1.9 分布式存储之 MFS

讲师：汪洋

目录

一、什么是 MFS ？

MFS：MooseFS 是一个具备冗余容错功能的分布式网络文件系统，它将数据分别存放在多个物理服务器或单独磁盘或分区上，确保一份数据有多个备份副本，然而对于访问 MFS 的客户端或者用户来说，整个分布式网络文件系统集群看起来就像一个资源一样，从其对文件系统的情况看 MooseFS 就相当于 UNIX 的文件系统

a.高可靠性：每一份数据可以设置多个备份（多分数据），并可以存储在不同的主机上

b.高可扩展性：可以很轻松的通过增加主机的磁盘容量或增加主机数量来动态扩展整个文件系统的存储量

c.高可容错性：我们可以通过对mfs进行系统设置，实现当数据文件被删除后的一段时间内，依旧存放于主机的回收站中，以备误删除恢复数据

d.高数据一致性：即使文件被写入、访问时，我们依然可以轻松完成对文件的一致性快照

- Master 目前是单点，虽然会把数据信息同步到备份服务器，但是恢复需要时间
- Master 服务器对主机的内存要求略高
- 默认 Metalogger 复制元数据时间较长（可调整）

内存使用问题：

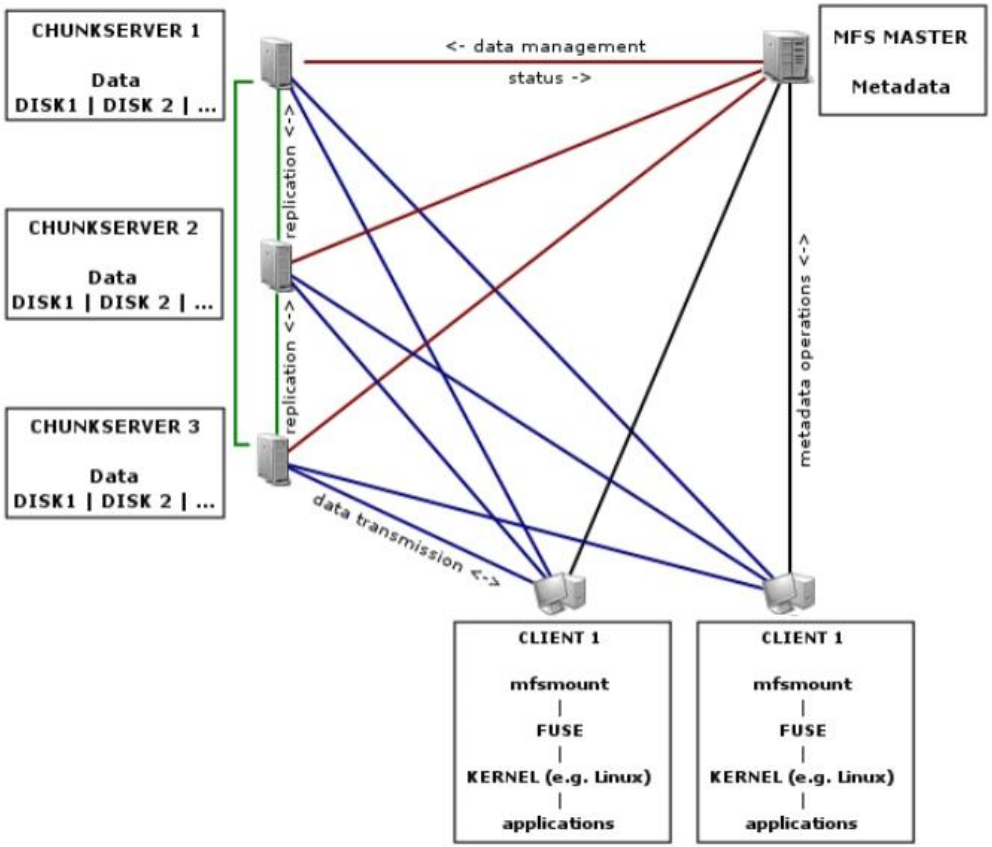
处理一百万个文件chunkserver，大概需要300M的内存空间。据此，推算如果未来要出来1个亿的文件chunkserver，大概需要30G内存空间

应用场景：

I. 大规模高并发的线上数据存储及访问（小文件，大文件都适合）

II. 大规模的数据处理，如日志分析，小文件强调性能不用 HDFS

二、MFS 组件说明


CHUNKSERVER 1
Data
DISK1 | DISK 2 | ...
CHUNKSERVER 2
Data
DISK1 | DISK 2 | ...
CHUNKSERVER 3
Data
DISK1 | DISK 2 | ...
<- data management
status ->
MFS MASTER
Metadata
<-> replication
<-> replication
metadata operations <->
data transmission <->
CLIENT 1
mfsmount
FUSE
KERNEL (e.g. Linux)
applications
CLIENT 1
mfsmount
FUSE
KERNEL (e.g. Linux)
applications

管理服务器 managing server 简称 master ：

这个组件的角色是管理整个mfs文件系统的主服务器，除了分发用户请求外，还用来存储整个文件系统中每个数据文件的 metadata 信息，metadate（元数据）信息包括文件（也可以是目录，socket，管道，块设备等）的大小，属性，文件的位置路径等

元数据备份服务器 Metadata backup servers 简称 metalogger ：

这个组件的作用是备份管理服务器 master 的变化的 metadata 信息日志文件，文件类型为 changelog_ml.*.mfs 。以便于在管理服务器出问题时，可以经过简单的操作即可让新的主服务器进行工作

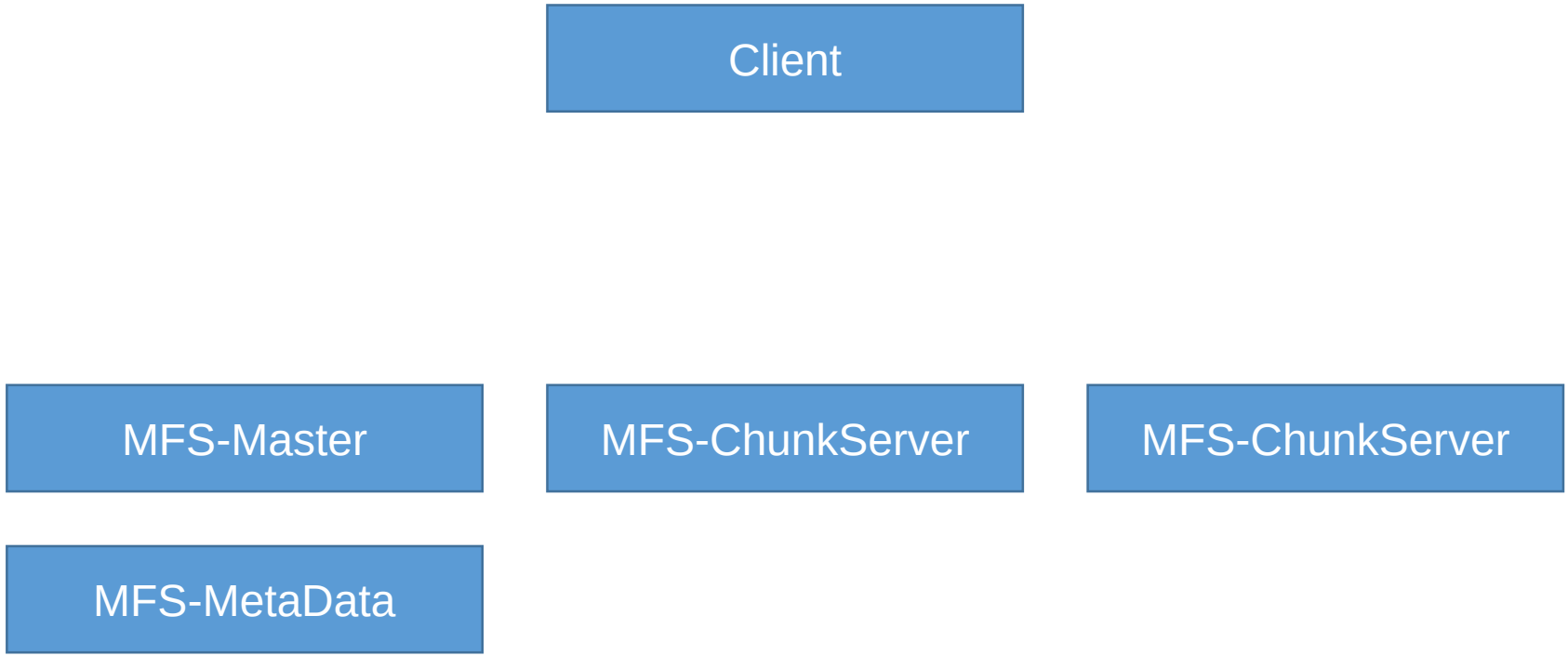
数据存储服务器组 data servers（chunk servers）简称 data：

这个组件就是真正存放数据文件实体的服务器了，这个角色可以有多台不同的物理服务器或不同的磁盘及分区来充当，当配置数据的副本多于一份时，据写入到一个数据服务器后，会根据算法在其他数据服务器上进行同步备份

客户机服务器组（client servers）简称 client：

这个组件就是挂载并使用 mfs 文件系统的客户端，当读写文件时，客户端首先会连接主管理服务器获取数据的 metadata 信息，然后根据得到的 metadata 信息，访问数据服务器读取或写入文件实体，mfs 客户端通过 fuse mechanism 实现挂载 mfs 文件系统的，因此，只有系统支持 fuse，就可以作为客户端访问 mfs 整个文件系统

三、增、删、改、读、遍历


Client
MFS-Master
MFS-ChunkServer
MFS-ChunkServer
MFS-MetaData

Client

MFS-Master

MFS-ChunkServer

MFS-ChunkServer

MFS-MetaData

Client

MFS-Master

MFS-ChunkServer

MFS-ChunkServer

MFS-MetaData

Client

MFS-Master

MFS-ChunkServer

MFS-ChunkServer

MFS-MetaData

Client

MFS-Master

MFS-ChunkServer

MFS-ChunkServer

MFS-MetaData

四、MFS 补充描述

Master 记录着管理信息，比如：文件路径|大小|存储的位置(ip,port,chunkid)|份数|时间等，元数据信息存在于内存中，会定期写入 metadata.mfs.back 文件中，定期同步到 metalogger，操作实时写入 changelog.*.mfs ，实时同步到 metalogger 中。Master 启动将 metadata.mfs 载入内存，重命名为 metadata.mfs.back 文件

文件以 chunk 大小存储，每 chunk 最大为 64M，小于 64M 的，该 chunk 的大小即为该文件大小（验证实际 chunk 文件略大于实际文件），超过 64M 的文件将被切分，以每一份（chunk）的大小不超过 64M 为原则；块的生成遵循规则：目录循环写入(00-FF 256 个目录循环，step 为 2)、chunk 文件递增生成、大文件切分目录连续

Chunkserver 上的剩余存储空间要大于 1GB（Reference Guide 有提到），新的数据才会被允许写入，否则，你会看到 No space left on device 的提示，实际中，测试发现当磁盘使用率达到 95% 左右的时候，就已经不行写入了，当时可用空间为 1.9GB

文件可以有多份 copy，当 goal 为 1 时，文件会被随机存到一台 chunkserver 上，当 goal 的数大于 1 时，copy 会由 master 调度保存到不同的 chunkserver 上，goal 的大小不要超过 chunkserver 的数量，否则多出的 copy，不会有 chunkserver 去存

五、构建 MFS 集群

MFS-Master
10.10.10.11

MFS-ChunkServer
10.10.10.13

MFS-ChunkServer
10.10.10.14

MFS-MetaData
10.10.10.12

MFS-Client
10.10.10.15

六、MFS 维护操作