语句一	(Select w_id, w_ytd from warehouse) except(select d_w_id, sum(d_ytd) from district group by d_w_id);
执行语句二	(Select d_w_id, d_id, D_NEXT_O_ID - 1 from district) except (select o_w_id, o_d_id.max(o_id) from oorder group by o_w_id, o_d_id);
执行语句三	(Select d_w_id, d_id, D_NEXT_O_ID - 1 from district) except (select no_w_id, no_d_id.3max(no_o_id) from new_order group by no_w_id, no_d_id);
执行语句四	select * from (select -(count(no_o_id)-(max(no_o_id)-min(no_o_id)+1)) as diff from new_order group by no_w_id, no_d_id) where diff != 0;
执行语句五	(select o_w_id, o_d_id, sum(o_ol_cnt) from oorder group by o_w_id, o_d_id) except(select ol_w_id, ol_d_id, count(ol_o_id) from order_line group by ol_w_id, ol_d_id);
执行语句六	(select d_w_id, sum(d_ytd) from district group by d_w_id) except(Select w_id, w_ytd from warehouse);

### 8.2.9 海量数据复杂查询、Join 查询性能测试(可选)

海量数据复杂查询、Join 查询性能测试规范对数据量、工具、方法和指标等测试要求见表 12, 该测试内容可根据实际情况选测。

表12 海量数据复杂查询、Join 查询性能测试要求(可选)

编号	仓数	测试工具	测试方法	测试指标
1	3000	TPC-H	初始化3000仓TPC-H数据, 并执行TPC-H各标准语句	TPC-H各语句的执行时间(单位: S)

### 8.2.10 168 小时稳定性测试(可选)

在基础数量 1000 仓和 100 并发用户, 持续运行系统 168 小时, 记录执行结果与服务器资源占用情况。168 小时稳定性测试对数据量、工具、方法和指标等测试要求见表 13, 该测试内容可根据实际情

况选测。

表13 168小时稳定性测试要求（可选）

编号	仓数	并发	测试工具	测试方法	测试指标
1	1000	100	benchmarks ql	持续运行系统168小时，执行操作：1. TPC-C场景进行负载测试；2. 记录工具执行结果、服务器资源占用情况（通过nmon每2小时记录一次，提交最终分析图）	能够顺利完成168小时压测，中途无异常

## 9 测试报告

本文件仅对测试方向作指引，不作强制约束。附录对测试过程中涉及的包括但不限于编译参数、测试配置、虚拟化比例等仅作参考，标准之外应当有具体的测试用例约束测评环境一致，测试人员对软硬件配置及参数调优应当在测试报告中详细列明。

依据本文件的测试报告应披露以下测试环境信息：

- a) 待测服务器硬件配置（包括但不限于处理器、内存、网卡、硬盘等）；
- b) 待测服务器软件配置（包括但不限于操作系统类型及版本、编译软件版本及参数、镜像或模板文件大小、镜像操作系统版本、应用软件版本等）；
- c) 待测服务器使用的分布式关系型数据库及其版本信息；
- d) 存储配置（包括但不限于物理存储资源厂商、类型、容量、读写能力等）；
- e) 网络配置（包括但不限于网络拓扑结构、部件厂商、转发速率、接口速率等）；
- f) 测试工具版本、配置以及编译参数；
- g) 测试过程软、硬件调优过程及参数。

参 考 文 献

- [1] SPEC CPU®2017 Run and Reporting Rules.
-