

Re: SCO OS 5.0.7 on Qemu

Source: <http://unix.derkeiler.com/Newsgroups/comp.unix.sco.misc/2006-01/msg00033.html>

- *From:* Rob <rob@xxxxxxxxxxxxxxxxxxxxxx>
 - *Date:* Tue, 10 Jan 2006 15:39:02 +0100
-

Bela Lubkin wrote:

> Roberto Zini wrote:

>

>

>>This time of the year allows me to experiment with things that otherwise
>>I wouldnt' have the time play with; during the last few days I gave Qemu
>> 0.8.0 + kqemu (running on SuSE Linux 10) a try. I was able to install
>>and successfully use Windows XP and I wanted to give SCO OS 5.0.7 a run.

>>

>>I started with a 300MB memory allocation and by creating a virtual disk
>>of 2GB; when I booted from the CD, I was able to get to the boot: prompt
>>but, after depicting the usual dots, I ended up having a blank screen
>>instead of the "laundry list" of devices.

>>

>>Qemu is able to emulate an old Cirrus Logic board and, since
>>I was aware of the OSS653A patch (the initial VGA recognition
>>supplement) so I tried to add it as a BTLN driver (which worked fine)
>>but again I ended up with a blank screen.

>>

>>So I gave OSS661A a try and I was able to get to the laundry list but I
>>had a "WARNING: hd: no root disk controller was found".

>>

>>This is strange since Qemu is able to emulate a simple IDE/EIDE
>>controller; the CD-ROM interface the installed detect is the real
>>/dev/cdrom device under Linux and the HD is an ATA-2 2GB HD configured
>>as ata0/master.

>

>

> This is much the same problem that OSR5 has previously had with VMware
> and then with MS Virtual PC. The OSR5 "wd" IDE driver has code in it to
> work around the quirks of _ancient_ "wd" family disk controllers. I'm
> talking about things like the Western Digital WD1003 (16-bit ISA
> controller used in PC/AT class machines), and even older relatives.
> These workarounds have been conditioned over the years to work correctly
> with _real_ MFM/RLL/ESDI/IDE/ATA/etc controllers, but they tend to
> trigger oddities in emulators.

>

> The driver is partly at fault (for pushing the "hardware" in weird,
> unexpected directions); and the emulator is partly at fault (for not

Re: SCO OS 5.0.7 on Qemu

> emulating every quirk of the real hardware it's supposed to represent).

Once more, Bela, thank you for taking the time to read this and for the above explanation.

>
>
>>I tried with "defbootstr Sdsk=wd(0,0,0) link=invd prompt" but that gave
>>me a panic during the hd_config stage.
>
>
> Although "wd" has a SCSI interface, that is only used for ATAPI devices
> (mostly CD/DVD drives, also tapes and a few other oddball items). IDE
> hard disks are not in any way seen as SCSI under OSR5. Here you told
> the kernel to look for a SCSI disk on a "wd" HBA. This is a valid
> formulation to access an ATAPI device that represents itself as a hard
> disk, e.g. an ATAPI Zip. It goes nowhere with a true IDE hard disk.
>

OK.

>
>>So I downloaded the WD supplement from SCO's FTP site and tried to use
>>it using the
>>
>> link="invd wd"
>>
>>bootstring. The loader correctly prompted me to replace the existing
>>"wd" driver but, again, I ended up having a "no root disk controller was
>>found".
>>
>>The laundry list reported
>>
>> adaptec 0x0170-0x0177 15 - type=IDE ctrl=secondary drv=wd
>>
>>which is the secondary IDE controller but failed to report the primary
>>(which is/should be at 0x1F0 - IRQ 14). I tried again by telling the
>>loader to search for a "wd" adapter here:
>>
>> adapter=wd(0x1F0,14,0)
>>
>>but unsuccessfully.
>
>
> "wd", in its role as a pseudo-SCSI HBA, announces "%adapter" lines for
> controllers on which it finds ATAPI devices. Your primary controller
> has only a hard disk (which it isn't going to see as SCSI), so it
> doesn't announce the HBA. If the hard disk itself is found, you get a
> "%disk" line that announces the same I/O addresses etc.
>
> No "adapter=", "Sdsk=" or any other OSR5 SCSI subsystem-related

Re: SCO OS 5.0.7 on Qemu

- > bootstrings are going to help here. What you need is to get the IDE
- > controller and hard disk detection over the hump.
- >
- >
- >>I'm pretty sure that (again – and he will be more than welcome :-) Bela
- >>will jump ip, scare me to death and provide a simple solution...
- >
- >
- > Hmmm, well, let's scare you with this:

You already did (but I was prepared :-)

- > I'm starting a new job Monday, at
- > VMware. I suspect that I should not do a whole lot of very visible
- > support work on open source products that compete directly with my
- > employer. But what the heck, at the moment this falls under "learning
- > about emulation" — it will make me more effective at my job.

Well, I wish you luck for your new life; I'm pretty sure folks at VmWare will gain from your expertise and knowledge but __please__ don't forget us poor guys still stuck to Oses from SCO :-)

- >
- > Next step I recommend:
- >
- > Boot
- > : defbootstr link=invd wd.debug=ugi
- >

Well, I used the following:

```
defbootstr link="invd wd" wd.debug=ugi
```

- >From inside the emulator I'm able to "switch" the floppy disk emulated image since I had to use 2 BTLDS (one for the invd driver and one for the newer wd one).
- > This makes the "wd" driver print as much debug output as it can; which
- > isn't very much, but should give us a vague sense of how far it's
- > getting.
- >
- > The sequence of operations that happen is, very crudely, something like
- > this:
- >
- > wd_ctlpres():
- > see if a ST506/IDE controller is present
- > /* if we got this far, the driver does think a controller exists,
- > but it also wants to know if it's one of the ancient weird ones
- > that it knows won't work, so: */
- > subject it to several weird tests
- > return false if those weird tests fail

- >
- > wd_drivepres():
- > see if a ST506/IDE hard disk is present
- > /* same deal */
- > subject it to several weird tests
- > return false if those weird tests fail
- >
- > You're going to get some debug output that may or may not clarify this.
- > If you're lucky, debug output will suggest an easy workaround. If not,

Well, I did that; problem is that debug messages fill up the screen so I'm not sure if the easy workaround you suggested actually appeared or not.

Anyway, what I can see on the screen is (hand written so prone to errors):

```
wd: if incorrect, add wd.udma=33 (or 66 or 100) to your bootstring.
wd 0/0: IDENTIFY DEVICE: Status 58, Error 00
wd 0/0: non-ATAPI device present
wd: Drive max supported mode = 4, BIOS selected mode = 5
wd: Drive using BIOS selected UDMA mode 5
wd 0/0: Common UDMA Mode 4, Max MBoard mode 6, Max dev supt 4
wd: Possible UDMA timing mismatch, bootstring may be required
wd: add wd.udma=auto for driver attempted auto sense or
wd: wd.udma=off to use Programmed I/O (PIO) instead of UDMA
wd 0/1: IDENTIFY_DEVICE: Status 00, Error 00 (no device)
wd 1/0: IDENTIFY_DEVICE: Status 41, Error 04
wd 1/0: ATAPI_IDENTIFY: Status 48, Error 00, ATAPI = yes
wd 1/1: IDENTIFY_DEVICE: Status 00, Error 00 (no device)
%adapter 0x0170-0x177 15 - type=IDE ctrl=secondary dvr=wd
....
....
WARNING: hd: no root disk controller was found
hd: a Boot-Time Loadable Driver may be required
%cd-rom - - type=IDE unit=0 ctrl=sec cfg=mst dvr=Srom->wd
```

I've tried by rebooting with "wd.debug=ui wd.udma=auto" but unsuccessfully; however, the following messages appeared (**):

```
wd: if incorrect, add wd.udma=33 (or 66 or 100) to your bootstring.
wd 0/0: IDENTIFY DEVICE: Status 58, Error 00
wd 0/0: non-ATAPI device present
wd: Drive max supported mode = 4, BIOS selected mode = 5
** wd: Reprogram ATA drive to UDMA mode 4 **
wd 0/0: Common UDMA Mode 4, Max MBoard mode 6, Max dev supt 4
wd: Possible UDMA timing mismatch, bootstring may be required
wd: add wd.udma=auto for driver attempted auto sense or
wd: wd.udma=off to use Programmed I/O (PIO) instead of UDMA
wd 0/1: IDENTIFY_DEVICE: Status 00, Error 00 (no device)
wd 1/0: IDENTIFY_DEVICE: Status 41, Error 04
wd 1/0: ATAPI_IDENTIFY: Status 48, Error 00, ATAPI = yes
wd 1/1: IDENTIFY_DEVICE: Status 00, Error 00 (no device)
```

```
%adapter 0x0170-0x177 15 - type=IDE ctrl=secondary dvr=wd
....
....
```

WARNING: hd: no root disk controller was found
hd: a Boot-Time Loadable Driver may be required

Next, I used wd.udma=off and some more lines appeared before the "wd: if incorrect..." one:

```
wd: didn't find any specific known PCI IDE controller, trying by class
wd: PCI bus/dev/func 0/1/1, vend/dev 8086/7010, prog_if 80, BMIBA 0000C001
wd: unknown but possibly usable PCI IDE controller; assuming UDMA 133
wd: if incorrect, add wd.udma=33 (or 66 or 100) to your bootstring
wd 0/0: IDENTIFY_DEVICE: Status 58, Error 00
wd 0/0: non-ATAPI device present
wd 0/0: Max MBoard UDMA mode=6, Max drive UDMA mode=4
wd 0/0: BIOS selected UDMA mode=5 for drive
** wd 0/0: UDMA disabled by bootstring **
wd 0/1: IDENTIFY_DEVICE: Status 00, Error 00 (no device)
wd 1/0: IDENTIFY_DEVICE: Status 41, Error 04
wd 1/0: ATAPI_IDENTIFY: Status 48, Error 00, ATAPI = yes
wd 1/1: IDENTIFY_DEVICE: Status 00, Error 00 (no device)
%adapter 0x0170-0x177 15 - type=IDE ctrl=secondary dvr=wd
....
....
```

WARNING: hd: no root disk controller was found
hd: a Boot-Time Loadable Driver may be required

Again, I tried with wd.udma=33: the drive tried by reprogram the ATA drive to UDMA mode 2 and again I had a UDMA timing mismatch. The same went with wd.udma=66 and wd.udma=100.

- > the next step will be to boot with "defbootstr link=invd maindebug".
- > That invokes scodb (kernel debugger) early in the kernel's life, then
- > you can set breakpoints inside "wd" and trace the recognition code, see
- > where it goes wrong.

Well, without further guidance, this is outside of my limited scope :-(

- >
- > =====
- >
- > BTW, are you sure it's "invd" and not "ivga"? "invd" works around a
- > fairly specific nVidia BIOS behavior; "ivga" is much more generic. (The
- > issues are completely different and could possibly both occur on the
- > same system, though I've never heard of that.) "ivga" works around a
- > far more common BIOS problem: incorrect setting of the CMOS video type
- > value. In fact, I see that Qemu uses the Bochs video BIOS, and that
- > this very bug was fixed in version 0.5c of that BIOS. See

Re: SCO OS 5.0.7 on Qemu

- > <http://www.nongnu.org/vgabios/>. That page has compiled BIOS images; you
- > should be able to download a newer one into your Qemu/bios directory.
- > (Update both vgabios.bin and vgabios-cirrus.bin, I'm not sure how to
- > tell which one Qemu will use on your machine.) Then you shouldn't need
- > "ivga".
- >
- > On further examination, I suspect that Qemu's BIOS may have both issues!
- > So you may really mean you are using "invd". Why would it not need
- > "ivga"? Because it gets lucky -- the wrong value that it leaves in CMOS
- > isn't wrong enough to bother OSR5.
- >
- > But you said "oss653a", which is the "ivga" supplement, so in the end my
- > guess is that you're using "ivga" and just typed "invd" into your post
- > by mistake.

Well, I'm pretty sure about that. Just to refresh my flaky memory, I went up again today and I started by using oss653a (link=ivga) but I ended up with a blank screen.

So I gave oss661a.btld a try (link=invd) and I was able to get to the laundry list and actually "see" its content so it's the invd which did the trick.

Concerning the vgabios, I used the default one (01 Dec 2004) and I also tried version 0.5d which, with Windows XP/2003, gave me some graphic problems (4 colours, low resolution); nonetheless, even version 0.5d of the VGA bios had problems with OS 5.0.7 so I ended up by reverting to the original one and using the oss661a BTLD.

- >
- > =====
- >
- > I've been running OSR5 under VMware Workstation for several years now.
- > There were some problems at first. These days I have great success with
- > it. VMware now makes a "player" available for free, lets you run
- > previously built virtual machines. You could probably install OSR5 onto
- > that, giving you another free virtual machine that can run OSR5 (without
- > all these issues). But without source -- so if your real goal here was
- > to study the emulation mechanics themselves, that won't do.

I think that, unless you have more ideas, I'll stick to VmWare Player for Linux (and this should sound good for you, right? :-); studying the emulation mechanics is way above my knowledge for now.

- >
- > There, is that scary enough? ;-}
- >

As usual, from you :-)

- >

Re: SCO OS 5.0.7 on Qemu

>>Bela<

Thanks,
Roberto

Roberto Zini - r.zini<@AT@>strhold.it

"Has anybody around here seen an aircraft carrier?"
(Pete "Maverick" Mitchell - Top Gun)

.

- *Follow-Ups:*

- ◆ [Re: SCO OS 5.0.7 on Qemu](#)

- ◆ *From:* Bela Lubkin

- *References:*

- ◆ [SCO OS 5.0.7 on Qemu](#)

- ◆ *From:* Rob

- ◆ [Re: SCO OS 5.0.7 on Qemu](#)

- ◆ *From:* Bela Lubkin

- Prev by Date: [Re: SCO 5.0.5 dd / clone](#)
- Next by Date: [Re: SCO 5.0.5 dd / clone](#)
- Previous by thread: [Re: SCO OS 5.0.7 on Qemu](#)
- Next by thread: [Re: SCO OS 5.0.7 on Qemu](#)
- Index(es):
 - ◆ [Date](#)
 - ◆ [Thread](#)