Go to the first, previous, next, last section, table of contents.

3. BIOSes and Firmware

"BIOS" (Basic Input/Output System) and "firmware" mean the same thing. However, on PC and PC98 based computers, the word BIOS is more common. On Apple Macintosh and Sun computers, the word "firmware" is more common. The BIOS or firmware program is built into a ROM chip inside your computer, that does memory checks, etc. You can not (easily) change this program. Since BIOSes today are generally compatible with BIOSes in use 20 years ago, these programs tend to have an antiquated design. Features have been added in counter-intuitive ways over the years, leading to overly complicated systems that cause a lot of confusion today.

3.1 The PC BIOS

There are a few popular PC BIOSes: AmiBIOS, Award, Phoenix, and others. They all work in a similar way. None of these BIOSes understand or know about partition tables. They affect partitioning indirectly.

The following occurs when your computer boots up:

1. These BIOSes load the first bit of the boot loader program, stored on the MBR (Master Boot Record) on the hard disk.
2. The BIOS executes the first bit of the boot loader.
3. The boot loader program uses the BIOS to load the rest of itself.
4. The boot loader uses the BIOS to load the operating system (or, perhaps another boot loader, in which case you go to step 2 again).
5. The operating system may or may not use the BIOS to do normal file system access (Windows usually does, Linux or BSD do not).

Steps (3) to (5) involve programs communicating with the BIOS, to ask it to talk to the disks. There are two ways of talking to the BIOS to do this: by using CHS (Cylinders Heads and Sectors) or LBA (Linear Block Addressing). Older BIOSes will only support CHS. New BIOSes usually support both LBA and CHS, although CHS support may be phased out in the future. (CHS is generally considered to be a horrible legacy system.)

Steps (3) and (4), performed by the boot loader, will always use the same access method -- either always LBA or always CHS. In the case of the Windows boot loader, this is determined by the LBA flag on the Windows boot partition (see section 2.4.16 set for information about about flags). In the case of Linux, you will probably be using either LILO or GRUB as your boot loader. GRUB uses LBA if it is available, and CHS otherwise. LILO requires that you choose when you install it (with the linear, or lba32 options).

Step (5) - IO done by the operating system - only Windows does its IO through the BIOS. [We still don't know enough about the problems, but it seems Windows can have its own problems with CHS mode. People have told us about occassions where Windows corrupts its own file systems, etc. It's really hard for us to know what's going on. We strongly recommend you use LBA, if you can!]

So, there are 3 possible situations, which are all covered here:

1. You are using CHS mode, and your BIOS only supports CHS mode.
2. You are using CHS mode, and your BIOS supports both CHS and LBA. So, you want to convert from CHS mode to LBA mode As Soon As Possible TM.
3. You are using LBA mode already.

3.1.1 Using Parted with CHS mode BIOS

Linux usually detects the BIOS geometry automatically. However, sometimes it gets it wrong. In this case, you should tell it yourself, by passing a parameter to Linux. For example, if Linux thinks your hard drive `` `/dev/hda' `` has geometry 256/64/63, but the BIOS setup program says the geometry is 512/32/63, then you would pass this parameter to Linux:

```
hda=512,32,63
```

Parameters are passed in different ways, depending what boot-loader you're using. You are probably using LILO. In this case, you add this line to `` `/etc/lilo.conf' ``: (You then need to run @command{/sbin/lilo}, and reboot for the change to take effect)

```
append="hda=512,32,63"
```

Parted can usually detect if Linux has detected the incorrect geometry. However, it cannot do this if there are no partitions on the disk. In this case, you should check yourself. It is very important that you do this.

Sometimes, Parted will complain that the partitions aren't aligned to cylinder boundaries. Parted will give you the option to ignore. If you ignore, then Parted will do some tricks with your partition table. Linux will have no problems. DOS and Windows will have no problems if they are using LBA mode. If DOS/Windows is using CHS mode, then reinstalling the the boot loader (see section 4.3 Legacy Microsoft Operating System Bootloaders) should solve any problems - but switching to LBA mode is preferrable (see see section 3.1.2 Converting from CHS to LBA mode BIOS).

Partitions that are involved in the boot process must end before cylinder 1024, if CHS mode is being used. See the section on boot loaders to determine if a partition is involved in the boot process. The typical setup is to have a small Linux `` `/boot' `` partition, a Windows partition and then the Linux partition(s).

3.1.2 Converting from CHS to LBA mode BIOS

To make the Windows boot loader + operating system use LBA mode, just set the LBA flag on all FAT partitions section 2.4.16 set. You should have no problems. If you do have problems booting Windows, then reinstalling the Windows boot loader will fix it section 4.3 Legacy Microsoft Operating System Bootloaders.

Linux doesn't use the BIOS for IO. However, the boot loader (LILO or GRUB) may. GRUB automatically uses LBA, if it is available. LILO requires the "linear" or "lba32" option. So, if LILO still boots after changing `` `/etc/lilo.conf' `` reinstalling with @command{/sbin/lilo}, then everything's done! See section 4.1 LILO: a bootloader for the Linux kernel. (If you have problems, just revert back to CHS, by removing "linear" or "lba32", and reinstalling lilo from a boot disk.)

Now that you're using LBA, read on...

3.1.3 Using Parted with LBA mode BIOS

LBA solves all of CHS's problems. However, there is no reliable way for Linux or Parted to know that you're using LBA, so Parted may give you warnings about cylinders not being aligned, or the BIOS geometry being inconsistent, etc. You can ignore these messages if you are using LBA mode. (Old versions of Parted had problems, which have all been solved).

When your disk is in LBA mode, Parted (and most other programs) will say your CHS geometry is X/255/63 - unless you have switched from CHS to LBA.

3.2 OpenFirmware of the Apple Macintosh

There are two major version of the PowerMac's OpenFirmware - one used on "old world", and one on "new world" PowerMac. "New world" refers to coloured PowerPCs manufactured since 1999. They have significant differences. However, both understand partition maps.

Both require that the user choose exactly one partition to be the boot partition (the partition with the boot loader). However, they use different mechanisms to do this.

3.2.1 Old World Macintosh OpenFirmware

The partition chosen to boot is set by various boot loaders, like Quik. So, you shouldn't need to do anything. We could add support for this in Parted, if someone screams loudly enough...

3.2.2 New World Macintosh OpenFirmware

New world OpenFirmware requires the boot partition to be HFS and marked as a boot partition. It uses a different mechanism to be marked as the boot partition. This is controlled with Parted's "boot" flag. For example:

```
(parted) set 2 boot on
```

3.3 The PC98 BIOS

The PC98 BIOS allows you to mark any number of partitions as bootable. You can mark or unmark a partition as bootable with Parted's "boot" flag. For example:

```
(parted) set 2 boot off
```

3.4 SGI / MIPS Firmware

The SGI / MIPS firmware allows booting from special boot files, which are managed by the partition table. In Parted, these boot files are treated as logical partitions inside an extended partition.

For example:

```
Disk label type: mips
Minor    Start       End     Type      Filesystem  Name        Flags
9          0.000      2.732  extended
17         0.002      0.002  logical               sgilabel
18         0.003      1.162  logical               symmon
19         1.404      1.670  logical               sash
1          2.732   8555.212  primary   xfs                     root
2       8555.213   8682.270  primary                           swap
```

Partition 9 is the extended partition (volume header in SGI/MIPS terminology) where the boot files may lie. Partitions 17, 18 and 19 are the boot files. There names can be manipulated with parted's name command. Partitions 1 and 2 are normal partitions. They can't have names.

Note that Linux doesn't see the boot files as partitions (maybe it should?). So /dev/hda17 doesn't exist in Linux. You should use dvhtool(8) to manipulate boot files.

Go to the first, previous, next, last section, table of contents.