

The Bootloader

How Grub works, booting other OSs

Nick Urbanik <nicku(at)nicku.org>

Copyright Conditions: GNU FDL (see <http://www.gnu.org/licenses/fdl.html>)

A computing department

Grub: Grand Unified Bootloader

- A bootloader can be very simple, provide minimal functionality
- Can pay money for tools such as System Commander, or BootMagic (with Partition Magic)
- Or use LILO, or better still, Grub.
- Grub aims to boot anything on an Intel 86 architecture

Booting Windows on an Intel Computer

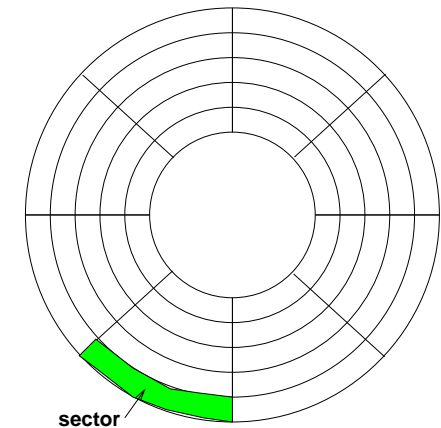
OSSI — Bootloader — ver. 1.1 — p. 1/11

- BIOS loads **MBR** (Master Boot Record), first 512 bytes of “first” hard disk to RAM
- BIOS jumps to this code
- That code (by default) reads first sector of first active partition, the **boot sector**
- boot sector code then loads the operating system

Disk sectors, cylinders

OSSI — Bootloader — ver. 1.1 — p. 2/11

- The rings are called **cylinders**:



Methods of Booting

- BIOS always loads MBR
- MBR can load a bootloader, such as grub's `stage2` or LILO
- Booting OS directly:
 - bootloader can load an operating system kernel directly
- *Chaining bootloaders*:
 - bootloader can load another bootloader which in turn loads an operating system

How grub works

- We install "stage1" of grub into MBR
- stage1 reads `stage1_5` or `stage2` from a list of disk blocks
- stage1 loads `stage1_5` from hard disk
- `stage1_5` can now read files on the hard disk, and loads `stage2`
- `stage2` of grub provides commands to support many features

Features of grub

OSSI — Bootloader — ver. 1.1 — p. 5/11

- support directly loading many different operating system kernels
- can directly read many file system formats
- can load a configuration file from disk
- can decompress files automatically
- can read any device that the BIOS recognises
- is independent of drive geometry
- can detect all RAM
- supports Logical Block Address mode (LBA)
- supports network booting
- supports remote (serial) terminals

OSSI — Bootloader — ver. 1.1 — p. 7/11

Installing grub into MBR from floppy

OSSI — Bootloader — ver. 1.1 — p. 6/11

- Create a grub installation disk
 - see the lab sheet to see how
- Boot computer with this disk, type:

```
grub> find /boot/grub/stage2
grub> root (hdx, y)
grub> setup (hdx)
```

OSSI — Bootloader — ver. 1.1 — p. 8/11

Installing grub into MBR from floppy — 2

- What happens when type

```
grub> find /boot/grub/stage2
```

- grub searches all partitions for that file
- lists the partitions that contain stage2 of grub

- What happens when type

```
grub> root (hdx, y)
```

- The “*x*” and “*y*” were found in previous step
- grub mounts the partition, and determines file system type

Installing grub into MBR from floppy — 3

- What happens when type

```
grub> setup (hdx)
```

- grub copies small stage1 to MBR of disk *x*
- installs a list of sectors containing the stage1_5 file
- aim is so stage1 can load stage1_5 when booting

Bootling using grub

- BIOS loads stage1 from MBR
- stage1 knows which sectors of hard disk contain stage1_5, so
 - loads stage1_5 into RAM,
 - jumps to that code
- stage1_5 can now read the file system containing stage2
- reads stage2 from hard disk
- stage2 can read the menu in /boot/grub/grub.conf
- stage2 can boot (almost?) any operating system from any disk BIOS can read