计算机研究与发展
Journal of Computer Research and Development
ISSN 1000-1239/CN 11-1777/TP
47(Suppl.): 433-437, 2010

# TaijiDB:一个双核云数据库管理系统

胡享梅 赵 婧 孟小峰 王仲远 史英杰 刘兵兵 王海平
(中国人民大学信息学院 北京 100872)
(hxm2008@ruc.edu.cn)

# TaijiDB: A Dual-Core Cloud-Based Database System

Hu Xiangmei, Zhao Jing, Meng Xiaofeng, Wang Zhongyuan, Shi Yingjie, Liu Bingbing, and Wang Haiping
(*School of Information, Renmin University of China, Beijing* 100872)

**Abstract** Taiji is a Chinese cosmological term, which means two modes can be uniform relatively. In order to leverage the advantages of cloud storage based on master-slave and p2p structures, we propose a project called Taiji, which is a dual-core cloud-based database system. This system can support SQL to manage the Big Data in the cloud.

**Key words** cloud computing; cloud-based database system

**摘 要** 太极是一个中国古代哲学术语——即两种模式可以相对统一.利用基于云存储的主从结构和点对点结构各自的优点,融合两种结构,构建了一个双核的云数据库管理系统——太极.系统支持使用SQL语言对云数据库系统中的海量数据进行管理.

**关键词** 云计算;云数据库系统

**中图法分类号** TP311.13

随着数据量的迅猛增长,如何存储和管理海量复杂数据已成为一个亟待解决的挑战性问题.云计算应运而生,它改变了数据存储的基础架构.现有的云计算系统包括:亚马逊的弹性云计算(EC2)[1]、IBM的蓝云[2]和谷歌的GFS[3].它们都采用了弹性资源管理机制并提供很好的可扩展性.另外,也有一些开源项目,譬如Apache Hadoop项目的HDFS[4]和HBase[5]以及Cassandra[6].HDFS和HBase是谷歌GFS和BigTable[7]的开源实现,Cassandra则是亚马逊Dynamo[8]分布式实现和Bigtable列簇数据模型的融合.

云计算系统通常有2种底层结构:主从结构和点对点结构.表1展示了基于上述两种结构的云计算系统在各方面的对比.

表1 基于主从与点对点结构的云计算系统比较

| | 主从结构 | 点对点结构 |
|---|---|---|
| CAP[9] | 通常关注于一致性和高可用性 | 通常关注于可用性和划分容错性 |
| 数据写操作 | 如果Region服务器意外停机,则在数据重新分布前,写操作会被阻止 | 每个节点都是平等的,因而"写操作永远不会失败" |
| MapReduce | 支持MapReduce框架 | 不支持MapReduce |
| 系统性能 | Master节点可能会成为瓶颈 | 在通信负载较大时,系统性能会迅速下降 |
| 应用场景 | 适合分析型数据管理应用 | 适合事务型数据管理应用 |

收稿日期:2010-06-25
基金项目:国家自然科学基金项目(60833005,60573091);国家"八六三"高技术研究发展计划基金项目(2007AA01Z155,2009AA011904);教育部博士学科点专项科研基金项目(200800020002)

从表 1 中可看出,这 2 种结构各具优势,但也存在自身结构性问题.因此,我们希望构建一个混合系统,充分利用上述 2 种底层结构的优势.

在另一方面,除了 HIVE[10] 系统,现有的云计算系统都不支持 SQL 语言,这就意味着原有的基于传统 DBMS 的数据管理系统无法平滑地迁移到云存储平台上,进而导致云存储无法大规模普及.同时,HIVE 系统仅支持较为简单的类 SQL 查询语句.因此,我们需要在云数据管理系统中解决这一挑战性问题.基于上述原因,我们提出构建一种"双核"的基于云的数据库管理系统.

太极是中国古代哲学术语,它包含阴阳两种状态.太极具有两层意思:一方面,阴阳两种状态相对统一;另一方面,阴阳轮转,派生万物,从混沌变为清晰.我们认为这两个特点与系统的目标极为相似:首先,我们希望结合不同架构,利用各自优势;其次,我们需要管理海量的纷繁复杂的数据,使之有条有理,便于查询.因此,我们以太极来命名我们的工程.

我们的贡献主要包括:1)太极建立了一个适配层来桥接底层的主从结构和点对点结构.上层通过统一的 API 调用进行操作,底层的差异对上层透明.2)太极支持部分 SQL 查询语句,包括建表、插入数据、选择数据、删除数据和数据表等.对 SQL 语句的支持有助于降低应用开发的难度,从而使云计算和云服务等到更好的推广和应用.

## 1 应用场景

基于云的数据库系统旨在支持下一代数据存储和管理.图 1 展示了移动应用领域中 WAP 应用框架.移动终端通过 WAP 服务器访问网络站点.WAP 服务器产生话单数据并通过 FTP 传输到话单

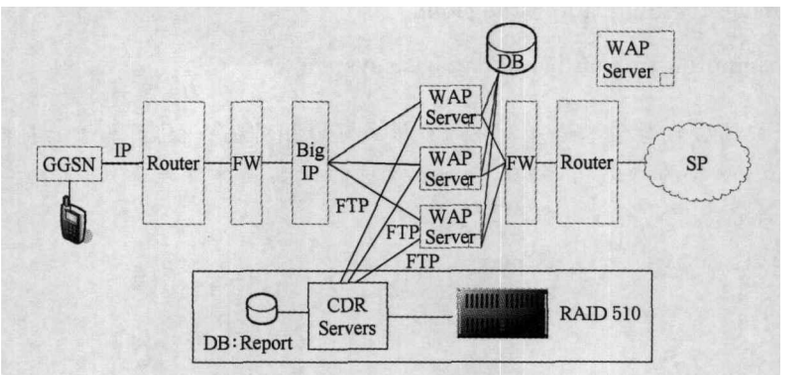


图 1 WAP 应用体系结构

数据服务器.话单数据在通信过程中产生,用于描述通信细节.该数据主要面向两类应用:1)个人通信详情查询;2)数据分析及决策支持.话单数据的规模巨大:每 5 min 产生的数据量近 4.5 GB.随着手机用户数量的增长,在不久的将来将会产生越来越多的话单数据.

如何存储和管理这些数据将是一个极具挑战性的难题.目前各数据中心普遍采用昂贵的高端数据服务器进行集中式存储,因此企业只能在磁盘中保留最近 3 个月的话单数据.

话单数据适于存储在基于云的 DBMS 中,基于云的 DBMS 可以部署在廉价、低端并且自适应的硬件上.为了满足数据导入和用户查询的需求,话单数据管理系统在数据读和写操作上都必须有高性能.

我们使用太极获取来自 WAP 服务器的话单数据并进行分析,分析结果交由报表数据库存储.话单数据表结构如表 2 所示:

表 2 话单数据结构表

| 属性 | 类型 |
|---|---|
| *service_ID* | string |
| *service_type* | int |
| **MSISDN* | string |
| *IP_address* | string |
| **ts_start* | datetime |
| *ts_end* | datetime |
| *ts_start_content_fetching* | datetime |
| *ts_edn_content_fetching* | datetime |
| *session_or_ID* | int |
| *fetched_URL* | string |
| *online_time* | long |

若用户想查询一天之中访问过的所有 WAP 站点,我们可通过执行如下 SQL 查询语句获得结果:

SELECT *MSISDN*,*ts_start*,*ts_end*,*fetched_URL*

FROM *COR*

WHERE *MSISDN*='1395451XXXX'

and *ts_start*≥'2009-09-15 00:00:00.000'

and *ts_end*<'2009-09-16 00:00:00.000'.

另一方面,若运营商需要统计一个 URL 每天被访问的时间总和,可执行以下 SQL 语句获得结果:

SELECT *fetched_URL*, *sum*(*ts_end* − *ts_start*)

FROM *CDR*

WHERE *ts_start* ≥ '2009-09-15 00:00:00.000'

and *ts_end*<'2009-09-16 00:00:00.000'

GROUPBY *fetched_URL*.

为了满足该应用场景的所有需要,我们设计了太极数据库管理系统.太极的框架主要包括 3 层:数据存储层、查询处理层和应用层,如图 2 所示:

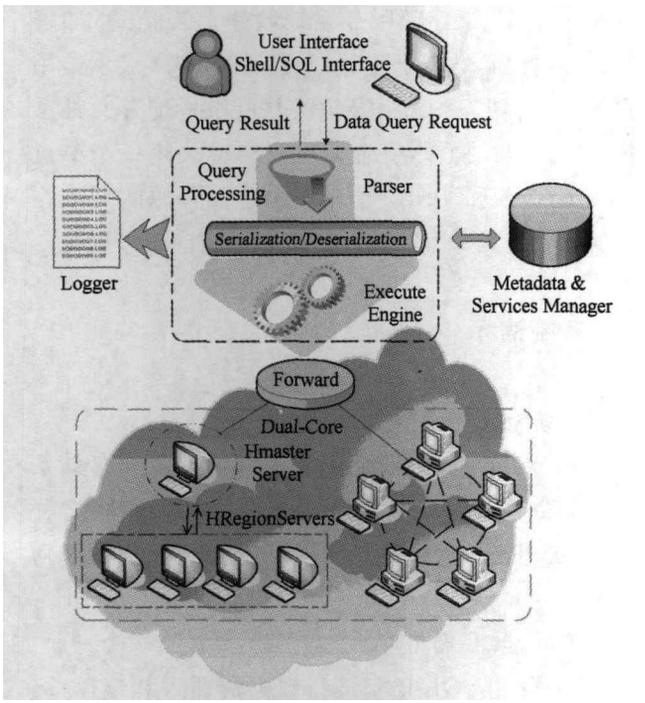


图 2 太极的框架图

顶层是应用层.太极支持 SQL 以便于复杂的数据管理,同时便于应用开发商实现其服务向云计算的无缝迁移.该层通过查询语言以及丰富的 API 支持多种 Web 应用.客户端可通过 shell 接口或 SQL 接口提交用户查询.

中间层是查询处理层.太极通过该层将 SQL 语句解析为原子操作序列,序列化或反序列化数据并调用执行引擎来完成操作.同时,该层包含元数据和服务管理器,日志等用于监控云数据库系统的状态.

底层是双核存储层.通过可配置的存储架构为用户提供灵活的存储管理数据方式.底层可使用主从和点对点的存储结构,综合二者的优点,为上层提供统一的 API 实现.该层支持云存储的特性:备份、并行性、容错性、主键划分和同步.

太极为云数据管理提供强大的双核模型,并为云上的应用开发提供便利的方式——标准的 SQL 支持.同时,充分利用双核设计的优势和根据应用需求(如事务、一致性和负载均衡)来自动选择合适的架构都极具研究意义,这将在本文的第 4 部分讨论.

## 2 系统架构

图 3 显示了太极的组成部分以及 Hadoop 和 Cassandra 的通信.太极主要包括 5 个组成部分:

1.前端接口模块."双核"云数据库管理系统提供 SQL 接口、Shell 和应用程序编程接口(API).用户不仅可通过 SQL 接口得到记录形式的结果,还可以执行文件级别的操作,如从文件中进行数据载入,以及使用 API 接口将数据导出到文件中.

2.查询处理模块.为前端接口提供两种查询接口:基于 SQL 的查询接口和基于编程的查询接口.当 SQL 语句被提出时,SQL 处理器进行 SQL 解析和查询优化,并且将 SQL 语句翻译成命令,调用统一的 API 和存储层进行通信.如果编程应用接口被用户调用,它们也会依照客户端库翻译成执行计划进行执行.

3.统一执行引擎模块.为上层提供统一的应用,而无需关注两种存储模式的不同处理方法.它从统一 API 提取标识参数,并通过 API 触发存储管理器.

4.存储管理器模块.负责控制数据存储位置.用户可以指定使用 HBase 或者 Cassandra 作为存储引擎.

5.运营维护模块.存储数据库元信息.元信息被 SQL 处理器和驱动器使用,还可用于监控系统的操作和运行状态.

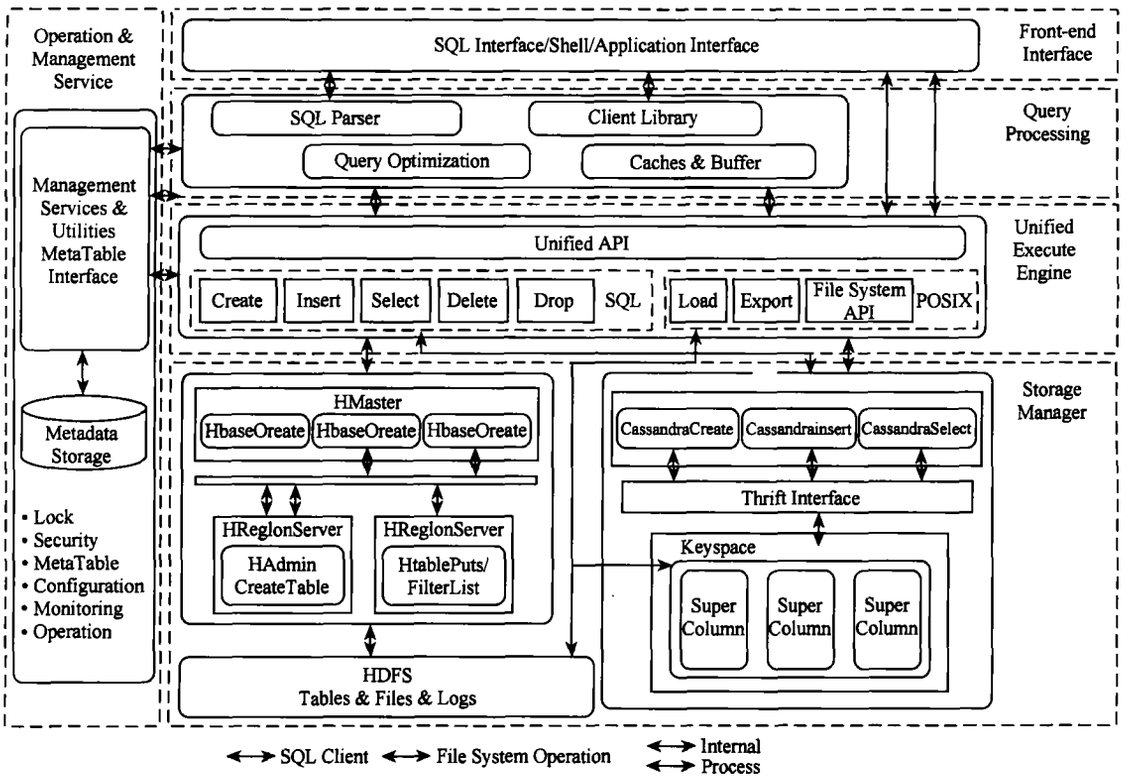


图 3 "双核"云数据库管理系统的体系结构

### 2.1 存储管理模块

太极拥有双核存储引擎. 它支持这些特性:备份、并行、容错、键值划分和同步. 在此部分中,有两种存储模式:主从模式和点对点模式. HBase 以主从模式组织,主服务器管理文件系统命名空间,控制用户的文件访问. 集群的数据节点负责执行具体读写操作,同时根据主节点的命令进行数据块的创建、删除和复制. Cassandra 基于点对点结构,使用 Gossip 协议管理集群成员. 在主从模式中,我们为一张表使用一个列族,表中的每一列与列族中一个限定词匹配. 在点对点模式中,整个数据库是一个关键词空间,表对应超级列,超级列下的列与用户表的属性列相对应. 在"双核"云数据库管理系统中,由于 Cassandra 无法动态添加列族,我们并未使用列族表示一张表.

### 2.2 运营维护模块

在"双核"云数据库管理系统中,运营维护模块负责元数据管理、操作管理和系统监控. 系统包含两种元数据:1)表结构信息,如表名、字段名、字段类型、表存储等;2)用户信息,如用户名、密码、权限等. 由于元数据规模有限且多服务于事务操作,我们采用 RDMBS——MySql 存储. 我们通过元数据接口对操作元数据. 管理服务系统收集监控信息,如资源、系统健康、数据配置等,并且通过 GUI 展示给用户. 同时它具备系统报警功能并可重置配置文件参数. 通过 GUI 操作,管理员可开启或关闭每个节点上的 DBMS 和 OS,创建基于角色的资源队列或在集群中动态地添加(或删除)节点以适应负载变化.

## 3 系统演示

系统演示主要包含以下 3 部分:

1. "双核"存储. 我们将展示太极的双核存储系统的有效性. 应用可在主从结构和点对点结构 2 种模式之间进行平滑地切换,无需对 API 调用做任何修改.

2. 功能. 我们将演示的功能主要包括 2 个方面:1)建表、使用 SQL 语句进行数据的插入和选择;2)通过 API 在系统和文件之间进行数据的导入和导出.

3. 性能. 我们将演示太极在电子通信场景下的应用. 我们使用话单数据作为我们的测试实例. 演示使用的话单数据超过 3.8TB,分布存储在 20 个节点上.

## 4 未来工作

太极目前正被应用在在电信领域管理不同类型的话单数据.用户可以在SQL语句中指定底层数据的存储结构——使用主从机构或者点对点结构.我们认为两种类型的数据模式和数据管理应该更加紧凑地融合在一起，而不仅仅是在存储层，同时，查询计划和存储模式的选择可以更加智能化.我们的未来工作主要包括：

1.建立一种根据列类型、列大小和已存储在表中的数据等来自动选择存储模式的服务.

2.目前，太极中一个表只能只采用一种存储结构.我们打算将表和备份数据本别采用不同模式进行存储，进而丰富我们的查询优化算法.

3.我们打算将论文文献[11]中提出的多维索应用到太极中以优化多列查询，同时采用该论文中的基于代价估计的索引更新策略来有效地更新索引结构.

## 参 考 文 献


[1] Lynch M. Amazon elastic compute cloud (Amazon ec2). [2010-06-25]. http://aws.amazon.con/ec2/

[2] IBM. IBM introduces ready-to-use cloud computing. [2010-06-25]. http://www-03.ibm.com/press/us /en/pressrelease/22613.wss

[3] Ghemawat S, Gobioff H, Leung S T. The google file system //Proc of SOSP'03. New York: ACM, 2003: 29-43

[4] HDFS. [2010-06-25]. http://hadoop.apache.org/hdfs/

[5] Hbase. [2010-06-25]. http://hadoop.apache.org/hbase/

[6] Cassandra. [2010-06-25]. http://incubator.apache.org/cassandra

[7] Chang F, Dean J, Ghemawat S, et al. Bigtable: A distributed storage system for structured data //Proc of the 7th Conf on USENIX Symp on Operating Systems Design and Implementation. Berkeley, CA: USENIX Association, 2006: 205-208

[8] DeCandia G, Hastorun D, Jampani M, et al. Dynamo: Amazons highly availbale key-value store //Proc of the 21st ACM Symp on Operating Systems Principles (SOSP'07). New York: ACM, 2007: 205-220

[9] Fox A, Brewer E A. Harvest, yield, and scalable tolerant systems //Proc of the the 7th Workshop on Hot Topics in Operating Systems. 1999: 174-178

[10] HIVE. [2010-06-25]. http://hadoop.apache.org/hive/

[11] Zhang X, Ai J, Wang Z, et al. An efficient multi-dimensional index for cloud data management //Proc of the CIKM Workshop on Cloud Data Management (CloudDB2009). New York: ACM, 2009: 17-24



**胡享梅** 女，1985年生，硕士研究生，主要研究方向为云数据管理.

**赵 婧** 女，1985年生，硕士研究生，主要研究方向为云数据管理.

**孟小峰** 男，1964年生，研究员，博士生导师，主要研究方向为Web数据管理、个人数据空间管理、XML数据管理、移动数据管理、闪存数据库技术以及云数据管理.

**王仲远** 男，1985年生，硕士，主要研究方向为数据集成、云数据管理.

**史英杰** 女，1983年生，博士研究生，主要研究方向为云数据管理.

**刘兵兵** 男，1987年生，硕士研究生，主要研究方向为云数据管理.

**王海平** 男，1987年生，硕士研究生，主要研究方向为云数据管理.