

Re: Grub hangs – two hard drives and a CD

Source: <http://linux.derkeiler.com/Newsgroups/comp.os.linux.setup/2005-11/0626.html>

From: Enrique Perez-Terron (*enrio_at_online.no*)

Date: 11/29/05

Date: Tue, 29 Nov 2005 18:02:16 +0100

On Tue, 29 Nov 2005 06:23:08 +0100, imotgm <imotgm_REM@invalid-yahoo.com> wrote:

> *On Mon, 28 Nov 2005 15:26:23 +0100, Enrique Perez-Terron wrote:*

>

>

>> *Grub> root (hd3,0) # the partition having "stage2"*

>> *Grub> setup (hd0) # the disk your Bios will access first.*

>>

>

>> *(hd0) /dev/hda*

>> *(hd3) /dev/hdc*

>>

>>

> *Enrique,*

>

> *What you wrote is all based on a wrong assumption that grub numbers drives*

> *based on their position on the IDE cables. It does not. Grub counts*

> *physical hard drives only, counts from (hd0), with (hd0) being the drive*

> *that the BIOS will call to boot an OS.*

>

> *Here is grubs device map when the BIOS is set to boot from the first hard*

> *drive, and when there is no hda, but drives are present on hdc, hde, and*

> *hdg, all master positions on a system with four IDE controllers. (My hard*

> *drives are in caddies, so this is an easy test)*

>

> *(fd0) /dev/fd0*

> *(hd0) /dev/hdc*

> *(hd1) /dev/hde*

> *(hd2) /dev/hdg*

>

> *When hde and hdg are removed, a drive is added to hda, and a usb drive is*

> *also plugged in, with grub in the MBR of hdc, the BIOS is set to boot from*

> *hdc, the device map gets automatically rewritten as below.*

>

> *(fd0) /dev/fd0*

> *(hd0) /dev/hdc*

> *(hd1) /dev/hda*

> *(hd2) /dev/sda*

comp.os.linux.setup: Re: Grub hangs – two hard drives and a CD

>
> *When (hd0) is moved to /dev/hda and (hd1) is moved to /dev/hdc, and the
> other drives are returned to hde, and hdg, the BIOS is again set for
> normal booting, the device map is again rewritten.*
>
> *(fd0) /dev/fd0
> (hd0) /dev/hda
> (hd1) /dev/hdc
> (hd2) /dev/hde
> (hd3) /dev/hdg
> (hd4) /dev/sda*
>
> *All of which illustrates that there is no fixed relationship between cable
> position, and grub's (hdX). Grub counts from (hd0), does not count CD-Roms
> or DVD-Roms, (My DVD-R/W is hdb, and hdd is my CD-R/W) only hard drives,
> and does not skip numbers. You can't have an (hd3) without having (hd0),
> (hd1), and (hd2) precede it.*

Thanks a lot for this info, I have been asking myself about this for some time, but forgot to consider this a possibility when I wrote my last post. I will use the opportunity to ask more questions related to this.

I suspect that this behavior of Grub is a reflection of how the Bios behaves, but I don't have any good source for the Bios behavior in different computers and circumstances. The Bios calls to read disk require the register DL to contain a code for the drive to be accessed, and that code is 0x80 for the boot disk, or is it for the first disk, or is it for the primary master? The texts I have seen describing the "int 13h" calls just say "first disk" without any further reference to the enumeration process.

I have observed that the standard Microsoft MBR code uses the value of the bootable flag in the partition table, as the drive designation when issuing the Bios call to read the "partition boot sector". This made me think that, for one, it should be possible to abuse of the partition table, having an entry that actually describes a partition of the second disk, and use the value 0x81 in the "bootable" flag byte. (I'll not go into the disastrous consequences this could have when other software believed this partition table entry without considering the low-order bits of the the bootable flag.)

Second, it makes me think that in computers where it is possible to boot off any other disk than the first, the Bios will likely renumber the disks so that the boot disk always has code 0x80. What you tell, fits neatly with this theory. I have not been able to test this because my computers could only boot from an unspecified "hard drive", as stated earlier. Does anyone have more info on this?

Another possible interpretation would be that a "bootable" flag in a partition on a disk other than the first, would have to be 0x80 + zero-based disk number. This would render the bootable flag invalid if disks are rearranged. It does not seem very attractive.

Prompted by this ng exchange, I have searched the web again, and found a "BIOS Boot Specification" from Compaq, Phoenix and Intel, 1996, where an appendix suggests that boot sector programs use the value that the Bios provides in the DL register, rather than the value in the bootable flag. That reminds me that I saw some reference to that in the source code for Grub's stage1. Grub seems to rely on this extension, although it has a code to modify the register DL by or'ing with 0x00, where the byte 0x00 can be patched to 0x80 when the stage1 is installed on a hard disk. If I understand this correctly,

- grub can boot from devices other than the "first" if whoever calls it, calls it with the device number to use in register DL. It could be called by the Bios, or by some other boot-loader that chains into grub.
- grub can boot off the "first" device if the caller leaves the DL set to zero, e.g., an older Bios.
- If your Bios or chaining boot loader leaves garbage in DL when calling grub's stage1, you are fried. It won't boot.

Here "first" is whatever the bios "int 13h" calls access when DL is 0x80. Here I am only talking about the stage1 part of the boot. The stages 1.5 and 2 can of course access any device supported by "int 13h".

> *The OP's drive is indeed (hd1), if it is not (hd0), and there are only two*

Ah, that explains it!

- > *hard drives in the machine. To know the exact interaction between grub and*
- > *Fedora, we need to see the partition table for (hd1), know for sure that*
- > *it is jumpered properly for whatever position it occupies in the IDE*
- > *cabling, and the contents of /boot/grub/menu.conf.*

So, with some luck, the OP can possibly achieve to boot off the second disk (300G) by referring to it as (hd1) in the device.map file and elsewhere.

But then it is not clear to me what happened when he connected the dvd as secondary master. If the bios (or is just grub) skips the dvd in the numbering of disks, the 300G disk in the secondary slave position should just be the same as before, and equally accessible?

That returns me to the hypothesis that there may be possible to access the 300G (with some error rate != 100%) even in the case that both the dvd and the 300G disk are jumpered as masters. Does anybody know if this is physically possible? It sounds improbable to me, what could possibly prevent the dvd from interfering in the communication, and destroying it? But in the world of actual engineering there are quite a few surprises. I don't know the ATA command repertoire, but I cannot exclude a priori that it be possible to issue a disable command to the master, in the hope that if one of them responds sufficiently faster, the other will abort or cancel the command and so remain accessible...

But another theory that seems much more likely is that the behavior you describe (hd0=boot device, hdn=nth existing non-boot *disk* device) is not carved in stone, but rather subject to bios-standard versions, producer idiosyncrasies, disagreements about what constitutes a disk, and right-out bugs. So, it would be interesting to hear from the OP, if he finds a way to get into a grub prompt independent of the exact set of ata device attachments, if he can use the "find" grub command under different circumstances and report what he finds.

Perhaps the fastest way is for the OP to use the knoppix to access the 300G disk, experiment with hd1, hd2 and hd3 in the devices.map and see if any of these works. He would have to run "grub-install /dev/hda" with each setting in device.map, then reboot. Updating grub.conf would be of secondary importance, as that file only affects what happens after stage2 is loaded.

Oh, yes, it was said that an USB floppy would not act as a bootable device, but that seems also uncertain. Some computers have Bioses that allow booting off usb disks, some do not. What I just found in the Bios boot specification seems to point to mechanisms for plug-and-play devices to install Bios extensions, and that opens for quite many possibilities. Whether any particular device or setup exploits the possibilities is anyone's guess. But of course, the OP should be warned that the USB floppy may not work as a boot device.

Regards,
Enrique