

4.3 文件系统的实现

现在从用户角度转到实现者角度来考察文件系统。用户关心的是文件是怎样命名的、可以进行哪些操作、目录树是什么样的以及类似的界面问题。而实现者感兴趣的是文件和目录是怎样存储的、磁盘空间是怎样管理的以及怎样使系统有效而可靠地工作等。在下面几节中，我们会考察这些文件系统的实现中出现的问题，并讨论怎样解决这些问题。

4.3.1 文件系统布局

文件系统存放在磁盘上。多数磁盘划分为一个或多个分区，每个分区中有一个独立的文件系统。磁盘的0号扇区称为主引导记录（Master Boot Record, MBR），用来引导计算机。在MBR的结尾是分区表。该表给出了每个分区的起始和结束地址。表中的一个分区被标记为活动分区。在计算机被引导时，BIOS读入并执行MBR。MBR做的第一件事是确定活动分区，读入它的第一个块，称为引导块（boot block），并执行之。引导块中的程序将装载该分区中的操作系统。为统一起见，每个分区都从一个启动块开始，即使它不含有一个可启动的操作系统。不过，在将来这个分区也许会有一个操作系统的。

除了从引导块开始之外，磁盘分区的布局是随着文件系统的不同而变化的。文件系统经常包含有如图4-9所列的一些项目。第一个是超级块（superblock），超级块包含文件系统的所有关键参数，在计算机启动时，或者在该文件系统首次使用时，把超级块读入内存。超级块中的典型信息包括：确定文件系统类型用的魔数、文件系统中数据块的数量以及其他重要的管理信息。

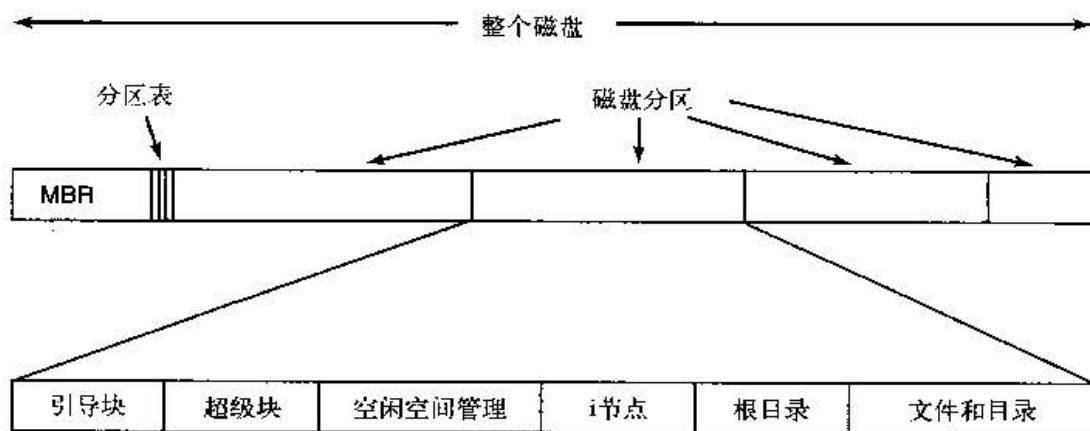


图4-9 一个可能的文件系统布局

接着是文件系统中空闲块的信息，例如，可以用位图或指针列表的形式给出。后面也许跟随的是一组i节点，这是一个数据结构数组，每个文件一个，i节点说明了文件的方方面面。接着可能是根目录，它存放文件系统目录树的根部。最后，磁盘的其他部分存放了其他所有的目录和文件。

4.3.2 文件的实现

文件存储的实现的关键问题是记录各个文件分别用到哪些磁盘块。不同操作系统采用不同的方法。这一节，我们讨论其中的一些方法。

1. 连续分配

最简单的分配方案是把每个文件作为一连串连续数据块存储在磁盘上。所以，在块大小为1KB的磁盘上，50KB的文件要分配50个连续的块。对于块大小为2KB的磁盘，将分配25个连续的块。

在图4-10a中是一个连续分配的例子。这里列出了头40块，从左面从0块开始。初始状态下，磁盘是空的。接着，从磁盘开始处（块0）开始写入长度为4块的文件A。紧接着，在文件A的结尾开始写入一个3块的文件B。

请注意，每个文件都从一个新的块开始，这样如果文件A实际上只有 $3\frac{1}{2}$ 块，那么最后一块的结尾会浪费一些空间。在图4-10中，一共列出了7个文件，每一个都从前面文件结尾的后续块开始。加阴影是为了容易表示文件分隔，在存储中并没有实际的意义。