

云存储在校园信息化中的应用与优化

熊聪聪¹, 冯 龙¹, 陈丽仙², 苏 静¹

(1. 天津科技大学计算机科学与信息工程学院, 天津 300222;
2. 天津科技大学信息化建设与管理办公室, 天津 300222)

摘要: 针对目前校园海量数据存储面临的服务器资源利用率低、管理维护成本高等问题, 在云存储技术的基础上提出了校园云存储服务的设计, 并进行了实验. 为减少读写速率的损失, 系统通过调用 HDFS (Hadoop distributed file system) 的 API 接口方式实现数据存储及对集群的状态监控和管理. 实验表明: 传输文件的大小和集群中数据节点的个数对云存储服务的数据读写速率的影响较大; HDFS 并行传输大量小文件的速率相对较低.

关键词: 校园网; 云存储; 海量数据; HDFS

中图分类号: TP399 **文献标志码:** A **文章编号:** 1672-6510(2012)05-0069-05

Application and Optimization of Cloud Storage Service for Campus Informatization

XIONG Congcong¹, FENG Long¹, CHEN Lixian², SU Jing¹

(1. College of Computer Science and Information Engineering, Tianjin University of Science & Technology, Tianjin 300222, China; 2. Informatization Construction and Management Office, Tianjin University of Science & Technology, Tianjin 300222, China)

Abstract: Concerning the existing problems such as the low utilization and high costs for operation of the massive data storage on campus, a cloud storage service for campus was designed based on the cloud storage technology, and an experiment was carried out about that. To achieve higher I/O efficiency, the system can implement data storage, and also monitor and manage cluster status by using HDFS (Hadoop distributed file system) API. Experimental results of the system performance show that the size of the files being transferred and the number of the cluster data nodes have great influence on the reading and writing speed of the cloud storage service, while the parallel transmitting rate of small files on HDFS is relatively low.

Key words: campus network; cloud storage; massive data; HDFS

随着校园信息化建设的发展, 校园网内各类数字信息的积累以及学校部门间设备采购缺乏统一管理等问题引发了服务器、存储和网络等资源利用率下滑和管理成本攀升的问题. 云存储技术^[1]的应用为解决该问题提供了可能. 云存储是云计算概念的延伸, 是指通过虚拟化技术、集群应用或分布式文件系统等功, 将网络中大量各种不同类型的存储设备通过应用软件集合起来协同工作, 并通过一定的应用软件或接口技术共同对外提供数据存储和业务访问功能^[2].

目前, 国内外众多的 IT 厂商纷纷推出云存储产

品^[3-5], 按照用户使用范围和接入方式可将其分为公有云存储和私有云存储. 这些产品多应用于商业目的, 服务对象多为政府、大中型企业或是个人用户, 而专门针对高等院校教育资源整合的成功案例不多. 倘若直接利用这些平台, 开发周期和项目成本固然会减少, 但原有教育资源的转移将会成为突出问题, 而且资源的安全性访问、传输效率等方面均比不上在本地建立云存储中心. 部分国内高校已成功地, 将云存储技术应用于学校图书馆数字资源整合, 如北京第二外国语学院图书馆^[6]. 而云存储技术在整体校

收稿日期: 2012-03-05; 修回日期: 2012-03-29

基金项目: 天津市高等学校科技发展基金计划项目(20100823); 天津市科技型中小企业创新基金项目(11ZXCXGX07700)

作者简介: 熊聪聪(1961—), 女, 四川人, 教授, xiongcc@tust.edu.cn.

园信息化建设中的应用大多尚处于研究探索阶段。

从云存储技术的基本思想出发,结合校园数字资源的特点,本文提出采用校园云存储服务方案来改变当前校园中配置专用服务器维护单一应用的现状,介绍了系统设计及部分主要模块的实现.并且,分别从传输文件的大小、集群节点的数量和数据块设置的大小三个角度进行实验,寻找提高集群数据传输速率的路径。

1 HDFS工作原理

HDFS (Hadoop distributed file system)^[7]以容错性好、可伸缩性强、代码开源等优势倍受关注,成为当前主流分布式文件系统之一. HDFS 是被设计成可以在大规模廉价机器上运行的分布式文件系统,其设计思想来源于 GFS (Google file system). 与现有分布式文件系统相比, HDFS 放宽了部分 POSIX (portable operating system interface of Unix) 约束,以实现流式读取文件系统数据的目的^[8]. HDFS 的设计目标使得它在异构软硬件平台间有较良好的可移植性,因此可以充分利用学校分散的不同型号的服务器资源和存储资源构建数据中心,为校园网上的不同应用按需提供底层支持。

1.1 工作模型

HDFS 采用主从式结构^[8],主要由 3 部分组成: NameNode (主节点), SecondaryNameNode (备份主节点) 和 DataNode (从节点). 每个集群中只有 1 台计算机作为 NameNode,负责记录整个系统的元数据(包括文件名、副本数、数据块的位置等)和文件系统的整体运行状况; 1 台计算机作为 SecondaryNameNode,用于备份 NameNode 上的系统信息;其他计算机作为 DataNode,存储用户数据。

整个系统的工作流程较复杂,以用户上传文件为例进行简述. DFSCient (客户端) 向 NameNode 发出任务请求, NameNode 在对客户合法性做验证后将文件按照系统配置分成数据块(默认 64 MB),接着依据集群中各 DataNode 状态进行任务分配,通过负载均衡算法决定数据块存储在哪些 DataNode 中,并告知客户端. 之后,客户端直接与 DataNode 通信,进行读写操作. 任务完成后, DFSCient 将结果反馈给 NameNode.

1.2 内部通信机制

HDFS 的内部通信主要是指集群内 NameNode

和 DataNode 之间、DataNode 之间,以及 NameNode 和 SecondaryNameNode 间的数据传输和通信. 相对于现有的网络 C/S 模型, HDFS 有所改进, NameNode 只负责与用户通信以及管理 DataNode 上的数据,而数据流直接在客户端与 DataNode 之间传输. 节点之间的进程间通信是通过借助远程过程调用(RPC)方式以实现^[9]. SecondaryNameNode 通过周期性保存 NameNode 上的 FSImage 和 EditLog 进行系统备份,减少 HDFS 的单点故障引发的问题。

2 系统设计与实现

校园网功能的纵向扩展使得校内各种数字信息量不断攀升,师生对校园网的依赖性日渐增强. 云存储服务的建立不仅要将校园内的存储资源进行整合,良好的用户体验和系统可靠性均要有所保证。

2.1 总体设计

校园云存储服务的系统架构如图 1 所示,分为 3 层,用户端通过 Web 浏览器对云存储系统进行访问, HdfsClient 层负责接受用户请求,并对不同的应用提供服务接口. HdfsClient 层的主要功能有: 用户管理、访问控制、业务逻辑控制和数据管理,是用户 (UI 层) 与 HDFS 集群 (存储层) 通信的桥梁,也是本系统实现的重点. 数据库主要用于保存系统内用户信息、权限和文件属性信息,本系统中将数据库与 HdfsClient 层放置于同一服务器上. 校园网内所有应用的数据(如用户上传的文档和音视频文件等)均存储在 HDFS 集群中。

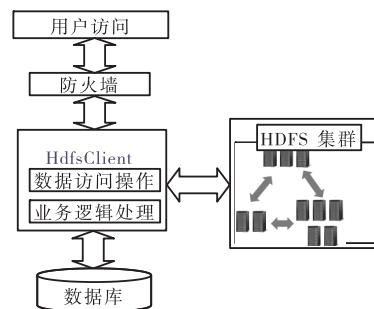


图 1 系统架构图

Fig. 1 System architecture diagram

2.2 功能模块设计

在本系统中 HdfsClient 层主要包含数据访问操作和业务逻辑处理两部分内容,具体分为用户访问、用户管理、部门权限管理和集群信息管理 4 个模块。

用户访问模块实现包括用户登录,修改个人信

息,文件的上传、下载、共享和加密,目录创建与编辑等功能。

用户管理模块实现用户信息管理、用户状态管理、用户所属部门的分配管理。

部门权限管理模块实现部门基础信息设置、系统权限管理和部门权限分配。

集群管理模块用于系统维护人员设置和管理集群内主从节点 IP、从节点数量,查看集群状态、任务状态。同时,该模块有存储空间余量预警、集群故障警报等功能。

系统后台管理的层次及各模块的功能见图 2。

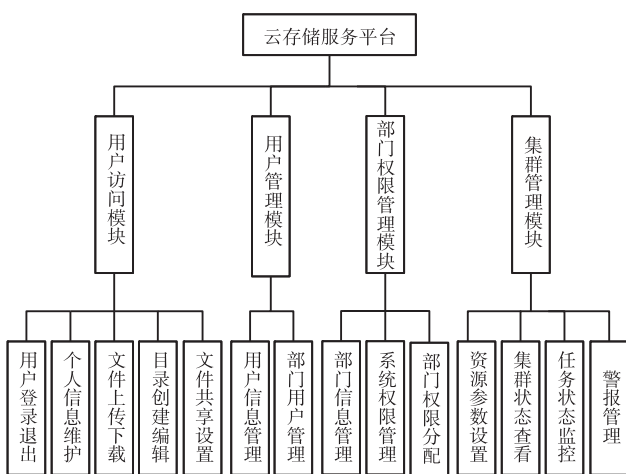


图 2 系统功能模块图

Fig. 2 Function module of the system

2.3 系统实现

HDFS 的调用方式有两种:一种是在客户端上调用 HDFS 的 API 对文件进行操作;另一种是使用 Fuse-DFS 将一个 Hadoop 文件系统通过 mount 命令挂载到 Linux 文件系统中,然后利用标准的 Unix 命令与之交互^[9]。由于使用 Fuse-DFS 后的读写操作会有 30%左右的效率损失^[10],所以在 HdfsClient 层采用直接调用 HDFS 的 API 接口的方式实现各功能。云存储平台包含的 4 个模块分别是:用户访问模块、集群管理模块、用户管理模块和部门权限管理模块。其中,前两个模块涉及到与 HDFS 通信。

2.3.1 用户访问模块

用户访问模块主要涉及的功能有:文件的上传、下载、共享和删除等操作,目录的创建、修改以及查看目录下的子目录和文件列表等操作。目录在客户端的浏览器上以文件夹的形式呈现。用户访问的实现过程需要调用 HDFS 的 API 接口进行操作,核心代码如下所示:

```

public bool CreateFile(String filename)
{//目录创建代码
    Configuration config = new Configuration();
    FileSystem fs = FileSystem.get(config);
    Path path = new Path(filename);
    FSDataOutputStream.outputStream = fs.create(path);
    ...
}
public static void Filelist(FileSystem fs, Path path)
{//遍历目录中所有文件
    try{
        FileStatus[] status2 = fs.listStatus(path);
        Path[] listedPaths = FileUtil.stat2Paths(status2);
        for(Path p: listedPaths){
            //文件路径信息
            ...
            if(fs.getFileStatus(p).isDir())
                Filelist(fs, p); //递归显示子目录内的信息
        }
    } catch (Exception e) {
        e.printStackTrace();
    }
    ...
}
//将用户提交的文件上传至 HDFS
Configuration config = new Configuration();
FileSystem fs = FileSystem.get(config);
fs = FileSystem.get(conf);
Path srcPath = new Path(src); //源路径
Path dstPath = new Path(dst); //目标路径
fs.copyFromLocalFile(srcPath, dstPath);
...
//下载文件
fs.copyToLocalFile(srcPath, dstPath);
...

```

2.3.2 集群管理模块

集群管理模块的主要功能是查看和维护集群中服务器信息,管理集群节点的规模,并对任务状态进行监控,对各类故障进行报警处理,主要方式为消息提示和定时发送电子邮件。查看集群中节点状况的主要代码如下:

```

FileSystem fs = FileSystem.get(conf);
DistributedFileSystem hdfs = (DistributedFileSystem) fs;
DatanodeInfo[] datanode = hdfs.getDataNodeStats();
...

```

```
String[] names = new String[datanode.length];
//节点名称数组
String[] status = new String[datanode.length];
for(int i = 0; i<datanode.length; i++)
{
    names[i] = datanode[i].getHostName();
    status[i] = datanode[i].getDatanodeReport();
    //节点信息
}
...
}
```

2.3.3 用户管理模块

该模块主要实现对注册用户管理的基本操作,只有通过审核并激活的用户才有对资源的申请及相应的文件和目录操作的权限. 界面主要包含注册用户的类别、空间的大小分配、激活状态、审批状态等.

2.3.4 部门权限管理模块

该模块的主要功能是通过用户对所属部门的权限进行设定来控制部门内各成员的系统使用权限. 用户的访问权限主要包含对文件上传、下载、共享和删除的权限,目录的创建和修改等操作的权限.

3 实验

为使用户得到更好的体验效果,要最大程度地提高系统传输性能. 通常,影响 HDFS 集群数据传输速率的因素主要包括:网络环境的状况、集群中硬盘的接口类型、客户端并发数量、HDFS 配置等. 在网络和服务器设备既定的情况下,设计 3 组实验,分别从传输文件的大小、集群节点的数量和数据块设置的大小 3 个方面寻找提高集群数据读写速率的途径.

3.1 实验环境

本系统集群环境是由在两台服务器上创建的 7 个虚拟机组成,每个虚拟机作为集群中的一个节点. 其中,1 个节点作为主节点,1 个节点作为客户端,其余节点作为从节点. 每个节点的软硬件环境均相同,其配置情况见表 1.

表 1 集群节点配置

Tab. 1 Configurations of each node

项目	规格
CPU	Intel® Xeon® CPU, E5506, 2.13 GHz, 2 CORE
内存	2 GB, 1 333 MHz
硬盘	SATA, 200 GB
OS	RedHat Enterprise Linux Server release 5.3
JDK	1.6.0_24
Hadoop	hadoop-0.20.1

3.2 实验结果

3.2.1 传输文件大小对集群读写速率的影响

在 HDFS 集群中对不同大小文件的读写速率进行实验. 读写速率是多组实验数据的平均值,部分实验结果如图 3 所示. 可以看出,读取速率大于写入速率,系统对小文件的读写速率较低,当文件大小为 128 MB 时,读写速率均达到最大值.

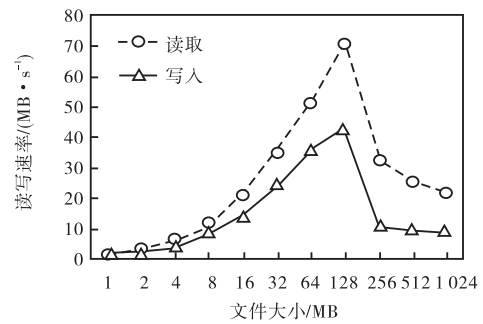


图 3 不同大小文件的读写速率

Fig. 3 Reading and writing rate of the system with different sizes of files being transferred

3.2.2 节点数量对集群读写速率的影响

对不同节点数量的读写速率进行实验,使用的传输文件大小为 128 MB,实验结果如图 4 所示. 可以看出,集群中的节点数量对读取速率影响相对较小,数据的写入速率会随着节点数量的增加而减小. 当节点数为 1 时,即数据块备份全部存储在一个从节点中,省去了从节点之间的通信及数据块的拷贝所消耗的时间. 所以,单个从节点的数据写入速率比多个从节点的数据写入速率明显偏大.

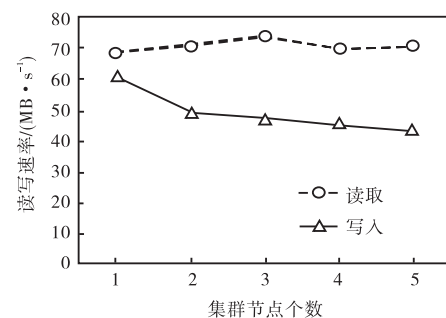


图 4 节点数量与读写速率的关系

Fig. 4 Relation between node numbers and reading-writing rate

3.2.3 数据块的大小对集群读写速率的影响

HDFS 集群将文件分割为数据块存储在从节点上,文献[9]对 HDFS 中块大小的设置进行了说明,并将 HDFS 默认的块大小设置为 64 MB. 为探讨数据块的大小对集群的读写速率的具体影响程度,将集群

中数据块大小分别设置为 16 MB, 32 MB, ..., 256 MB, 传输文件大小设定为 128 MB, 进行实验. 实验结果如图 5 所示. 可以看出, 数据块的大小对集群的数据读写速率影响不大.

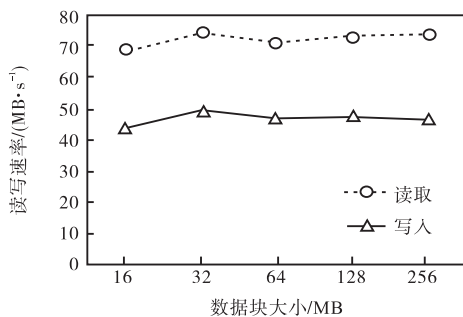


图 5 数据块大小与读写速率的关系

Fig. 5 Relation between block sizes and reading-writing rate

由以上 3 个实验可知, 集群节点数量的增加会造成文件写入速率的小幅下降, 集群中数据块大小的设定对系统读写速率影响偏小, 但传输文件的大小对读写速率的影响较大. 针对此问题, 可以配置系统允许用户上传文件大小的最大值为系统写入速率达到峰值时的文件大小(本实验条件的测试结果为 128 MB), 以提高用户提交数据块的写入速率. 当用户提交的单个大文件或批量小文件的数据量超过此阈值时, 系统 HdfsClient 层的文件上传控件先将用户待上传的资料分割压缩成固定大小, 然后按照多文件上传形式依次提交. 同时, 集群节点数的扩充有利于扩充集群存储容量, 但会对数据写入速率造成一定影响, 因此, 对不同的应用环境要进一步权衡.

4 结 语

本文在云存储技术的基础上提出了校园云存储服务的设计, 并进行了实验, 以分析影响系统读写速率的因素. 从实验结果可看出, 在小文件传输方面 HDFS 的读写速率相对较低. 所以, 相对于校内论

坛、学生博客等以小数据量传输为主的应用, 本系统更适合于影视在线等涉及较大文件传输的校内网版块. 与校园现有的单个服务双机热备策略相比, 云存储服务平台可以成功地整合校内多种存储资源, 将它们的存储空间化零为整, 从而提高资源利用率, 降低运行和维护成本. 目前, 本系统使用的底层 HDFS 集群在传输小文件的读写速率上还不够高, 应当继续对 HDFS 的通信机制和作业调度算法进行改进, 使之能够更好地为师生服务, 促进绿色存储.

参考文献:

- [1] Zhao Yuelong, Ou Kairi, Zeng Wenying, et al. Research on cloud storage architecture and key technologies[C]// Proceedings of 2nd International Conference on Interaction Sciences: Information Technology, Culture and Human. New York: Association for Computing Machinery, 2009: 1044-1048.
- [2] 刘金芝, 余丹, 朱率率. 一种新的云存储服务模型研究[J]. 计算机应用研究, 2011, 28(5): 1869-1872.
- [3] 孙香花. 云计算研究现状与发展趋势[J]. 计算机测量与控制, 2011, 19(5): 998-1001.
- [4] 张先锋, 邹蕾. 云计算技术及其应用研究[J]. 计算机与数字工程, 2011, 39(10): 194-197.
- [5] 王佳隼, 吕智慧, 吴杰, 等. 云计算技术发展分析及其应用探讨[J]. 计算机工程与设计, 2010, 31(20): 4404-4409.
- [6] 刘斌, 朱海涛, 王晓翠. 基于 XCP 的虚拟化云平台建设研究[J]. 现代图书情报技术, 2011(6): 59-65.
- [7] Venner J. Pro Hadoop[M]. New York: Apress, 2009.
- [8] 金松昌. 基于 HDFS 的多用户并行文件 IO 的设计与实现[D]. 长沙: 国防科学技术大学, 2010.
- [9] White T. Hadoop: The Definitive Guide[M]. Sebastopol: O'Reilly Media Inc, 2009: 42-43, 49-50.
- [10] Brock Noland. Mounting HDFS[EB/OL]. [2012-01-20]. <http://wiki.apache.org/hadoop/mountableHDFS>.

责任编辑: 常涛