**[linux硬盘分区讲解](http://blog.csdn.net/engerled/article/details/6182323)**

2011-02-13 14:08 1918人阅读 [评论](http://blog.csdn.net/engerled/article/details/6182323#comments)(0) [收藏](javascript:void(0);) [举报](http://blog.csdn.net/engerled/article/details/6182323#report)

[linux](http://www.csdn.net/tag/linux)[command](http://www.csdn.net/tag/command)[扩展](http://www.csdn.net/tag/%e6%89%a9%e5%b1%95)[磁盘](http://www.csdn.net/tag/%e7%a3%81%e7%9b%98)[ide](http://www.csdn.net/tag/ide)[action](http://www.csdn.net/tag/action)

 　 在学习 Linux 的过程中，安装 Linux 是每一个初学者的第一个门槛。在这个过程中间，最大的困惑莫过于给硬盘进行分区。虽然，现在各种发行版本的 Linux 已经提供了友好的图形交互界面，但是很多的人还是感觉无从下手。这其中的原因主要是不清楚 Linux 的分区规定，以及它下面最有效的分区工具― Fdisk 的使用方法。

     首先我们要对硬盘分区的基本概念进行一些初步的了解，硬盘的分区主要分为**基本分区（primary partion）**和**扩充分区(extension partion)**两种，**基本分区和扩充分区的数目之和不能大于四个**。且 基本分区可以马上被使用但不能再分区。扩充分区必须再进行分区后才能使用，也就是说它必须还要进行二次分区。那么由扩充分区再分下去的是什么呢？它就是逻 辑分区（logical partion），况且逻辑分区没有数量上限制. （硬盘分区为基本分区和扩充分区，扩充分区下有可以分为逻辑分区）。

      对习惯于使用dos或windows的用户来说，有几个分区就有几个驱动器，并且每个分区都会获得一个字母标识符，然后就可以选用这个字母来指定在这个分 区上的文件和目录，它们的文件结构都是独立的，非常好理解。但对这些初上手 red hat linux的用户，可就有点恼人了。因为对red hat linux用户来说无论有几个分区，分给哪一目录使用，它归根结底就只有一个根目录，一个独立且唯一的文件结构。red hat linux中每个分区都是用来组成整个文件系统的一部分，因为它采用了一种叫“载入”的处理方法，它的整个文件系统中包含了一整套的文件和目录，且将**一个分区和一个目录联系起来**。这时要载入的一个分区将使它的存储空间在一个目录下获得。

       对windows用户来说，操作系统必须装在同一分区里，它是商业软件! 所以你没有选择的余地！对red hat linux来说，你有了较大的选择余地，你可以把系统文件分几个区来装（必须要说明载入点），也可以就装在同一个分区中（载入点是“/”）。

下面从这两个方面入手，来讲解这个困扰大家的问题。

Linux 的分区规定

　　 1. 设备管理  （设备命名规则）

 在 Linux 中，**每一个硬件设备都映射到一个系统的文件**（一个硬件对应一个设备文件结点），对于硬盘、光驱等 IDE 或 SCSI 设备也不例外。Linux 把各种 IDE 设备分配了一个由 hd 前缀组成的文件；而对于各种 SCSI 设备，则分配了一个由 sd 前缀组成的文件。

对于ide硬盘，驱动器标识符为“hdx~”,其中“hd”表明分区所在设备的类型，这里是指ide硬盘了。“x”为盘号（a为基本盘，b为基本从 属盘，c为辅助主盘，d为辅助从属盘）,“~”代表分区，前四个分区用数字1到4表示，它们是主分区或扩展分区，从5开始就是逻辑分区。例，hda3表示 为第一个ide硬盘上的第三个主分区或扩展分区,hdb2表示为第二个ide硬盘上的第二个主分区或扩展分区。对于scsi硬盘则标识为 “sdx~”，scsi硬盘是用“sd”来表示分区所在设备的类型的，其余则和ide硬盘的表示方法一样，不在多说。

例如，第一个 IDE 设备，Linux 就定义为 hda；第二个 IDE 设备就定义为 hdb；下面以此类推。而 SCSI 设备就应该是 sda、sdb、sdc 等。

         2. 分区数量

         要进行分区就**必须针对每一个硬件设备进行操作（对一个硬盘设备来说）**，这就有可能是一块IDE硬盘或是一块SCSI硬盘。对于每一个硬盘（IDE 或 SCSI）设备，Linux 分配了一个 1 到 16 的序列号码，这就代表了这块硬盘上面的分区号码。

　　例如，第一个 IDE 硬盘的第一个分区，在 Linux 下面映射的就是 hda1，第二个分区就称作是 hda2。对于 SCSI 硬盘则是 sda1、sdb1 等。

3. 各分区的作用

在 Linux 中规定，每一个硬盘设备最多能有 4个主分区（其中包含扩展分区）构成，任何一个扩展分区都要占用一个主分区号码，也就是在一个硬盘中，主分区和扩展分区一共最多是 4 个。  
　　对于早期的 DOS 和 Windows（Windows 2000 以前的版本），系统只承认一个主分区，可以通过在扩展分区上增加逻辑盘符（逻辑分区）的方法，进一步地细化分区。

主分区的作用就是计算机用来进行启动操作系统的，因此每一个操作系统的启动，或者称作是引导程序，都应该存放在主分区上。

这就是主分区和扩展分区及逻辑分区的最大区别。

我们在指定安装引导 Linux 的 bootloader 的时候，都要指定在主分区上，就是最好的例证。  
　　Linux 规定了**主分区（或者扩展分区）占用 1 至 16 号码中的前 4 个号码**。以第一个 IDE 硬盘为例说明，主分区（或者扩展分区）占用了 hda1、hda2、hda3、hda4，而**逻辑分区占用了 hda5 到 hda16 等 12 个号码**。  
　　因此，**Linux 下面每一个硬盘总共最多有 16 个分区**。  
　　对于逻辑分区，Linux 规定它们必须建立在扩展分区上（在 DOS 和 Windows 系统上也是如此规定），而不是主分区上。  
　　因此，我们可以看到扩展分区能够提供更加灵活的分区模式，但不能用来作为 操作系统 的引导。 除去上面这些各种分区的差别，我们就可以简单地把它们一视同仁了。

　　4. 分区指标

       对于每一个 Linux 分区来讲，分区的大小和分区的类型是最主要的指标。容量的大小读者很容易理解，但是分区的类型就不是那么容易接受了。分区的类型规定了这个分区上面的文件系统的格式。  
　　Linux 支持多种的文件系统格式，其中包含了我们熟悉的FAT32、FAT16、NTFS、HP-UX，以及各种 Linux 特有的 Linux Native和 Linux Swap分区类型。  
　　在 Linux 系统中，可以通过分区类型号码来区别这些不同类型的分区。各种类型号码在介绍Fdisk的使用方式的时候将会介绍。

   5  常用分区

/boot分区，它包含了操作系统的内核和在启动系统过程中所要用到的文件，建这个 分区是有必要的，因为目前大多数的pc机要受到bios的限制,况且如果有了一个单独的/boot启动分区，即使主要的根分区出现了问题，计算机依然能够 启动。这个分区的大小约在50mb—100mb之间。但是如果想用lilo启动red hat linux系统的话，含有/boot的分区必须完全在柱面1023以下。又由于8gb后的数据lilo不能读取，所以red hat linux要安装在8gb的区域以内。

/usr分区，是red hat linux系统存放软件的地方，如有可能应将最大空间分给它。

/home分区，是用户的home目录所在地，这个分区的大小取决于有多少用户。如 果是多用户共同使用一台电脑的话，这个分区是完全有必要的，况且根用户也可以很好地控制普通用户使用计算机，如对用户或者用户组实行硬盘限量使用，限制普 通用户访问哪些文件等。其实单用户也有建立这个分区的必要，因为没这个分区的话，那么你只能以根用户的身份登陆系统，这样做是危险的，因为根用户对系统有 绝对的使用权，可一旦你对系统进行了误操作，麻烦也就来了。

/var/log分区，是系统日志记录分区，如果设立了这一单独的分区，这样即使系统的日志文件出现了问题，它们也不会影响到操作系统的主分区。

/tmp分区，用来存放临时文件。这对于多用户系统或者网络服务器来说是有必要的。 这样即使程序运行时生成大量的临时文件，或者用户对系统进行了错误的操作，文件系统的其它部分仍然是安全的。因为文件系统的这一部分仍然还承受着读写操 作，所以它通常会比其它的部分更快地发生问题。

/bin分区，存放标准系统实用程序。

/dev分区，存放设备文件。

/opt分区，存放可选的安装的软件。

/sbin分区，存放标准系统管理文件。

上面介绍了几个常用的分区，一般来说我们需要一个swap分区，一个/boot分区，一个/usr分区，一个/home 分区，一个/var/log分区。当然这没有什么规定，完全是依照你个人来定的。但记住至少要有两个分区，一个swap分区，一个/分区。

Fdisk 使用详解

下面通过介绍 Fdisk 的使用方法，来巩固上面所学到的各种关于 Linux 分区的知识。  
　　Fdisk 是各种 Linux 发行版本中最常用的分区工具，是被定义为 Expert 级别的分区工具，它让初学者有点望而却步。  
　　1. Fdisk 参数说明  
　 　运行Fdisk的时候，首先映入眼帘的是欢迎界面，用户通过在这个界面中输入命令参数来操作Fdisk。用户通过提示键入 “m”，可以显示 Fdisk 命令各个参数的说明。 读者可以看到 Fdisk 有很多参数，可是经常使用的就是几个，如果读者熟练掌握这几个参数就可以流畅地运用 Fdisk，对 Linux 的硬盘进行分区。我们先简单介绍各个参数的意义，然后详细说明几个重点参数。  
　　用户在 Linux 中进行分区的时候，最常用的参数分别是 d、l、m、n、p、q、t、w 等。

2.用 Fdisk 进行分区

在 Linux 分区过程，一般是先通过 p 参数来显示出硬盘分区表信息，然后根据信息确定将来的分区。如果想完全改变硬盘的分区格式，就可以通过 d 参数一个个删除存在的硬盘分区。

例如 d1，d2。 删除完毕，就可以通过 n 参数来增加新的分区。当按下 “n” 后，我们就可以看到新增的分区。这里要选择新建的分区类型，**是主分区还是扩展分区**；然后选择 p 或是 e。它们的区别在上文中已经说明。然后就是设置分区的大小。需要提醒注意的是，如果硬盘上有扩展分区，就只能增加逻辑分区，不能增加扩展分区了**，（这里的意思是说一个盘只能有一个扩展分区，可以有多个逻辑分区），**在 增加分区的时候，其类型都是默认的 Linux Native，如果需要把其中的某些分区改变为其它类型，例如 Linux Swap 或 FAT32 等，可以通过命令 t 来改变， 当按下 “t” 改变分区类型的时候，系统会提示要改变哪个分区，并且改变为什么类型（如果想知道系统所支持的分区类型，键入 l）。Linux 所支持的分区类型号码和其对应的分区类型，可以参考表 2（这些信息可以用 l 命令得到）。改变完了分区类型，就可以按下 “w”，保存并且退出。如果不想保存，那么可以选择 “q” 直接退出。

fdisk 是一款强大的磁盘操作工具，来自util-linux软件包，我们在这里只说他如何查看磁盘分区表及分区结构；参数 -l ，通过-l 参数，能获得机器中所有磁盘的个数，也能列出所有磁盘分区情况；

[root@localhost beinan]# fdisk -l  
Disk /dev/hda: 80.0 GB, 80026361856 bytes  
255 heads, 63 sectors/track, 9729 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System  
/dev/hda1   \*           1         765     6144831    7  HPFS/NTFS  
/dev/hda2             766        2805    16386300    c  W95 FAT32 (LBA)  
/dev/hda3            2806        7751    39728745    5  Extended  
/dev/hda5            2806        3825     8193118+  83  Linux  
/dev/hda6            3826        5100    10241406   83  Linux  
/dev/hda7            5101        5198      787153+  82  Linux swap / Solaris  
/dev/hda8            5199        6657    11719386   83  Linux  
/dev/hda9            6658        7751     8787523+  83  Linux

在上面Blocks中，表示的是分区的大小，Blocks的单位是byte ，我们可以换算成M，比如第一个分区/dev/hda1的大小如果换算成M，应该是6144831/1024=6000M，也就是6G左右，其实没有这么 麻烦，粗略的看一下把小数点向前移动三位，就知道大约的体积有多大了；  
    System 表示的文件系统，比如/dev/hda1 是NTFS格式的；/dev/hda2 表示是fat32格式的文件系统；.

在此例中，我们要特别注意的是/dev/hda3分区，这是扩展分区；他下面包含着逻辑分区，其实这个分区相当于一个容器；从属于她的有 hda5,hda6,hda7,hda8,hda9 ；

我们还注意到一点，怎么没有hda4呢？为什么hda4没有包含在扩展分区？一个磁盘最多有四个主分区； hda1-4算都是主分区；hda4不可能包含在扩展分区里，另外扩展分区也算主分区；在本例中，没有hda4这个分区，当然我们可以把其中的一个分区设 置为主分区，只是我当时分区的时候没有这么做而已；

再仔细统计一下，我们看一看这个磁盘是不是还有空间？hda1+hda2+hda3=实际已经分区的体积，所以我们可以这样算 hda1+hda2+hda3=6144831+16386300+39728745 = 62259876 (b)，换算成M单位，小数点向前移三位，所以目前已经划分好的分区大约占用体积是62259.876(M)，其实最精确的计算 62259876/1024=60800.67（M）；而这个磁盘大小是80.0 GB （80026361856byte)，其实实际大小也就是78150.744（M）；通过我们一系列的计算，我们可以得出这个硬盘目前还有使用的空间；大 约还有18G未分区的空间；

我们也可以指定fdisk -l 来查看其中一个硬盘的分区情况；

[root@localhost beinan]# fdisk -l /dev/sda

Disk /dev/sda: 60.0 GB, 60011642880 bytes  
64 heads, 32 sectors/track, 57231 cylinders  
Units = cylinders of 2048 \* 512 = 1048576 bytes

   Device Boot      Start         End      Blocks   Id  System  
/dev/sda1               1       57231    58604528   83  Linux

通过上面情况可以知道，在/dev/sda 这个磁盘中，只有一个分区；使用量差不多是百分百了；

我们还可以来查看 /dev/hda的

[root@localhost beinan]# fdisk -l /dev/hda

自己试试看？

我们也可以通过: cat /proc/partitions查看目前机器中的所有磁盘及分区情况

**df 命令**；  
df 是来自coreutils 软件包，系统安装时，就自带的；我们通过**这个命令可以查看磁盘的使用情况以及文件系统被挂载的位置**；

举例：

[root@localhost beinan]# df -lh

Filesystem            容量  已用 可用 已用% 挂载点

/dev/hda8              11G  6.0G  4.4G  58% /

/dev/shm              236M     0  236M   0% /dev/shm

/dev/sda1              56G   22G   35G  39% /mnt/sda1

我们从中可以看到,系统安装在/dev/hda8 ；还有一个56G的磁盘分区/dev/sda1挂载在 /mnt/sda1中；

其它的参数请参考 man df

本文来自CSDN博客，转载请标明出处：<http://blog.csdn.net/aaronychen/archive/2008/04/09/2270048.aspx>

应用fdisk创建U盘分区的过程：

[root@ubuntu](mailto:root@ubuntu):~# fdisk /dev/sdb1                                       //此处的设备是/dev/**sdb1**

WARNING: DOS-compatible mode is deprecated. It's strongly recommended to  
         switch off the mode (command 'c') and change display units to  
         sectors (command 'u').

Command (m for help): p

Disk /dev/sdb1: 516 MB, 516567552 bytes  
16 heads, 62 sectors/track, 1017 cylinders  
Units = cylinders of 992 \* 512 = 507904 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x6f20736b

**This doesn't look like a partition table  
Probably you selected the wrong device.           //注意这两条语句的提示，也由此注意到设备文件错误.应该是/dev/sdb**

     Device Boot      Start         End      Blocks   Id  System  
/dev/sdb1p1   ?      784412     1935127   570754815+  72  Unknown  
Partition 1 has different physical/logical beginnings (non-Linux?):  
     phys=(357, 116, 40) logical=(784411, 3, 11)  
Partition 1 has different physical/logical endings:  
     phys=(357, 32, 45) logical=(1935126, 8, 51)  
Partition 1 does not end on cylinder boundary.  
/dev/sdb1p2   ?      170050     2121692   968014120   65  Novell Netware 386  
Partition 2 has different physical/logical beginnings (non-Linux?):  
     phys=(288, 115, 43) logical=(170049, 14, 47)  
Partition 2 has different physical/logical endings:  
     phys=(367, 114, 50) logical=(2121691, 4, 42)  
Partition 2 does not end on cylinder boundary.  
/dev/sdb1p3   ?     1884962     3836603   968014096   79  Unknown  
Partition 3 has different physical/logical beginnings (non-Linux?):  
     phys=(366, 32, 33) logical=(1884961, 2, 30)  
Partition 3 has different physical/logical endings:  
     phys=(357, 32, 43) logical=(3836602, 7, 39)  
Partition 3 does not end on cylinder boundary.  
/dev/sdb1p4   ?     2908953     2909009       27749+   d  Unknown  
Partition 4 has different physical/logical beginnings (non-Linux?):  
     phys=(372, 97, 50) logical=(2908952, 12, 25)  
Partition 4 has different physical/logical endings:  
     phys=(0, 10, 0) logical=(2909008, 11, 33)  
Partition 4 does not end on cylinder boundary.

Partition table entries are not in disk order

Command (m for help): q

[root@ubuntu](mailto:root@ubuntu):~# fdisk /dev/sdb                                                   **//注意此处的设备文件名**  
You will not be able to write the partition table.

WARNING: DOS-compatible mode is deprecated. It's strongly recommended to  
         switch off the mode (command 'c') and change display units to  
         sectors (command 'u').

Command (m for help): p

Disk /dev/sdb: 516 MB, 516685824 bytes  
17 heads, 62 sectors/track, 957 cylinders  
Units = cylinders of 1054 \* 512 = 539648 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0xe32bfb4e

   Device Boot      Start         End      Blocks   Id  System  
/dev/sdb1   \*           1         958      504460+   b  W95 FAT32  
Partition 1 has different physical/logical beginnings (non-Linux?):  
     phys=(0, 1, 0) logical=(0, 3, 46)  
Partition 1 has different physical/logical endings:  
     phys=(1001, 16, 0) logical=(957, 7, 40)  
Partition 1 does not end on cylinder boundary.

Command (m for help): n                                          //试图加入分区  
Command action  
   e   extended  
   p   primary partition (1-4)  
p                                                                               //选择加入主分区  
Partition number (1-4): 1                                         //选择加入的分区号为1  
Partition 1 is already defined.  Delete it before re-adding it.  //系统提示分区号为1的分区已定义，重加前请删除

Command (m for help): d                                        //删除分区  
Selected partition 1                                                //系统自动的删除分区1

Command (m for help): p                                       //再次显示系统分区

Disk /dev/sdb: 516 MB, 516685824 bytes  
17 heads, 62 sectors/track, 957 cylinders  
Units = cylinders of 1054 \* 512 = 539648 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0xe32bfb4e

   Device Boot      Start         End      Blocks   Id  System

Command (m for help): n                                      //选择添加系统分区  
Command action   
   e   extended  
   p   primary partition (1-4)  
p                                                                           //添加主分区  
Partition number (1-4): 1                                     //添加的主分区号为1  
First cylinder (1-957, default 1):                          //选择1号分区的起始cylinder  
Using default value 1                                           //选择默认值  
Last cylinder, +cylinders or +size{K,M,G} (1-957, default 957): +200M    //选择1号分区的结束cylinder(或者1号分区的大小)

Command (m for help): n                                    //选择继续添加分区  
Command action  
   e   extended  
   p   primary partition (1-4)  
e                                                                         //选择添加extended分区  
Partition number (1-4): 2                                   //extended分区的分区号为2  
First cylinder (391-957, default 391):                //选择2号分区的起始cylinder  
Using default value 391                                     //选择默认值  
Last cylinder, +cylinders or +size{K,M,G} (391-957, default 957):   //选择此分区的结束cylinder   
Using default value 957                                    //选择默认值

Command (m for help): p                                   // 再此显示分区列表

Disk /dev/sdb: 516 MB, 516685824 bytes  
17 heads, 62 sectors/track, 957 cylinders  
Units = cylinders of 1054 \* 512 = 539648 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0xe32bfb4e

   Device Boot      Start         End      Blocks   Id  System  
/dev/sdb1               1         390      205499   83  Linux  
/dev/sdb2             391         957      298809    5  Extended

Command (m for help): n                                   // 选择添加分区  
Command action  
   l   logical (5 or over)                                         //此处已经没有extended分区了，可以选择创建于extended分区下的logic分区  
   p   primary partition (1-4)  
l                                                                         //选择创建logic分区  
First cylinder (391-957, default 391): +100M

Value out of range.  
First cylinder (391-957, default 391): **//在创建logic分区过程中没有让我们再次选择分区号，logic分区号是由系统指点的**  
Using default value 391  
Last cylinder, +cylinders or +size{K,M,G} (391-957, default 957): +100M

Command (m for help): n  
Command action  
   l   logical (5 or over)  
   p   primary partition (1-4)  
l  
First cylinder (586-957, default 586):   
Using default value 586  
Last cylinder, +cylinders or +size{K,M,G} (586-957, default 957):   
Using default value 957

Command (m for help): p                                            //再此显示分区表

Disk /dev/sdb: 516 MB, 516685824 bytes  
17 heads, 62 sectors/track, 957 cylinders  
Units = cylinders of 1054 \* 512 = 539648 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0xe32bfb4e

   Device Boot      Start         End      Blocks   Id  System  
/dev/sdb1               1         390      205499   83  Linux  
/dev/sdb2             391         957      298809    5  Extended  
/dev/sdb5             391         585      102734   83  Linux  
/dev/sdb6             586         957      196013   83  Linux

Command (m for help): w                                              //把新的分区表写入U盘

Unable to write /dev/sdb                     //写入不成功，没找到解决办法，可能是因为我用的U盘做实验的，我的U盘不支持重写分区表  
[root@ubuntu](mailto:root@ubuntu):~#