Google三驾马车读后感

18301101 刘宇帆

阅读完Google的：GFS，MapReduce，BigTable后，对其搜索引擎的架构设计有了一定的了解。从文件系统（GFS），到数据模型（BigTable），再到MapReduce（算法），最后到应用，其自下而上的设计方式令我受益匪浅。

**1.Google File System (GFS)--2003**

Google在2003年发布了Google文件系统文件，这是一种可扩展的分布式文件系统，用于访问大量数据的大型分布式应用程序。它在廉价的常规硬件上运行，并提供容错能力。从根本上讲：文件被拆分为多个块，这些块冗余地存储在商用计算机集群中。

GFS由一个主服务器和大量的块服务器组成。 Google设置了一个主目录来存储目录和索引信息，以简化系统结果并提高性能，但这可能导致主目录成为单点故障或瓶颈。为了消除主要的单点故障，Google将每个块（chunk）设置为较大的大小（64M）。因此，由于代码访问数据的局部性，减少了应用程序与主服务器之间的交互，即减少了元数据（metadata）和流量访问。

另外，主机的所有信息都存储在内存中，并且在启动时从块服务器中获取信息。提高主机的性能和吞吐量，也帮助主机切换到主机之后的主机。客户端和块服务器不会分别缓存文件数据，而是使用linux文件系统自己的缓存。

# 2.MapReduce--2004

随后是2004年发表的MapReduce论文。该论文描述了大数据的分布式计算方法。主要思想是分解任务，然后在处理能力较弱的多个计算节点中同时处理它，然后合并结果。完成大数据处理。

传说Google使用它来计算他们的搜索索引。 Mikio L. Braun认为其工作模式应为：Google将所有已爬网的页面放在其集群上，然后每天使用MapReduce重新计算它们。

MapReduce是一组用于分布式并行计算的编程模型。当时流行通过提升CPU频率的方式提高程序的性能，方向也主要是多核，超频等。但是，以前的程序不会自动获得多核的好处。只有编写并发程序，才能真正体验到多核的好处。分布式计算也是如此。

MapReduce由Lisp的Map和reduce组成。 Map是一个映射，并且指令已分发给多个工作程序。 Reduce是一种协议，并且将Map工作者计算的结果合并在一起。 MapReduce使用GFS来存储数据。

# 3.BigTable--2006

Bigtable于2006年发布，启发了无数NoSQL数据库，例如Cassandra，HBase等。 Cassandra架构的一半模仿Bigtable，包括数据模型，SSTables和预先写入日志（另一半模仿Amazon的Dynamo数据库，使用对等集群）。

BigTable建立在GFS和MapReduce之上。每个表都是多维稀疏图。为了管理巨大的表，将表按行划分，并将这些划分的数据统称为平板电脑（tablets）。每个平板电脑大约有100-200 MB，每台机器存储大约100个平板电脑，基础架构是GFS。

由于GFS是分布式文件系统，因此在使用Tablets机制后，它可以实现良好的负载平衡。举个栗子，可以将频繁响应的表移至另一台空闲计算机，然后快速重建它。

参考资料

[1] The Google File System; <http://labs.google.com/papers/gfs-sosp2003.pdf>

[2] MapReduce: Simplifed Data Processing on Large Clusters; <http://labs.google.com/papers/mapreduce-osdi04.pdf>

[3] Bigtable: A Distributed Storage System for Structured Data;<http://labs.google.com/papers/bigtable-osdi06.pdf>

[4] <https://www.youtube.com/watch?v=WLad7CCexo8> BigTiger官方频道视频

(bilibili搬运:

<https://www.bilibili.com/video/BV1iV41167GL?from=search&seid=712578621857251827>)