

盘配额文件中提取出来的。当所有文件关闭时，该记录被写回配额文件。

当在打开文件表中建立一新表项时，会产生一个指向所有者配额记录的指针，以便很容易找到不同的限制。每一次往文件中添加一块时，文件所有者所用数据块的总数也增加，引发对配额硬限制和软限制检查。可以超出软限制，但硬限制不可以超出。当已达到硬限制时，再往文件中添加内容将引发错误。同时，对文件数目也存在着类似的检查。

当用户试图登录时，系统核查配额文件，查看该用户文件数目或磁盘块数目是否超过软限制。如果超过了任一限制，则显示一个警告，保存的警告计数减1。如果该计数已为0，表示用户多次忽略该警告，因而将不允许该用户登录。要想再得到登录的许可，就必须与系统管理员协商。

这一方法具有一种性质，即只要用户在退出系统前消除所超过的部分，他们就可以在一次终端会话期间超过其软限制；但无论什么情况下都不能超过硬限制。

#### 4.4.2 文件系统备份

比起计算机的损坏，文件系统的破坏往往要糟糕得多。如果由于火灾、闪电电流或者一杯咖啡泼在键盘上而弄坏了计算机，确实让人伤透脑筋，而且又要花上一笔钱，但一般而言，更换非常方便。只要去计算机商店，便宜的个人计算机在短短一个小时之内就可以更换（当然，如果这发生在大学里面，则发出订单需3个委员会的同意，5个签字要花90天的时间）。

不管是硬件或软件的故障，如果计算机的文件系统被破坏了，恢复全部信息会是一件困难而又费时的的工作，在很多情况下，是不可能的。对于那些丢失了程序、文档、客户文件、税收记录、数据库、市场计划或者其他数据的用户来说，这不啻为一次大的灾难。尽管文件系统无法防止设备和介质的物理损坏，但它至少应能保护信息。直接的办法是制作备份。但是备份并不如想象得那么简单。让我们开始考察。

许多人都认为不值得把时间和精力花在备份文件这件事上，直到某一天磁盘突然崩溃，他们才意识到事态的严重性。不过现在很多公司都意识到了数据的价值，常常把数据转到磁带上存储，并且每天至少做一次备份。现在磁带的容量大至几十甚至几百GB，而每个GB仅仅需要几美分。其实，做备份并不像人们说得那么烦琐，现在就让我们来看一下相关的要点。

做磁带备份主要是要处理好两个潜在问题中的一个：

- 1) 从意外的灾难中恢复。
- 2) 从错误的操作中恢复。

第一个问题主要是由磁盘破裂、火灾、洪水等自然灾害引起的。事实上这些情形并不多见，所以许多人也就不以为然。这些人往往也是以同样的原因忽略了自家的火灾保险。

第二个原因主要是用户意外地删除了原本还需要的文件。这种情况发生得很频繁，使得Windows的设计者们针对“删除”命令专门设计了特殊目录——“回收站”，也就是说，在人们删除文件的时候，文件本身并不真正从磁盘上消失，而是被放置到这个特殊目录下，待以后需要的时候可以还原回去。文件备份更主要是指这种情况，这就允许几天之前，甚至几个星期之前的文件都能从原来备份的磁带上还原。

为文件做备份既耗时间又费空间，所以需要做得又快又好，这一点很重要。基于上述考虑我们来看看下面的问题。首先，是要备份整个文件系统还是仅备份一部分呢？在许多安装配置中，可执行程序（二进制代码）放置在文件系统树的受限制部分，所以如果这些文件能直接从厂商提供的CD-ROM盘上重新安装的话，也就没有必要为它们做备份。此外，多数系统都有专门的临时文件目录，这个目录也不需要备份。在UNIX系统中，所有的特殊文件（也就是I/O设备）都放置在/dev目录下，对这个目录做备份不仅没有必要而且还十分危险——因为一旦进行备份的程序试图读取其中的文件，备份程序就会永久挂起。简而言之，合理的做法是只备份特定目录及其下的全部文件，而不是备份整个文件系统。

其次，对前一次备份以来没有更改过的文件再做备份是一种浪费，因而产生了增量转储的思想。最简单的增量转储形式就是周期性地（每周一次或每月一次）做全面的转储（备份），而每天只对当天更改的数据做备份。稍微好一点的做法只备份自最近一次转储以来更改过的文件。当然了，这种做法极大地缩减了转储时间，但操作起来却更复杂，因为最近的全面转储先要全部恢复，随后按逆序进行增量转储。为了方便，人们往往使用更复杂的增量转储模式。

第三，既然待转储的往往是海量数据，那么在将其写入磁带之前对文件进行压缩就很有必要。可是