第 6 卷第 2 期
1995 年 2 月

软 件 学 报
JOURNAL OF SOFTWARE

Vol. 6, No. 2
Feb. 1995

异构分布式文件系统 GSHDFS 名字服务的设计与实现*

康德华　杨学良

(中国科学院研究生院计算机系，北京 100039)

摘要　名字服务是异构分布式文件系统的核心部分. 在参阅国内外众多分布式文件系统的基础上，提出一种新的命名方案——输出共享方式，在保留原文件系统命名语法的前题下，将本地文件系统与远程文件系统有机地结合起来，并采用混和的分布方式实现了名字服务员，达到了位置透明性要求.

关键词　异构分布式文件系统，名字服务，名字服务员，位置透明性.

1. 异构分布式计算机系统

分布式计算机系统是计算机研究领域中的一个新分支，是计算机科学与工程应用技术不断向前发展的必然结果. 分布式计算机系统需要解决的一个核心问题就是分布式操作系统的设计. 分布式操作系统以全局方式管理系统中的所有资源，向用户提供透明的服务，使用户感受不到底层的分布式硬件，而象是运行一个虚拟的单机系统. 系统中各个节点保持高度自治并通过分布式操作系统透明地使用其它机器上的资源.

至今，国内外已有数 10 个分布式操作系统问世，其中绝大多数是同构分布式操作系统，即分布式系统中各个节点机器都是同种类型的，并各自运行同一个分布式操作系统的内核. 而现实的应用环境中，用户接触到的不同类型的多种计算机和操作系统. 因此，将这些异构的计算机通过局域网互联起来组成一个异构的分布式计算机系统，使系统资源充分利用，将是特别有意义的.

异构分布式操作系统用于管理各种异构计算机资源，协调各结点计算机的分工与合作. 设计一个异构分布式操作系统除了要解决一般分布式系统所面临的各种问题，还要解决由于系统的异构性所带来的种种问题. 分布式系统的异构性是用户有效和透明访问全局资源的障碍，是异构分布式系统所要解决的主要问题之一.

2. 异构分布式操作系统 GSHDOS

GSHDOS (Graduate School Heterogeneous Distributed Operating System)是由中国科

* 本文 1992－12－21 收到，1993－05－24 定稿

本课题受国家自然科学基金资助.作者**康德华**，1966 年生，助教，主要研究领域为分布式计算机系统，多媒体系统. **杨学良**，1936 年生，教授，主要研究领域为分布式计算机系统，分布式多媒体系统.

本文通讯联系人：杨学良，北京 100039，中国科学院研究生院计算机系

学院研究生院计算机学部研究开发的异构分布式操作系统.该系统是在借鉴原有的同构分布式操作系统的基础上设计研制的.整个异构分布式系由 3 台 ALTOS68000、3 台 IBM－PC 和 1 台 VAX－11/750 通过美国 Proteon 公司的 Pronet 网络联结组成.在这些机器上分别运行 UNIX、DOS、VMS 操作系统.

在我们的系统中,异构主要表现为两个方面:计算机硬件体系结构的异构和各计算机所用操作系统的异构性.通过 GSHDOS 的设计,我们较好地解决了系统的异构性问题.

GSHDOS 异构分布式系统按照层次模型结构可分为 5 层:硬件层、本地操作系统与网络协议层、RPC 层、异构分布式文件系统及应用环境层.异构分布式操作系统由本地操作系统与网络协议层、RPC 层、异构分布式文件系统 GSHDFS 组成.

3. 异构分布式文件系统 GSHDFS

异构分布式文件系统 GSHDFS(Graduate School Heterogeneous Distributed File System)是异构分布式操作系统 GSHDOS 的重要子系统.

GSHDFS 的设计充分利用了 GSHDOS RPC 子系统支持的顾客—服务员模型.系统设计按请求与提供服务的关系分为名字服务、文件服务和命令服务的设计,见图 1.本文主要讨论 GSHDFS 名字服务的有关问题.

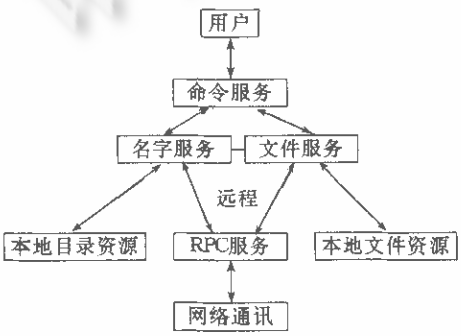

图1 GSHDFS结构

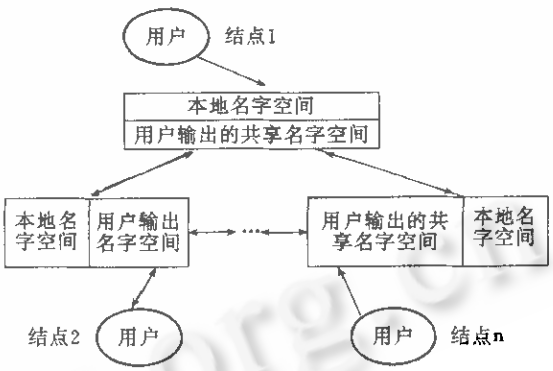

图2 输出共享方式的名字空间组织方式

4. GSHDFS 名字服务

在文件系统中,用户通过文件名来对文件进行访问.各种文件系统对文件名都有它自己的名字语法规划,由系统提供的名字服务(目录服务)完成由文件名到文件位置的映射.

在 GSHDFS 中,由于本地操作系统的异构性,各异构文件系统的文件命名规则也有很大的差异.构成 GSHDFS 的 UNIX、XENEX 和 DOS 虽都采用了树形目录结构,但它们都采用了不同的文件命名语法.象 UMIX,XENIX 操作系统中的文件/usr/kdh/quickc/my.c 在 DOS 操作系统中表示为 C:\USR\KDH\QUICKC\MY.C.因此,如何在不改变本地文件命名规则的情况下,将这些异构的文件系统合理地组织起来,同时完成顾客进程的名字服务请求,是设计异构分布式文件系统 GSHDFS 名字服务所要解决的主要问题.GSHDFS 名字服务包括命名方案的设计和名字服务的实现.

1　GSHDFS 命名方案

在分布式文件系统中,目前主要有 3 种命名方案:(1)超级虚根方式;(2)逻辑安装方式;

(3)全局方式.有关它们的讨论见文献[9].

在GSHDFS中我们设计一种新的命名方案—输出共享方式.这种方式将整个分布式文件系统的文件分为两类:一类是用户本地文件,另一类是用户输出的共享文件,见图2.

对各异构文件系统,我们保留本地文件系统的命名规则.在UNIX(XENIX)文件系统中,用户输出的共享目录在目录/HDFS中,其它为本地文件系统的文件,见图3.

在DOS文件系统中,用户输出的共享文件在目录C:\HDFS下,其它为本地文件系统的文件,见图4.

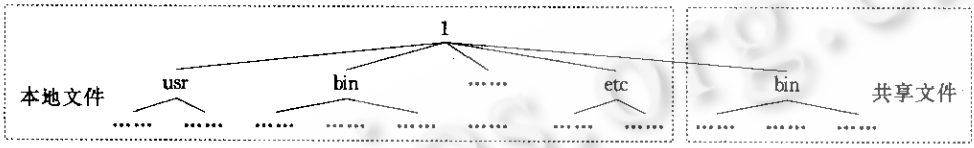

图3 UNIX(XENIX)名字空间组织方式

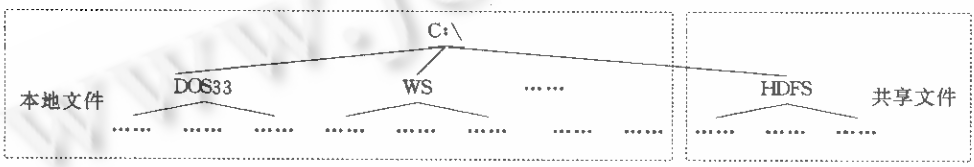

图4 DOS名字空间组织方式

在异构分布式系统任何一个结点上的用户,可以使用命令expdir把他希望所有结点共享的目录输出到异构分布式文件系统中.在每个结点上,系统把用户输出的目录设置在HDFS子目录中,用户使用本地文件系统的命名语法就可访问整个分布式文件系统的文件.如在UNIX(XENIX)文件系统中,用户使用"/hdfs+路径"即可访问其它结点输出的共享文件,在DOS文件系统中,用户使用"C:\HDFS+路径"可访问到共享文件.对于HDFS目录下的文件,用户并不知道它们来自哪个结点,其中有的可能就是本地结点输出的,有的可能是由远程结点输出的.对于用户来讲,HDFS中的文件如同本地子目录中的文件一样,实现了位置透明的要求.

输出共享方案的提出是由于下面的考虑:

1.在各个结点的本地文件系统中,有许多目录和文件是只供本地文件系统使用的.因此,建立一个全局文件系统将所有的文件统一管理起来很不必要,即使能统一管理起来,维持一个庞大的全局文件系统的开销也是非常巨大的.我们观点是异构分布式文件系统只管理各结点希望共享的目录和文件.

2.一些用户的文件都放在各结点的本地磁盘上,这些文件包括一些执行文件、库函数、数据文件和其它一些私用文件.对于这些文件,文件属主并不希望其它用户共享,因此有必要将本地文件与共享文件区别开来.

采用输出共享方式的命名方案,GSHDFS把整个异构分布式文件系统中的共享文件有效地组织起来.

2 GSHDFS名字服务的实现

分布式系统的名字服务可以用3种方式来实现:(1)集中方式;(2)非冗余的分布方式;

(3)冗余的分布方式. 有关这 3 种实现方式的评述见文献[9].

我们采用了分布方式来实现系统的名字服务. 采用这种方式,在分布式系统的每个结点上都有一个名字服务员,它是系统初启时创建的,它管理本地结点输出的共享目录信息和其它结点输出共享目录的根目录信息,响应本地顾客进程的名字服务请求并通过在远程结点注册的服务例程(以 RPC 作为与远程服务例程的通讯手段)完成对远程结点的名字服务请求. 为了加快本地顾客进程对远程结点输出目录的访问速度,每个名字服务员都管理一个远程目录缓冲区(remote directory cache). 缓冲区中存放远程结点的输出目录文件. 因此,一些输出目录文件分布在不同的结点上,从名字信息冗余的角度看,这部分信息是冗余的,但每个结点的名字服务员并没有维护系统中所有输出目录的全部信息,因此它即不同于冗余的分布式,也不同于非冗余的分布式,我们称之为混合分布方式.

2.1　用户输出共享目录

当结点名为 YANG 上的一个用户想要系统共享其本地文件系统的一个目录时,他可以用 expdir path export_name 命令把结点 YANG 上本地路径名为 path 的目录输出到整个分布式文件系统中. 其它结点的用户,如果在 UNIX(XENIX)文件系统中,可以用/hdfs/yang/export_name 路径名来访问结点 YANG 上路径为 path 目录中的子目录和文件;DOS 文件系统中的用户可以用 C:\HDFS\YANG\EXPORT_NAME 路径名来访问该输出目录.

通过系统内部信息文件—输出目录索引文件、输出目录子目录索引文件和输出目录文件等文件,系统完成共享目录的输出.

2.2　GSHDFS 名字服务员

在 GSHDFS 中,由名字服务员完成由文件或目录名字到文件或目录位置的映射. 这里的位置是指结点地址以及文件或目录在该结点上的本地路径名. 由于各结点文件系统类型不同,由名字服务员给出的本地路径名应符合文件或目录所在结点的文件系统命名语法.

名字服务对顾客进程提供名字服务原语 lookup(path,&absolute_path,&node_address,&mode). lookup 是个接口程序,在 UNIX 中它与服务员进程以消息方式进行通讯,在 DOS 中则以子程序调用方式进行参数传递,它提供给服务员进程顾客进程需要服务的名字信息,并将服务结果传递到顾客进程.

lookup 返回 LOCAL、REMOTE、EXPORT、SPECIAL 或 ERROR.

(1)lookup 返回 LOCAL,表明 path 为本地结点一个未输出的目录或文件. 此时,absolute_path为本地路径名,mode 为文件或目录的属性.

(2)lookup 返回 EXPORT,表明 path 为本地一个输出的目录或文件,此时 absolute_path 为该文件或目录的本地路径名,mode 为文件或目录的属于性.

(3)lookup 返回 REMOTE,表明 path 为远程结点的一个输出的目录或文此时 absolute_path为该文件或目录在远程结点的"本地路径名",node_address 为远程结点的结点地址,mode 为该文件或目录的属性.

(4)lookup 返回 SPECIAL,表明 path="根目录"、"根目录+hdfs"或"根目录+hdfs+结点名".

(5)lookup 返回 ERROR,表明 path 语法错或指定的文件或目录不存在.

lookup 是个内部服务原语,是供 GSHDFS 文件服务和命令服务使用的.

2.3 名字服务员对远程目录缓冲区的管理

为了提高名字服务员对远程结点名字服务请求的响应速度,GSHDFS 名字服务员将远程结点的输出目录文件副本传输到本地远程目录缓冲区中,加快对远程结点输出目录访问速度,同时减少网络通讯量.

当名字服务员收到一个对远程结点输出目录名字的服务请求,而该输出目录文件又不在本地远程目录缓冲区时,名字服务员需要更新远程目录缓冲区,这时对远程目录缓冲区的管理方法如下:

(1)如果缓冲区中尚有空闲缓冲区记录时,将远程结点的输出目录文件传输至本地,将远程输出目录的有关信息填写缓冲区记录结构中.

(2)如果缓冲区中无空闲缓冲区记录时,表明缓冲区已满.系统根据 LRU 算法淘汰缓冲区中最近一次访问时间最早的远程目录.在远程目录缓冲区的每个记录中有一个域 last_access_time,用来记录最近一次访问该远程目录的时间.名字服务员将其中最近一次访问时间最小的记录用于存放新的远程输出目录信息,同时将远程结点输出目录文件副本传输至本地.将被淘汰的远程输出目录文件删掉.

(3)每当有远程输出目录文件副本传输至本地或从本地删除时,名字服务员都要通过 RPC 请求,通知远程结点的服务例程修改输出目录文件中有关它的副本位置信息.这些副本位置信息主要用于保证输出目录文件与它的副本的一致性.

2.4 系统对输出目录文件一致性的保证

由前几节可以看到,输出目录文件不仅存在于输出目录的结点上,而且也存在于其它结点的目录缓冲区中.当这些结点对输出目录中的文件和目录进行创建、删除操作时,势必会修改输出目录文件的内容.如果这些输出目录文件的内容不一致,将会使顾客进程得到不正确的名字服务,因此系统应当采取一定的措施保证输出目录文件的一致性.

系统采用主结点封锁法来解决输出目录文件的一致性问题.这里的主结点是指输出一个目录的结点,其它在远程目录缓冲区中含有该输出目录文件副本的结点称之为副结点.主结点封锁法过程如下:

1.副结点对输出目录文件的修改过程

(1)当副结点需要对输出目录文件内容进行修改时,首先向主结点发封锁主结点输出目录文件请求.当封锁请求不能满足时,修改操作不能进行.当封锁请求得到满足时,副结点修改输出目录文件的副本文件,同时请求主结点对输出目录文件进行相应的修改.

(2)主结点收到修改请求后,根据输出目录文件中有关副本的位置信息,向所有副结点(除请求修改的副结点)发出修改输出目录文件的请求.

(3)主结点修改输出目录文件.

(4)主结点收到各副结点修改成功的应答消息后,向发出封锁请求的副结点发修改成功的应答消息.

(5)副结点收到主结点发回的应答消息后,请求主结点将输出目录文件解锁.

(6)主结点将输出目录文件成功解锁后,修改过程结束.

2.主结点对输出目录文件的修改过程

(1)主结点封锁输出目录文件.

(2)主结点修改输出目录文件.

(3)主结点根据输出目录文件中有关输出目录文件副本的位置信息,向所有副结点发修改输出目录文件的请求.

(4)主结点收到各副结点对输出目录文件成功修改的消息后,将输出目录文件解锁.修改过程结束.

上述的结点间通讯都是通过 RPC 子系统来进行的,RPC 子系统的执行模型和它的容错机制能保证消息的成功传递和服务例程的成功执行.

采用主结点封锁法,是通过封锁主结点的输出目录文件来实现顾客进程对输出目录文件并发修改操作的串行化的,因而可以解决各结点用户对输出目录文件的并发修改操作而引起的不一致性问题.由于在主本和所有副本文件中只对主本文件进行封锁,因而不会出现死锁问题.

3 结　论

GSHDFS 是在松耦合的异构分布式计算机环境中开发的一个分布式文件系统.在分析国外众多命名方案的基础上,我们提出了一种新的命名方案—输出共享方式.下面是该方案与国外主要命名方案的比较.

1.与 NFS 采用的逻辑安装方式相比,使用方便.在 NFS 中,用户使用任何一个远程目录,需要登入到远程文件系统中,查看/etc/net/export 文件中有哪些目录可以共享,如果拥有足够权限,可以使用 mount 命令将其中的共享目录安装到本地文件系统的一个空目录下才可以使用,使用起来很不方便.在 GSHDGS 中,任何一个结点的共享目录均在本地 HDFS 子目录中,用户访问共享文件,只需在 HDFS 目录下查找,使用本地的一个路径名即可访问远程共享目录.

2.与 Locus 使用的全局方式相比,系统开销小.Locus 维护一个全局文件系统,系统采用统一的命名语法,系统中只有一个/etc、/bin、/usr 目录,位置透明性非常好.但系统维护一个一致的全局名字空间所花的开销是巨大的.在 GSHDFS 中,系统只维护各结点输出的共享目录,因而系统开销比 Locus 小.同时,每个结点既可以只有几个用户输出的共享目录,也可以由超级用户将一个结点的根目录输出使其它结点共享整个文件系统,因此 GSHDFS 具有非常好的灵活性.

从名字服务的实现上看,GSHDFS 采用的混合的分布方式避免了集中方式低效率与低可靠性问题.与非冗余的分布方式比较,减少了查找名字信息的通讯开销以及由此引起的响应速度慢的问题.以非冗余的分布方式实现的名字服务进行远程名字服务时,要向所有名字服务员广播名字服务请求,具有该资源名字信息的远程结点名字服务员向请求结点的名字服务员发送含有名字服务结果的应答信息.GSHDFS 的名字服务员保留了各结点输出目录的根目录信息,通讯上使用基于连接的点点通讯,同时由于使用了远程目录缓冲区,一些名字服务可以在本地进行,加快了对远程名字服务的响应速度,减少了网络通讯量.与冗余的分布方式相比,系统保证冗余名字信息一致性所花的开销要小.冗余的分布方式在每个结点上拥有所有结点的名字信息,因而它的响应速度非常快.但系统为保证这些名字信息的一致

性所花的开销巨大的.而 GSHDFS 只需维护远程目录缓冲区中的输出目录文件,因此系统开销要小,但响应速度比冗余的分布方式慢.

GSHDFS 名字服务是系统文件服务和命令服务的基础,它的成功设计为文件服务和命令服务提供了较好的支持.

参考文献

1 Tanenbaum A S, Renesse Robert van. Distributed operating system. ACM Computing Surveys, 1985,17(4).

2 Levy Eliezer, Silberschatz Abraham. Distributed file system: concepts and examples. ACM Computing Surveys, 1990,22(4).

3 Coulouris G F, Dollimore Jean. Distributed systems: concepts and design. Addison—Wesley Publishing Company, 1988.

4 Weddle H F, Korel Bogdam, Brown W G *et al*. Transparent distributed object management under completely decentralized control. IEEE Trans. on Parallel and Distributed Computing, 1988.

5 Notkin David, Black A P, Lazowska E D *et al*. Interconnecting heterogeneous computer systems. Communications of the ACM, 1988,31(3).

6 Triantafillou Peter. Distributed name management in internet systems: a study of design and performance issue. Journal of Parallel and Distributed Computing, 1990,(9):357—368.

7 Curtis Ronald, Willie Larry. Global naming in distributed systems. IEEE Trans. on Software, 1984.

8 Levine P H. The apollo DOMAIN distributed file system. Distributed Operating System Theory and Practice, 1987. 241—60.

9 康德华,杨学良.分布式文件系统的透明性研究.计算机研究与发展,1993,30(2).

THE DESIGN AND IMPLEMENTATION OF NAMING SERVICE OF HETEROGENEOUS DISTRIBUTED FILE SYSTEM GSHDFS

Kang Dehua Yang Xueliang

(*Department of Computer Science, Graduate School, The Chinese Academy of Sciences, Beijing* 100039)

Abstract Name service is the kernel of heterogeneous distributed file system. On the basis of analyzing many distributed file systems, a new naming scheme has been proposed, that is export—and—share mechanism with which they can perfectly combine the remote file system with local file system in terms of reserving the local file system naming syntax. They implement the name server distributively and satisfy the demand of location transparency.

Key words Heterogeneous distributed file system, naming service, name server, location transparency.