一块brick被误从glusterfs volume里删除。但是原volume里在这个brick上的数据都还在。具体情况为：

data01:/gv0

data02:/gv0

其中data02:/gv0被误删除。现从local fs里直接访问data01和data02的gv0的brick目录

data01: /data/local/gv0

data02:/data/local/gv0

这两个目录的树结构是一致的，但是每个子目录中的文件是不同的。欲合并这两个目录，恢复出gluster gv0卷的数据。首先将原始目录拷出来备份到同一台机器上，分别为

gv0-data01/

gv0-data02/

目录，新建合并后的目录

gv0-merged/

理论上，执行 cp –r gv0-data01/\* gv0-data02/\* gv0-merged/即可合并

可以用 cp –r –link gv0-data01/\* gv0-data02/\* gv0-merged/ ，这是用硬连接来避免文件内容的复制，速度很快。

但是实际操作时，gv0-data01/ 和 gv0-data02/ 目录中，存在部分重名文件，经查勘，重名文件的实际内容在某目录，而另一目录只是设置了sticky bit权限的空文件，应该是glusterfs处理DHT冲突的一种手段。因此在合并之前，首先查看两目录中的sticky权限的空文件

find gv0-data01/ -type f –perm 1000 | xargs –l > 1.txt

find gv0-data02/ -type f –perm 1000 | xargs –l > 2.txt

然后执行cp –r –link gv0-data01/\* gv0-data02/\* gv0-merged-test/

会输出冲突文件的覆盖提示（手工敲了几百个n，怎么默认为n，没查命令）

比对1.txt 2.txt以及上述的覆盖提示，确认所有文件match，确认所有权限为1000的文件都为空，之后删除这些空文件

find gv0-data01/ -type f –perm 1000 | xargs rm

find gv0-data02/ -type f –perm 1000 | xargs rm

执行真正的合并目录

cp –r –link gv0-data01/\* gv0-data02/\* gv0-merged/

合并后的数据拷贝或打包到某处存储，完工。