GFS读后感

通过阅读概要部分，我了解到GFS是一个面向大规模数据密集型应用的、可伸缩的分布式文件系统。GFS虽然运行在廉价的普遍硬件设备上，但是它依然了提供灾难冗余的能力，为大量客户机提供了高性能的服务。GFS完全满足了Google对存储的需求。GFS 作为存储平台已经被广泛的部署在 Google 内部，除了存储服务产生和处理的数据，同时还用于那些需要大规模数据集的研究和开发工作。目前为止，最大的一个集群利用数千台机器的数千个硬盘，提供了数百 TB 的存储空间，同时为数百个客户机服务。足见GFS堪称整个Google公司的基础实现，起到了举足轻重的重要地位，也当之无愧是三驾马车之一。

在简介中，作者详细陈述了开发GFS的实际需求和现实意义，首先，组件失效被认为是常态事件，而不是意外事件。GFS 包括几百甚至几千台普通的廉价设备组装的存储机器，同时被相当数量的客户机访问。GFS 组件的数量和质量导致在事实上，任何给定时间内都有可能发生某些组件无法工作，某些组件无法从它们目前的失效状态中恢复。其次以通常的标准衡量，所需要存储的文件数量本身就是非常巨大的，且绝大部分文件的修改是采用在文件尾部追加数据，而不是覆盖原有数据的方式。对文件的随机写入操作在实际中几乎不存在。一旦写完之后，对文件的操作就只有读，而且通常是按顺序读。而正是为了解决这一些列的问题，GFS应运而生。读了简介也是深深感受到，GFS之所以能产生，Google本身的庞大以及所产生的需求之迫切，也成为了GFS开发的培养基，相比与小公司小团队，Google本身拥有庞大的数据量，高新硬件设备，这一切都是技术迭代发展的培养基。

在第二部分的设计概述中，作者展开描述了简介中所列举的需求以及相对应的解决方案，例如针对常态化的组件失效，系统必须持续监控自身的状态，它必须将组件失效作为一种常态，能够迅速地侦测、冗余并恢复失效的组件。同时针对所需要实现的功能，展示了所设计的用户接口以及结构图。

在第四部分，作者主要介绍了GFS开发的一个重要原则，就是最小化所有操作和 Master 节点的交互。带着这样的设计理念，描述了客户机、Master 服务器和 Chunk 服务器如何进行交互，以实现数据修改操作、原子的记录追加操作以及快照功能。同时，为了提高网络效率，开发者们采取了把数据流和控制流分开的措施。在控制流从客户机到 主Chunk、然后再到所有二级副本的同时，数据以管道的方式，顺序的沿着一个精心选择的Chunk 服务器链推送，这样就能够充分利用每台机器的带宽，避免网络瓶颈和高延时的连

接，最小化推送所有数据的延时。

第五部分主要介绍了容错与诊断，处理频繁发生的组件失效。组件的数量和质

量让这些问题出现的频率远远超过一般系统意外发生的频率，也就是不完全依赖机器的稳定

性，也不完全相信硬盘的可靠性，所以GFS 自带工具诊断系统故障。

而后主要就是论文的收尾阶段，也就是写了测试性能以及一些开发过程中的感悟，相比于阅读一些学生出产的论文，Google的论文从深度广度上都是极难阅读的，实在是感觉所涉及的技术多且难，底层且复杂，但是也能学习到发展的过程就是在应对不同的问题与需求中所产生的，其中所展现出的设计思想和理念十分值得学习。