文章编号:1001-9081(2011)09-2449-04

doi:10.3724/SP. J. 1087.2011.02449

TPC-DS 性能测试工具的实现

陈 旦,叶晓俊,施 霖

(清华大学 软件学院,北京 100084)

(thsschendan@gmail.com; yexj@tsinghua.edu.cn)

摘 要:阐述了新一代面向决策支持的数据库性能测试基准 TPC-DS 的数据模型、业务模型、执行模式和度量方法,设计了一个支持不同数据库管理系统(DBMS)性能对比测评的 TPC-DS 测试框架,阐述了配置文件、查询执行控制、数据维护机制等关键实现技术。最后通过对任务配置和语法配置的实践,验证了工具对不同 DBMS、不同优化配置的对比测试可用性。

关键词:数据库管理系统;决策支持系统;数据库测试;TPC-DS 性能基准;测试工具中图分类号: TP311.13;TP311.52 文献标志码:A

Implementation of performance testing for TPC-DS benchmark

CHEN Dan, YE Xiao-jun, SHI Lin

(School of Software, Tsinghua University, Beijing 100084, China)

Abstract: The data model, business model, execution schema and performance metric of TPC-DS benchmark for next generation Decision Support System (DSS) application performance evaluation were introduced. The implementation architecture and key technologies for a configurable TPC-DS performance testing tool were put forward, including configuration file, query execution control and data maintenance mechanism. By testing practices in different Database Management Systems (DBMSs), the configurability and usability of the proposed tool for implementation strategies were verified.

Key words: Database Management System (DBMS); Decision Support System (DSS); database test; TPC-DS benchmark; testing tool

0 引言

数据仓库是决策支持系统(Decision Support System, DSS)的重要组成部分,其核心组件——数据库管理系统(Database Management System, DBMS)的性能解决方案(如物化视图、分区表、位图索引等)及商务智能套件(如 OLAP, ETL)影响 DSS 的可用性。因此事务处理性能委员会(Transaction Processing Performance Council, TPC)已制定了一系列面向决策支持系统的数据库性能基准[1]。

目前在用的 TPC-H 基准已很难模拟日益复杂的 DSS 业务需求。例如 TPC-H 的数据模型是第三范式,不是目前普遍采用的星座模型,它的业务类型(22 个查询)很难体现位图索引、物化视图等现代 DBMS 查询引擎的优势,而且它的数据表数据特征单一(如数据不倾斜),其数据维护功能(rfl, rf2)仅仅限制了潜在的对索引的过度使用,而没有测试 DBMS 执行真实数据维护操作——数据提取、转换和加载(Extraction-Transformation-Loading, ETL)功能的能力^[2]。为此,TPC 组织在推出了新一代的面向决策应用的 TPC-DS 基准^[3]。

TPC-DS 基准将帮助加快硬件和数据库技术的发展来鼓励研究和发展海量数据查询的优化技术。TPC-DS 也会被硬件厂商和数据库厂商使用来说明他们产品对 DSS 的支持能力,被客户作为某个购买决定的重要因素之一。因此,如何基于 TPC-DS 测试 DBMS 对 DSS 应用的支持能力,衡量其数据仓库解决方案及其各种技术的可用性等,需要一个满足不同DBMS 性能评测的支持工具。本文结合笔者承担的国产DBMS 对比测评需求,介绍了基于 TPC-DS 基准实现的性能测试框架,给出了基于这个基准的实验验证测试方法。

1 TPC-DS 基准

TPC-DS 基准意图提供一个公平和诚实的业务和数据模型,以衡量 DBMS 系统的整体性能,如 CPU 与内存使用,I/O 负载特征以及操作系统和 DBMS 完成海量数据复杂计算等。它主要关注^[3]:

- 1)共享维度的多雪花模式:24 个平均含有 18 列的数据 库表及丰富的主外键约束。
- 2) 随机替换的 99 个不同的 SQL 查询: 随机的、报告的、 迭代的、抽取的查询。
- 3) 更具代表性的不对称的倾斜的测试数据: 非事实数据 表的子线性比例划分。
 - 4)类 ETL 数据维护等。

1.1 数据模型

TPC-DS 基准模拟零售企业三种销售渠道(店铺、网络、目录)的销售和退货业务,包含了客户和典型销售渠道的产品数据订单等。基准设计了7张事实表、17张维度表和104个保持数据完整性的外键约束条件。7张事实表为:3种销售渠道的销售和退回业务数据表以及库存销售业务数据表。

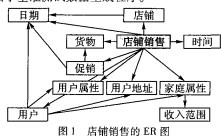
TPC-DS 基准采用雪花模型和星座模型混合方式来表示被测数据仓库中事实和维的关系。图 1 展示了店铺销售的雪花模型,店铺销售为事实表,日期、时间、店铺、商品、用户和促销等为维度表,用户统计、用户地址、家庭统计和收入范围对上述纬度表进行层次和粒度上的分解,这样就形成了以店铺销售事实表为中心的雪花模型。

基准的测试数据集规模通过离散的比例因子分 7 级。数据集内的比例划分有两种不同的特点:一种是数据集中的元

收稿日期:2011 - 03 - 21;修回日期:2011 - 05 - 24。 基金项目:国家"核高基"科技重大专项(2009ZX01045-004-001-03)。

作者简介:陈旦(1989-),男,江西南昌人,硕士研究生,主要研究方向:数据库安全、数据库测试; 叶晓俊(1964-),男,江苏泰兴人,教授,主要研究方向:数据库测试; 施霖(1981-),男,福建泉州人,硕士研究生,主要研究方向:数据库测试。

组数量是扩展的,但是其对应的值的集合(域)是静态的;另 一种是元组的数量是固定的,但是产生元组的值的域是扩展 的。TPC-DS 基准选择使用混合的方法。很多表的列都采用 数据集比例划分而不是域的比例划分,尤其是事实数据表。 TPC 给出了基准测试数据生成程序。



1.2 业务模型

TPC-DS 基准提供了两种重要的业务模型:用户查询和数 据维护。查询是把操作性的事实转化为商业情报,而数据维 护操作是将数据仓库数据与操作数据库进行同步。

TPC-DS 基准业务[3] 如图 2 所示:1) 基准提供了被测试数 据库测试数据生成程序生成测试数据文本文件,测试人员通 过使用 ETL 工具将其装载到被测 DBMS;2) 基准提供 99 个查 询模板及查询语句生成程序,这些查询模板模拟迭代的 OLAP 查询、数据挖掘工具的抽取查询、即席(ad-hot)查询以 及大量定制的常见的报表查询;3)基准提供了5种方法模拟 现实应用的 ETL 过程,包括不需保留历史记录的维度数据维 护,需保留历史记录的维度数据维护,事实表数据插入和数据 删除维护以及库存表数据删除维护。

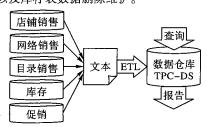


图 2 TPC-DS 基准业务

1.3 执行模式

TPC-DS 测试分为:测试数据加载、查询顺序执行(Power) 和并行执行(Throughput)测试。测试数据加载主要包括:被 测系统准备、数据文件生成、测试数据库创建、基础表创建、数 据加载、约束验证、辅助数据结构(如索引)创建、表和辅助数 据统计分析等。Power 测试是用于评测数据库对单个查询流 的处理能力。Throughput 测试是用于测试 DBMS 对多个查询 流并发查询和操作的处理能力,分为数据查询和数据维护各 两个子步骤。所以一个完整的 TPC-DS 测试执行模式如图 3 所示。

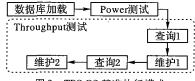


图 3 TPC-DS 基准执行模式

数据查询是 TPC-DS 基准测试的核心。在 Power 测试中, 99个数据查询形成的查询流只执行一次,用于评测数据库对 单个查询流的处理能力。在 Throughput 测试中,数据查询执 行两次,每次执行至少20个以上的并行查询流,模拟多个用 户同时对数据库的查询操作。为了适应多查询流,数据库优 化组件必须依据数据规模进行规划,每个用户也要有足够的 临时空间来保存大量的中间结果集。这样 DBMS 能找出并行 用户的最佳执行计划。表 1 展示了 TPC-DS 基准查询所使用 的 SQL 特征及数量。

表 1 TPC-DS SQL 查询特征

查询数量
31
15
76
78
64
9
11
5
17
2
1
24
5

1.4 度量方法

TPC-DS 基准定义了两个评价指标:1)反映每秒的有效查 询数据量的性能指标,其计算公式为: QphDS@ SF(SF 是数据 库规模因子,单位为 GB,最小为 100 GB),值越大说明性能越 高;2)反映每秒每查询数据量的性价比指标,值越小说明性 价比越高,其计算公式为\$/QphDS@SF。

1.5 测试框架

基于 TPC-DS 对 DBMS 进行对比测试需要支持的工具应 具备测试环境、连接方式、SQL 语句等可配置能力。图 4 是一 个可配置 TPC-DS 测试工具功能结构示意图,包括面向测试 管理的功能界面、DBMS 功能正确性验证以及覆盖 TPC-DS 规 范的性能测试功能。例如数据库加载是先使用 TPC 组织提 供的 DBGEN 程序生成测试数据,再采用数据库的 ETL 工具 将其装载到测试数据库;数据查询是使用 TPC 组织提供的 QGEN 程序把查询模板中的语句转化为可执行查询语句,并 通过查询解析(parse_query),生成待提交的查询语句;数据维 护阶段,DBGEN程序生成更新数据文件,并通过自定义的更 新脚本/函数进行数据维护。

- 界面功能 —————						
界面	通信	配置				
参数配置	发送	任务				
状态显示	请求	配置				
错误提示	返回 结果	语法 配置				

正确性验证功能

查询验证	事务特性测试
装载1GB	原子性测试
验证数据	一致性测试
执行查	隔离性测试
询验证	持久性测试

C TPC-DS基准性能测试功能

数据库加载	数据查询	数据维护	性能计算	日志管理
生成文	生成查	生成更	性能计算	记录执
本文件	询文件	新文件		行步骤
读取配	解析查	加载更		记录耗
置参数	询语句	新数据		用时间
加载	执行	数据	报告 生成	记录错
数据	查询	维护		误信息

图 4 TPC-DS 测试框架

2 测试工具实现

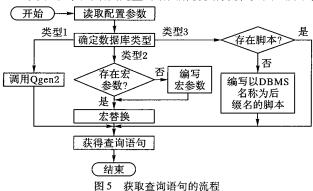
基于基准的对比测试是指依据上述要求设计测试程序, 搭建测试环境进行性能测试,把测试结果与相同测试程序的 运行结果相比较的整个过程。

2.1 测试环境配置设计

TPC-DS 基准测试环境配置分为任务配置和语法配置。 任务配置用于维护不同被测 DBMS 测试任务参数,例如数据 库连接、数据加载脚本和维护脚本的地址等,配置参数以XML方式存储在外部文件中。语法配置用于维护不同 DBMS的 TPC-DS 数据库定义和查询执行特有的 SQL 语法特性。在执行测试时,测试工具根据任务配置中的数据库类型,从语法配置中获取对应的查询文件。通过对 TPC-DS 基准的查询语句进行分析和归纳,我们将语法配置分为三个抽象类型:

- 1) TPC-DS 基准支持的 DBMS 和标准 ANSI 类型 SQL 语句: Driver 直接使用相应的参数调用 TPC 组织的 QGen2 程序生成查询文件。
- 2) 部分通用的 SQL 语句: 针对不同 DBMS 而定制的宏参数,测试时通过语法配置引擎的宏替换,即可得到针对不同数据库的查询文件。
- 3)不通用的 SQL 语句:语法配置引擎通过文件名后缀的方式,分别加载具有针对性的查询文件。

本测试工具对语法配置的语法解析执行流程如图 5 所示。



2.2 体系结构设计

测试工具体系结构如图 6 所示。被测系统(System under test, SUT)包括硬件设备(如服务器、网络设备)和软件设备(操作系统、DBMS 以及连接 Driver 的一系列软件)。测试程序(Driver)分三层:展示层负责 Driver 客户端与用户之间的交互,用户通过图形界面进行任务和语法的参数配置,测试时Driver 客户端向用户展示测试进度和测试结果,最终生成测试报告;业务层负责 TPC-DS 基准定义的性能测试业务处理;基于 JDBC/ODBC 技术的控制层实现与多种 DBMS 数据库间的相关操作。

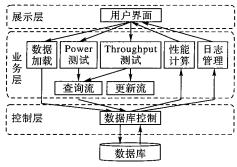


图 6 TPC-DS 测试工具体系结构

2.3 实现关键技术

2.3.1 配置文件描述

任务配置和语法配置文件采用 XML 格式进行描述。例如语法配置主要包含不同数据库的语法参数,如查询前n行记录、日期计算、类型转换、字符串处理等。

语法配置示例如下:

- < Grammar replace = "_LIMITA" > select * from(< /Grammar >
- < Grammar replace = "_LIMITB" > </Grammar >
- < Grammar replace = "_LIMITC" >)

where rownum <= % d </Grammar >

```
< Grammar replace = " cast(' [ SALES_DATE] ' as date) " >
to_date(' [ SALES_DATE] ', ' yyyy - mm - dd ')
```

2.3.2 查询执行控制

查询前,调用 QGen2 程序生成与用户数等量的查询文件。 工具采用多线程模拟多用户并发访问,这样,每个线程执行一个查询文件,每个查询文件包含 99 个查询语句。为防止 SQL 语句及数据被数据库缓存,查询语句绑定不同变量且每次调用的次序都是随机生成的。基准要求记录每个线程中每个查询语句的执行时间,因此执行过程中会对查询文件进行解析,分解出每条查询语句并记录查询语句的编号和执行时间。

通过对 TPC-DS 生成的查询文件的分析发现,查询语句都是以"start query"开始、"end query"结尾,且注释部分带有模板编号。因此工具实现时以这两个标志词为分隔符,从查询文件中分解出查询语句,并提取注释中的模板编号作为查询语句的编号,将查询执行的结果写到跟踪日志文件中,以便于后面的统计分析。

查询语句示例如下:

 $--\,\text{start}$ query 1 in stream 0 using template query 96. tpl

select * from(select count(*)

from store_sales

- , household_demographics
- . time dim. store

where ss_sold_time_sk = time_dim. t_time_sk

and ss_hdemo_sk = household_demographics. hd_demo_sk

and ss_store_sk = s_store_sk

and time_dim. t_hour = 8

and time dim. t minute > = 30

and household_demographics. hd_dep_count = 2

and store. s_store_name = 'ese'

order by count(*))

where rownum <= 100;

-- end query 1 in stream 0 using template query96. tpl

2.3.3 数据维护机制

TPC-DS 基准提供了 5 种数据维护方法及其伪代码。在实现时,依据 DBMS 的 ETL 接口将伪代码转化为针对具体 DBMS 的可执行脚本放到工具语法配置文件指定的子目录即可。

为实现测试工具的通用可配置,在测试前首先在任务配置中定义调用脚本的地址,然后编写针对具体 DBMS 的数据维护脚本,测试时测试工具会自动调用配置的脚本文件进行数据维护。调用维护脚本流程如图 7 所示。

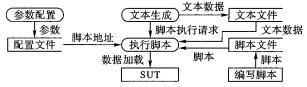


图 7 数据加载或维护数据流图

2.3.4 异步通信机制

在数据库测试过程中,数据库初始化、复杂查询、数据更新等都是一些非常耗时的操作,不能以阻塞方式执行,否则就会导致用户界面死锁。工具通过建立命令队列,引入异步通信机制。这样,在初始化时建立一个调度器,在收到异步调用请求后,把调用参数放入调度器的命令队列中,然后就可返回。此时调度器的命令队列处理流程被激活,开始处理队列中的请求,当耗时的请求被处理完毕时,调度器把处理结果再反映到用户界面上。

2.3.5 查询正确性验证

查询验证的目的是验证性能测试时使用的 DBMS 功能是

正确的,这也是性能对比测试的前提。查询验证流程如下:

- 1)测试数据库生成:使用 DBGen2 生成1 GB 验证数据, 并将其加载到测试数据库中。
- 2)功能正确性验证:使用 QGen2 生成 DBMS 相关的 99 条 SQL 查询验证语句,在测试数据库上执行上述语句,比较执行结果和预期结果的一致性。
- 3)事务特性验证:依据基准要求执行被测 DBMS 的 ACID 事务特性测试程序。

3 性能测试实验

数据库性能基准测试一般是指检测、测量一个 DBMS 可以达到的最佳性能。本实验旨在验证测试工具的可配置性。首先测试任务和语法的可配置性,即对待测 DBMS 执行查询验证,验证待测 DBMS 满足正确性要求和 ACID 特性;然后测试工具的可用性,即分别从不同的 DBMS 和同一 DBMS 的不同配置两个方面进行可用性测试。

3.1 可配置 DBMS 实验

对三种主流 DBMS(Oracle 10g R2、DB2 9.7、SQL Server 2005)进行性能测试,验证工具的可配置性。测试采用曙光服务器(Intel(R) Xeon(R) CPU X7460 @ 2.66 GHz,6 核 ×4;32 GB 内存;450 GB × 16 块硬盘; Windows 2008 操作系统),数据库规模因子为 100 GB,20 个并发用户,DBMS 均采用默认配置。实验结果如图 8 所示。

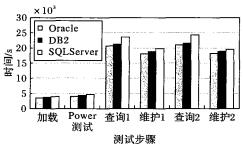


图 8 可配置实验结果

3.2 DBMS 优化技术实验

对同一 DBMS(实验采用 Oracle 10g R2)的不同配置进行性能测试。实验采用三种不同的配置:1)默认配置;2)对7张事实表进行水平分区;3)对7张事实表进行压缩存储,并使用水平分区和压缩存储优化技术后的性能测试。测试环境同3.1节。实验数据如图9所示,从实验结果可以看出,使用优

化技术后(除了使用压缩存储技术后数据加载和维护耗时变长外)耗时普遍减少,数据库性能得到提升。

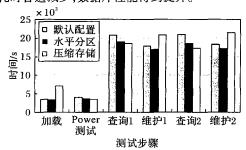


图 9 DBMS 优化实验结果

4 结语

本文通过对 TPC-DS 基准的研究和分析,阐述了通用可配置的 TPC-DS 性能测试工具的设计思想与实现技术,并通过对不同的 DBMS 和同一 DBMS 在不同配置下的性能测试,验证了测试工具的可用性。

为保证工具在国产数据库对比测试中的可用性,今后的工作包括:1)分析研究商业 DBMS 的决策支持优化技术,包括面向应用的 ETL 技术,并基于测试工具进行实践,为企业选择优化技术提供数据参考;2)基于多维模型的在线数据分析已经集成多商业 DBMS 中,如何对 MOLAP 引擎性能进行评测,是我们扩展 TPC-DS 物理数据存储模型和基于多维查询语言的业务模型实现努力的方向。

参考文献:

- [1] NAMBIAR R O, LANKEN M, WAKOU N, et al. Transaction Processing Performance Council (TPC) twenty years later—a look back, a look ahead [C]// TPCTC 2009: First TPC Technology Conference on Performance Evaluation and Benchmarking, LNCS 5895. Berlin: Springer-Verlag, 2009: 1-10.
- [2] NAMBIAR R O, POESS M. The making of TPC-DS [C]// Proceedings of the VLDB. [S.1.]: VLDB, 2006: 1049 1058.
- [3] DRAFT Specification Revision 32, TPC BENCHMARK DS [S/OL]. [2011 02 16]. http://www.tpc.org/tpcds/spec/tpcds32.pdf.
- [4] POESS M, NAMBIAR R O, WALRATH D. Why you should run TPC-DS: a workload analysis [C]// Proceedings of the VLDB. [S. l.]: VLDB, 2007: 1138-1149.
- [5] 施霖. TPC-DS 基准测试工具实现[D]. 北京: 清华大学, 2010.

(上接第2443页)

参考文献:

- [1] BOURHFIR C, DSSOULI R, ABOULHAMID E, et al. Automatic executable test case generation for extended finite state machine protocols[C]// IWTCS'97: International Workshop for Protocol Test Systems. Cheju Island, Korea: [s. n.], 1997: 75 - 90.
- [2] KALAJI A, HIERONS R M, SWIFT S. Automatic generation of test sequences form EFSM models using evolutionary algorithms [R]. London: Brunel University, School of Information Systems, Computing and Mathematics, 2008.
- [3] 易国洪,王渊峰. 基于 EFSM 模型的等价类测试[J]. 计算机科学,2007,34(1):281-284.
- [4] 舒挺, 魏仰苏, 吴柏青, 等. EFSM 可执行状态验证序列的生成 [J]. 北京邮电大学学报, 2007, 30(2): 84-88.
- [5] LEFTICARU R, IPATE F. Automatic state-based test generation using genetic algorithms [C]// SYNASC '07: Proceedings of the Ninth International Symposium on Symbolic and Numeric Algorithms for Scientific Computing. Washington, DC: IEEE Computer Society, 2007: 188-195.

- [6] ZHAO R, HARMAN M, LI Z. Empirical study on the efficiency of search based test generation for EFSM models [C]// ICSTW: Software Testing, Verification, and Validation Workshops. Washington, DC: IEEE Computer Society, 2010: 222 - 231.
- [7] YU B, QIN Y M, LI F, et al. Tabu search and genetic algorithm for generating test data of class testing [C]// International Conference on Information Engineering and Computer Science. Piscataway, NJ: IEEE, 2009: 1-6.
- [8] 张涌, 钱乐秋, 王渊峰. 基于扩展有限状态机测试中测试输入数据自动选取的研究[J]. 计算机学报, 2003, 26(10): 1295-1303.
- [9] 端木传毅, 邓洪敏, 胡晓勤, 等. 基于双价值函数的测试用例生成技术[J]. 四川大学学报: 自然科学版, 2010, 47(1): 61-66.
- [10] DIAZ E, TUYA J, BLANCO R. Automated software testing using a metaheuristic technique based on tabu search [C]// Proceedings of 18th International Conference on Automated Software Engineering. Washington, DC: IEEE Computer Society, 2003: 310-313.
- [11] MCMINN P. Search-based software test data generation: a survey: research articles [J]. Software Testing-Verification and Reliability, 2004, 14(2):105-106.