

文件编号: ALIPAY-BI-QC-002

文件版本: v7.0

SqlEngine 测试报告

文档编号 BI20100722001
起草人 官辉 审核

版本号 V6.0
审定

变更日志

[illegible]

目 录

1. 引言	4
1.1. 目的	4
1.2. 背景	4
1.3. 参考资料	4
2. 测试概述	4
3. 测试方法和范围	5
3.1. 测试方法	5
3.2. 测试范围	5
4. 测试结果	6
4.1.1. 缺陷的分布情况图表	6
4.1.2. 功能点测试结果	8
5. 结论	9
附件 1 缺陷详细描述	10
附件 2 缺陷等级的描述	11

1. 引言

1.1. 目的

对 0.8 版本的 sqlengine 中所有能实现的功能点以及阿里适配进行详细的功能测试，看能否达到预期的效果,以便为后续的改写 TCL 程序和数据迁移做好铺垫。

1.2. 背景

SQLEngine 是构建在 Youchao Storage 之上的分布式数据库服务，它包含：服务的运行时环境和一组客户端接口（用于数据的写入、读取和查询）。SQLEngine 的设计目标如下：

- 海量级别结构化数据的存储
- 大规模的离线数据分析
- 高效的在线实时查询

目前要把数据仓库所有数据迁移到飞天的 SqlEngine 下，并保证目前仓库所有 TCL 程序在 SqlEngine 环境下能够全部运行通过。

1.3. 参考资料

SQLServer 手册

2. 测试概述

产品名称/版本	Sqlengine version 0.8
测试类型	功能测试
测试标准	GB/T 8567-2006
测试方法	功能点全覆盖

测试环境	硬件环境	数据库服务器：7 台 pc server 集群 应用服务器：无 客户端：联想 PC
	软件环境	数据库服务器：SqlEngine version 0.8 应用服务器：无 客户端：Windows Xp SP3 + secure CRT version 6.2, Ubuntu 10.0.4 + 远程桌面查看器 , Red Hat Enterprise Linux Server release 5.4 (Tikanga), tcl 8.4
	测试平台	服务器端采用 7 台 PC Server 集群，单机版客户端 CPU:Pentium® Dual Core CPU E5500@2.80GHz 内存:1.8G

3. 测试方法和范围

3.1. 测试方法

本测试使用功能点全覆盖测试方法对 SqlEngine 使用文档中所列举的功能以及利用 tcl 脚本对阿里适配进行了全面的测试。

3.2. 测试范围

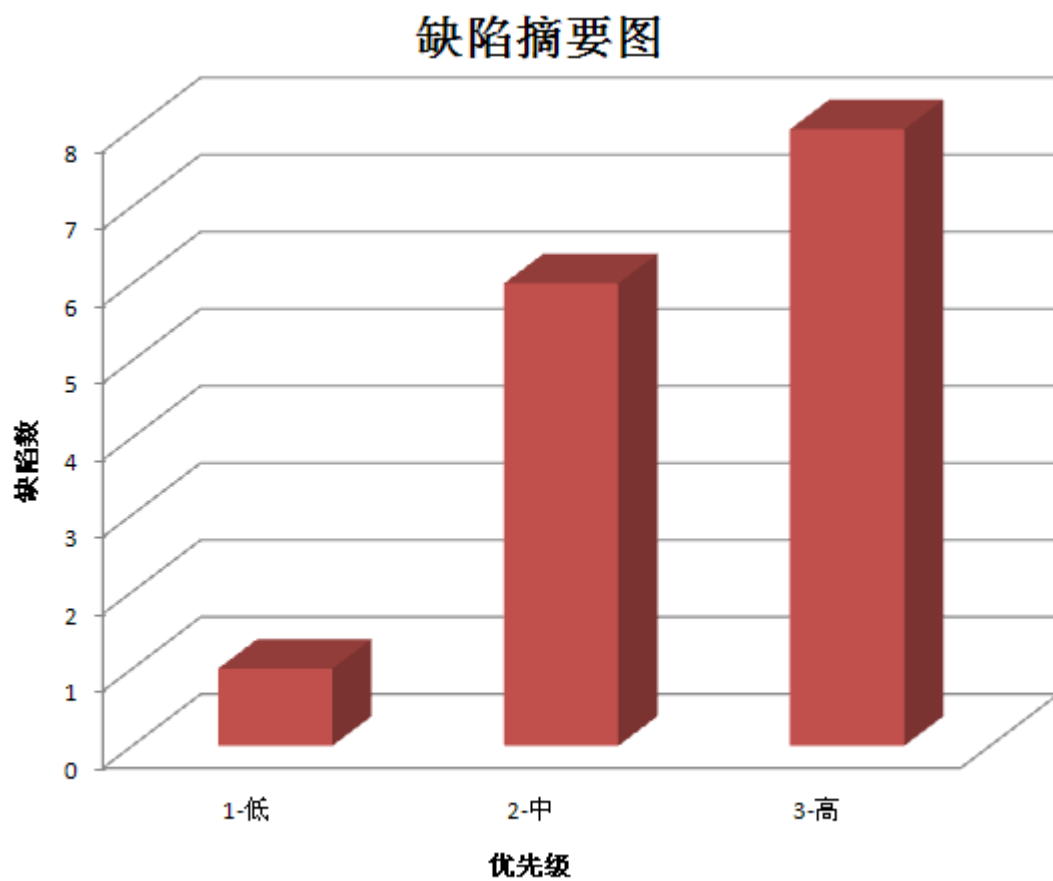
测试项	子测试项	测试项描述	测试对象
功能性测试	准确性	测试 sqlengine 中的所有支持的 SQL 语句能否达到实际的效果	GETDATE,DATEPART,DATETRUNC 等时间函数, REGEXP_SUBSTR 正则表达式等函数 BITAND,BITOR,等数学函数 create table,drop table, select into 等

性能测试	响应时间	在 console 里运行简单的 SQL 测试其从运行到执行结果出来期间所耗时间	Sqlengine offline query
性能测试	并发性	三个用户同时在各自的 sqlengine console 下同时启动运行同一个 SQL	Sqlengine 并发访问同一张表

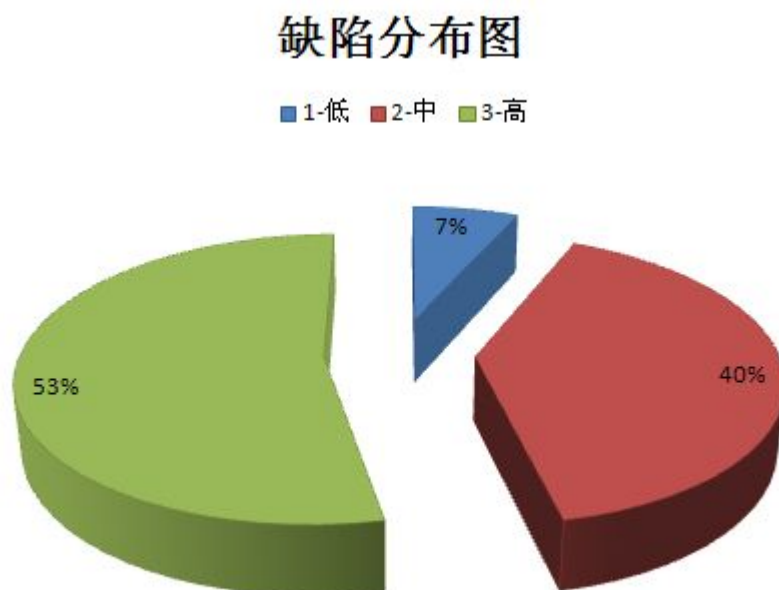
4. 测试结果

测试项	子测试项	测试结果	说明
功能性测试	准确性	截止目前 sqlengine0.8version 共发现 16 个缺陷,其中: 1 个缺陷问题不严重, 7 个缺陷很严重优先高, 8 个缺陷优先中等。	严重的 BUG 直接影响后续的测试
性能测试	响应时间	响应时间较快	简单 select 语句可以在半分钟内完成
性能测试	并发性	三个用户的 console 界面均返回成功	并发性较差, 基本是串行执行任务
备注: 建议对处理 MERGE INTO 语句的相关函数进行重写或者做大的校正			

4.1.1. 缺陷的分布情况图表



图一 缺陷等级 (X 轴) /缺陷个数 (Y 轴)



图二 缺陷等级(低, 中, 高)在所有缺陷中的分布

4.1.2. 功能点测试结果

系统名称	测试功能点（即需求）	测试结果（通过/不通过）
sqlengine	删除表 drop table	通过
	创建表 create table	通过
	SELECT GETDATE() INTO 表名	通过
	Select DATETRUNC() into 表名	通过
	Select DATEPART() into 表名	通过
	Select into 表名（多用户并发访问同一张表）	通过
	Select into 表名（表字段不加别名,但目标表确实创建了）	不通过
	Select BITAND() into 表名	通过
	Select BITOR() into 表名	通过
	Group by	通过
	Having	通过
	On 连接条件	通过
	MERGE INTO	不通过
	TOCHAR（）函数	不通过
	TODATE()函数	不通过
	CHAR（）函数	不通过
	Merge into REGEXP_INSTR	不通过
	Merge into REGEXP_SUBSTR	不通过
	Left join	通过
	Right join	通过
	Full join	通过
	Case when 多层嵌套	通过
	Console 的 Insert TABLENAME	通过
	Console 的 update TABLENAME	通过
	Console 的 source 执行 SQL 文件	通过
	sqlengine_console2 -f test2.sql	通过
	sqlengine_console2 < test.2sql	通过
	Console 的 load_data 命令帮助查看	不通过
	Console 的 load_data select into 导入文本数据	通过
	Console 的 load_data merge into 导入文本数据	通过
	Console 的 export_data 导出数据	通过

5. 结论

注:遗留问题是指在以前的测试报告中已反映了该问题，但在本次测试中发现该问题依然存在

1. 创建删除表

本次测试过程中创建表，删除表都没问题。

2. select into 语句

即使 SELECT INTO 语句执行失败，sqleng 也会生成临时表。

3. 时间函数，正则表达式等

本次测试了 GETDATE,DATETRUNC,DATEADD,TODATE,DATEPART 函数,除了 todate 函数外，其他时间类函数均表现良好

不支持正则表达式和 CAST 函数

4. merge into

本次测试中发现,merge into 可以将数据插入到任意类型的字段中

5.sqlengine console(遗留问题)

load_data 有一个小 BUG，在输入 load_data 时看到的却是 export_data 的帮助

通过此次测试发现了 14 个缺陷，其中 1 个优先级低，8 个优先级高，6 个中等；在测试过程中发现 sqlengine0.8 版本的稳定性较以前版本有了很大的提高。

附件 1 缺陷详细描述

缺陷等级	测试功能点	缺陷描述
3-高	MERGE INTO 类型	该语句在向表中插入数据时不检查字段类型, 可以将数据插入到任意类型字段中
3-高	MERGE INTO TABLE	当 MERGE INTO 的源表和目的表是同一个表时, 无论 SELECT 子句的字段是什么, 结果都是复制源表的所有字段
3-高	MERGE INTO	该语句在插入数据时, 会将上一行数据移到当前行, 然后将待插入的数据放到当前行的行尾, 该语句有严重 BUG, 建议做大的校正或者重写
3-高	类型	在 BOOLEAN 字段上插入字符串, SQLENG 不报错, 且将 BOOLEAN 字段置为 FALSE
3-高	CHAR	该函数不支持 UTF-8, 仅支持 ASCII 码
3-高	TOCHAR	该函数对 double 类型参数会产生四舍五入, 且会截去末尾的不影响参数大小的 0
3-高	CONVERT	该函数得不到正确的结果. 结果集为空
3-高	MOD	不支持 MOD 函数
2-中	DATEDIFF	抛出 char*异常
2-中	CAST	CAST 函数的参数中的 AS 和 SQL 语句中的 AS 重合
2-中	ROUND	如果当前小数点后有 A 位, 无法精确到小数点后 B 位, 其中 $B > A$
2-中	正则表达式 (REGEXP_LIKE, REGEXP_INSTR, REGEXP_SUBSTR, REGEXP_REPLACE)	该类函数均通过语法检查, 但执行结果却返回 failed
2-中	DATASTAMPPART	该函数实现了其承诺的功能, 但当传递的参数错误时, 控制台依然返回成功而不是 failed
2-中	TODATE 函数	format 选用 YYYY-MM-DD 或者 YYYY-MM-DD HH:MI:SS 格式依然会以 YYYY-MM-DD HH:MI:SS. XXX 格式显示数据
1-低 (遗留问题)	控制台 load_data 命令	在控制台下输入 load_data 命令显示的却是 export_data 的帮助信息。

附件 2 缺陷等级的描述

等级	描述	说明
5-紧急	发现可重复出现的致命问题	——导致系统崩溃； ——导致程序模块丢失； ——主业务流程出现断点； ——内存泄漏； ——导致死机
4-非常高	发现可重复出现的严重问题	——被测功能不能正确实现； ——软件错误导致数据丢失； ——被测数据处理错误； ——用户需求未实现。
3-高	一般性的错误或功能实现有 不完美处	——操作界面错误； ——打印内容、格式错误； ——简单的输入限制未放在前台进行控制； ——删除操作未给出提示。
2-中	细小的错误	——界面不规范； ——辅助说明描述不清楚； ——输入输出不规范； ——长操作未给用户提示； ——提示窗口文字未采用行业术语。
1-低	建议类错误	需求说明书、用户手册中未说明，但影响用户对软件使用的方便性等