1.flush-table mysql为了提高其性能,部分数据缓存在内存中,因此要刷新表(清除缓存),就需要使用 flushtables.如果是需要备份数据库,同时防止备份时候有新数据写入,且备份是最新的,就需要,flush-tables with read lock.flush privilege 常用,但对于并发很高,缓存很大的数据库,在备份时,都需要加上锁并刷新.

2.高可用系统都是做了主从复制和热备,备份直接让数据库做备份功能,另外磁盘镜像也是一个很好的备份方案.备份就是逐行执行sql.

3. Waiting for table flush,遇到慢查询,很多mysql线程处于此状态,查询语句一直被阻塞,只能通过kill进程解决.

4.官网解释:线程正要刷新表,等待所有的线程关闭表,或者是某个线程收到通知,表明表结构已经发生变化,需要重新打开表,获取新的表结构.但是如果要重新打开表,就必须等到所有的线程都关闭了相关的表.

如果一个线程使用了相关语句,就会产生这种错误:

flush tables tbl\_name,alter table,rename table ,repair table, analyze table,optimize table. 刷新表索引,插入表,重命名表,修复表,分析表,优化表

5.flush table 就是关闭查询的表,并且刷新查询缓存,如果有LOCK TABLES ... READ存在则不允许.如果需要同时进行flush和实现READ LOCK,可以使用FLUSH TABLES tbl\_name with READ LOCK.

我们考虑使用innodb\_filr\_per\_table方式建立的INNODB表,每个表都有一个相应的数据文件idb,格式文件frm,但是某些数据还是存储在共享表空间的,.如果我们的线程要访问这个表的数据必须要首先打开这些文件然后通过pread()/read()/seek()等系统调用进行文件位置寻找和查询.那么我们打开文件的系统调用应该是open()系统调用.

这些必要有一个Linux系统及编程基础,每个进程都包含PCB(PCB在内核态空间,进程间是共享的),PCB是一个进程存在的标识,存放着进程运行的相关信息,内核中他实际上是一个结构体,其中包含了很多信息, PID,PPID其中PCB中包含了一份文件描述,我们很容易在内核代码task\_struct结构体中,也就是PCB中找到信息.............

MySQL是一个进程多个线程,同样也同样执行Linux同样的.

MySQL打开表,底层必定调用 OPEN()函数

结论:flush\_table会关闭MYISAM和CSV的文件描述符,同时会写脏数据到文件,同时关闭文件描述符,关闭文件.

flush\_table 在innodb 中不会真正的关闭文件描述符,同时也不会写脏数据,但是flush\_table确实会由于innodb中对文件操作而造成堵塞.堵塞等待未futex系统调用.对innidb的用处不大.