



10.1: 都会创建失败。虽然磁盘空间已经释放,但指向该空间的条目依然存在。在创建新文件时,文件系统会认为磁盘该区域仍然存在一个为原名的文件。尽管实际上该文件已经从磁盘上被删除了。这会导致这一片磁盘空间永远不能再使用。

解决方法: 删除文件时,不立即释放空间,而是检查是否还有其他链接指向该文件。如果存在则不完全删除文件。

提供事务支持: 引入事务机制, <sup>(释放空间)</sup> 确保文件删除和链接删除是一个原子操作, 如果只进行了一个操作, 那么整个操作回滚, 以保证系统的一致性。

10.2. 应该每个用户拥有一张文件打开表, 理由如下:

每个用户拥有自己的文件打开表, 对文件的操作不会影响其他用户。

每个用户的使用权限不同, 每个用户使用有一张文件打开表更容易实现对不同用户权限的控制。

需要有单独的条目。不同用户的权限可能不同。只有一个条目法实现两个不同权限用户同时合法的对同一个文件进行访问。同时只有一个条目操作系统无法对各进程或用户进行管理。<sup>按他们权限</sup>

11.1 FAT结构提高了系统的可靠性。FAT结构提供了对链表完整性的检查, 这样系统可以更容易检测出文件系统由于硬件或软件所引起起的指针错误。

FAT相对于传统链式结构相对加快了随机访问速度, 因为不需要





一个读块只需要读FAT表就可以快速找到。

11.2.

可以给较大文件分配大的块,来减少管理开销,给小文件分配小块来减少内碎片。

修改空闲空间管理方案以使其支持多种大小的块

11.3

由于没有重定位或基址寄存器,可以创建一个新的文件空间,将原文件拷贝连续分配到新的空间,注意同时需要更新各种引用,最后释放掉原始空间

原因1: 整理和重定位涉及昂贵的开销,需要占用大量系统资源

原因2: 整理和重定位会使得磁盘的读写头大量移动,会影响磁盘寿命。

