

Re: 2 wondows versions detected on pc startup?

Source:

http://www.tech-archive.net/Archive/WinXP/microsoft.public.windowsxp.setup_deployment/2007-04/msg00026.htm

- *From:* "Timothy Daniels" <TDaniels@xxxxxxxxxxxxxx>
 - *Date:* Tue, 3 Apr 2007 21:10:01 -0700
-

"Wesley Vogel" wrote:

I may have some of this wrong. But, the way I understand it on an x86-based computer, after the POST, the BIOS locates and reads the configuration information stored in CMOS looking for the order of what drive to look at for an operating system.

So how does the configuration information get into the CMOS if not put there by the BIOS? I believe the BIOS looks at the HDs connected to the controller and makes a vector (ordered list) of the HDs using the default HD boot order as a checklist, or that vector is set in the BIOS by the user who orders the IDs of the HDs manually via keyboard input to the BIOS. The HD at the head of the resulting vector is referred to as "rdisk(0)" in boot.ini's ARC path entries, the next existing HD is referred to as "rdisk(1)", etc.

If the first bootable disk is a fixed disk the BIOS examines the very first sector of the disk for a Master Boot Record (MBR). (A Master Boot Record is made up of two parts – the partition table which describes the layout of the fixed disk and the partition loader code which includes instructions for continuing the boot process.) With a valid MBR loaded into memory the BIOS transfers control of the boot process to the partition loader code. The BIOS is done.

The partition loader (or Boot Loader) examines the partition table for a partition marked as active (or bootable). The Boot Loader then searches the very first sector of that partition for a Boot Record. The active partition's boot record is checked for a valid boot signature and if found the boot sector code is executed as a program.

This is where Windows and NTLDR comes in, the partition loader code loads NTLDR. NTLDR is the Boot Loader for Windows NT. NTLDR wants to see a boot.ini. Supposedly, if a boot.ini is not present it will

Re: 2 wondows versions detected on pc startup?

default to C:\Windows. I am not about to delete my boot.ini to test this. ;—)

If you have multiple HDs, each with an OS, in the system, you can use the BIOS's HD boot order to choose with HD (and thus which OS) to boot. That way, you can repair mangled files in a partition whose OS won't boot..

Ntldr reads the Boot.ini file to determine the location of the operating system boot partition. For systems that use a single-boot configuration, Ntldr initiates the hardware detection phase by starting Ntdetect.com. For multiple-boot configurations you get the Boot Menu, to pick an operating system from.

Windows uses the Advanced RISC Computing Specification (ARC) Path Naming Convention.

The Advanced RISC Computing Specification
<http://www.netbsd.org/Documentation/Hardware/Machines/ARC/riscspec.pdf>

Disk (rigid). Where r stands for rigid. So, rdisk is the physical hard disk. fdisk is a FloppyDisk. Where f is for floppy.

I "think" that the path(s) in the boot.ini tells NTLDR where to look for any operating systems.

From my boot.ini...
multi(0)disk(0)rdisk(0)partition(1)\WINDOWS

The Path is: Adapter\Controller\Peripheral\Partition\Filepath

multi(0) is a MultiFunction Adapter.
disk(0) is the Controller for the disk, a hard disk in this case (rdisk denotes a hard disk).
rdisk(0) is the Peripheral, which is a hard disk.
partition(1) is the Partition on the hard disk.
WINDOWS is the Filepath.

rdisk(0) is Primary Controller channel 0 (Master)
rdisk(1) is Secondary Controller channel 0 (Master)
rdisk(2) is Primary Controller channel 1 (Slave)
rdisk(3) is Secondary Controller channel 1 (Slave)

In my Phoenix Technologies BIOS (a very common BIOS), the HD boot order (if there were 4 HDs) would default to:

Re: 2 wondows versions detected on pc startup?

Re: 2 wondows versions detected on pc startup?

rdisk(0) = Master, IDE ch. 0
rdisk(1) = Slave, IDE ch. 0
rdisk(2) = Master, IDE ch. 1
rdisk(3) = Slave, IDE ch. 1

If there were only 2 HDs, for instance, and they were Slave on ch. 0 and Master on ch 1, the default HD boot order for such an arrangement of HDs would have
rdisk(0) = Slave on ch. 0,
rdisk(1) = Master on ch. 1.

For SATA HDs, the HD boot order defaults to:

rdisk(0) = SATA ch. 0
rdisk(1) = SATA ch. 1
rdisk(2) = SATA ch. 2
rdisk(3) = SATA ch. 3

(For mixed PATA and SATA controllers, one or the other types may take precedence in the default case, according to what I interpret from the newsgroups.)

But this default HD boot order may be adjusted by the user via keyboard input to the BIOS immediately after POST, and it's that resulting HD boot order which will define the meanings of "rdisk(x)", and it will persist until the user changes it or the battery for the CMOS dies or is removed.

So there is some sort of vector prepared by the BIOS and stored in CMOS memory which is accessible by NTLDR so that NTLDR can understand which HD is meant by a given value of "x" in "rdisk(x)". Then, when NTLDR sees "rdisk(0)" in a boot.ini ARC path, it knows to load the MBR from the HD which is at the head of (i.e. at position 0 in) the HD boot order. If NTLDR sees "rdisk(1)" in a boot.ini ARC path, it knows to load the MBR from the HD that is in the next position (i.e. at position 1) in the HD boot order. So "x" in "rdisk(x)" stands for the position of the HD in the HD boot order relative to the head of the list. You could even think of "rdidk()" as designating the "relative disk position".

The numbering of partitions, though, starts with "1", and it proceeds through the primary partitions, then the logical drives in the extended partition (if one exists). With this numbering scheme, the OS can reside and be loaded from a logical drive in an extended partition. As a matter of fact, multiple OSes can reside in an extended partition – one OS per logical drive – and each of them can be bootable.

Re: 2 wondows versions detected on pc startup?

Re: 2 wondows versions detected on pc startup?

The last component in an ARC path is the name of the folder which contains the OS – which defaults to "WINDOWS" during installation if the user doesn't designate some other name to give the folder.

Notice that the boot files (i.e. NTLDR, boot.ini, and NTDETECT.COM) don't have to be in the same partition as the OS. They just have to be in the primary partition that is marked "active" on the HD at the head of the HD boot order. The MBR of that HD will find the "active" primary partition, and the boot sector in that partition will then find the boot files in the partition.

This I've gleaned from days of brute force experimentation. What remains unknown, though, is whether the HD vector which defines the meaning of "rdisk(x)" is the same for all PC BIOSes. It may be that NTLDR adjusts its interpretation or assumed location of the HD vector according to what BIOS it detects. That may be what Phoenix Technologies means when it says to "ask Microsoft". But I don't know, and Microsoft doesn't document it for the public.

TimDaniels

Timothy Daniels hunted and pecked:

Well, let me say this on that: You know more about what's in the registry than *I* do. :-)

The core of my question, though, lies in whether msconfig only verifies that the ARC path entries in boot.ini each lead to an installed Windows OS, or whether it can actually find OSes and then create corresponding entries in boot.ini for them. I guess the latter is yet to be seen, although if I had the energy and time, I could run some experiments which would answer the question. I'll let you know if I come up with anything.

There also seems to be more that ntldr gets from the BIOS because ntldr has to interpret the meaning of "rdisk(0)" and "rdisk(1)", etc., when it goes looking for the OSes. There must be a vector (i.e. an ordered list) of HD locations that the BIOS exposes to ntldr so that the ARC path entries in boot.ini can be understood by ntldr so that ntldr can find the HDs that contain an OS. But Phoenix Technologies has refused to say anything about how the BIOS does this it except to say "ask Microsoft". I have not been able to understand the reason for this stonewall, but it exists.

TimDaniels

Re: 2 windows versions detected on pc startup?

"Wesley Vogel" wrote:

You got pretty much what I know about it, Tim. Check All Boot Paths reads the boot.ini. How? I have no idea. I also have no idea how checking or unchecking Boot Options adds or removes switches in the boot.ini. I do know that it is safer to let msconfig edit the boot.ini, especially for first timers, than directly editing it in Notepad.

As far as I know, there are only a couple of things that Windows even knows about a machine's BIOS.

HKEY_LOCAL_MACHINE\HARDWARE\DESCRIPTION\System
Value Name: SystemBiosVersion
Data Type: REG_MULTI_SZ
Value Data: This should show the BIOS version.

Value Name: SystemBiosDate
Data Type: REG_MULTI_SZ
Value Data: This should show the BIOS date.

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Biosinfo
Value Name: InfName
Data Type: REG_SZ
Value Data: biosinfo.inf

Value Name: SystemBiosDate
Data Type: REG_SZ
Value Data: This should show the BIOS date.

Biosinfo.inf: Windows file that lists machines on the Autoenable APM list and the Disable APM list, and also lists the BIOS detection sequences used to match them.

Biosinfo vs. Machine-Specific Info. The results of machine-specific detection versus Biosinfo.inf are stored in HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\BIOSInfo\APM. If the "Attributes" value is 1, the machine's APM BIOS is "known good" and the machine is on the Autoenable APM list. If the value 2, the machine's APM BIOS is "known bad" and the machine is on the Disable APM list.

Re: 2 wondows versions detected on pc startup?

Otherwise, the machine is neutral.

C:\WINDOWS\inf\biosinfo.inf

If this is any help to you. My boot.ini.

```
[boot loader]
timeout=5
default=multi(0)disk(0)rdisk(0)partition(1)\WINDOWS
[operating systems]
multi(0)disk(0)rdisk(0)partition(1)\WINDOWS="Microsoft
Windows XP
Professional" /noexecute=optin /fastdetect /sos
```

multi(0)disk(0)rdisk(0)partition(1) appears in my registry.

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control
Value Name: SystemBootDevice
Data Type: REG_SZ
Value Data: multi(0)disk(0)rdisk(0)partition(1)
```

```
/noexecute=optin /fastdetect /sos also appear in that key.
Value Name: SystemStartOptions
Data Type: REG_SZ
Value Data: NOEXECUTE=OPTIN FASTDETECT SOS
```

FYI, NOEXECUTE=OPTIN Enables DEP for core system images and those specified in the DEP configuration dialog. Added by SP2.

FASTDETECT Default boot option for Windows. Causes NTDETECT to skip parallel and serial device enumeration, so that NTDETECT can support booting Windows NT 4. Note that ntdetect.com detects installed hardware components when XP boots.

SOS I added this. Causes Windows to list the device drivers marked to load at boot time and then to display the system version number (including the build number), amount of physical memory, and number of processors. Maybe I should have just said, "You got pretty much what I know about it", and left it at that, :-)

--

Hope this helps. Let us know.

Re: 2 wondows versions detected on pc startup?

Wes
MS-MVP Windows Shell/User