# 删除

不像一些本地文件系统，只需要释放元数据就可以继续在原位置继续覆盖写入新数据。CephFS 作为一个分布式存储系统，需要将数据和元数据都删除，才能保证不浪费空间。然而当有大量删除的时候，数据的删除会是一个非常耗时的操作，所以实际上 Ceph 实现了延迟删除，先删除元数据，数据并没有被马上删除，而是被移动到了一个名叫 stray 的特殊目录，后续再后台一点点删除。

# 空间统计

CephFS 使用文件的扩展属性 xattr 去实现上述需求，将目录包含的子文件和子目录的使用空间等统计信息保存在xattr 中，当需要获取某个目录的空间使用情况时，只需要通过 [getfattr](https://link.zhihu.com/?target=https%3A//man7.org/linux/man-pages/man1/getfattr.1.html) 获取这个目录的xattr就可以了。

root@node-3:/# getfattr -d -m ceph.dir.\* .

# file: .

ceph.dir.entries="1"

ceph.dir.files="0"

ceph.dir.rbytes="708822611968"

ceph.dir.rctime="1457535283.09430737000"

ceph.dir.rentries="3633611"

ceph.dir.rfiles="3608532"

ceph.dir.rsubdirs="25079"

ceph.dir.subdirs="1"

各字段含义见:

表格

描述已自动生成

# Hardlink

当存在硬链接时，第一个指向Inode的Dentry被称为Primary Dentry，后续的被称为Remote Dentry。

为了访问Remote Dentry， 人们提出了Anchor，它包括Path，Inode，Parent，Ref.前三者显而易见，Ref是被Inode引用的次数（即2.4节提到的计数器）。当进行目录重命名时，可能会影响整个链上的Inode，此时就需要一个事务来保证整个链上相关的Inode同时进行更新，将旧的Ref计数减少，新的Ref计数删除。如果Ref为0，说明Inode已经没有硬链接，可以从Anchor表删除，对Dentry的增加修改同理。

我理解上面的描述应该是之前的版本，最新版本已经废弃了上述的方案？

# 日志

1）更新首先会写入到MDS的日志中；

2）将有改变的元数据标记为“dirty”，并在MDS缓存中置为“pinned”。

3）最终修改会更新到具体元数据对象中，但同时也会做延时处理直到从日志中剪掉，这使得日志可以变得非常大（数百兆）。

# 其他

chattr/lsattr 文件特殊属性 这些属性存放在文件的那个字段中？