# – LPI101 –

# Install a boot manager [1]

## (Linux Professional Institute Certification)

a

```
  .~.
  /V\       by: Andrew Eager
 // \\         andy@linuxivr.com
 @._.@
```

```
$Id: gl1.102.2.slides.tex,v 1.2 2003/05/30 05:01:31 waratah Exp $
```

---

# Install a boot manager [1]

## Objective

Candidate should be able to select, install and configure a boot manager. This objective includes providing alternative boot locations and backup boot options (for example, using a boot floppy).

# Install a boot manager [1]

## Key files, terms, and utilities

```
/etc/lilo.conf
lilo
/boot/grub/grub.conf
grub-install
MBR
superblock
first stage boot loader
```

## Resources of interest

**LinuxGazette GRUB article** :

```
http://www.linuxgazette.com/issue64/kohli.html:
```

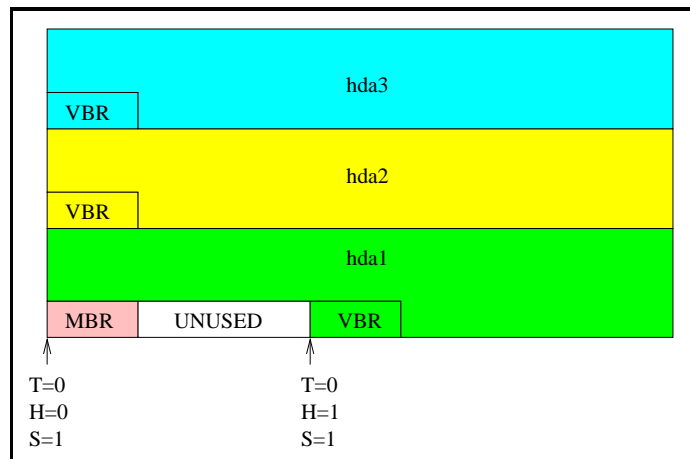**The GNU Grub Site** :

```
http://www.gnu.org/software/grub/
```

# Disk Organisation

- A disk is organised into:

  - Cylinders

  - Heads

  - Sectors

- All sectors of Cyl 0, Head 0 are reserved:

# **Disk Organisation**

- Fdisk on a floppy confirms this:

```
Disk /dev/fd0: 2 heads, 18 sectors, 80 cylinders
Units = cylinders of 36 * 512 bytes
```

| Device | Start | End | Blocks |
|---|---|---|---|
| /dev/fd0p1 | 1 | 1 | 9 |
| /dev/fd0p2 | 2 | 2 | 18 |
| /dev/fd0p3 | 3 | 3 | 18 |
| /dev/fd0p4 | 80 | 80 | 18 |

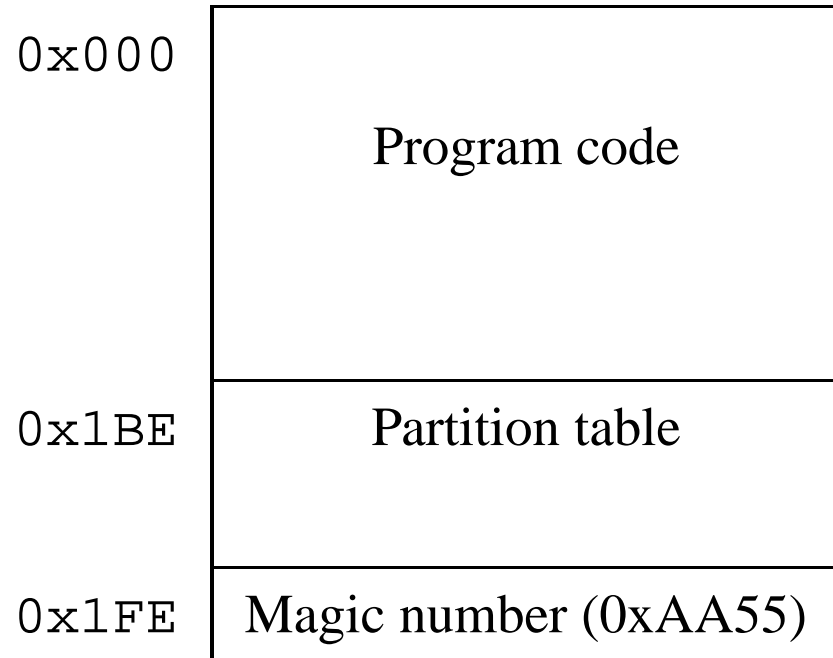- Note the first partition is smaller than the rest

# The DOS Boot Process

- A Power On Self Test is performed.

- Control is passed to the boot device

- The first sector on disk is loaded into memory

- The MBR code is executed:

  - It reads the partition table

  - Looks for an active partition

  - Loads the boot sector of active partition

  - Executes the code in that boot sector

# The Master boot record

- The MBR is the first block (sector) on the disk

- It contains:

  - Boot Code: 446 bytes

  - Partition Table 64 bytes

  - MBR Signature 2 bytes (0x55AA)

  - TOTAL 512 bytes

# The Master boot record

```
0x000   ┌─────────────────────────┐
        │                         │
        │      Program code       │
        │                         │
        │                         │
0x1BE   ├─────────────────────────┤
        │      Partition table    │
        │                         │
0x1FE   ├─────────────────────────┤
        │  Magic number (0xAA55)  │
        └─────────────────────────┘
```

# Boot Loaders

A boot loader is a program that is responsible for selecting the operating system to be booted. Once the desired OS is selected for boot, the boot loader must transfer control to it. Typical boot loaders are:

- The DOS boot loader. Not interactive

- LILO: An interactive boot loader

- GRUB: An interactive boot loader

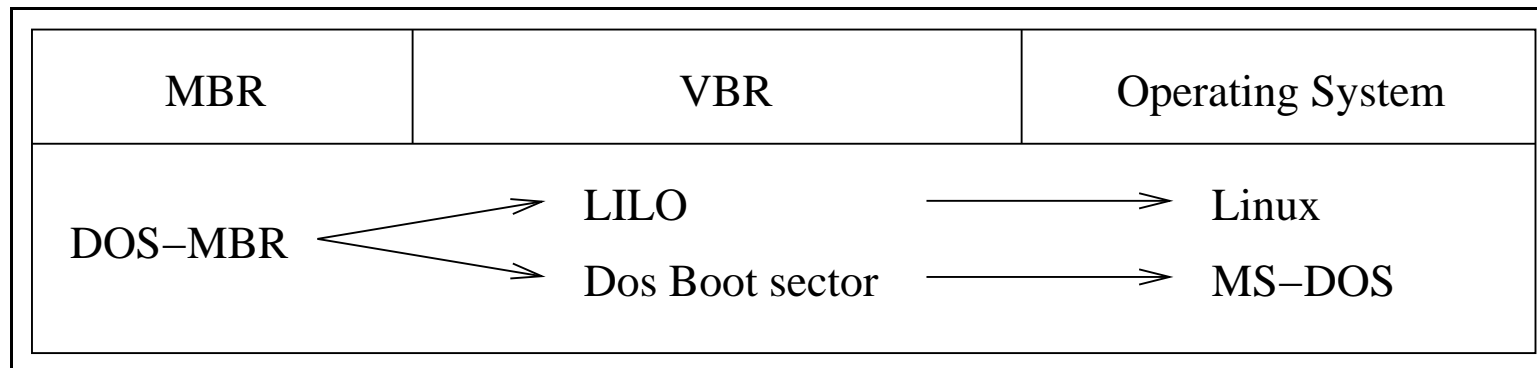- Boot Manager: An interactive boot loader

# LILO - The Linux Loader

- LILO is a two stage loader:

- Stage 1 is 446 bytes long & is located in *either*

  - The MBR (Master boot record. Eg in /dev/hda) *OR*

  - The VBR (Volume boot record. Eg in /dev/hda1)

- Stage 2 is a file (/boot.b) located in /boot it contains the bulk of the lilo functionality

# LILO Boot sequence

## LILO in the VBR

When LILO is installed in the VBR, it is up to the MBR to transfer control to it. This is normally done by setting the partition where LILO is installed to 'active'. You would use fdisk (either dos or linux) to do this:

| MBR | VBR | Operating System |
|-----|-----|------------------|

DOS−MBR →→ LILO ——————→ Linux

DOS−MBR →→ Dos Boot sector ——————→ MS−DOS

# LILO Boot sequence

## LILO in the MBR

When LILO is installed in the MBR, the BIOS transfers control directly to LILO's 1st stage boot loader. In this case, the VBR is not used and it does not matter which partition is marked active in the partition table:
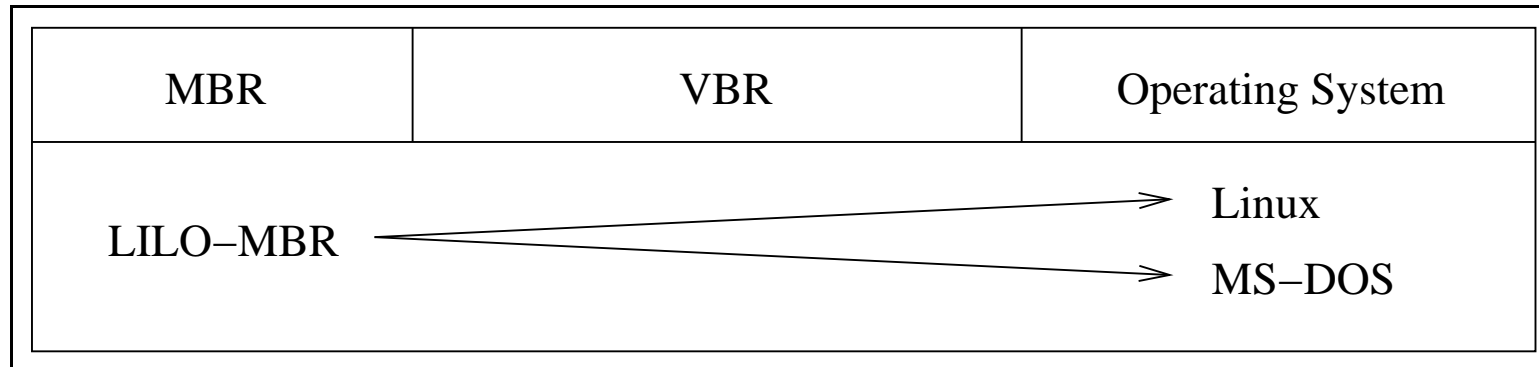
| MBR | VBR | Operating System |
|-----|-----|------------------|
| LILO–MBR | | Linux |
| | | MS–DOS |

# LILO

## LILO and the BIOS

LILO uses the BIOS INT13 calls to read the following files:

- /boot/boot.b - The second stage loader

- /boot/map - The map file

- All kernels

- The Volume Boot sectors of all OS's it boots

- The boot message if one has been set

# LILO

## Configuring LILO

LILO is entirely configured within the file /etc/lilo.conf. This file contains three areas.

- Global Options (eg signon message, delay time etc)

- General 'per operating system' options

- Linux kernel options (Linux only - eg root device, ro etc)

# LILO

## Installing LILO

Once LILO has been configured, it can be installed by running **/sbin/lilo** as root from the command line. This will do the following:

- Make a backup copy of the boot sector (MBR or VBR)

- Create a /boot/map file with the locations of:
    - /boot/boot.b
    - The kernel
    - /boot/message
    - /boot/chain.b

- Make a boot sector containing the location of the map file

- Write the boot sector to either the MBR or VBR.

## LILO

## LILO diagnostics

When LILO runs, it prints the word 'L I L O' in the top left hand corner of the screen, one letter at a time:

- 'L' - Stage 1 loaded & running.

- 'LI' - Stage 2 was loaded but could not run

- 'LIL' - Stage 2 was loaded but could not locate the descriptor table

- 'LIL?' - Stage 2 was loaded at an incorrect address

- 'LIL-' - Stage 2 was loaded but descriptor table is corrupt

- 'LILO' - LILO has started up correctly.

# GRUB

GRUB operates in a similar manner to LILO. It is a two stage loader that can load stage 1 into either the MBR or VBR.

The main differences between LILO and GRUB are:

- Stage 2 can be located beyond the 1024 cylinder limit.

- GRUB can boot from network devices using tftp

- GRUB has many more configuration options

- GRUB is effectively a mini shell and can be configured 'on-the-fly' by giving it commands directly at boot time.

## GRUB

## Hard Disk Naming Conventions

GRUB uses a different naming convention to that of the rest of the Linux world. In general, the syntax used is: hdN,X where: N is the hard disk number (0 for the first, 1 for the second...) X is the partition number (0 for the first, 1 for the second...)

Examples: (hd0,1) refers to the 1st physical drive, 2nd primary partition (hd1,6) refers to the 2nd physical drive, 3rd logical partition (fd0) refers to the floppy drive

A file within a partition can be referred to by appending a path to it: EG: (hd0,0)/boot/vmlinuz, refers to the file /boot/vmlinuz on the 1st drive, 1st partition

## GRUB

## Installing GRUB

Once the configuration file has been created, there are two ways to install
GRUB:

Option 1: Run grub from the command line:

```
[root@cds grub]# /sbin/grub
grub> root (hd0,6)
grub> setup (hd0)
        or;
grub> setup (hd0,4)
```

**GRUB**

## Installing GRUB

Option 2: Use the grub-install command: To install into the MBR:

```
[root@cds grub]# grub-install /dev/hda
```
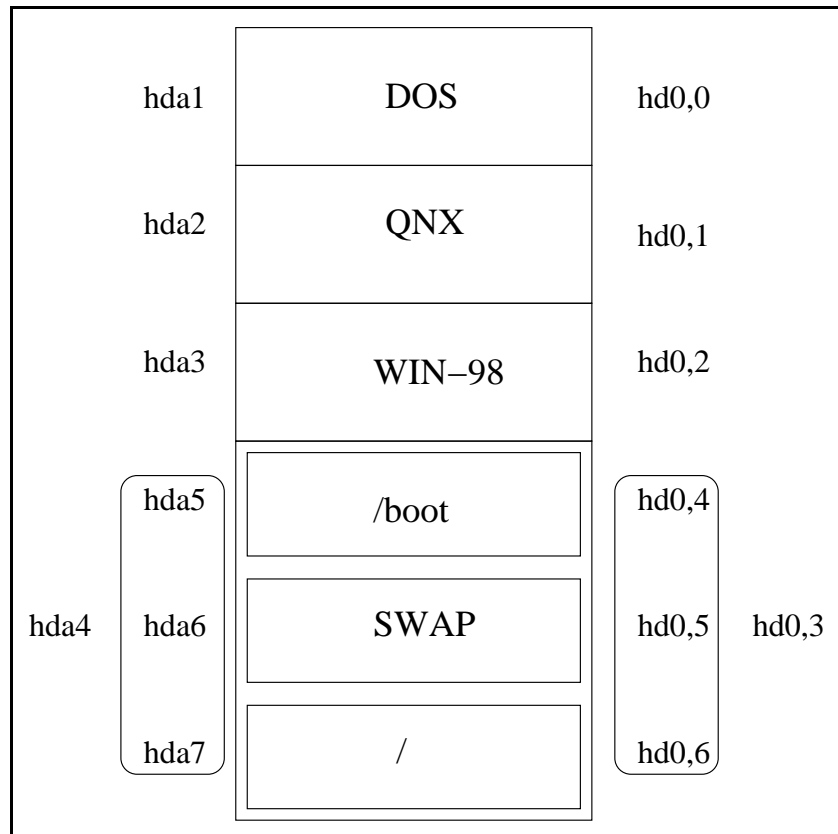
To install into the VBR:

```
[root@cds grub]# grub-install /dev/hda5
```

If you have a separate /boot partition, then you need to specify boot directory:

```
[root@cds grub]#
grub-install --root-directory=/boot /dev/hda5
```

# Sample Installation

The diagram below shows the disk layout used for the sample configuration scripts.

| | | |
|---|---|---|
| hda1 | DOS | hd0,0 |
| hda2 | QNX | hd0,1 |
| hda3 | WIN–98 | hd0,2 |
| hda5 | /boot | hd0,4 |
| hda4    hda6 | SWAP | hd0,5    hd0,3 |
| hda7 | / | hd0,6 |

# Sample Installation

## LILO configuration

```
# Global configuration options
boot=/dev/hda
timeout=20
message=/boot/message
prompt
default=linux
vga=normal
map=/boot/map
install=/boot/boot.b
```

```
# Per image options
other=/dev/hda3
    label=WIN98


other=/dev/hda2
    label=QNX


other=/dev/hda1
    label=DOS


image=/boot/vmlinuz
    label=linux
    root=/dev/hda7
    read-only
```

# Sample Installation

## GRUB configuration

```
default=0
timeout=10
fallback=1
splashimage=(hd0,0)/boot/grub/splash.xpm.gz

title RedHat 7.2
        root (hd0,4)
        kernel /vmlinuz ro root=/dev/hda7
        initrd /initrd.img
```

```
title Windows
        root (hd0,2)
        makeactive
        chainloader +1

title QNX
        root   (hd0,1)
        makeactive
        chainloader +1

title DOS
        root   (hd0,0)
        makeactive
        chainloader +1
```