重要数据文件的恢复技术①

周维贵, 褚 剑, 王海慰, 赵 萌

(西昌卫星发射中心, 西昌 615000)

摘 要: 在进行硬盘数据恢复时, 如果每次都从头到尾地进行扫描, 需要耗费大量的时间, 而且恢复的数据大部分是不需要的. 为了节约时间提高效率, 应对特定的重要数据文件进行恢复. 论文从硬盘启动及文件读写的原理出发, 分析文件被修改或删除时操作系统所做的操作, 对比分析文件分配表(FAT)和文件目录表(FDT)在特定文件修改或删除时产生的变化, 针对现有的数据恢复技术无法自动恢复不连续簇文件的问题, 通过 FAT 表内连续簇的分布规律来进行特定文件恢复, 并在实验中进行验证.

关键词: 数据恢复, 文件分配表, 文件目录表, 特定文件恢复

Important Data File Restoring Technology

ZHOU Wei-Gui, CHU Jian, WANG Hai-Wei, ZHAO Meng

(Xichang Satellite Launch Center, Xichang 615000, China)

Abstract: During data recovery, it would cost a lot of time to scan from begin to end, and most of the recovered data would be not required. Restoring the particular data files can be useful to save time and improve efficiency. Paper begins with presenting the principle of file reading and writing, and analyzes the OS' actions when the file is modified or deleted, as well as the changes of the File Associate Table (FAT) and File Directory Table (FDT). Aiming at the problem that the current data recovery method is not in effect for the lost files without continuous clusters, we can do the job for the particular file according to its clusters' distribution characteristics, and paper tests it in some experiment.

Key words: data recovery; FAT; FDT; particular file recovery

1 引言

在试验任务和日常办公的过程中, 各种数据被存储在计算机的硬盘里, 它们一般以文件的形式存在, 比如日常办公用的 word 文档、火箭飞行弹道数据文件、气象信息数据文件等等. 在组成计算机的各部件中, 只有硬盘是机械式部件, 其使用寿命相对其他电子部件而言要短. 正常情况下, 各类数据文件在计算机断电后不会丢失, 但在硬盘出现故障无法启动, 或者用户进行了误操作的情况下, 就可能导致数据文件丢失[1], 如果存有关键数据的文件丢失, 很可能对试验任务的数据判读和历史数据分析造成影响.

为了保证重要数据和重要文件的安全以及保密的需要, 单位成立了数据恢复与销毁中心, 专门针对硬盘等存储介质进行销毁或数据恢复. 目前, 数据恢复与销毁中心对数据恢复采用的方法一般是将硬盘或硬盘的某一个分区从头到尾扫描一遍, 来查找其中所包含的文件, 但在工作中发现, 这种扫描效率是很低的, 如果硬盘可以正常工作, 其扫描速度约是 5Mb/s, 那么 100G 的硬盘需要 6 个小时, 而现在一般的硬盘都达到 500G 的容量, 扫描一遍就需要 30 小时; 如果硬盘较老化或坏道较多, 其读写速度急剧下降, 约为 58Kb/s, 那么 100G 的硬盘就需要 478 个小时, 即 20 天, 500G 的硬盘扫描下来需要的时间更无法忍受, 很明显, 这样的扫描方法不现实, 而且, 扫描结束后, 扫描到的数据大部分并不是用户想要的, 比如将硬盘送来恢复的用户往往关注的是盘内的 word、excel 等办公

①收稿时间:2014-03-17;收到修改稿时间:2014-04-23

© 中国科学院软件研究所 http://www.c-s-a.org.cn

文档, 或者是固定位置的几个存有重要数据的文件等, 而扫描出来的文件很多是系统运行时产生的零碎文件, 从中查找用户需要的文件效率也非常低, 所以, 需要用其他的方法来针对特定的重要文件进行准确定位, 有针对性地进行恢复.

文件数据丢失的原因主要分两大类, 一类是硬盘无法启动或损坏, 从硬盘启动到文件读写, 这一过程中涉及到五个环节, 任何一个环节出问题都可能导致文件数据丢失, 可以针对每一个环节进行恢复, 确保文件能够读写成功. 另一类是人为地误操作导致, 现在有很多用于恢复此类数据的软件, 比如 Easy Recovery, FinalData 等, 但这类软件都存在一个共同的问题, 对文件目录表(FDT)里文件起始簇号被修改、以及不连续簇的文件无法恢复, 需要用其他方法来进行恢复.

2 硬盘无法读取文件的分析与恢复

2.1 从硬盘启动到文件读写的过程

硬盘从启动到读取其中的文件, 这一过程涉及到五个环节: 读取主引导扇区(Boot Sector)、读取操作系统引导记录(DBR)、读取文件目录表(FDT)、读取文件分配表(FAT)、读取文件数据存储区[2][3]. 其过程如图 1 所示:

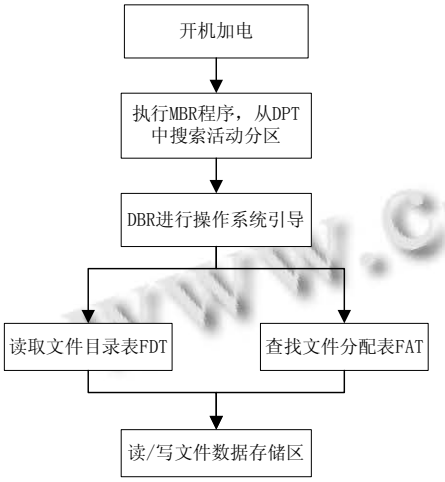

图 1 从硬盘启动到文件读写的流程图

在开机加电自检后, CPU 从内在地址 0ffff:0000 处开始执行, 将硬盘第一个扇区(0 面 0 磁道 1 扇区)读入内存地址 0000:7c00 处, 然后执行 MBR 中的程序, 即在主分区表(DPT)中搜索活动分区, 找到 DBR 进行操作系统引导, 从 DBR 中找到 BPB 中记录的本分区的起始扇区、结束扇区、文件存储格式、根目录等等信息, 并进入系统[4].

在成功进入操作系统后, 读写某一个文件时, 需要先从文件目录表(FDT)中找到该文件的目录项, 获取该文件的起始簇号, 并通过文件分配表(FAT)中的相应簇号信息找到下一簇号, 直至文件结束, 然后到数据存储区按这一系列簇号进行数据读写.

这个过程中的任何一个环节出问题, 都会导致文件读取失败, 因此, 可以针对每一个环节对硬盘进行恢复.

2.2 主引导扇区(Boot Sector)的分析与恢复

主引导扇区也就是硬盘的第一个扇区(0 面 0 磁道 1 扇区), 它由主引导记录(MBR)和硬盘主分区表(DPT)两部分组成.

其中, MBR 里存放的系统主引导程序主要用于检查分区表是否正确, 确定哪个分区为引导分区, 在 Dos 系统下用 fdisk /mbr 可以重建标准的主引导记录程序.

分区表(DPT)占用 64 个字节, 记录了磁盘的基本分区信息, 分为四个分区项, 分别记录每个主分区的信息, 当分区表遭到破坏时, 系统找不到主分区不能引导系统, 常用的重建分区表的软件有 Partition Magic, DiskGenius 等, 可以自动检测硬盘分区参数, 修复分区表[5].

2.3 操作系统引导记录(DBR)的分析与恢复

操作系统引导记录(Dos Boot Record)通常位于硬盘的 0 磁道 1 柱面 1 扇区, 是操作系统可以直接访问的第一个扇区, 它包括一个引导程序和一个本分区参数记录表(BPB). 引导程序的主要任务是当 MBR 把系统控制权交给它时, 判断本分区根目录前两个文件是不是操作系统的引导文件, 如果是, 将其读入内存, 并把控制权交给该文件. BPB 参数块记录着本分区的起始扇区、结束扇区、文件存储格式、硬盘介质描述符、根目录大小、FAT 个数、分配单元的大小等重要参数.

在分区进行格式化时, 一般都会在六个扇区对 DBR 做一个备份, 当 DBR 损坏时, 可以使用 DiskEdit、WinHex 等硬盘编辑软件对 DBR 所在的扇区进行恢复.

2.4 文件分配表(FAT)的分析与恢复

为了实现文件的链式存储, 硬盘上必须准确地记

© 中国科学院软件研究所 http://www.c-s-a.org.cn

录哪些簇已经被文件占用，还必须为每个已经占用的簇指明存储后继内容的下一簇号，对一个文件的最后一个簇，则要指明本簇无后续簇. 这些都是由 FAT 表来保存的，FAT 表中有很多表项，每项占四字节，用于记录一个簇的信息.

由于 FAT 表对于文件管理的重要性，所以 FAT 表有一个备份，即在原 FAT 表的后面再建一个同样的 FAT 表，当主用 FAT 表遭到破坏时，可以用备份的 FAT 表进行恢复.

2.5 文件目录表(FDT)的分析与恢复

FDT 所在的位置并不固定，任何一个子目录都有它自己的目录区，该子目录下的所有文件的相关信息都在该目录区中存储. 每一个目录项占 32 字节，用于存储一个文件的信息，包括文件名、文件扩展名、文件属性、建立时间、起始簇号、文件长度等等. 其中值得注意的是起始簇号是分两小块来存储的，在相对位移为 14H－15H 的两个字节里存储的是文件起始簇号的高 16 位，在 1AH-1BH 两个字节里存储的是文件起始簇号的低 16 位，这两块儿组合起来共同构成文件起始簇号.

对于 FDT 表，操作系统并没有进行备份，因此当 FDT 表遭到破坏时，只能手动地根据实际情况来对其进行恢复，实际上，对于 FDT 表的第一个表项，只需要参考其他项恢复其重要的几个字节的信息即可，一般是采用小范围的扫描来重新获取这一块区域里曾经存在的目录项.

3 误操作丢失文件的分析与恢复

针对一个特定的文件，当出现误操作致使文件丢失时，如误删除、格式化等，文件的数据并没有丢失，只是修改了 FDT 和 FAT 等地方的相关项，所以，只要采取相应的措施，就可以将误操作丢失的文件恢复.

这里对文件的新建、修改和删除操作进行实验，通过 winhex 实时观察硬盘在上述文件操作时发生的变化，并分析如何对文件进行恢复.

3.1 新建文件的过程分析

在硬盘的一个分区里新建一个 test 目录，并在该目录下新建一个文件 test-text.txt，在文件里输入文本，大小为 8736B. 用 winhex 打开该分区，可以看到在该分区的根目录表下有一项为 test，并从该目录项中得到其起始簇号为 000E 0124H，即十进制的 917796，由于是目录，所以其文件大小区域为全 0. 进入 FAT 表里对应该簇号的表项，可见该簇标记为已用，并且标记为最后一个簇，即该 test 子目录的下级文件目录项占用这一个簇. 如图 2:

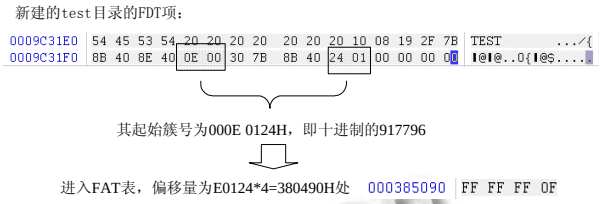

图 2　新建文件夹的目录项及 FAT 表项

进入硬盘的第 917796 号簇，即可看到子目录 test 的下层文件目录项结构. 由 FDT 可知，其下属的文件 test-text.txt 的目录项就在 test 目录的内容里，由于文件名较长，系统自动将其文件名以 Unicode 类型单独存放在该目录项的前面.

通过目录项里的数据，相对偏移为 14H-15H 处为 0E 00，1AH-1BH 处为 13 03，这里低位在前，高位在后，因此组合起来，就得到该文件的起始簇号为 000E 0313 H. 相对偏移为 1C-1FH 处为 20 22 00 00，表示该文件的大小为 2220H 字节，即十进制的 8736 字节.

转到 FAT 表，从 FAT 表的开始向后定位偏移量为 E0313*4=380C4C 字节处，可以看到 test-text.txt 文件对应的 FAT 表项，可以看出，文件的起始簇号为 918291，即 0000E0313H，指向下一块簇号为 000E0314H，即 918292，最后一块的簇号为 918293，在 FAT 表里即是 FFFFFF0F. 转到该文件的起始簇（第 918291 号），就可以查看文件的内容了. 如 Error! Reference source not found.：

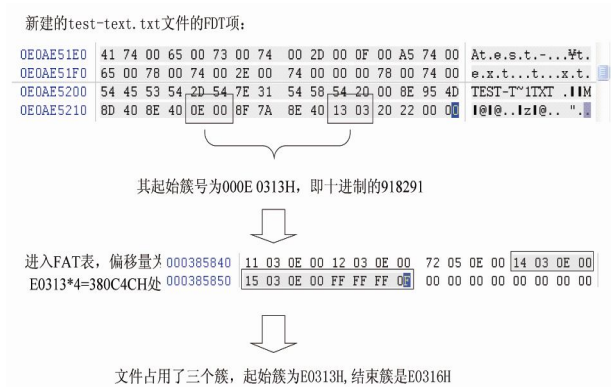

图 3　新建文件的目录项及 FAT 表项

至此，我们可以得知，在新建了该目录和文件后，系统在硬盘里自动新建了相关目录项和修改 FAT 表里

© 中国科学院软件研究所 http://www.c-s-a.org.cn

的相关信息.

通过类似的实验，可以得知，当文件内容被修改时，比如内容扩充，系统会自动地从 FAT 中找到没有使用的簇，并将其填入扩充的内容，同时在 FAT 表里进行标记，先将原文件的最后一个簇用扩充的内容填满，然后链接至剩余新内容的第一个簇，并在文件结束时标记为结束簇，以此维持该文件的链式结构. 而在 FDT 目录项中，除了文件大小和修改时间等信息发生变化外，其他信息如起始簇号信息不变.

3.2 删除文件的过程分析与文件恢复

很多文献资料指出删除文件时所做的操作仅仅是将文件 FDT 目录项的首字节置为 E5H，表示该项为空闲，同时，置该文件对应的所有 FAT 表项为空，但在实验的过程中发现有时会有所偏差，有时 FDT 项里文件对应的起始簇号的高位被清 0，在这里，将 test-text.txt 文件删除进行实验，可以得知，在文件被删除后，数据区里的数据原封不动，而 FDT 和 FAT 表里做了相应的改变，其中, FAT 里涉及到该文件的所有簇对应的表项都被清空，FDT 里除了第一字节被置为 E5 外，其 14H－15H 偏移处也被清空. 如图 4:

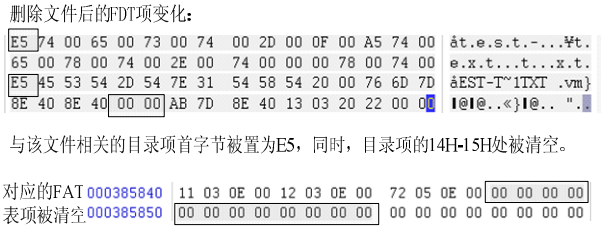

图 4 删除文件后的目录项及 FAT 表项

此时，若采用一般软件如 easyrecovery 或 finaldata 来恢复，就会得到一个文件大小为 8736 字节恢复就无法找到文件数据区的起始簇号的高位 2 字节被清空，如果按一般的做法来也无法恢复真正的数据.

针对这种情况进行文件恢复时，需要将该目录项中的 14H－15H 偏移处的两字节进行推测恢复. 由于硬盘在给文件分配空间时，是以尽量完整并且较近的原则来分配的，而 14－15H 偏移处所代表的是起始簇号的高 16 位，因而，硬盘在分配空间时起始簇号的高 16 位一般不变，所以可以借鉴该目录下最新的文件的目录项的这两个字节，直接将其填充至待恢复文件的目录项中.

此时，再用软件或者手动恢复，就可以找回原来的文件了. 但遇到特殊情况如分配空间时刚好需要翻页，即之前的文件刚好占满 FFFF 个簇，即 650MB 时，这个起始簇号的高 16 位会改变，，这种方法需要做适当的修改，如果借用最新文件的这两个字节后找到的起始簇号不对，那么可以尝试将这两个字节的数字加一，再进行尝试，这样的情况出现概率极小，可以忽略.

另外，根据实验得知，在刚删除一个文件之后，再新建文件，很有可能会马上覆盖掉刚刚删掉的文件，而且，由于文件的空间分配是以簇为单元，而擦写过程是以扇区为分配单元，因此不管新建的文件有多小，都会至少占用一个簇的空间，至少覆盖一个扇区的数据，若新文件的大小小于一扇区，即 512B，系统的操作过程是将新文件的内容写入该扇区，并将该扇区剩余的空间置为 0. 因此，在误操作删掉文件之后，千万不要再向该硬盘分区里写入任何数据，应在第一时间对该文件进行恢复.

4 针对不连续簇文件的恢复

在实验过程中我们发现，当文件较大，利用的硬盘空间不连续时，用一般的数据恢复软件如 Easy Recovery, FinalData 等进行恢复，往往得不到完整的文件，原因在于它们都只是从 FDT 项中读取到起始簇号和文件大小，然后去数据区对应的起始簇开始复制与文件大小相同的长度的数据，并没有考虑文件的簇没有连在一起的情况. 而这种情况目前尚没有统一的解决方案，本文在实验中尝试找到这种文件的簇在分配和删除时的规律，进行一定条件下的文件完整恢复.

系统在对文件进行分配空间时，先遵循力争完整的规则，若进行文件扩充时，扩充的新内容也力争完整地放在一块区域，因此，对于扩充的文件，其对应的 FAT 表也是由不同的整块整块区域组成的. 在数据密度较大的硬盘区域，当删除一个不连续簇的文件时(以两个簇块儿的文件为例), FAT 表里对应的表项就会很有规律地清零.

实验过程如下:

将 test-text.txt 文件先扩充为 53212B，在 FAT 表项中可以看到其空间分为两个隔得不远的簇块儿，将该文件删除后, FAT 表内与文件相关的项均被清空，如图 5.

如图，不连续簇文件在删除后，会在 FAT 表里留下这样的痕迹，在进行这类文件的恢复时，可以遵循如下

© 中国科学院软件研究所 http://www.c-s-a.org.cn

流程:

(1)通过 FDT 目录项找到文件的起始簇号;

(2)然后转到 FAT 表，查看该簇号所对应的文件分配表项是否为空，计算连续为空的簇数，并用文件大小相减，得到剩下的不连续的空间大小;

(3)继续查找 FAT 表，找到下一块儿清空的簇空间，将其大小与文件剩余空间大小进行对比，若差距在一个簇的范围以内，则暂定此簇块儿为候选块儿;

(4)尝试将其内容拷贝出来，查看恢复出来的文件是否正确，若不正确，则重复向后查找，直至找到剩余簇为止.

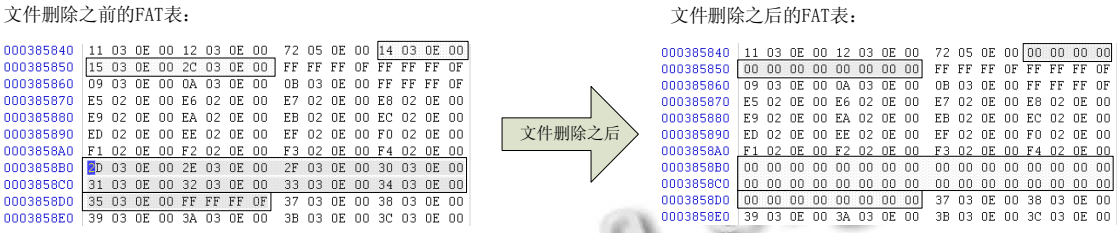

图 5　非连续簇文件删除前后的 FAT 表对比

根据以上流程，可以将指定文件的 FDT 和 FAT 表项都进行恢复，重新启动 explorer 进程，即可发现 test 目录中的 test-text.txt 文件又出现了，打开后文件内容正确，说明该方法有效.

5　结论

在试验任务和日常办公中会产生很多数据文件，有些文件非常重要，若丢失则可能影响到试验任务的进行，因此当重要文件丢失时需要对其进行恢复. 但目前所采用的文件恢复技术一般是对硬盘进行扫描，效率低，成功率也不高，为了节约时间提高效率，需要针对特定的重要数据文件进行恢复，论文分析了从硬盘启动到文件读写的过程，并通过一系列实验分析了文件新建和删除时操作系统所做的操作，对比文件分配表(FAT)和文件目录表(FDT)在特定文件修改或删除时产生的变化，针对现有的数据恢复技术无法自动恢复 FDT 起始簇号被清空和不连续簇文件的问题，找到 FDT 中分配起始簇号的特性和 FAT 表内连续簇的分布规律，并以此来进行特定文件的恢复，通过实验证明该方法行之有效，能有效地从数以万计的文件中快速恢复特定的重要文件，能大大提高恢复的效率，为数据恢复工作节省大量的时间.

参考文献

1 张彬.软硬兼施——硬盘固件维修及数据恢复实战.北京:清华大学出版社,2010.

2 Bhushan B. Tribology and Mechanics of Magnetic Storage Devices. Springer-Verlag, New York, 1990.

3 李培.硬盘数据恢复技术的研究实践.制造业自动化,2010,32(12).

4 傅建明,彭国军,张焕国.计算机病毒分析与对抗.武汉:武汉大学出版社,2004.

5 陈培德,殷莉芬.硬盘主引导扇区分析及恢复.云南大学学报(自然科学版),2010,32.

© 中国科学院软件研究所 http://www.c-s-a.org.cn