

Two loaders widespread

- LILO – LInix LOader
- Grub – Grand Unified Boot Loader

© David Morgan 2004

/sbin/lilo

- Input
 - file containing a kernel image (e.g., vmlinux)
 - file containing a boot loader (e.g., boot.b)
- Output
 - map file (w/coordinates of kernel file)
 - boot loader code (w/coordinates of map file)
- at the direction of /etc/lilo.conf

© David Morgan 2004

lilo.conf options*

- Some: install time role
- Others: boot time role

```
Boot=/dev/hda
delay=10

image=/boot/vmlinux      }
    root=/dev/hda4      } stanza
    label=linux          }

other=/dev/hda1          }
    label=Windows        } stanza
```

*GRUB conceptually same

© David Morgan 2004

/sbin/lilo

- Determine coordinates (eg, x,y,z) of to-be-loaded kernel image file (eg, /boot/vmlinux)
- write them as data into a file, /boot/map
- determine coordinates of /boot/map (eg, a,b,c)
- compose code:

```
go to a,b,c and read coordinates
run code at those coordinates
telling it to mount hda4
```

- put it in /dev/hda (ie, MBR)

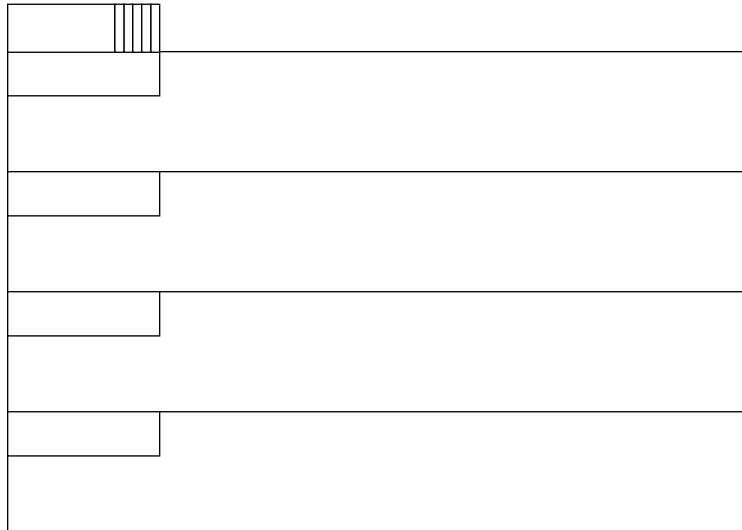
```
Boot=/dev/hda
delay=10

image=/boot/vmlinux
    root=/dev/hda4
    label=linux

other=/dev/hda1
    label=Windows
```

© David Morgan 2004

Hard disk layout

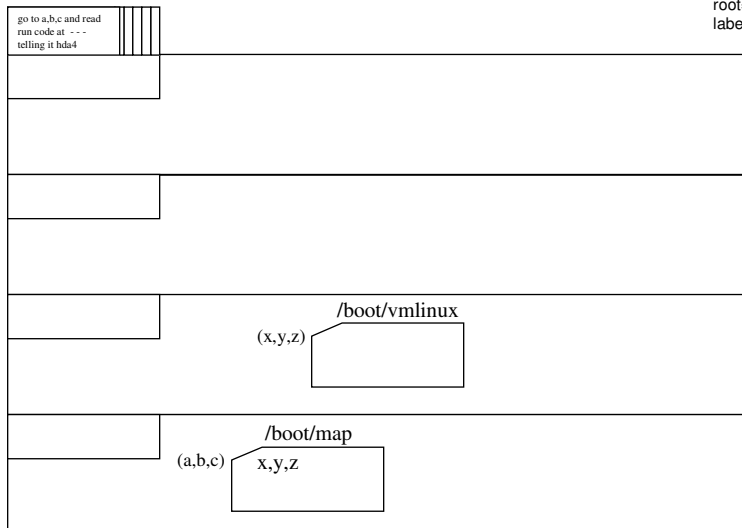


© David Morgan 2004

Install Time

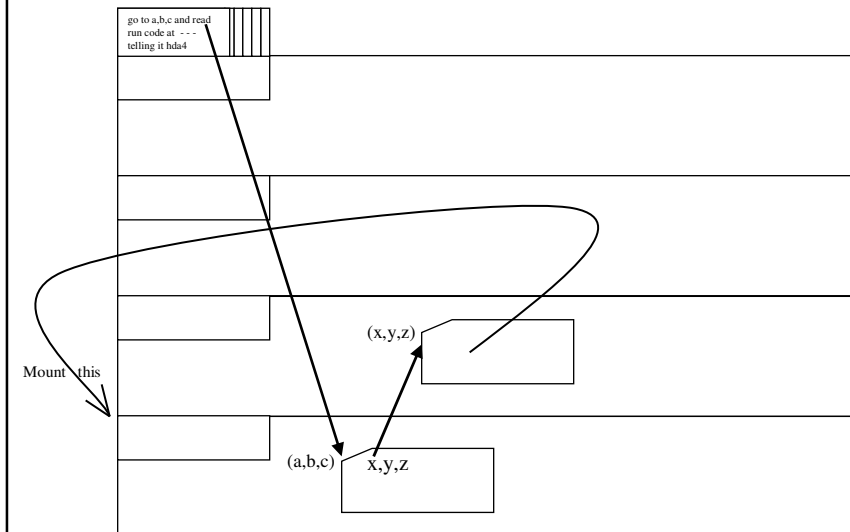
```
Boot=/dev/hda  
delay=10
```

```
image=/boot/vmlinux  
root=/dev/hda4  
label=linux
```



© David Morgan 2004

Boot Time



© David Morgan 2004

Grub files

- `/usr/share/grub/i386-pc/`

```
[root@EMACH1 i386-pc]# ls
e2fs_stage1_5  ffs_stage1_5  minix_stage1_5  stage1  vstafs_stage1_5
fat_stage1_5   jfs_stage1_5  reiserfs_stage1_5  stage2  xfs_stage1_5
```

- `/usr/share/doc/grub/`

```
[root@EMACH1 grub-0.92]# ls
AUTHORS  COPYING  ChangeLog  NEWS  README  TODO  menu.lst
```

- `/boot/grub/`

```
[root@EMACH1 grub]# ls
splash.xpm.gz
```

- `/sbin/grub`

© David Morgan 2004

Grub consists of images

- 2 essential stages
 - stage1
 - stage2
- optional stages
 - stage 1.5
- files found in `/usr/share/grub/i386-pc/`

© David Morgan 2004

Stage 1

- function: load stage2 or stage1.5 from local disk
- size: 512 bytes to fit MBR or boot sector

© David Morgan 2004

Stage 1.5

- optional bridge between stages 1 and 2
 - stage1 loads stage1.5
 - stage 1.5 loads stage2
- understands filesystems (per-filesystem images)
- allows for stage2 to reside in one

© David Morgan 2004

Stage 2

- core image of grub
- does everything but actual booting
- much larger (130K) than stage1 (512 bytes)
- usually put in a filesystem (ie, in a disk file)

© David Morgan 2004

Grub code you could run

- /sbin/grub
- grub code on a floppy
- grub code in hard disk's MBR

- all yield an interactive "grub>" prompt

© David Morgan 2004

Formula to effect a boot

- Set GRUB's root device to the drive where the OS images are stored, by the command "root"
- Load the kernel image, by the command "kernel"
- Run the command "boot"

© David Morgan 2004

Grub's disk nomenclature

- hda -> hd0 (disk: primary master)
hdb -> hd1 (disk: primary slave)
- hda1 -> (hd0,0) (partition: primary master's 1st)
hdc2 -> (hd2,1) (partition: secondary master's 2nd)

© David Morgan 2004

Disk nomenclature

	traditional labeling
first	hda
second	hdb
third	hdc
fourth	hdd

	GRUB's labeling
first	hd0
second	hd1
third	hd2
fourth	hd3

© David Morgan 2004

Disk nomenclature

traditional naming	
hda	primary master
hdb	primary slave
hdc	secondary master
hdd	secondary slave

But GRUB's labeling doesn't correspond to physical disks this way.

© David Morgan 2004

Determining nomenclature

```
Tera Term - 192.168.3.7 VT
File Edit Setup Control Window Help
[root@EMACHI root]# fdisk -l
Disk /dev/hdc: 30.0 GB, 30020272128 bytes
16 heads, 63 sectors/track, 58168 cylinders
Units = cylinders of 1008 * 512 = 516096 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/hdc1 *           1           507     255496+   83  Linux
/dev/hdc2             508        30000    14864472   5  Extended
/dev/hdc5             508         1500     500440+   82  Linux swap
/dev/hdc6            1501        30000    14363968+  83  Linux

Disk /dev/hda: 8455 MB, 8455200768 bytes
255 heads, 63 sectors/track, 1027 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/hda1 *           1           500     4016218+  83  Linux
/dev/hda2             501         533     265072+   82  Linux swap
[root@EMACHI root]#
```

This machine has a primary master (hda) and secondary master (hdc) only. There is no hdb.

Put uniquely named files on partitions. Let GRUB's "find" command report where they reside, to reveal nomenclature.

Placed on hda1: /file_on_hda1
 hdc1: /file_on_hdc1
 hdc6: /file_on_hdc6

© David Morgan 2004

Determining nomenclature

Placed on hda1: /file_on_hda1
hdc1: /file_on_hdc1
hdc6: /file_on_hdc6

```
Tera Term - 192.168.3.7 VT
File Edit Setup Control Window Help

GRUB version 0.93 <640K lower / 3072K upper memory>
[ Minimal BASH-like line editing is supported. For the first word, TAB
  lists possible command completions. Anywhere else TAB lists the possible
  completions of a device/filename.]

grub> find /file_on_hda1  ← So GRUB calls hda "hd0"
(hd0,0)

grub> find /file_on_hdc1  ← but calls hdc "hd1" (not "hd2")
(hd1,0)

grub> find /file_on_hdc6
(hd1,5)
```

© David Morgan 2004

Multiple config files

```
Tera Term - 192.168.3.7 VT
File Edit Setup Control Window Help

GRUB version 0.93 <640K lower / 3072K upper memory>
[ Minimal BASH-like line editing is supported. For the first word, TAB
  lists possible command completions. Anywhere else TAB lists the possible
  completions of a device/filename.]

grub> find /boot/grub/grub.conf
(hd0,0)
(hd1,5)
```

Two of them on this system.
Be mindful which copy you are working with.

© David Morgan 2004

Getting grub into mbr

- run `/sbin/grub`
- manually execute:

```
grub> root (hd0,0)*  
grub> setup (hd0)
```

- writes bootloader code into MBR of `hd0`

* if `(hd0,0)` is the partition containing "the stuff" (`stage1`, `stage2`, `grub.conf`) in `/boot/grub`

© David Morgan 2004

Get Grub onto a floppy

```
# dd if=/usr/share/grub/i386-redhat/stage1 of=/dev/fd0 count=1  
# dd if=/usr/share/grub/i386-redhat/stage2 of=/dev/fd0 seek=1
```

© David Morgan 2004

Manual boot from “grubbed” floppy

- boot linux

```
grub> root (hd0,2)
grub> kernel /boot/vmlinuz-2.4.19-12m root=/dev/hda3
grub> initrd /boot/initrd-2.4.19-12m.img
grub> boot
```

- boot windows

```
grub> root (hd0,0)
grub> makeactive
grub> chainloader +1
grub> boot
```

© David Morgan 2004

Automating Grub

- copy all files
 - from /usr/share/grub/i386-pc
 - to /boot/grub/ (but, which one?)
- fashion /boot/grub/grub.conf

© David Morgan 2004

Make an auto-boot floppy

```
Tera Term - 192.168.3.26 VT
File Edit Setup Control Window Help
[root@monarch root]# fdformat /dev/fd0 > /dev/null
[root@monarch root]# mkfs /dev/fd0 > /dev/null
mkfs 1.32 (09-Nov-2002)
[root@monarch root]# mount /dev/fd0 /mnt
[root@monarch root]# mkdir -p /mnt/boot/grub
[root@monarch root]# cp /usr/share/grub/386_robhat.stage1 /mnt/boot/grub/
[root@monarch root]# cat > /mnt/boot/grub/grub.conf
title Boot Something
  root (hd0,1)
  kernel /boot/vmlinuz root=/dev/hda2
[root@monarch root]# grub
```

GRUB wants this directory

Containing files
stage1
stage2
grub.conf

first disk,
second partition

Now we're ready to write
all this stuff to the floppy
(next slide)

© David Morgan 2004

Make an auto-boot floppy

```
Tera Term - 192.168.3.26 VT
File Edit Setup Control Window Help

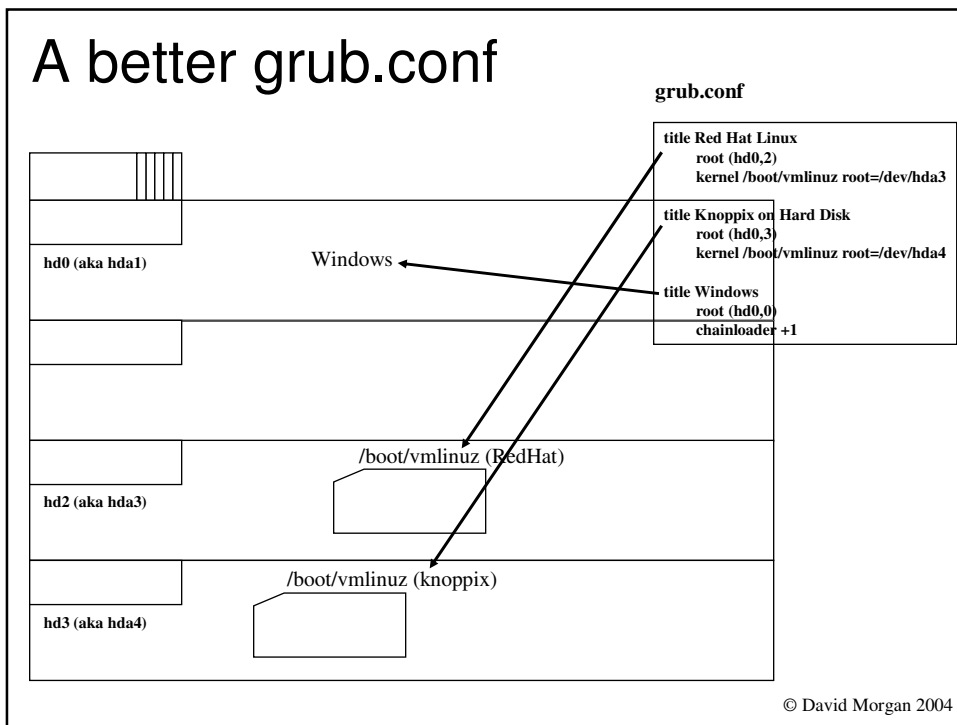
GRUB version 0.93 <640K lower / 3072K upper memory>
[ Minimal BASH-like line editing is supported. For the first word, TAB
  lists possible command completions. Anywhere else TAB lists the possible
  completions of a device/filename.]
grub> root (fd0) ← Tells GRUB to get stuff, and where from (floppy)
Filesystem type is ext2fs, using whole disk
grub> setup (fd0) ← Tells GRUB to put stuff, and where to (floppy)
Checking if "/boot/grub/stage1" exists... yes
Checking if "/boot/grub/stage2" exists... yes
Checking if "/boot/grub/e2fs_stage1_5" exists... no
Running "install /boot/grub/stage1 (fd0) /boot/grub/stage2 p /boot/grub/grub.c
onf"... succeeded
Done.
grub> █
```

You can now boot from the floppy, with screen prompt "Boot Something,"
to the copy of linux on first disk second partition

to boot copies of operating systems in other partitions, with better
screen prompts, rewrite the /boot/grub/grub.conf that's on the floppy

© David Morgan 2004

A better grub.conf



Put it on the hard disk

First pre-position the “stuff” on 1st disk’s 1st logical

```

Tera Term - 192.168.3.26 VT
File Edit Setup Control Window Help

GRUB version 0.93 (640K lower / 3072K upper memory)
[ Minimal BASH-like line editing is supported. For the first word, TAB
  lists possible command completions. Anywhere else TAB lists the possible
  completions of a device/filename. ]
grub> root (hd0,4) ← Tells GRUB to get stuff, and where from
Filesystem type is ext2fs, partition type 0x83 (1st disk's 1st logical)
grub> setup (hd0)
Checking if "/boot/grub/stage1" exists... yes
Checking if "/boot/grub/stage2" exists... yes
Checking if "/boot/grub/e2fs_stage1_5" exists... no
Running "install /boot/grub/stage1 (hd0) /boot/grub/stage2 p /boot/grub/grub.c
onf "... succeeded
Done.
grub> ← Tells GRUB to put stuff, and where to
(1st disk's MBR)
  
```

Disk specified in “root ()” directive becomes the one whose config file (/boot/grub/grub.conf by default) will be read and followed each boottime

© David Morgan 2004

How to get config file on disk?

- Purpose of this exercise: to enable booting
- Ergo: you can't boot
- Need to put config file into a place you can't get to in order to get to it
- This is circular! You're stuck!

One (good) solution

Boot an in-memory copy of linux from CD. From that platform, mount the partition(s) where you need to put stuff and put it there.

Try Knoppix - <http://www.knoppix.com>

Or other "live CD" solutions -

<http://www.frozentech.com/content/livecd.php>

© David Morgan 2004

A grub.conf

```
# Note that you do not have to rerun grub after making changes to this file
# NOTICE: You have a /boot partition. This means that
#   all kernel and initrd paths are relative to /boot/, eg.
#   root (hd0,1)
#   kernel /vmlinuz-version ro root=/dev/hda3
#   initrd /initrd-version.img
#boot=/dev/hda
default=1
timeout=10
splashimage=(hd0,2)/grub/splash.xpm.gz
title linux15
    root (hd0,2)
    kernel /boot/vmlinuz-2.4.18-14 ro root=/dev/hda3
    initrd /boot/initrd-2.4.18-14.img

title RH8
    root (hd1,1)
    kernel /boot/vmlinuz-2.4.18-14 ro root=/dev/hdb2 init=/sbin/init
    initrd /boot/initrd-2.4.18-14.img
```

© David Morgan 2004

A default grub.conf

```
# grub.conf generated by anaconda
#
# Note that you do not have to rerun grub after making changes to this
# NOTICE: You do not have a /boot partition. This means that
#   all kernel and initrd paths are relative to /, eg.
#   root (hd0,2)
#   kernel /boot/vmlinuz-version ro root=/dev/hda3
#   initrd /boot/initrd-version.img
#boot=/dev/hda
default=0
timeout=10
splashimage=(hd0,2)/boot/grub/splash.xpm.gz
title Red Hat Linux (2.4.20-8)
    root (hd0,2)
    kernel /boot/vmlinuz-2.4.20-8 ro root=LABEL=/1 hdc=ide-scsi
    initrd /boot/initrd-2.4.20-8.img
```

© David Morgan 2004

Biblio

- Grub manual
 - <http://www.gnu.org/software/grub/manual/>
- “Multibooting with Grub,” Linux Magazine, 11-12/2003
 - http://www.linux-mag.com/2003-11/guru_01.html
 - http://www.linux-mag.com/2003-12/guru_01.html

© David Morgan 2004