



基于 Hadoop 的信令详单实时查询平台构建方法

陈 丽¹, 王 锐²

(1. 广东交通职业技术学院计算机工程学院 广州 510650; 2. 中国移动通信集团广东有限公司 广州 510623)

摘 要: 随着电信行业 4G 通信网络技术的普及, 移动端互联网应用正蓬勃发展, 随之带来的数据流量呈爆炸式增长, 传统的处理方法难以支撑海量信令数据的实时查询需求, 为此, 提出了一种基于 Hadoop 的大数据实时查询平台, 以支撑超大规模的信令详单的处理。通过数据采集、数据建模、rowkey 设计、数据入库、二级索引、数据压缩与存储等实现方法, 提供了实时数据查询服务。最后, 在现网实验环境进行了验证, 能够取得非常好的收益。

关键词: 信令; 大数据; Hadoop; HBase; 实时

doi: 10.11959/j.issn.1000-0801.2015186

Method of Building Signaling Big Data Real-Time Query Platform Based on Hadoop

Chen Li¹, Wang Rui²

(1. Department of Computer Engineering, Guangdong Communication Polytechnic, Guangzhou 510650, China;

2. China Mobile Group Guangdong Co., Ltd., Guangzhou 510623, China)

Abstract: With the popularity of 4G communication technology, and development of mobile internet application, the data flow is booming, and traditional processing method is difficult to support the massive signalling data real-time query requirement. Therefore, a kind of data real-time query platform based on Hadoop was put forward, to support the details of a single large scale signaling processing. Through data collection, data modeling, rowkey design, data compression and storage, etc., real-time data query service was provided. Finally, the experimental environment of the current network was verified, which could get very good income.

Key words: signaling, big data, Hadoop, HBase, real-time

1 引言

随着电信行业 4G 技术的普及以及移动互联网应用的不断发展, 特别是手机、Pad、电脑等各种智能终端的广泛使用, 每人每时每刻都在产生海量的数据。其中, 数据记录了使用移动网络的时间、访问地址、位置信息和通信记录

等。对于电信运营商来说, 一方面, 可以对这些数据进行分析, 如分析网络瓶颈, 开展网络优化等, 为用户提供更好的服务; 另一方面, 可以为用户提供流量提醒、流量详单等实时查询, 避免因操作不规范、恶意软件等导致的流量过度消费, 提高用户体验感知。因此, 运营商需要构建一个适用于 PB 级甚至 EB 级大数据的查询处理平台, 以提供更

收稿日期: 2015-04-04; 修回日期: 2015-07-13

论文引用格式: 陈丽, 王锐. 基于 Hadoop 的信令详单实时查询平台构建方法. 电信科学, 2015186

Chen L, Wang R. Method of building signaling big data real-time query platform based on Hadoop. Telecommunications Science, 2015186



好的网络服务及用户体验。

传统的方案是使用传统关系型数据仓库,对海量数据进行采集、入库、计算等,提供数据分析和查询服务。一方面由于受限于其技术架构,主要采用二维表的数据存储方式,其数据访问性能在面对 TB 级以上数据量时存在较大瓶颈,而且数据量越大,瓶颈越明显;另一方面,扩展能力有限,无法支撑 PB 级甚至 EB 级的数据规模;另外,由于主要为商用产品,其扩展成本也非常昂贵。因此,传统数据仓库主要用于 TB 级的数据分析,用于支撑企业核心分析型应用。

目前电信业务发展迅猛,以某省运营商为例,其每天的信令数据量就达上百 TB。面对如此海量信令数据,以前由于缺少较好的解决方案,主要提供按计费流量单位(如 2 Mbit/s 或 10 Mbit/s)合并后的清单,数据的精确性和实时性均无法有效解决用户投诉问题。在用户体验至上的今天,不仅带来了用户满意度下降以至造成业务损失,而且在网络优化、业务优化、市场营销方面的信令数据分析缺失也已成为电信业务发展的核心瓶颈之一。

为了更有效地利用数据,发掘商业价值,提高网络质量,快速有效处理用户投诉,需要构建一种适用于 PB 级、EB 级的信令详单实时查询及分析的平台。

2 现有方案及缺点

面对 PB 级、EB 级海量数据的实时查询,现有技术方案主要采用传统关系型数据仓库。为了支撑更大的数据量、更高的查询速度,主要是通过分布式技术架构部署、增加硬件资源、采用高性能硬件设备等方式进行。其缺点如下。

- 数据入库效率差、并发能力差。使用关系型数据仓库的集群数量有限,并不能很好地扩展,而且大部分集群使用传统 SAN 存储,存储的总 I/O 吞吐量有限,难以满足每天数十 TB 的数据处理要求,同时也无法应对将来更大数据量的处理需求。
- 数据统计查询效率低。传统的架构主要依赖于高端设备的 I/O 能力及并发能力,但对于信令详单数据百亿级的数据量,传统集中式的架构查询统计效率无法满足要求。
- 系统扩展能力差。由于受限于系统软件架构,关系型数据仓库集群的节点数有限,并行扩展能力不强,无法适用大数据的倍数增长。
- 软硬件成本高。由于采用商用软件,硬件一般采用

高端硬件设备以提高系统性能及吞吐量,软硬件成本和维护费用非常高。

因此,传统关系型数据仓库主要解决 TB 级的数据分析应用,无法支撑信令详单 PB 级、EB 级的大数据应用需求。

3 基于 Hadoop 的大数据实时查询平台

大数据实时分析查询的需求就是快速从数十亿甚至上百亿条数据中,查询出一定条件的数据,并进行输出展示。随着互联网的迅猛发展,大数据的处理需求也越来越多,本文提出一种基于 Hadoop^[1]的大数据实时查询平台,用来支撑超大规模的信令详单的处理。

Hadoop 是 Apache 软件基金会旗下的一个开源分布式计算平台,可编写和运行分布式应用来处理大规模数据,其最核心设计就是:HDFS 和 MapReduce。HDFS 提供了大数据的存储,MapReduce 提供了对大数据的计算。相对于传统架构,Hadoop 集群具有高并发、低成本、高存储等众多优势,目前已在互联网行业得到广泛应用。其中,HBase^[2,3]是在 Hadoop 上构建的一种数据库,用来快速随机读写大规模数据。

针对电信行业信令详单数据实时查询的数据规模及业务需求,本文利用 Hadoop 相关技术制定解决方案,以 HBase 作为详单核心数据存储及处理技术,实现了基于 Hadoop 的大数据的实时平台,如图 1 所示。

本文方案通过说明如何快速、准确地从存放信令详单的 FTP 服务将数据入库,并构建适合业务场景的数据模型、物理存储、二级索引等,提供数据的高效存储及访问能力,支撑大数据的实时查询。

4 具体技术实现方法

平台实现的主要技术方法包括数据采集、数据建模、rowkey 设计、数据入库、二级索引、数据压缩与存储等,实现信令数据从数据源入库到 HBase 数据库,并按照数据模型进行数据存储、索引建立及计算,最后通过数据查询接口,提供实时数据查询服务。

4.1 大数据建模

HBase 是构建于 HDFS 上可伸缩、分布式、列式动态模式的 Hadoop 数据库。HBase 的模型结构^[3]在 Schema 声明的时候创建,与传统关系型数据库的差异在于没有固定的结构化。HBase 表的基本结构由 rowkey、column family(列族)、column(列族中的列)、time stamp(时间戳)组成。每个

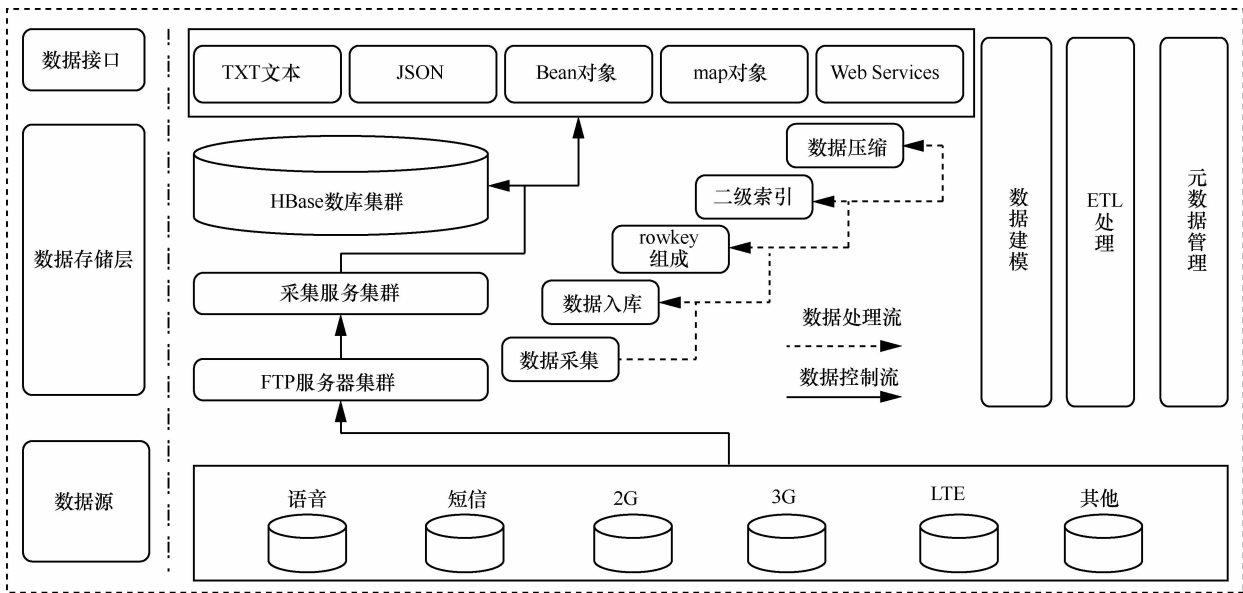


图 1 基于 Hadoop 的大数据实时查询平台

数据单元格是由{rowkey、column、version}组成,每组单元格内容是不可分割的字节数组。在构建 HBase 这种列式 key-value 模型时,尽量少创建列族,以避免数据空缺差异很大,导致不均匀分布,影响查询性能。数据建模需要根据业务需求和集群硬件资源情况来设计,一般可以分为以下两种。

- 高表:多行少列(可以存储大对象,也可以存储小对象),方便管理,耗费的资源多,有利于做 MapReduce 分析。
- 宽表:少行多列(如客户关系中某用户办理的业务),有利于提升查询性能,但 MapReduce 分析处理相对较繁琐。

以 CDR 的位置更新(CS_MM)数据为例来实现数据建模:首先需要设计 Schema,指定表名、列族名、压缩方式、版本等设计如下:

```
create 'TableName', { NAME => 'f', COMPRESSION => 'snappy', VERSIONS=3, BLOCKSIZE => '65536', IN_MEMORY => 'false', TTL => '2147483647' }
```

其中,列族名必须指定,数据对应的列字段名可以任意松散式添加,但建议工整规则,不需要指定数据类型,存入的数据可以为任意数据类型。模型设计根据需要分别设计成高表、宽表两类结构。

高表结构的模型设计如图 2 所示。

从文本数据至 HBase 表结构的映射如图 3 (图中粗字体表示组成 rowkey 部分的字段)所示。

- rowkey 组成:号码、时间、位置信息(必查条件选项)

rowkey	时间戳	column-family1
		column1
反序号码+时间+小区编号	插入时间	列族名称
		列名

图 2 模型设计-高表结构

组合。

- 时间戳:自动生成,按照入库时间生成,可以手工设置。
 - column-family|column1:列族(f:c1,value="文本所有值"),没有结构,全部将所有的值放入一个列中。
- 宽表结构的模型设计如图 4 所示。

从文本数据至 HBase 表结构的映射如图 5 (图中粗字体表示组成 rowkey 部分的字段)所示。

- rowkey 组成:号码、时间、位置信息(必查条件选项)组合;
- 时间戳:自动生成,按照入库时间生成,可以手工设置;
- column-family|column1:列族(f:c1,value="文本所有值"),文本第一个值;
- column-family|column2:列族(f:c2,value="文本所有值"),文本第二个值;
- column-family|column3:列族(f:c3,value="文本所有值"),文本第三个值。

4.2 大数据采集

移动用户打电话、发短信、访问网站均会产生相应的信令数据,目前现在各运营商均已完成语音短信 CS 域信

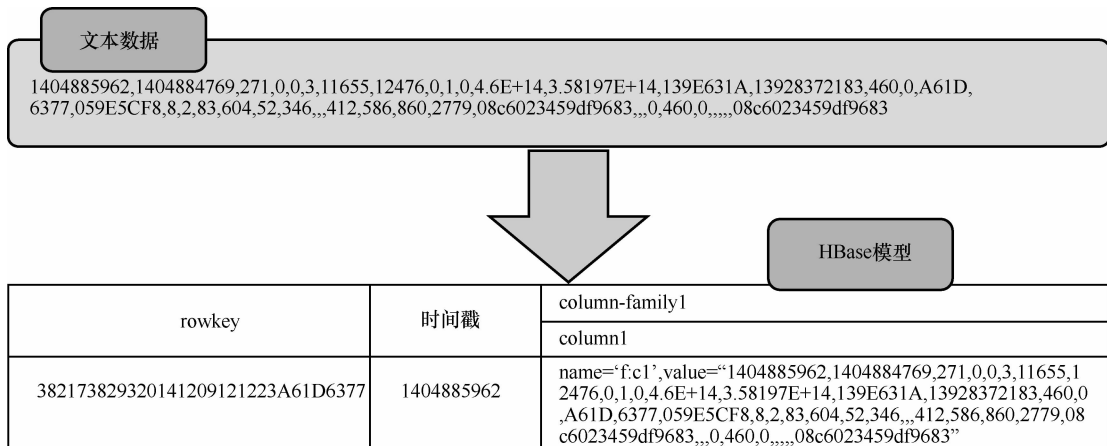


图3 模型设计-高表映射

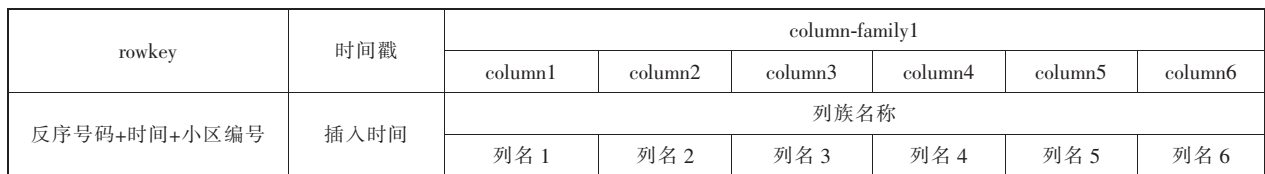


图4 模型设计-宽表结构

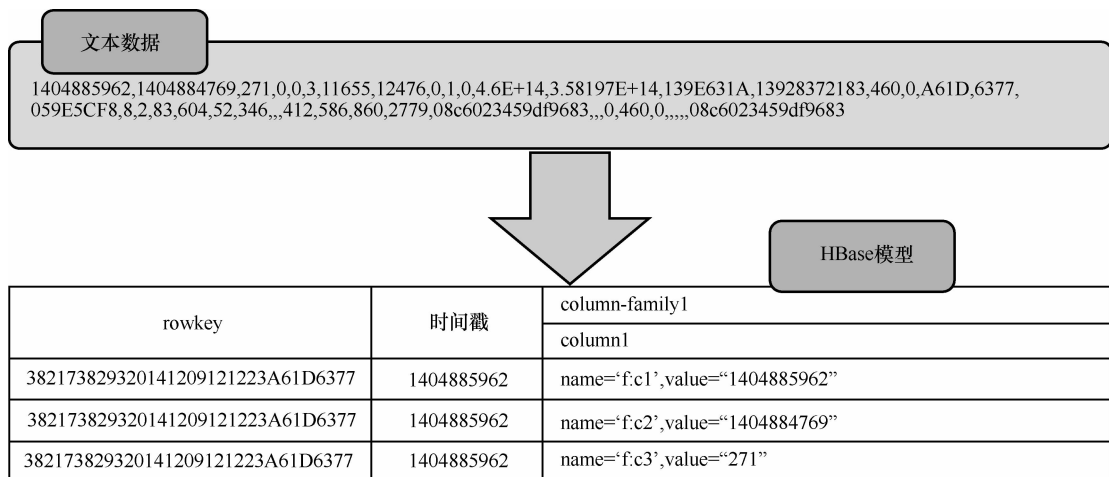


图5 模型设计-宽表映射

令数据及流量 PS 域数据的存储^[4]。大数据采集架构如图 6 所示。

信令详单数由各大厂商通过 FTP 定期将数据推送至指定目录（目前是以 5 min 作为推送周期）。客户端根据业务分配一定的服务器来集中接收各大厂商提供的信令数据。为了确保采集的效率及准确性,避免出现漏采、错采、重采之类的问题,在 HBase 数据库及 FTP 服务器之间设置转移数据存储,用于缓存信令详单数据。采集服务器集群规模按如下算式进行测算:

采集服务器数量=1 h FTP 服务器新增文件总量/3 600/

网卡速率。

在采集服务器上部署采集程序,实时从 FTP 服务器上采集详单文件。考虑涉及的数据类型多、数据量大,在采集服务器上划分不同的存储区域,建立信令数据存储规则(文件目录规则、文件命名规则、被采集的数据备份规则),实现每 n min 扫描一次 FTP 服务器、创建 FTP 连接池、实现 FTP 连接失败自动重连功能、多线程并行从采集队列中进行数据采集以及监控是否采集成功、没采集成功下次自动补采等功能。按照业务作为主体,将采集文件信息按指定的规则自动进行持久化保存。

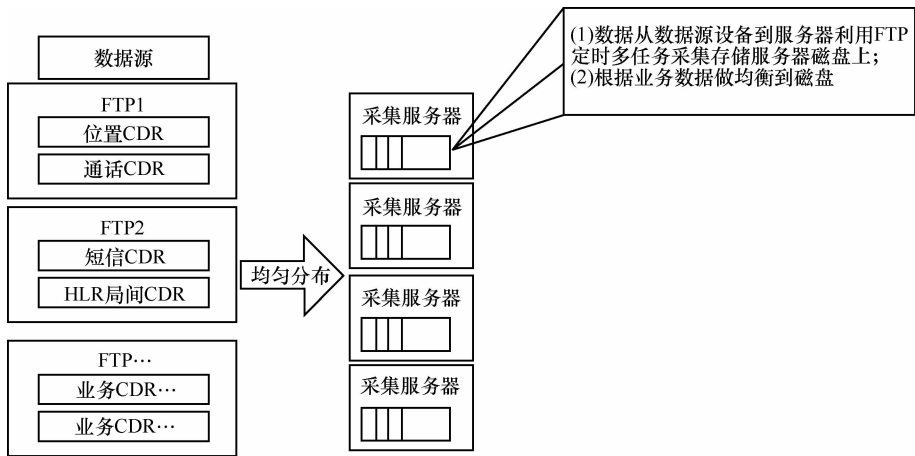


图 6 大数据采集架构

4.3 大数据入库

海量数据从 FTP 服务器采集到采集服务器集群后,按设定的文件存储原则,将数据文件存放在不同的存储空间(如图 7 所示)。采集服务器上配置入库进程,指定扫描周期,定期扫描数据,并按照元数据配置以元数据驱动自动实现 rowkey 生成、噪音数据过滤、数据匹配、入库字段选择等功能,数据自动识别进入相应的 HBase 表中,从而实现大数据的可配置式快速入库。

首先建立元数据配置信息,包括 HBase 中存放数据的表与需要入库的信令数据文件在采集服务器集群的磁盘位置及命名规则信息、文件数据字段描述与配置信息及文件字段与表字段对应关系等。

在完成元数据配置信息后,采集程序可每个周期扫描、被扫描的目录文件信息加载至内存并且对加载的文件进行标记,确保在内存中是唯一存在的;加载完成,程序通

过多线程对文件进行消费,每张表对应多个入库的线程,每个线程一次性固定消费 1~n 个文件;在读取文件的过程中,获取字段元数据信息进行匹配,按照配置进行 rowkey 生成,截取要入库的字段,过滤噪音数据,按照指定模型调用 HBase API 以 put 的方式入库。

4.4 rowkey 设计

HBase 中 rowkey 是主键,其设计也至关重要,入库数据会更根据 rowkey 进行排序,数据分布式按照 rowkey 范围来划分。为了将数据均匀分布到每台机器上,防止热 region 导致数据堆积在某一台或者多台机器上,rowkey 中的号码组成部分需要反序,这样数据分布均匀程度会高很多,同时随机插入的数据会被分配到对应 region 中。

HBase 的表数据按照 rowkey 规则分布,分布规则如图 8 所示。

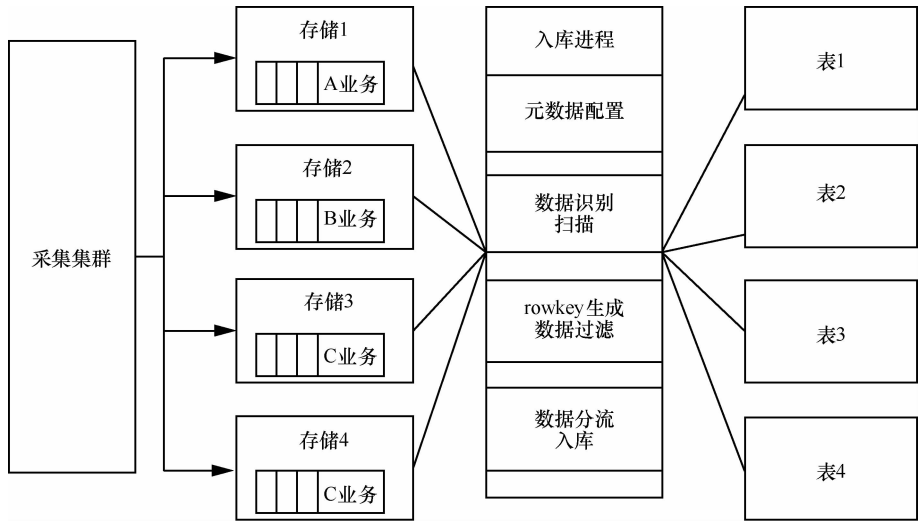


图 7 大数据入库架构

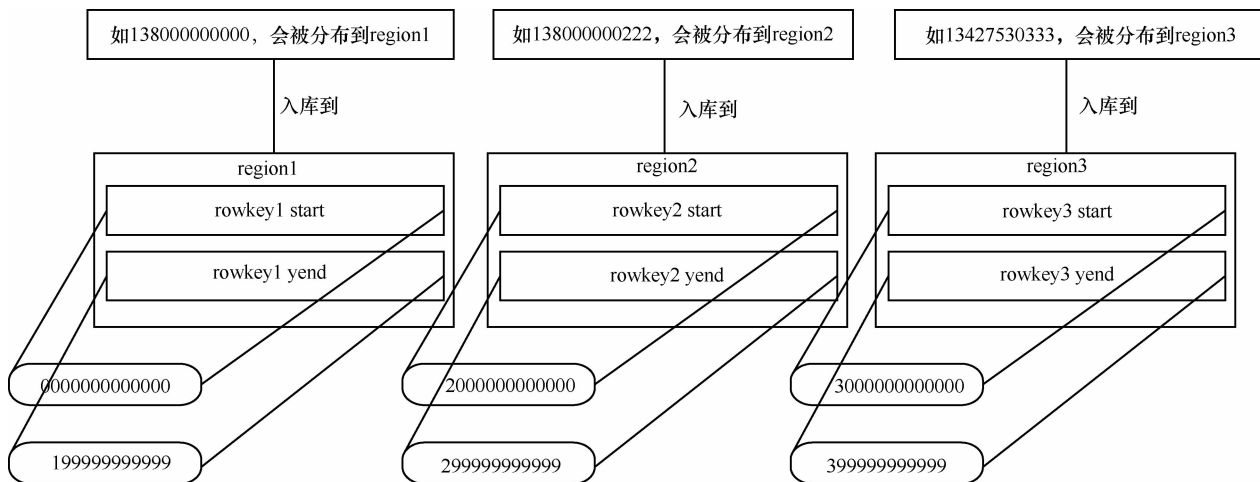


图8 rowkey 分布规则

HBase 的表由 region 组成,region 默认均匀分布在不同机器上。region 二个重要的属性: StartKey 与 EndKey,表示这个 region 维护的 rowkey 范围,当要读/写数据时,如果 rowkey 落在某个 start-end key 范围内,那么就会定位到目标 region 并且读/写到相关的数据。

对于 rowkey 的设计,还需要结合数据实际查询场景,通过 rowkey 能满足主要的查询需求。以 CS 域位置更新(CS_MM)数据为例,数据查询场景见表1。

通过分析,该类数据的查询以号码及时间是必选条件,但是小区编号用的比较少,所以放到 rowkey 的后段,被查询的条件越频繁越置前,即 rowkey 设计为:反序号码+时间(年月日时分秒)+小区编号。

4.5 二级索引设计

rowkey 设计一般可以满足单一的查询条件的性能,但实际应用场景中,单一的查询条件无法覆盖所有的查询需求,尽管这些查询需求的量较少,但足以引起非常不好的用户体验。

以信令数据位置更新(CS_MM)数据为例,位置更新需要支持多条件查询,rowkey 组成:号码+时间+位置,但是号码不一定是必选,时间+位置也会用到,然而时间+位置查询

会非常慢。为了提升用户体验,支撑满足 5%使用率的查询功能性能要求,因此需要创建二级索引,其实现方法一般有如下两种。

(1)实现 HBase 的 Coprocessor,将非必要的查询条件入库到另外的 HBase 表中,映射出 rowkey,达到毫秒级别的响应。该方法适用于数据量不大且查询条件比较固定的。具体实现如下。

- 位置更新表需要对应一张索引表,创建该索引表,rowkey 为时间+位置,值为号码。在录入位置更新表的同时将数据录入该索引表。
- 在查询选择了时间+位置信息的时候,自动走索引表,查找对应的号码,获取号码后,再组成位置更新表中的 rowkey 进行自动查询。
- 经过索引表查询出相应的 rowkey 进行查询,速度依然可以达到毫秒级别,该方法的缺点是增加入库负担和压力。

(2)实现基于 solr^[5]等搜索引擎的方式,将查询条件和 rowkey 以搜索引擎创建索引的方式存储到磁盘,前端访问通过接口的方式访问查询条件对应的 rowkey,再进行 HBase 的 rowkey 查询,达到毫秒级别。适用于数据量

表1 业务场景设计

场景	查询条件	涉及数据来源字段	场景比率
场景 1	按号码+时间(年月日) 按号码+时间(年月日時) 按号码+时间(年月日時分) 按照多号码+跨度时间范围 按单号码+跨时间范围	号码、开始时间	85%
场景 2	按号码+时间+小区编号	号码、开始时间、LAC、CI	10%
场景 3	按号码+时间+小区编号	开始时间、LAC、CI	5%

表 2 压缩方式对比

压缩格式	split	native	压缩率	速度	是否 Hadoop 自带	Linux 命令
gzip	否	是	很高	比较快	是,直接使用	有
lzo	是	是	比较高	比较快	否,需要安装	有
snappy	否	是	比较高	很快	否,需要安装	没有
bzip2	是	否	最高	慢	是,直接使用	有

表 3 实验结果对比

对比指标	GreenPlum	HBase	实验描述
入库性能	60 Mbit/s	150 Mbit/s	并行入库
并发	5~100 次/s	超过 20 000 次/s	多线程并发请求
查询性能	平均 30 s	平均 500 ms	100 并发主键查询
单点故障	无(集群无法工作)	有(自动切换、照常运行)	在线宕掉主节点
可扩展性	弱(重启集群生效)	强(修改配置、启动新添加节点、立即生效)	在线热插拔节点
二级索引	强(可创建联合索引)	弱(需要借助 solr 构建二级索引)	构建联合索引支撑

大,并且查询比较多、较复杂的组合条件的查询。具体实现如下。

- HBase 集群中搭建 solr,采用分布式。
- 实现多线程入库 solr 接口,利用 HBase 的提供的 Coprocessor,实现 endpoint 和 observer,endpoint 相当于关系型数据库的存储过程,而 observer 则相当于触发器。利用 observer 在记录 put 前后进行数据处理,例如将时间+位置+其他组合查询条件+号码形成一条记录一起插入 solr 中。同时采用先数据缓存,后批量写入 solr,提升入库效率。
- 查询过程中,通过时间+位置字符调用 solr API 进行全文搜索,搜索出来的内容进行过滤解析,提取号码,再组成位置更新表中的 rowkey 自动进行查询,从而实现毫秒到秒级别的查询效率。

其中,方法 1 是通过空间换时间的方法,适用于数据量小且查询条件较固定的方式,适用率不高;方法 2 通过加入全文搜索引擎的方式,建立新的索引,灵活高效、适用率高,并能达到毫秒级别的响应时间。

4.6 数据存储和压缩

由于信令详单数据量巨大,数据中也存在大量的重复数据,存放在 HBase 数据库中的 HFile 一般可进行压缩以节省存储空间,提升入库性能,HBase 支持 gzip、lzo、snappy、bzip2 等压缩方式。各种压缩方式的压缩比及数据访问速度对比见表 2。

经过测试,综合压缩率及速度两项指标,选用 snappy^[6]是最佳的压缩方式。项目实践证明,使用 snappy 压缩方法,对信令数据的压缩比可以达到 1:3 左右。

5 平台实验结果

通过以上技术,实现了基于 Hadoop 的分布式的大数据实时查询平台。实验案例采用 HBase 集群,对比传统数据库集群(以 GreenPlum 为例)^[7],进行了入库性能、并发量、查询性能、单点故障、扩展性的对比分析,结果显示 HBase 具有明显优势。

实验环境:某省语音话单业务数据和上网业务数据,业务数据表中最小超过 4 亿条记录/天,最大超过 100 亿条记录/天,数据总量超过 20 TB/天,数据保存一个月,总存储量达到 PB 级别,7×24 h 不间断进行数据处理,GreenPlum 和 HBase 集群机器均衡配置各 40 台机器。对比结果见表 3。

相比传统解决方案,利用相同的硬件成本能够达到远优于传统架构的处理效率与能力,取得非常好的效益。

6 结束语

随着移动互联网的发展,信息量及用户量将继续不断增长,传统运营商必将面临在大数据处理方面更多的压力。基于对信令数据特点及查询统计需求的分析,通过创建数据模型、设计高效的 rowkey 及二级索引,使用合理的数据压缩方法等技术建立的详单查询平台,可以有效地解决海量信令数据详单等大数据的实时业务需求,解决传统技术架构的瓶颈。

参考文献

1 White T. Hadoop: The Definitive Guide. Sebastopo: O'Reilly Media, 2012

(2015186-8 接排在 2015178-5 后)



参考文献

- 1 列数国外智慧城市经典案例. <http://www.doc88.com/p-9005797157579.html>, 2015
Foreign smart city classic case. <http://www.doc88.com/p-9005797157579.html>, 2015
- 2 青岛成“智慧城市”双试点 首批共 20 个城市. http://blue.qingdaonews.com/content/2013-10/10/content_10035621.htm, 2013
Qingdao into a “smart city”, the first pilot has 20 cities. http://blue.qingdaonews.com/content/2013-10/10/content_10035621.htm, 2015
- 3 卢风顺, 宋君强, 银福康等. CPU/GPU 协同并行计算研究综述. 计算机科学, 2011, 38(3)
Lu F S, Song J Q, Yin F K, *et al.* Survey of CPU/GPU synergetic parallel computing. Computer Science, 2011,38(3)
- 4 Hong C, Chen D, Chen W, *et al.* MapCG: writing parallel program portable between CPU and GPU. Proceedings of the 19th International Conference on Parallel Architectures and Compilation Techniques, Vienna, Austria, 2010: 217~226
- 5 He B, Fang W, Luo Q, *et al.* Mars: a MapReduce framework on graphics processors. Proceedings of the 17th International Conference on Parallel Architectures and Compilation

Techniques, Toronto, Canada, 2008: 260~269

- 6 Enmyren J, Kessler C W. SkePU: a multi-backend skeleton programming library for multi-GPU systems. Proceedings of the 4th International Workshop on High-Level Parallel Programming and Applications, New York, NY, USA, 2010: 5~14

[作者简介]



杨德利,男,现就职于中国电信股份有限公司广州研究院,主要研究方向为视频监控、大数据。



张涛,男,现就职于中国电信股份有限公司广州研究院,主要研究方向为视频监控、大数据。

2015178-5

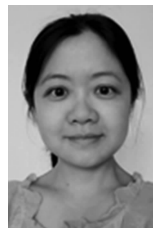
(上接 2015186-7)

- 2 George L. HBase: The Definitive Guide. Canada:O'Reilly Media, 2011
- 3 Dimiduk N. HBase in Action. Greenwich:Manning Publications, 2012
- 4 汤雅妃,魏进武,张云勇. 基于大数据的信令监测系统研究.邮电设计技术,2014(7):47~52
Tang Y F, Wei J W, Zhang Y Y. Research on big data based signaling monitoring platform. China Unicom Research Institute, 2014(7):47~52
- 5 傅巍玮,李仁发,刘钰峰等. 基于 solr 的分布式实时搜索模型研究与实现.电信科学,2011,27(11):51~56
Fu W W, Li R F, Liu Y F, *et al.* Study and implementation of distributed real-time search engine model based on solr. Telecommunications Science,2011,27(11):51~56
- 6 陈娜,张金娟,刘智琼等. 基于 Hadoop 平台的电信大数据入库及查询性能优化研究.移动通信,2014(7):58~63
Chen N, Zhang J J, Liu Z Q, *et al.* Research on loading and querying performance tuning of telecom big data based on Hadoop platform. Mobile Communications, 2014(7):58~63
- 7 张雨,蔡鑫,李爱民等. 分布式文件系统与 MPP 数据库的混搭架构在电信大数据平台中的应用.电信科学,2013,29(11):

12~16

Zhang Y, Cai X, Li A M, *et al.* Application of distribute file system & MPP database mashup architecture in telecom big data platform. Telecommunications Science, 2013, 29(11):12~16

[作者简介]



陈丽,女,广东交通职业技术学院讲师,主要研究方向为大数据技术及应用、信息系统管理。



王锐,男,中国移动通信集团广东有限公司高级工程师,主要研究方向为大数据/数据库技术及应用、网管支撑系统。

2015186-8