读后感

GFS是一个大规模的分布式文件系统。 GFS 具有与传统分布式文件系统相同的许多设计目标，例如性能、可扩展性、可靠性和可用性。  
          
      客户端是 GFS 提供给应用程序的访问接口，是作为库文件提供的一组专用接口。 应用程序直接调用这些库函数并将其链接到它们。 Master 是 GFS 的管理节点，在逻辑上只有一个节点，负责维护系统元数据、管理整个文件系统以及 GFS 文件系统的核心。 Chunk Server 负责特定的存储任务。 客户端是 GFS 提供给应用程序的访问接口，是作为库文件提供的一组专用接口。 应用程序直接调用这些库函数并将其链接到它们。 Master 是 GFS 的管理节点，在逻辑上只有一个节点，负责维护系统元数据、管理整个文件系统以及 GFS 文件系统的核心。 Chunk Server 负责特定的存储任务。

主要介绍我们一个大规模的分布式文件系统，该系统由一个主服务器、多个块服务器和多个客户端组成，通过特殊的交互。

Cloud BigTable 是谷歌使用的分布式存储系统，可以归类为非关系数据库系统。BigTable 主要用于可扩展性。它通常适用于分布在数千台计算机上的 PB 数据。关于 BigTable 的细节， 没有太多的公开信息， 因为它是谷歌的所有者。关于它最权威的信息是它的论文[1]。基于原始纸张的开源实现是 Apache HBase。谷歌现在提供了 BigTable 作为其云 NoSQL 数据库服务。

在 BigTable 中，表被划分为几个平板电脑，每个片剂都是连续行的子集。平板电脑是一个数据分发和负载平衡单元。表中的不同平板电脑可以分配给不同的平板电脑服务器。平板电脑以map中构造的融合树的形式存储。此外，BigTable 允许客户创建本地组。局部组是表中列的子集。BigTable 将为每个区域组创建一个单独的 SSTable，这将提高此区域组的读取性能。

正如文件系统需要数据库来存储结构化数据一样，GFS 也需要 Bigtable 来存储结构化数据。BigTable 构建于 GFS、调度器、锁定服务和地图减少。 每个表都是多维稀疏卡，为了管理巨大的表，表按行划分，除以数据统称为平板电脑。每台平板电脑的存储约 100-200 MB，每台机器存储约 100 片。底层体系结构是：GFS。由于 GFS 是一个分布式文件系统，因此可以使用平板电脑机制获得良好的负载平衡。例如，您可以移动经常响应另一台空转计算机的表并快速重建它。

主要含义是，通过提高主频CPU快速结束来提高程序性能，CPU设计的方向主要是多核、超线程等，但以前的程序不能自动获得多核的优势，只需准备一个联合程序即可获得多核优势。 分布式计算也是如此。

MapReduce 由Map和 Reduce、Lisp 和地图组成，这些地图被暗示，指令分发给多个工人，Reduce 是法律，卡工作者计算并组合了结果。Google MapReduce 实现使用 GFS 来存储数据。MapReduce 可用于分布式 grep、FREQUENCies 访问 URL 的数量、反向 Web 链接图、不变量拼写和反向索引