

Hands on: Putting the boot in

Linux partitions explained, and find out how to dual-boot with Windows

Barry Shilliday, Personal Computer World 09 Feb 2007

For most people new to Linux, their biggest concern will undoubtedly be how to install it alongside Windows. Most people will understandably want to dip their toes in before wiping out what they're accustomed to, and many may still need to use specific Windows applications even if they do decide to switch to Linux full time.

Thanks to what began with the Knoppix distribution, many Linux distributions can boot directly from CD or DVD, allowing users to try things out before committing in any way. These work entirely from CD and don't need to modify the hard drive at all.

Although things are considerably easier than they were a few years ago, when it comes to Linux installation it's a good idea to understand a few concepts about dual-booting if you want to go down that route. Installation programs are generally good enough, but a basic understanding is helpful to ensure you know what you are doing.

If you don't need to dual-boot with Windows and are happy to wipe the hard drive clean, then you'll find that a Linux installation is no more difficult than a Windows one. If anything it's actually easier, especially as driver downloads are generally not needed at all.

Partition basics

On a PC, the hard drive needs to be split up into partitions before it can be used. A partition acts, to all intents and purposes, as a separate hard drive. Each partition can have its own file system, or indeed its own operating system. Linux does, in fact, allow you to use an entire hard drive for data without any partitions, and it even allows you to use other partitioning schemes (such as those used by Macs), but here we will concentrate on a traditional PC setup.

A hard drive can be split into four bootable partitions. These are known as primary partitions. Initially, back in the days of MS-Dos, this was the upper limit. In order to allow for more partitions, it was made possible for one of those primary partitions to be marked as an extended partition.

This itself could then be split up into any number of logical partitions. As a result, a hard drive can have one-to-four primary partitions. Alternatively, it can have one-to-three primary partitions and a number of extra logical partitions. Primary and logical partitions behave the same.

The information about partitions is stored on the master boot record, commonly referred to as the MBR. This is 512 bytes of data, stored in the first sector of the hard drive, outside of any partitions.

The MBR also holds executable code, since the PC must use this to start an operating system. If you install a copy of Windows, the traditional Dos MBR is installed. This is a simple piece of code: it checks each primary partition and whichever is marked as active is then booted.

Every partition can have executable code installed in its first sector, and this code is run on whichever partition is active. One limitation of the Dos MBR is that it can only boot primary partitions.

Linux partitioning

In Microsoft Windows, each partition the operating system can read receives a drive letter (such as C: or D:). These are fairly arbitrary and can change if you add or remove new hardware. They are also mixed in with network drives and other hardware. Any file system that isn't readable won't receive a drive letter at all.

In Linux, the situation is totally different. Every partition is referenced by a device name, which refers directly to the partition number on the hard drive. Primary partitions are numbered one to four, and logical partitions are five and up. Since any hard drive can have up to four primary partitions, the first four numbers are reserved, whether they're in use or not. In other words, if you have two primary partitions and three logical partitions, these will count as partitions 1, 2, 5, 6 and 7.

The device name starts with a prefix of 'hd' for IDE hard drives and 'sd' for any other – such as SCSI, serial ATA (Sata) or USB – hard drives. This is followed by a letter to distinguish one physical drive from another. See box on previous page for a list of common device names.

The sum of all partitions

Combined together, this gives a device name that refers directly to a specific partition – hda2 would be the second primary partition on an IDE drive, hda5 the first logical partition, and sda3 would be the third primary partition on the first Sata or USB drive.

It is the first Sata hard drive, so the device is /dev/sda. The drive has a 42GB primary partition (sda1), and the rest of the drive is an extended partition (sda2), represented with a cyan border. This is subdivided into five logical partitions (sda5 to sda9). Note that the first partition is marked with the 'boot' flag. This means the partition is active, which in turn means a Dos MBR would boot this partition directly.

Typical setups

As an example of a common situation, let's take a PC with an 80GB hard drive and Windows XP installed. The drive consists of a single primary partition, which takes up the entire drive. It is an IDE drive, in the typical primary master position, and so the device name is hda and the partition is hda1.

When starting up the PC, the Bios runs the code on the MBR, which finds that the first partition is active. This is then booted and Windows XP starts up.

We have decided to put a copy of Ubuntu Linux onto the PC. Ubuntu needs its own partition, and we don't want to delete our installation of Windows. For this to work the current partition must be reduced in size in order to make space for Ubuntu.

The Ubuntu desktop CD will do this for us, and we reduce the first partition to 40GB and create two new logical partitions: one 39GB partition for Ubuntu itself and a 1GB partition for virtual memory (swap). The disk is now laid out quite differently: hda1 contains the copy of Windows; hda5 has a Linux file system; and hda6 contains a partition for virtual memory.

At the end of the installation, we need to install the Linux bootloader (grub). This can go in one of two places: the MBR or the boot sector of a partition. In our case, it is impossible to put grub onto a partition. Remember the Dos MBR can boot only a primary partition, and we have only one, which is occupied with Windows. While it would be possible to install grub onto the Linux file system (hda5), it would be impossible to boot this with the Dos MBR installed.

Once grub is installed onto the MBR, the new setup is complete. When the PC is switched on, the Bios executes the MBR code as usual, which now starts up the Linux bootloader. This in turn allows us to choose whether to boot Linux or Windows: our dual-boot system is ready.

Grub problems solved

Dual-booting is normally configured to work automatically, and our typical example is one where little or no knowledge is needed to get everything up and running. There are, however, some important issues to bear in mind when running a dual-boot system from the MBR.

If you reinstall Windows, it impolitely overwrites the MBR and marks its own partition as active. That means Windows is guaranteed to boot up, but it also prevents you from booting Linux afterwards. For this reason, it is essential to install Linux after Windows.

If you want to remove Linux from a dual-booting PC, do not simply remove the Linux partitions. This is because grub works in two parts: the boot code that sits on the MBR, and the remaining code sitting on the Linux partition. Grub cannot start without both, so removal of the Linux partition will prevent the computer from booting.

One solution to this is to run the Windows installation CD, and choose only to reinstall the MBR. There is also Linux software available that will install a Dos-compatible MBR, such as ms-sys.

Avoid installing grub onto the MBR of your boot drive if you have Linux installed on an external drive. This is for the same reasons as above. If your external drive is not plugged in, grub will be unable to boot the PC.

The most sensible way to avoid all of these problems is to place grub on a primary partition instead of the MBR. As long as Linux is installed onto a primary partition, it isn't necessary to modify the Dos MBR. Simply mark the Linux partition or the Windows partition as active to switch between bootloading with grub or booting directly into Windows. A reinstall of Windows would require switching the active partition back to Linux and nothing more.

Common Linux device names

Primary master - hda

Primary slave - hdb

Secondary master - hdc

Secondary slave - hdd

First Sata/USB - sda

Second Sata/USB - sdb