Morten Jørgensen's Boot Menu

A product from

FENES™ *Off-line*

Introduction

Purpose

The purpose of this product is to enable PC users to alternate between several operating system. A PC can have several operating systems installed on different hard-drives or hard-drive partitions. The boot menu takes control of the PC's boot sequence an enables the user to select what operating system to boot.

Features

Some operating systems come with their own boot manager programs. These include OS/2, Linux and to a certain degree Windows NT and Solaris. These are often tied to their specific operating systems and restrictions usually apply to the way they can be installed, configured and used. However, similar commercial products are available and provide more flexible solutions for multi-OS PCs. One very good example is System Commander from V Communications. The drawbacks with this program are firstly that it costs money, and also its large size and awkward installation.

The Fenes Off-line boot menu is 100% free and provides the user with the following:

Feature	**MJ bootmenu**	**LILO**	**System Commander**
Menu or GUI for OS selection	√		√
Installs from OS-less floppy	√		
Takes up no space in OS partitions	√	√	
No re-partitioning required	√	√	√
Max # of operating systems	16	16	100^+
Pass parameters to Linux	√	√	
Boot MS-DOS/Win95 from 2nd hard-drive	√	√	
Multiple versions of Solaris on single HD	√		√
Includes repartitioning tool			√
Free	√	√	

Availability

The boot menu can be down-loaded from the following sites:

http://home.sol.no/~psycodad
ftp://sunsite.unc.edu/pub/Linux/system/boot/dualboot
ftp://bundy.hibo.no/~morten/bootmenu

The files you will need to down-load are:

bootmenu.exe,	to make install floppy from MS DOS/Win
bootmenu.bin,	to make install floppy from UNIX
bootmenu_A4.ps.gz,	postscript documentation, A4
bootmenu_US.ps.gz,	postscript documentation, US letter

Please send any comments, suggestions or questions about the boot menu to either of these addresses:

morten.jorgensen@ireland.sun.com
mjoergen@c2i.net
geir-aj@online.no

Disclaimer

The entire risk arising out of the use of the boot menu software and documentation remains totally with the user. In no event shall Fenes Off-line or another supplier of the software be held liable for any damages whatsoever (including, without limitation, damages for loss of business profits, business interruption, loss of business information or any other pecuniary loss) arising out of the use of inability to use the software and/or supplied documentation, even if the developer or the supplier has been advised of the possibility of such damages. Or, in plain English: Use at your own risk!

Installation

Preparing your computer for the boot menu

There is normally no need to prepare your computer for the boot menu. Read about hard-drive partitioning in the Problem Solving chapter if you are not familiar with the way PC hard-drives are structured.

Creating an installation floppy

An installation diskette can be created either from the MS-DOS executable bootmenu.exe or from the binary image bootmenu.bin.

- If you are primarily using **OS/2**, **MS-DOS** or any version of **Windows,** you should down-load the `bootmenu.exe` file from one of the sites listed in the previous chapter. Insert a 1.44 MB diskette into your floppy drive and run the command `BOOTMENU A:` from the command shell (MS-DOS prompt). If you are running Windows in DOS-mode you might have to run the command `LOCK A:` to enable direct disk access. If you are using a resident anti-virus program or if your BIOS has some virus prevention feature you might get a warning message and/or the write will fail. Turn off the anti-virus protection and try once more if you discover such problems. Even though an anti-virus program will see the bootmenu.exe as such, the program is not some sort of a Trojan horse. The boot menu must locate itself to a part of the floppy where viruses normally reside. That is the reason why an anti-virus program may complain.

- If you are primarily using a version of **UNIX**, such as **Linux** or **Solaris**, you should down-load the `bootmenu.bin` file from one of the sites listed in the previous chapter. Log on as root (if you don't have root access you probably shouldn't be using this program) and insert a 1.44 MB diskette into your floppy drive. Do not mount the diskette! The transfer the binary image to the floppy running a command similar to:

  `cat bootmenu.bin >> /dev/fd0` or
  `dd if=bootmenu.bin of=/vol/dev/rdiskette0/unlabeled`

Installing from the floppy

To install the boot menu, simply leave the installation diskette in your floppy drive and reboot the computer. Do NOT attempt to read the diskette while still running your operating system, as the diskette has its own file and miniature operating system. This operating system will start once the PC is booted from the floppy.

When the installation program starts it will perform a series of tests on your computer:

- First a search for previous installations of the boot menu is performed. If a previous installation is found, you will be prompted and asked if you still want to proceed with the installation. Reinstalling the program will erase any existing boot menu configuration. Note that the partition-table is left intact – no partitions will be touched.

- Then the installation program runs a check for available space. There is normally some free space in the very beginning of a hard-drive. This space is normally just wasted and left for viruses to play around in. The boot menu installation program will attempt to store the boot menu in this space. This prevents you from having to repartition the hard-drive to leave space for the boot menu. The boot menu needs 15 disk sectors of storage space, adding up to a total of only 7680 bytes.

- The existing master boot record (MBR) is backed up (not done during re-installation). If reading the MBR fails you will be asked if you still want to proceed with the installation. The backup is needed to properly uninstall the boot menu from the configuration program. The backup is strictly not needed if you can restore the original MBR by running the MS-DOS command "`fdisk /mbr`" or similar. Still this warning might be a preliminary warning that there is something wrong about the way your hard-drive is configured and it is not recommended that you carry on with the installation.

Some information is printed on the screen as soon as the installation is done. The configuration module of the boot menu will then start. This menu is very simple and should be fairly easy to understand. If you are having problems using it you should read the following chapter on configuration.

Configuration

Configuration menu

The configuration menu is invoked automatically when the PC is first booted after installation. You can enter the menu later by pressing the [F1] key while within the main menu. Here is a brief explanation to the entries in the menu:

Add partition lets you select one partition from your hard-drives and add it to the boot menu. You will be asked to enter a description/name for the operating system installed on this partition.

Delete partition removes one entry from the boot menu. This does not remove the actual partition from your disk. Only the boot menu configuration data is removed; the partition itself is left untouched. You need a program such as fdisk or System Commander to add, remove and resize actual partitions.

Rename partition will let you change the description/name for one of the operating systems. Use this option also to change the parameters sent to LILO/Linux

Set default partition is used to specify which operating system that will be booted by default. The timeout delay must be set for this to work.

Set timeout delay is used to define the delay before the default partition is booted. Set the delay to 0 to disable this feature.

Uninstall will let you uninstall the boot menu by restoring the original master boot record. Why would you want to do something like that?

Discard changes & exit and

Save changes & exit are self-explanatory, I think.

OS-specific configuration issues

This chapter describes any preparations that may have to be done to properly boot some operating systems. It also describes how to get around some of the shortcomings of MS-DOS and Solaris.

Linux

You will still need LILO to boot Linux. If you want the boot menu to coexist with LILO you need to have LILO installed on the Linux root partition and not in the MBR. Then the boot menu will first load LILO, which in turn loads Linux.

The boot menu can be configured to pass parameters to LILO and Linux. This is handy in case you have multiple Linux kernel versions installed on a single partition. The parameter string sent to LILO must be either empty or consist of the LILO name-tag for the partition followed by LILO/kernel parameters. Refer to the LILO documentation for a full list of these parameters.

Lets say that you have two Linux kernels installed. You first add these two kernels to your /etc/lilo.conf file, like this:

```
boot=/dev/hda3
# Entry for first Linux kernel
image=/boot/kernel-2.2.5
   label=Linux22
   root=/dev/hda3
   read-only
# Entry for second Linux kernel
image=/boot/kernel-2.0.32
   label=Linux20
   root=/dev/hda3
   read-only
```

When you install entries in the bootmenu for the two kernels you will need to pass the kernel label as the first parameter (in this case either "`Linux22`" or "`Linux20`"). You can add additional parameters that will be passed to the actual kernel, such as "`vga=3`" to use a 50-line VGA text mode for the consoles.

MS-DOS

MS-DOS can initially boot only from the first primary partition on the first hard-drive. The boot menu has a feature that can do a logical swap the first two hard-drives on your PC at boot time. This will allow you to boot MS-DOS from the first primary partition of the second hard-drive by fooling MS-DOS to believe that it is booted from the first hard-drive. The configuration menu will prompt you for logical disk swapping when you add such a MS-DOS partition to the boot menu. You must also make sure that the MS-DOS partition on the second hard-drive is bootable. You can do this by booting from an MS-DOS floppy and then running the command "`sys A: D:`".

This feature is handy if you want both Windows and MS-DOS installed on a single PC. Games and other CPU-intensive applications run faster under MS-DOS, but Windows is fairly user friendly, so a combination of the two is often desirable.

There are a few limitations to this approach. If you are using some specific IDE, SCSI or hard-disk drivers in your config.sys file you might not be able to perform the swapping properly. This is because the resident part of the boot menu that handles the disk swapping is overwritten by such drivers. Also a virus might intercept and corrupt the resident part of the boot menu.

Windows 95

Windows 95 can only be booted from the first primary partition on the first hard-drive. Early versions of Windows 95 can be forced to boot from the second hard-drive[1]. See the previous section on MS-DOS for a description on how to set this up.

Windows NT

Windows NT can be booted from the boot menu as long as it does not coexist with Windows 95 or MS-DOS. If you have Windows NT installed with Windows 95 and/or MS-DOS you will probably have to boot Windows 95 first and then proceed to boot Windows NT. A way of overcoming this is to hide the Windows 95 partition when installing Windows NT, then un-hide the Windows 95 partition and install the boot menu.

OS/2

Some OS/2 partitions can only be booted from the OS/2 boot manager. The boot menu can be used to boot the OS/2 boot manager which then fires up OS/2.

Solaris

Solaris can only reside in primary partitions, and a hard-drive can only hold one installation of Solaris[2]. The boot menu will let you overcome the latter limitation. If you have several copies of Solaris installed on the hard-drive the boot menu will make only one visible at boot-time. This is performed by first hiding all Solaris partitions on the hard-drive you are booting from (by giving the partitions system id 0xfd), and then unhiding only the one partition you have selected (by giving this single partition the proper Solaris system id 0x82[3]).

---

[1] The early versions of Windows 95 used the CHS BIOS to access hard disks purely through the BIOS. The logical swap of hard-drive 1 and 2 can only be done if the OS and applications stick purely to BIOS read/write functions.

[2] This is because Solaris assumes that the only installation of Solaris is the first Solaris partition on the disk. If you have two installations of Solaris on one disk and you boot the second Solaris partition, Solaris will still read data from the first partition.

[3] Some will probably recognise this as the Linux Swap partition id. This system id is unfortunately shared between Solaris and Linux Swap. If you intend to have Solaris and Linux swap partitions on the same drive you should make sure that all Linux Swap partitions are extended partitions. In that way the boot menu will leave them alone and they will keep their original system id of 0x82.

Problem solving

Hard-drive partitioning

The partition table in the master boot record[4] contains pointers to the beginning of the partitions on a hard-drive. The boot menu stores these pointers in its configuration data block when you add a partition to the menu. The configuration will become invalid if you resize/move/remove a partition. Before resizing, moving or removing a partition you should remove it from the boot menu. The configuration program will use the new values in the partition table if you later decide to add the partition to the menu.

Read the manual for your fdisk, Partition Magic or System Commander (or similar) partitioning program for further information on hard-drive partitioning.

Hard-drive addressing limitations

There can be limitations to the way your PC can access hard-disks. The boot menu will not help you overcome any of these limitations. This chapter is supplied only to provide readers with an understanding of problems they might encounter when using the boot menu.

Standard CHS BIOS

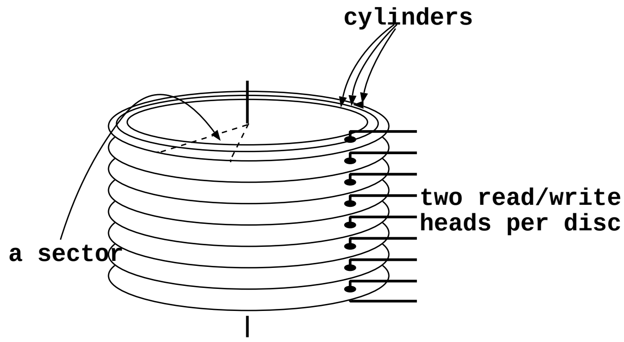
There are may ways in which programs can access the hard-drives on a PC. The original way the PC BIOS[5] presents the hard-drives to the operating system is tied to the physical structure of the drive. The drive has normally two read/write heads for each disc, each disc has a set of tracks (also referred to as cylinders[6]) and each track is divided into sectors[7]. When the operating system or an application wants to access the drive it must present the BIOS with an address in the form of cylinder, head and sector (CHS) numbers.

[4] The master boot record (MBR) is the first 512-byte sector on a hard-drive. This sector contains a 4-entry partition table and a small program that loads the operating system. This small program is replaced by a piece of the boot menu program when the installation is run.

[5] The BIOS (Basic Input/Output System) is a piece of software stored on a chip on your PC's motherboard. This software initialises the PC when you turn it on and it gives operating systems and programs access to the hardware. It is normally the BIOS that enables operating systems to access your hard-drives.

[6] "Cylinder" is normally used to refer to a set of equal tracks on multiple disc.

[7] A sector is a "pie-slice" of the hard-disk. Often the word "sector" is used to denote only one arc of a track on one disk. One such arc/sector can store 512 bytes.



The BIOS interface for this addressing is limited. The highest value for cylinders is 1023, the highest value for heads is 256 and the highest value for sectors is 63. Also, very few hard-drives have more than 16 heads (meaning it has 8 discs with 2 heads each). The maximum addressable size of a hard-drive is then:

$$1024\, cylinders * 16\, heads * 63\, sectors * 512\, bytes = 504\, MB \approx 528.5 \times 10^6$$

This means that if you have a standard CHS BIOS you will not be able to boot operating systems that reside on partitions that start beyond the first 504 MB of a drive.

Translating CHS BIOS

Translating CHS BIOSes will translate the physical geometry of a drive into a logical geometry that can be addressed through the CHS addressing interface. This translation is usually done by multiplying the number of heads by a factor of 2 and then dividing the number of cylinders by the same number. An 800 MB drive with a physical geometry of 1536 cylinders, 16 heads and 63 sectors can be made addressable through the BIOS using a logical geometry of 768 cylinders, 32 heads and 63 sectors. But there are limitations to this approach as well. The BIOS still has the same CHS interface, and the maximum addressable size of a hard-drive is then:

$$1024\, cylinders * 256\, heads * 63\, sectors * 512\, bytes = 7.875\, GB \approx 8.4 \times 10^9$$

This means that if you have a standard CHS BIOS you will not be able to boot operating systems that reside on partitions that start beyond the first 504 MB of a drive.

Logical Block Access BIOS

An LBA BIOS will translate the 3-dimensional physical structure of the hard-drive into a linear 1-dimensional array of 512-byte sectors. The operating system will address the drive using only a single number representing an absolute sector address (the address is an index in an array of sectors). The sector address is a

32-bit number, so the maximum size of a drive administered by an LBA BIOS is:

$$2^{32} \times 512\,bytes = 2\,terabytes \approx 2 \times 10^{12}$$

If you have a LBA BIOS you shouldn't have to worry about a thing for the next few years.

The boot menu and the BIOS

The boot menu has been implemented to utilize the best available addressing scheme. The ability to boot partitions on your drives is therefore only restricted by the limitations of your BIOS.

Viruses

When you turn on the PC the BIOS takes control and runs a series of tests[8]. When the tests are gone through with and everything is found to be working well, the BIOS loads the program located in the master boot record. This program loads a similar program in the boot record[9] of a pre-determined partition on the hard-drive. It is the responsibility of this program to load the operating system located in this partition. The boot menu overwrites the original code in the master boot record. It is no long pre-determined what partition that will be booted, as the boot menu allows the user to select what operating system that should be loaded.

Unfortunately, both the master boot record and partition boot records are favored spots for a certain breed of viruses. These viruses intercept the boot sequence, hides itself somewhere in memory, boots the operating system boot, while it still encapsulates some of the functions of the BIOS. In that way the virus can reside in memory while the operating system appears to be running normally. The virus can trap certain calls to the BIOS (such as hard-drive and floppy access calls) and in that way reproduce itself and most possibly cause irreparable damage. Such viruses may cause the boot menu to stop functioning properly, or they may cause particular functions of the boot menu to stop working properly.

Anti-virus software

Anti-virus software often monitor reads and writes to/from the master boot record. The boot menu performs such operations frequently and can therefore be reported as a virus. Turn off MBR protection in your anti-virus software before using the boot menu. This also applies to virus detection mechanisms in the BIOS.

[8] This is referred to as POST – Power On Self Test.

[9] The first 512-byte sector of every partition of a hard-drive is referred to as the boot sector.

Future plans

These are features planned for future releases of the boot menu. Send an e-mail to any of the addresses listed in the introductory chapter to have us add anything to this list:

Configuration password protection

One earlier release of the boot menu allowed password protection of the configuration menu. This earlier version of the boot menu was released especially for the University of Bonn, where the boot menu was used in the university labs to alternate between Windows NT and Linux. Password protection was removed from the boot menu to allow room for other features. I will work to further compress the program code in the boot menu to free space and allow for this feature to be added again.

Display boot menu over a serial line

Solaris allows the console of a Solaris 2.6^{+} system to be re-directed to a serial line. The console can then be controlled through a console server or directly over a tip-line. The boot menu should be enhanced to send a display of the menu to an RS232-port and similarly accept input from the same line. This feature is handy for remote lab-machines with several OSes – especially several versions of Solaris.

Windows configuration tool

Just forget about it. There will never, ever be an installation or configuration tool tied to any specific operating system. Especially not an MS one.