

Windows的安全性和功能。NTFS使用64位磁盘地址并且（理论上）能够支持最大 $2^{64}$ 字节的磁盘分区，尽管还有其他因素会限制磁盘分区大小。

因为NTFS文件系统是一个带有很多有趣的特性和创新设计的现代文件系统，在本章中我们将针对NTFS文件系统进行讨论。NTFS是一个大而且复杂的文件系统；由于篇幅所限，我们不能讨论其所有的特性，但是接下来的内容会使读者对它印象深刻。

### 11.8.1 基本概念

NTFS限制每个独立的文件名最多由255个字符组成；全路径名最多有32 767个字符。文件名采用Unicode编码，允许非拉丁语系国家的用户（如希腊、日本、印度、俄罗斯和以色列）用他们的母语为文件命名。例如， $\phi\tau\lambda\epsilon$ 就是一个完全合法的文件名。NTFS完全支持区分大小写的文件名（所以foo与Foo和FOO是不同的）。Win32 API不完全支持区分大小写的文件名，并且根本不支持区分大小写的目录名。为了保持与UNIX系统的兼容，当运行POSIX子系统时，Windows提供区分大小写的支持。Win32不区分大小写，但是它保持大小写状态，所以文件名可以包含大写字母和小写字母。尽管区分大小写是一个UNIX用户非常熟悉的特性，但是对一般用户而言，这是很不方便的。例如，现在的互联网在很大程度上是不区分大小写的。

与FAT32和UNIX文件不同，NTFS文件并不只是字节的一个线性序列，而是一个文件由很多属性组成，每个属性由一个字节流表示。大部分文件都包含一些短字节流（如文件名和64位的对象ID），和一个包含数据的未命名的长字节流。当然，一个文件也可以有两个或多个数据流（即长字节流）。每个流有一个由文件名、一个冒号和一个流名组成的名字，例如，foo:stream1。每个流有自己的大小，并且相对于所有其他的流都是可以独立锁定的。一个文件中存在多个流的想法在NTFS中并不新鲜。苹果Macintosh的文件系统为每个文件使用两个流，一个数据分支（data fork）和一个资源分支（resource fork）。NTFS中多数数据流的首次使用是为了允许一个NT文件服务器为Macintosh用户提供服务。多数数据流也用于表示文件的元数据，例如Windows GUI中使用的JPEG图像的缩略图。但是，多数数据流很脆弱，并且在传输文件到其他文件系统，通过网络传输文件甚至在文件备份和后来恢复的过程中都会丢失文件。这是因为很多工具都忽略了它们。

与UNIX文件系统类似，NTFS是一个层次化的文件系统。名字的各部分之间用“\”分隔，而不是“/”，这是从MS-DOS时代与CP/M相兼容的需求中继承下来的。与UNIX中当前工作目录的概念不同的是，作为文件系统设计的一个基础部分的链接到当前目录（.）和父目录（..）的硬连接，在Windows是作为一种惯例来是实现的。系统仅在其中的POSIX子系统里支持硬连接，正因为这样，NTFS支持对目录的遍历检查（UNIX中的“x”权限）。

从Windows Vista开始，NTFS才开始支持符号链接。为了避免如Spoofing这样的安全问题（当年在UNIX 4.2BSD第一次引入符号链接时就遇到过），通常只允许系统管理员来创建符号链接。在Vista中符号链接的实现用到一个叫再解析点（reparse points）的NTFS特性（将在本节后续部分讨论）。另外，NTFS也支持压缩、加密、容错、日志和稀疏文件。我们马上就会探讨这些特性及其实现。

### 11.8.2 NTFS文件系统的实现

NTFS文件系统是专门为NT系统开发的，用来替代OS/2中的HPFS文件系统的。它是一个具有很高复杂性和精密性的文件系统。NT系统的大部分是在陆地上设计的。从这方面看，NTFS与NT系统其他部分相比是独一无二的，因为它的很多最初设计都是在一艘驶出普吉特湾的帆船的甲板上完成的（严格遵守上午工作，下午喝啤酒的作息协议）。

接下来，我们将从NTFS结构开始，探讨一系列NTFS特性，包括文件名查找、文件压缩、日志和加密。

#### 1. 文件系统结构

每个NTFS卷（如磁盘分区）都包含文件、目录、位图和其他数据结构。每个卷被组织成磁盘块的一个线性序列（在微软的术语中叫“簇”），每个卷中块的大小是固定的。根据卷的大小不同，块的大小从512字节到64KB不等。大多数NTFS磁盘使用4KB的块，作为有利于高效传输的大块和有利于减少内部碎片的小块之间的折中办法。每个块用其相对于卷起始位置的64位偏移量来指示。

每个卷中的主要数据结构叫MFT（主文件表，Master File Table），该表是以1KB为固定大小的记录