计算机海量查询的技术研究

冯毓生

长春理工大学计算机科学技术学院，中国 ·吉林 长春 130022

摘要：大数据时代下，海量查询具有很重要意义，在互联网的应用背景下，DNS 的查询技术得到广泛的应用。在现有的海量查询中，使用较多的是分布式查询技术和云端查询技术。本文详细讨论了分布式查询技术和云端查询技术。海量查询的核心意义在于：从规模庞大但结构相对规整 (多为结构化) 的数据集中，高效提取关键信息、识别统计规律 , 将始数据转移为可支撑决策的结构化情报，解决“存得多却用不上”的瓶颈，有三种核心价值维度（1）信息提炼与降维。（2）模式发现与趋势洞察。（3）决策科学化支撑。

关键词：大数据；DNS；海量查询；分布式查询技术；云端查询技术

The computer Massive query technology research

Feng Yusheng

Changchun University of Science and Technology School of Computer Science and Technology, China Jilin Changchun 130022

Abstract: Under big Data era, massive query has important application. In the application of internet, DNS query. technology have wide application. In today massive use,most usage is distribute techuology and cloude-technology in this paper, distribute query and cloude technolgy are discussed clearly. Massive query has core sense in application. In large scale,structure is most order.(most structure).data set. This technology extract high key information recognize statistics ruler. It convert original data, it's form is surpport decision struture informaion. It can solve bottleneck of store but nerer use. It have three core values dimension. (1)Information extract and dimension vally reduction. (2)pattern find andl trend see clearly, (3)So Support of science dlecision.

Keywords: Big data; DNS; Massive query; Distribute; Cloud query

1 海量数据的存储结构

1.1 海量数据存储结构没有单一标准答案

核心是根据数据类型和询问模式选择合适的存储模型、物理组织方式和分布式架构，常见包括行存储 , 列存储、LSM 树、对象存储、分布式文件系统。

1.2 常见存储结构有哪些

（1）行式存储：传统关系型数据库（如 MysqlL,Qracle) 主要用这种 , 把一行数据连续存放，适合 OLTP 场景（增删改查整行记录）。

（2）列式存贮：同一列数据连续存放，压缩率高，I0 少，造合 OLAP 分析场影 (如 CLickHouse,HBASE 部份场景 ,Parquet 文件格式)。

（3）键值存储：简单高效 , 如 reclis,hBASE 底层逻辑也接近 Key/velue ）适合海量数据随机读等。

（4）文档存贮：每行可以有不同属性列 ,Schema Free, 适合半结构化数据，如 MongoDB,Tablestore 表模型。

（5）对象存贮：以对象为单位管理，适合海量非结构化数据。(图片、视频 , 文件) 如 Ceph，云厂商对象存贮服务。

（6）时序存储：专门优化时间序列数据 , 如 InfluxDB. TDengine 等。

1.3 底层用什么数据结构

（1）B+ 树 : 传统数据库索引常用，造合随机读写，但写入性能在海量场景有瓶颈。

（2）LsM 树 (log-structured-merge-Tree) HBase、levelDB、许多云数据库的核心存贮引擎 , 把随机写、转为顺序写、写入吞吐量高，但读取需写 Compaction 合并。

（3）SSTable/LSM 变种：HBase 每个 store 对应列族 , 包含 Memstore 和 Hfile, 通过 WAL(wire-Ahead-log) 保障高可用。

（4）列式编码压缩：列存储中常用 RLE,Delta 编码压缩技术，压缩可达传统方法的 10 倍。

1.4 分布式怎么搭

（1）HDFS 分布式文件系统：NameNode 管理元数据，

DataNode 存储实际数据块。默认块大小 128MB，通过多副本，实现高可用。

（2）HBASE 架构：构建在 HDFS 之上。HMaster 负责元数据管理、Region 分配；RegionServer 管理多个 Region；Region 是分布式存储和负载均衡的最小单元。

（3）存算分离：新兴趋势，计算资源和存储资源解耦，可独立弹性伸缩，降低复杂度。

（4）存储分层：热数据放高性能存储，冷数据放低成本存储，实现冷热分离。

（5）多副本与容错：节点失效时，自动将数据重新分配到其他节点，保证下载和读取完成。

1.5 概论

简单说，海量存储数据结构的核心是“按需选择”。

根据数据类型（结构化 / 半结构化 / 非结构化）、访问模式（读多写多 / 分析为主 / 事务为主）、规模和成本要求，组合使用。行存储、列存储，LsM 树，分布式文件系统、对象存储技术。

2 海量数据查询模式

海量数据查询模式并非单一固定术语，而是指在 TB/PB 级数据规模下 , 为平衡性能、成本与实时性所采用的分布式存储架构、计算范式及访问策略所组合体系。

2.1 核心定义与关键特征

海量数据查询模式核心在于放弃传统单机教据库的即时全程扫描逻辑，转而依赖分布式存储，并行计算及特定索引结构。基本是根据业务对延迟 (实时 / 离线)、精度 (精确 / 近视) 和查询类型 (点查 / 分析 / 搜索) 的需求，匹配相应的技术栈。

（1）架构基础：基于 HDFs、对象存储或分布式 NosQL(如 HBase，Cassandra)，通过分片 (sharding) 实现水平扩展。

（2）计算范式：区分离线批处理 (Mapreduce/Spark、高吞吐、秒线至分钟级延迟) 与实时 / 流式查询（FLINK/ClickHouse）低延迟，高单发。

（3）优化第略：依赖列式存储（OLAp）倒排索引 (全文搜) 预聚合（物化软网）及游标 / 流式读取以避免内存流出。

2.2 主流查询模式分类

（1）离线批分析模式

根据主同场景与技术实现 , 主要分为以下四类模式。

①适用场景：历史数据报表、多维统计、月表离线分析。

②技术代表：Hive，Spark SQL，Presto。

③特点：高吞吐、高延迟（秒 – 分级），支持复杂 SQL；

依赖 Map-Reduce 或向量化执行引擎。

（2）实时点查与范围查询模式

①适用场景：订单状态查询、用户画像检索、监控指标实时读取。

②技术代表：HBase，Cassandra，Redis，TiKV。

③特点：基于主键 (Rowkey) 的毫秒线随机读写；

不支持复杂关联分析，强调高并发、低延迟。

（3）全文检索与多维分析模式

①适用场景：日志关键词检索、电商商品搜索，复杂条件过滤。

②技术代表：Elasticsearch、Solr、ClickHouse。

③特点：利用倒排索引实现全文秒搜；或列式存储实现海量数据下的快速聚合分析。

（4）概率与近似查询模式

①适用场景：超大规模数据下的快速估算（如 UV 统计，趋势预判），允许少量误差以换取极致速度。

②技术代表：HyperLog，Bloom Filter，Probery 算法。

③特点：通过概率算法牺牲部分查全率，将查询复杂度从线性降为常数级。

2.3 关键实施策略

（1）数据分片与路由：按时间、哈希、或范围将数据拆分至不同节点；查询时仅扫描相关分片。

（2）索引优化：针对查询条件构建复合索引、覆盖索引或倒排索引，避免全表扫描。

（3）读写分离与预计算：将高频复杂查询结果预先计算并存储（物化视图），查询时直接读取结果集。

（4）流式 / 游标读取：大数据量返回时采用流式传输或游算分页（Cursor-based），防止内存溢出及读取时间衰减。

2.4 若涉及具体工具

如 Excel 批量查询软件中时“所有文件 / 当前文件夹”模式，则属于特定应用层的文件遍历策略，非大数据领域的通用架构模式。

3 分布式存储结构

3.1 分布式存储系统

是将数据分散存储在多台独立的设备上。传统的网络存储系统采用集中的存储服务器存放所有数据，存储服务器成为系统性能的瓶颈，也是可靠性和安全性的焦点，不

能满足大规模存储应用的需要。分布式网络存储系统采用可扩展的系统结构，利用多台存储服务器分担存储负荷，利用位置服务器定位存储信息，它不但提高了系统的可靠性，可用性和存储效率，还易于扩展。

3.2 关键技术

（1）元数据管理

在大数据环境下，元数据所体量也非常大，元数据所存贮性能是整个分布式文件系统性能的关键。常见所元数据管理可以分为集中式和分布式元数据管理架构。集中式元数据管理架构：采用单一元数括服务器实现简单，但是存在单点故障等问题。分布式元数据管理架构则将元数据分散在多个结点上，然而解决了元数据服务器的性能瓶颈等问题，并提高了元数据管理架构的可扩展性，但实现较为复杂，并引入了元数据一致性的问题。

另外，还有一种无元数据服务器的分布式架构，通过在线算法组织数据，不需要专门的元数据服务器。但是该架构对数据一致性的保障很困难，实现较为复杂。文件目录和元数据的操作效率低下，并且缺乏文件系统的全局监控管理功。

（2）系统弹性扩展技术

在大数据环境下，数据规模和复杂度的增加往往非常迅速，对系统的扩展性能要求较高。实现存储系统的高扩展性，首先要解决两个方面的重要问题，包含元数据的分配和数据的透明迁移。元数据的分配主要通过静态子树划分技术实现。后者则侧重数据迁移算法的优化。此外，大数据存储体系规模庞大，结点失效率高，因此还需要完成一定的自适应管理功能。系统必须能够根据数据量和计算的工作量估算所需要的结点个数，并动态地将数据结点迁移，以实现负载均衡；同时，结点失效时，数据必须可以通过副本等机制进行恢复，不能对上层应用产生影响。

（3）存储层级内的优化技术

构建存储系统时，需要基于成本和性能考虑，因此存储系统通常采用多层不同性能比的存储器件组成存储层次结构。大数据的规模大，因此可以构建高效合理的存储层次结构，可以在保证系统性能的前提下，降低系统能耗和构建成本；利用数据访问局部性原理，可以从两个方面对存储层次结构进行优化。从提高性能的角度，可以通过分析应用特征，识别热点数据并对其进行缓存或预取，通过高效的缓存预取算法和合理的缓存容量配比，以提高访问性能。从降低成本的角度，采用信息生命周期管理方法，将访问频率低的冷数据迁移到低速廉价存储设备上，可以在小幅牺牲系统整体性能的基础上，大幅降低系统的构建成本和功耗。

（4）针对应用和负载的存储优化技术

传统数据存储模型需要支持尽可能多的应用，因此需要具备较好的通用性。大数据具有大规模、高动态及快速处理等特性，通用的数据存储模型通常并不是最能提高应用性能的模型。而大数据存储系统对上层应用性能的关注远远超过对通用性的追求。针对应用和负载来优化存储，就是将数据存储应用耦合。简化或扩展分布式文件系统的功能，根据特定应用、特定负载、特定的计算模型对文件系统进行定制和深度优化，使应用达到最佳性能。这类优化技术在谷歌、Facebook 等互联网公司的内部存储系统上，管理超过千万亿字节级别的大数据，能够达到非常高的性能。

3.3 考虑因素

（1）一致性

分布式存储系统需要使用多台服务器共同存储数据，而随着服务器数量的增加，服务器出现故障的概率也在不断增加。为了保证在有服务器出现故障的情况下系统仍然可用。一般做法把一个数据分成多份存储在不同的服务器中。但是由于故障和并行存储等情况的存在，同一个数据的多个副本之间可能存在不一致的情况。这里称保证多个副本的数据完全一致的性质为一致性。

（2）可用性

分布式存储系统需要多台服务器同时工作。当服务器数量增多时，其中的一些服务器出现故障是在所难免的。我们希望这样的情况不会对整个系统造成太大的影响。在系统中的一部分节点出现故障之后，系统的整体不影响客户端的读 / 写请求称为可用性。

（3）分区容错性

分布式存储系统中的多台服务器通过网络进行连接。但是我们无法保证网络是一直通畅的，分布式系统需要具有一定的容错性来处理网络故障带来的问题。一个令人满意的情况是，当一个网络因为故障而分解为多个部分的时候，分布式存储系统仍然能够工作。

4 云存储架构

4.1 云存储架构是一种依托云计算技术构建的数据存储体系

通过虚拟化技术整合物理资源，形成可弹性扩展的虚拟资源池。它通常采用分层设计，将底层硬件抽象为统一的服务接口，支持用户按需获取存储资源。

4.2 分层架构由哪些部分组成

云存储架构一般从下至上分成四个核心层次，各层分工明确，以保障系统的高可用性与扩展性。

（1）数据存储层：由大量的物理存储设备（如硬盘、固态硬盘）组成，通过网络连接形成分布式存储集群，负责实际的数据持久化。

（2）基础管理层：是架构的核心，利用分布式文件系统和集群技术实现多节点协同，负责数据分配，无数据管理、监控维护及故障迁移。

（3）应用接口层：提供统一的访问接口，如 RESTful API，SDK 等，支持不同业务场景调用存储服务。

（4）用户访问层：授权用户通过 Web 界面，命令行或应用程序接入系统，进行上传、下载、共享等操作。

4.3 主要存储类型有什么区别

云存储根据数据访问方式的不同，主要分为对象存储、块存储和文件存储三类，适用场景各自迥异。

（1）对象存储：以对象为单位管理数据，支持海量非结构化数据（如图片、视频等），通过 HTTP/HTTPS 协议访问，扩展性极强。

（2）块存储：提供裸磁盘访问，类似物理硬盘，需挂载后格式化才能使用；适合对 IOPS 要求高的数据库或高性能计算场景。

（3）文件存储：支持 NFS/SMB 协议，提供共享文件系统；适合企业内部文件共享、多机器协同操作等场景。

4.4 关键技术如何保障可靠性

为了保障数据不丢失且访问高效，云存储架构采用多种关键技术：

（1）分布式架构：将数据分散存放在多个节点上，避免单点故障；支持线性扩展，扩容容易。

（2）数据冗余：采用多副本机制（如同一份数据 3 个副本）或者纠删码技术；即使部分节点损坏，也可以恢复出数据。

（3）元数据管理：通过分布式元数据管理文件索引，快速定位数据存储位置，提升访问效率。

（4）智能分层：自动识别数据冷热状态，在标准、低频归档，等存储类型间动态迁移，降低存储成本。

参考文献：

[1] More is less: a bag of long words for the Compression Project.ACM Transaction on Computer System(TOCS). 8(4),353-374.

[2] The Sprite network operating system. IEEE Computer, 21.(2),23-36.

[3] Amazon's highly av lable key-value store. ACM SIGOPS Operating System Review 41(6),205-220.

[4] Ita,A.(2016).An analysis of performance management in distribute storage system. International Journal of Management Science and Business Administration, 3(11),45-53.

[5] 胡文波，徐造林. 分布式存储方案的设计与研究[J]. 计算机技术与发展，2010, 20(4):65-68.

[6] 侯孟书. 基于 P2P 的分布式存储及其相关技术研究[D]. 电子科技大学，2005.

[7] 王志伟，李晓东，张宇. 云存储技术研究综述[J]. 计算机工程与科学，2015,36(2):425-432.

[8] 陈明，张志勇，刘建勇. 云存储技术及其应用研究[J]. 计算机工程与科学，2016,38(2):285-292.

[9] 李宁，王庆，李晓东. 云存储系统架构及关键技术方案分析[J]. 计算机工程与设计，2017,38(15):5375-5380.

[10] 刘志勇，刘建勇，陈明. 基于云计算的云存储技术研究[J]. 计算机技术与发展，2015,25(4):1-5.

作者简介：冯毓生（1967.05-），男，汉族，吉林长春，硕士，长春理工大学计算机科学与技术学院，讲师，研究方向：计算机科学技术。