



# (12) 发明专利

(10) 授权公告号 CN 110046098 B

(45) 授权公告日 2022. 12. 13

(21) 申请号 201910266160.3

(22) 申请日 2019.04.03

(65) 同一申请的已公布的文献号  
申请公布号 CN 110046098 A

(43) 申请公布日 2019.07.23

(73) 专利权人 河南源网荷储电气研究院有限公司

地址 461000 河南省许昌市魏都区五一路  
街道许继大道1706号

专利权人 许继集团有限公司  
许继电气股份有限公司

(72) 发明人 张玉中 浮明军 王鹏 张超永  
常卫 刘昊昱 周水斌 李松合  
左群业 刘仁千 匡海燕 黄显果  
谷威 杨丰源 王振 刘静静  
刘沛 李亚爽 时晓宁 从波  
宋文博

(74) 专利代理机构 郑州睿信知识产权代理有限公司 41119

专利代理师 陈浩

(51) Int.Cl.  
G06F 11/36 (2006.01)

(56) 对比文件  
CN 106874189 A, 2017.06.20  
CN 109189661 A, 2019.01.11  
US 6951013 B1, 2005.09.27  
US 2006242167 A1, 2006.10.26  
安思成等. 适用于广域测量系统的实时数据  
并发访问同步算法.《中国电机工程学  
报》.2014, 第34卷(第19期), 全文.

Kyoung-Don Kang. A Real-Time Database  
Testbed and Performance Evaluation.《13th  
IEEE International Conference on Embedded  
and Real-Time Computing Systems and  
Applications (RTCSA 2007)》.2007, 全文.

审查员 刘娜

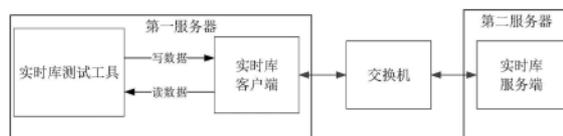
权利要求书1页 说明书4页 附图1页

## (54) 发明名称

一种实时库测试方法及系统

## (57) 摘要

本发明涉及一种实时库测试方法及系统,属于自动化测试领域,实时库测试方法,包括如下步骤:在进行并发测试时,根据所选择的参数,生成相应个数的线程,每个线程均调用实时库客户端写接口,向实时库客户端数据交互接口写入测试数据;通过实时库客户端数据交互接口接收到实时库服务端返回的数据后,调用实时库客户端读接口读取返回的数据,判断各线程中写入的测试数据与对应返回的数据是否相同,统计相同或者不同的线程数量。该方法非常便捷,所依赖的不过若干线程、实时库客户端的几种基本接口,不依赖非常完整的测试环境,提供了一种简单、便捷的测试方案,而且测试过程简单、准确、效率高,进一步的保证测试的质量。



1. 一种实时库测试方法,其特征在于,包括如下步骤:

在进行并发测试时,根据所选择的参数,通过实时库测试工具生成相应个数的线程,每个线程均调用实时库客户端写接口,向实时库客户端数据交互接口写入测试数据,所述实时库客户端数据交互接口用于与待测试的实时库服务端交互数据;

通过实时库客户端数据交互接口接收到实时库服务端返回的数据后,调用实时库客户端读接口读取返回的数据,判断各线程中写入的测试数据与对应返回的数据是否相同,统计相同或者不同的线程数量,从而完成并发测试。

2. 根据权利要求1所述的实时库测试方法,其特征在于,所述实时库测试方法还包括:在进行性能测试时,若选择为读数据测试,统计从下达读操作命令开始,到读到最后一条数据之间的时间,为读数据操作时间。

3. 根据权利要求1所述的实时库测试方法,其特征在于,所述实时库测试方法还包括:在进行性能测试时,若选择为同步写数据测试,统计从下达同步写操作命令开始,到读到实时库客户端同步写接口的返回值之间的时间,为同步写数据操作时间。

4. 根据权利要求1所述的实时库测试方法,其特征在于,所述实时库测试方法还包括:在进行性能测试时,若选择为异步写数据测试,统计从下达异步写操作命令开始,到读到最后一条数据之间的时间,为异步写数据操作时间。

5. 根据权利要求1-4中任一项所述的实时库测试方法,其特征在于,所述实时库测试方法还包括:在测试结束后,生成测试报告。

6. 一种实时库测试系统,其特征在于,该实时库测试系统包括第一服务器和交换机,所述第一服务器部署实时库测试工具和实时库客户端;交换机用于连接第一服务器和被测试的实时库服务端;所述第一服务器包括存储器和处理器,所述处理器执行存储在所述存储器上的计算机程序以实现权利要求1-5中任意一项所述的实时库测试方法。

## 一种实时库测试方法及系统

### 技术领域

[0001] 本发明涉及一种实时库测试方法及系统,属于自动化测试领域。

### 背景技术

[0002] 随着大数据、云服务等新技术的出现,原有传统型数据库很难满足新系统对数据操作的需求,为提高数据访问速度,内存数据库、非关系型数据库—实时库在新系统中的应用越来越广泛。实时库支持hash(哈希)、string(字符串)、set(集合)和list(链表)四种主要存储结构,根据新系统实际业务需求选择不同的存储结构。实时库中字段的访问频率是不同的,有些访问频率比较低,选取了hash存储结构体;有些字段访问频率较高,选取了string存储结构体;有些字段需要快速进行遍历,以获取某值是否存储,选取set存储结构体;有些字段需要按照先后顺序进行处理,选取list存储结构体。为了维护实时库的正常运行,实时库客户端封装了连接/断开实时库、删除某个键,给某个键设置生命周期等功能。

[0003] 目前,实时库测试的难点在于:在Linux系统上没有成熟实时库测试工具;由于使用实时库的新系统规模庞大,开发周期长,短期内很难搭建集成测试环境,即搭建完整的测试环境的时间投入和成本很高,特别是对应并发测试,现有技术依赖完整的测试环境,缺乏有效、便捷的手段进行测试。

### 发明内容

[0004] 本发明的目的是提供一种实时库测试方法,用以解决现有技术缺乏有效、便捷的手段测试实时库的问题;同时,本发明还提供了一种实时库测试系统,用以解决现有技术缺乏有效、便捷的手段测试实时库的问题。

[0005] 为实现上述目的,本发明提出一种实时库测试方法,包括如下步骤:

[0006] 在进行并发测试时,根据所选择的参数,通过实时库测试工具生成相应个数的线程,每个线程均调用实时库客户端写接口,向实时库客户端数据交互接口写入测试数据,实时库客户端数据交互接口用于与待测试的实时库服务端交互数据;

[0007] 通过实时库客户端数据交互接口接收到实时库服务端返回的数据后,调用实时库客户端读接口读取返回的数据,判断各线程中写入的测试数据与对应返回的数据是否相同,统计相同或者不同的线程数量,从而完成并发测试。

[0008] 另外,本发明还提出一种实时库测试系统,该实时库测试系统包括第一服务器和交换机,第一服务器部署实时库测试工具和实时库客户端;交换机用于连接第一服务器和被测试的实时库服务端;第一服务器包括存储器和处理器,处理器执行存储在存储器上的计算机程序以实现上述实时库测试方法。

[0009] 有益效果是:并发测试的好坏直接体现实时库的稳定性,该方法实现了对实时库的并发测试,若统计写入的测试数据与返回的数据相同的线程数量多,则表明实时库的稳定性好,反之则稳定性差。而且该方法非常便捷,所依赖的不过若干线程、实时库客户端的几种基本接口,不依赖非常完整的测试环境,提供了一种简单、便捷的测试方案,而且测试

过程简单、准确、效率高,进一步的保证测试的质量。

[0010] 进一步的,上述实时库测试方法及系统还包括:在进行性能测试时,若选择为读数据测试,统计从下达读操作命令开始,到读到最后一条数据之间的时间,为读数据操作时间。

[0011] 进一步的,上述实时库测试方法及系统还包括:在进行性能测试时,若选择为同步写数据测试,统计从下达同步写操作命令开始,到读到实时库客户端同步写接口的返回值之间的时间,为同步写数据操作时间。

[0012] 进一步的,上述实时库测试方法及系统还包括:在进行性能测试时,若选择为异步写数据测试,统计从下达异步写操作命令开始,到读到最后一条数据之间的时间,为异步写数据操作时间。

[0013] 有益效果是:这种对实时库进行性能测试时,测试的方法,简单、容易实现、实用性强,测试结果能够满足测试要求。

[0014] 进一步的,上述实时库测试方法及系统还包括:在测试结束后,生成测试报告。

[0015] 有益效果是:测试报告将测试结果进行记录,可以用于对实时库进行分析。

#### 附图说明

[0016] 图1为本发明实时库测试系统的结构简图;

[0017] 图2为本发明实时库测试方法的测试流程图。

#### 具体实施方式

[0018] 实时库测试系统实施例:

[0019] 本实施例提出的实时库测试系统如图1所示,包括两台服务器,分别为第一服务器和第二服务器,还包括一台交换机,第一服务器部署实时库测试工具(下文中也简称为测试工具)和实时库客户端,第二服务器部署实时库服务端,两台服务器通过交换机连接,实时库选用开源软件redis,存储结构包括哈希、字符串、集合三种存储结构体。当然,实时库软件也可以是其他软件。

[0020] 第一服务器还包括存储器和处理器,处理器执行存储在存储器上的计算机程序(即测试工具和实时库客户端,该实时库客户端不需要完成所有功能,只需要具备基本的实时库客户端读/写接口,实时库客户端数据交互接口即可)以完成测试。

[0021] 在测试工具进行测试之前,先启动redis服务,然后启动测试工具,按照用户的需求自动生成测试用例(即测试数据),如测试用例中每条数据多少字节,生成多条数据,生成的测试用例用于对实时库服务端进行检测。

[0022] 该实时库测试系统按照如图2所示的方法进行实时库测试:

[0023] 1) 选择测试类型,包括功能测试、性能测试,并发测试,以及选择测试方式,包括同步测试和异步测试,相应的,还需要选择一些参数,例如:并机测试时选择同时并发多少测试用例。

[0024] 通过人工可以向测试工具输入上述各种选择。

[0025] 下面依次对各种测试类型进行说明。

[0026] 同步测试与异步测试分别对应客户端提供的实时库客户端同步写接口和实时库

客户端异步写接口(即实时库客户端写接口包括实时库客户端同步写接口和实时库客户端异步写接口),实时库只有在写测试数据时才分同步和异步,主要区别是同步写测试数据时,写测试数据接口会有返回值,通过返回值可以判断实时库是否写成功。

[0027] 2)若选择同步功能测试时,首先测试工具通过调用实时库客户端同步写接口向实时库数据交互接口写数据,再通过实时库客户端读接口读出实时库服务端中的数据,然后测试工具对比读写的数据是否一致,如果一致,则说明实时库服务端读和写功能正确;如果不一致,说明读或写功能异常,需要进一步定位问题,将测试结果打印出来,这样测试工具自动判定测试结果的同时,测试人员也可以人工抽查判定,能够增加测试的可靠性,而且功能测试时,三种存储结构可以同时测试。

[0028] 测试工具进行异步功能测试时,与上述过程不同仅在于,其调用的是实时库客户端异步写接口,其他过程相同。

[0029] 3)若选择性能测试时,首先测试工具向实时库客户端数据交互接口写数据,并统计写操作的时间,然后测试工具通过实时库客户端读接口读出实时库服务端中的数据,并统计读操作的时间,为用户提供性能方面的数据支撑;性能测试时,由于测试结果的打印会影响性能,为了提高测试的精度,可以不打印测试结果数据,而且性能测试时,需要统计性能数据,只能选择一种存储结构进行测试。

[0030] 在性能测试时,如果是同步写测试,统计从下达同步写操作命令开始,到读到实时库客户端同步写接口的返回值之间的时间,为同步写数据操作时间。

[0031] 如果是读测试,统计从下达读操作命令开始,到读到最后一条数据之间的时间,为读数据操作时间。

[0032] 同步写测试和读测试都能够得到准确的测量结果。

[0033] 如果异步写测试,统计从下达异步写操作命令开始,到读到最后一条数据之间的时间,为异步写数据操作时间。对于写上百级的数据,操作一条数据的时间可以忽略,因此,也是能够满足测试精度要求的。

[0034] 4)测试工具进行并机测试时,根据选择的参数,工具生成相应个数,例如选择的参数为50,则建立50个线程,同时调用实时库客户端写接口,向实时库客户端的数据交互接口(用于与服务端进行交互的接口)写入测试数据;这相当于模拟了50个客户端同时向实时库服务端请求数据;接着实时库服务端返回所请求的数据,测试工具通过实时库客户端读接口读出实时库服务端的数据,将读出的数据与写入的数据进行比对,判断是否有某些线程对应的数据有异常(即写入的数据与返回的数据不对应),对异常情况进行统计,从而判断服务器的并发能力,出现异常的线程越少,并发能力强。而且线程数量多,异常情况少,并发能力越强。在判断读或写功能的同时,还要进行并发时间的统计(参照性能测试的方法即可)。

[0035] 5)生成测试报告,测试报告的形式是根据用户定制自动生成word形式,比如:按测试类型分,可以生成性能测试报告,功能测试报告,并发测试报告;按测试用例分,可以生成单个用例的测试报告或多个用例的测试报告。总之,测试报告的形式多种多样,用户可以灵活定制。当然,生成测试报告是便于对实时库进行分析,在只需测试的情况下,该步骤也可以没有。

[0036] 该方法非常便捷,所依赖的不过若干线程、实时库客户端的几种基本接口,不依赖

非常完整的测试环境,提供了一种简单、便捷的测试方案,而且测试过程简单、准确、效率高,进一步的保证测试的质量。

[0037] 实时库测试方法实施例:

[0038] 本实施例提出的实时库测试方法,包括如下步骤:

[0039] 在进行并发测试时,根据所选择的参数,通过实时库测试工具生成相应个数的线程,每个线程均调用实时库客户端写接口,向实时库客户端数据交互接口写入测试数据,实时库客户端数据交互接口用于与待测试的实时库服务端交互数据;

[0040] 通过实时库客户端数据交互接口接收到实时库服务端返回的数据后,调用实时库客户端读接口读取返回的数据,判断各线程中写入的测试数据与对应返回的数据是否相同,统计相同或者不同的线程数量,从而完成并发测试。

[0041] 实时库测试方法的具体实施过程在上述实时库测试系统实施例中已经介绍,这里不做赘述。

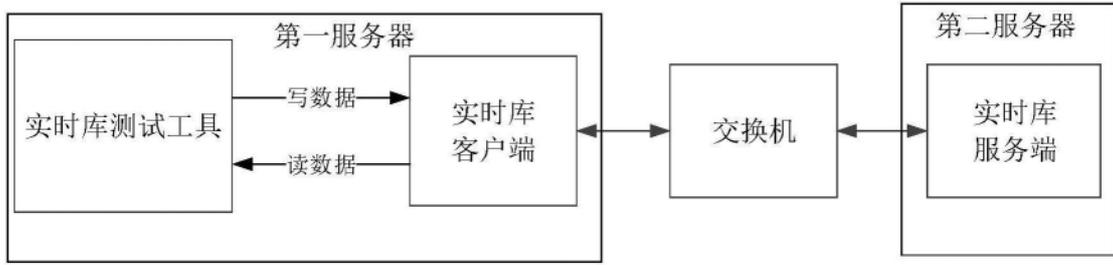


图1

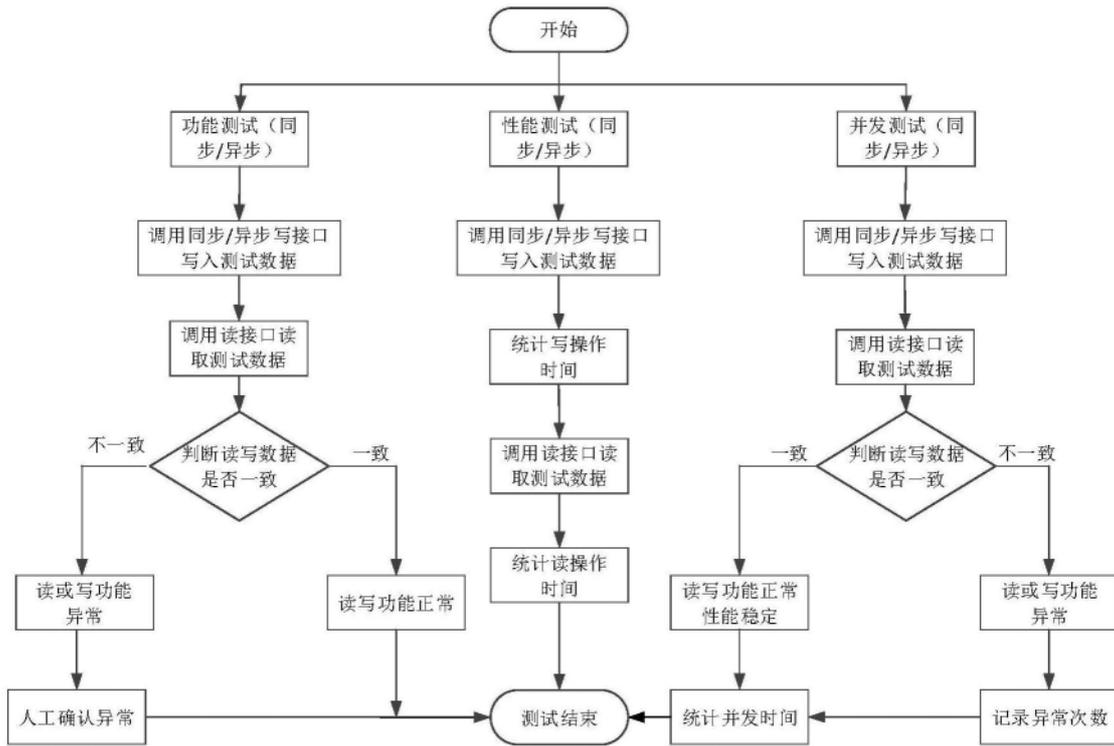


图2