

第4章 文件系统

所有的计算机应用程序都需要存储和检索信息。进程运行时，可以在它自己的地址空间存储一定量的信息，但存储容量受虚拟地址空间大小的限制。对于某些应用程序，它自己的地址空间已经足够用了；但是对于其他一些应用程序，例如航空订票系统、银行系统或者公司记账系统，这些存储空间又显得太小了。

在进程的地址空间上保存信息的第二个问题是：进程终止时，它保存的信息也随之丢失。对于很多应用（如数据库）而言，有关信息必须能保存几星期、几个月，甚至永久保留。在使用信息的进程终止时，这些信息是不可以消失的，甚至，即使是系统崩溃致使进程消亡了，这些信息也应该保存下来。

第三个问题是：经常需要多个进程同时存取同一信息（或者其中部分信息）。如果只在一个进程的地址空间里保存在线电话簿，那么只有该进程才可以对它进行存取，也就是说一次只能查找一个电话号码。解决这个问题的方法是使信息本身独立于任何一个进程。

因此，长期存储信息有三个基本要求：

- 1) 能够存储大量信息。
- 2) 使用信息的进程终止时，信息仍旧存在。
- 3) 必须能使多个进程并发存取有关信息。

磁盘（magnetic disk）由于其长期存储的性质，已经有多年的使用历史。磁带与光盘虽然也在使用，但它们的性能很低。我们将在第5章学习更多有关磁盘的知识，但目前我们可以先把磁盘当作一种固定块大小的线性序列，并且支持如下两种操作：

- 1) 读块 k ；
- 2) 写块 k 。

事实上磁盘支持更多的操作，但只要有了这两种操作，原则上就可以解决长期存储的问题。

不过，这里存在着很多不便于实现的操作，特别是在有很多程序或者多用户使用着的大型系统上（如服务器）。在这种情况下，很容易产生一些问题，例如：

- 1) 如何找到信息？
- 2) 如何防止一个用户读取另一个用户的数据？
- 3) 如何知道哪些块是空闲的？

就像我们看到的操作系统提取处理器的概念来建立进程的抽象，以及提取物理存储器的概念来建立进程（虚拟）地址空间的抽象那样，我们可以用一个新的抽象——文件来解决这个问题。进程（与线程）、地址空间和文件，这些抽象概念均是操作系统中最重要的概念。如果真正深入理解了这三个概念，那么读者就迈上了成为一个操作系统专家的道路。

文件是进程创建的信息逻辑单元。一个磁盘一般含有几千甚至几百万个文件，每个文件是独立于其他文件的。文件不仅仅被用来对磁盘建模，以替代对随机存储器（RAM）的建模，事实上，如果能把每个文件看成一种地址空间，那么读者就离理解文件的本质不远了。

进程可以读取已经存在的文件，并在需要时建立新的文件。存储在文件中的信息必须是持久的，也就是说，不会因为进程的创建与终止而受到影响。一个文件应只在其所有者明确删除它的情况下才会消失。尽管读写文件是最常见的操作，但还存在着很多其他操作，其中的一些我们将在下面加以介绍。

文件是受操作系统管理的。有关文件的构造、命名、存取、使用、保护、实现和管理方法都是操作系统设计的主要内容。从总体上看，操作系统中处理文件的部分称为文件系统（file system），这就是本章的论题。

从用户角度来看，文件系统中最重要的是它在用户眼中的表现形式，也就是文件是由什么组成的，怎样给文件命名，怎样保护文件，以及可以对文件进行哪些操作等。至于用链表还是用位图来记录空闲