



# The Pegasos book

« Le livre du Pegasos »

Compilation of articles on the Pegasos  
Volume 4 : Linux, MacOSX and other systems

May 2007 edition - By Geoffrey CHARRA (V2.3)

Translated from french by Geoffrey CHARRA (December 2007),  
With the help of Eric WALTER, Thibault JEANSON, Denis HILLIARD (translation)  
and Fulvio PERUGGI (corrections)

Sponsored by





## Contents

<u>1 Linux.....</u>	<u>4</u>
1.1 Global information.....	4
1.1.1 Linux distributions running on Pegasos.....	4
1.1.2 General information on Linux usage for beginners.....	5
1.1.3 Hard disk partitioning under Linux on Pegasos.....	5
1.1.4 Multi-boot with bootcreator.....	7
1.1.5 Using a Linux System from Another Linux System by Changing root.....	8
1.1.6 NFS : Network File System and Remote Compiling.....	8
1.2 Installation of Linux Debian on Pegasos I and II.....	10
1.2.1 Installation types.....	10
1.2.2 Installation .....	10
1.2.3 Tricks and hints.....	11
1.3 Installation of Linux Ubuntu on Pegasos II.....	13
1.3.1 Pre-requisite.....	13
1.3.2 Start-up and install.....	13
1.3.3 Post-installation steps needed on a Pegasos.....	17
1.3.4 First boot and Xorg update.....	17
1.4 Installation of Linux Suse on Pegasos II.....	19
1.4.1 Preparing.....	19
1.4.2 Setting up the installation environment .....	21
1.4.3 YaST startup and starting of the graphical installation.....	22
1.4.4 Making Linux SUSE bootable from hard disk .....	25
1.4.5 Troubleshooting.....	26
1.4.6 Update from a previous version.....	26
1.5 Media Center with Linux Gentoo on Pegasos II.....	27
1.5.1 Objective.....	27
1.5.2 Presentation of Genesi's HMCS .....	27
1.5.3 Necessary hardware .....	28
1.5.4 General information on Gentoo .....	28
1.5.5 Easy install with "EZPegTV" CD.....	30
1.5.6 Customized install, step by step.....	31
1.5.7 Backup of the system .....	52
1.6 Installation of Linux Fedora Core on Pegasos II.....	53
1.6.1 Preparing.....	53
1.6.2 Installation.....	53
1.6.3 Update.....	56
1.7 Installation of Linux Yellow Dog on Pegasos II.....	57
1.7.1 Preparing.....	57
1.7.2 Installation.....	57
1.8 Installation of Linux Crux on Pegasos II.....	60
1.8.1 Preparing.....	60
1.8.2 Installation of packages.....	61
1.8.3 System configuration.....	61
1.8.4 Compilation of the kernel.....	61
1.8.5 Starting Crux PPC.....	62
<u>2 MacOS X.....</u>	<u>63</u>
2.1 Running principles on Pegasos.....	63

2.2 Installation of PegXMac on Pegasos II.....	64
2.2.1 Pre-requisite.....	64
2.2.2 Starting PegXMac.....	64
2.2.3 Configuring Mac-On-Linux.....	65
2.2.4 MOL-Menu : additional configuration.....	66
2.2.5 Installation of MacOS X.....	68
2.2.6 Troubleshooting.....	70
2.3 Installation of MacOS X on Pegasos II from a minimal Linux Debian.....	72
2.3.1 Introduction.....	72
2.3.2 Installation of a minimal Linux distribution.....	72
2.3.3 Installation of Mac-On-Linux.....	72
2.3.4 Installation of MacOS X.....	74
2.4 Installation of MacOS X on a Pegasos II from Linux Ubuntu.....	76
2.4.1 Preparing the system.....	76
2.4.2 Compilation of MOL modules.....	76
2.4.3 Installation of MOL.....	77
2.4.4 Running Mac-On-Linux as a normal user.....	77
2.5 Configuring and running Mac-on-Linux.....	77
2.5.2 Configuring Network.....	78
2.5.3 FAQ.....	78
2.6 Installation of MacOS X with MolK.....	79
2.6.1 Installation.....	79
2.6.2 MolK setup.....	79
2.6.3 Installation of MacOS X.....	81
2.7 Configuring a Freebox with Mac-On-Linux (MacOS X 10.3).....	82
3 Other systems with LiveCD.....	83
3.1 AROS-Live.....	83
3.2 Freevo-Live.....	83
3.3 CDGA (CD Game Arcade).....	84
3.4 AmigaOS/CD32-Live.....	85
3.5 Others LiveCD.....	85
4 Tricks and hints under Linux with a Pegasos.....	86
4.1 Use a Nintendo Wiimote with a Pegasos.....	86
4.1.1 Cwiid Driver Installation.....	86
4.1.2 Bluetooth Configuration.....	87
4.1.3 Recognising the Wiimote.....	88
4.1.4 wminput, wmgui and wmdemo.....	88
4.1.5 And now what can we do with it ?.....	88
4.2 Using a cellular as a modem.....	88
5 Addendum.....	89
5.1 Complementary information on Linux Debian for Pegasos I and II.....	89
5.1.1 Installation of Linux Debian on Pegasos I.....	89
5.1.2 Networks problems with some Pegasos I.....	89
5.1.3 Loss of network support (Pegasos I or II).....	90
5.2 Complementary information about Linux Ubuntu for Pegasos II.....	90
5.2.1 Base install of Edgy 6.10 version (with GNOME window manager).....	90
5.2.2 Base install of Dapper 6.06 version (with GNOME window manager).....	91
5.2.3 Base install of Breezy 5.10 version (with GNOME window manager).....	96
5.2.4 Installing EasyUbuntu.....	100
5.2.5 Installing KDE.....	100
5.2.6 Troubleshooting.....	101
5.2.7 Installation of Xfce.....	101
5.2.8 Installation of Enlightenment e17.....	102
5.2.9 Installation of MacOSX.....	106

# 1 Linux

## 1.1 Global information

### 1.1.1 Linux distributions running on Pegasos

Linux is an OpenSource operating system of UNIX type created by Linus Torvalds and continuing to evolve thanks to many developers in the world.

Linux became very popular because it has many interesting features :

Free (OpenSource)

Great range of software available for free.

Very stable as based on the UNIX system.

Very efficient for network applications.

Multi-user.

Multi platforms.

It has nevertheless some weak points:

Not very easy to use, nor to install.

Quite heavy, mostly on boot.

Not very efficient for graphic applications.

Several distributions of Linux are available on Pegasos :

Distribution	Lien(s)	PegI	PegII	Easy install (*)	Interest (*)
Gentoo / Luminocity	<a href="http://www.gentoo.org/">http://www.gentoo.org/</a>	?	X	1/10	10/10
Ubuntu	<a href="http://www.ubuntulinux.org">http://www.ubuntulinux.org</a>		X	9/10 (v5.10) 5/10 (v6.x)	8/10
OpenSuse	<a href="http://www.opensuse.org/Download">http://www.opensuse.org/Download</a> <a href="http://www.opensuse.org/PPC:Boot_pegasos">http://www.opensuse.org/PPC:Boot_pegasos</a>		X	7/10	7/10
Debian	<a href="http://www.debian.org/CD/http-ftp">http://www.debian.org/CD/http-ftp</a>	X	X	6/10	7/10
Fedora Core	<a href="http://fedora.redhat.com/">http://fedora.redhat.com/</a>	?	X	10/10	6/10
Yellow dog	<a href="http://www.yellowdoglinux.com/">http://www.yellowdoglinux.com/</a> <a href="http://www.amipeglin.de.vu/">http://www.amipeglin.de.vu/</a> (installation notes for Pegasos I)	X	X	9/10	5/10
Crux	<a href="http://cruxppc.sunsite.dk/">http://cruxppc.sunsite.dk/</a>	?	X	2/10	4/10
PegXLin		?	X		
Holonlinux	<a href="http://www.holonlinux.com/">http://www.holonlinux.com/</a>				
Mandriva	<a href="http://www.mandriva.com/">http://www.mandriva.com/</a>				
Montavista	<a href="http://www.mvista.com/">http://www.mvista.com/</a>				
NiktariX / Knoppix	Knoppix : <a href="http://www.knoppix.org">http://www.knoppix.org</a> <a href="http://pegasos.jinak.cz/clanky/niktariX.html">http://pegasos.jinak.cz/clanky/niktariX.html</a>	X	X		
Skolelinux	<a href="http://www.skolelinux.org/">http://www.skolelinux.org/</a>				

For additional information, see <http://www.linux.org/>.

PenguinPPC (Linux PPC port home page) [www.penguinppc.org](http://www.penguinppc.org).

PegasosPPC the home of the Pegasos computer <http://www.pegasosppc.com/>.

Kernel.org (where you will find the latest Linux kernel) <http://www.kernel.org/>.

(\*) This is a personal opinion and some people might not agree. My favorite distributions are Ubuntu (easy install except since 6.x, nice distribution based on Debian update system) and Gentoo (highly customisable and very up to date for Pegasos ppc). For rookies, I would recommend Fedora Core which is really easy to install.

### 1.1.2 General information on Linux usage for beginners

#### *User « root »*

The « root » user is the administrator, which signifies that the root has all the privileges. In a console, to go to root mode, type "su" and then the root's password.

#### *Help about a command*

To have help about a command : in a console, type « man the\_command\_name ».

#### *Keyboard shortcuts and commands in Xorg/XFree (graphic user interface of Linux)*

To go from the text mode (console) to Xorg/XFree, press on "CTRL+ALT Fx" (X = 1 to 6, 7=XFree). For easily mounting/un a dismounting a disk, use KwikDisk from KDE menu.

To kill XFree (in the event of a crash): press on "CTRL+ALT+BACKSPACE".

To kill a software: launch "xkill" in a console and click on the unpleasant program !

To kill a program in text mode, use "kill" command ("man kill" for more help).

### 1.1.3 Hard disk partitioning under Linux on Pegasos

#### 1.1.3.1 Partitioning on Pegasos

Smartfirmware on the Pegasos, does not provide the RTAS methods which allow Linux to modify NVRAM settings and make its own partitions bootable: you will have to do that manually.

If your root is on a Logical Volume Managed (LVM) partition, then /boot should be on a separate disk to be able to boot.

Smartfirmware on the Pegasos numbers its partitions incorrectly starting from 0 instead of 1 according to the CHRP specification. So /dev/hda1 is hd:0 ; disk 0 is named "hd" and disk 1 "hd0".

#### IDE:

/dev/hda = first port IDE, master -> unit 0.

/dev/hdb = first port IDE, slave -> unit 1.

/dev/hdc = second port IDE, master -> unit 2.

/dev/hdd = second port IDE, slave -> unit 3.

/dev/hda1 = first port IDE, master, first partition.

/dev/hda2 = first port IDE, master, second partition.

/dev/hdb1 = first port IDE, slave, first partition.

/dev/hba5 = first port IDE, slave, fifth partition.

/dev/hbc2 = second port IDE, master, second partition.

/dev/hbd3 = second port IDE, slave, third partition.

#### SCSI :

/dev/sda -> unit 0...

#### 1.1.3.2 Preliminary advice to format the partitions

It is possible to create the partitions in Linux, like we are going to see with "parted", but it is also possible to create them in MorphOS so that they are recognized by the two systems. It is necessary in addition to delete the Master Boot Record if the hard disk has been used in a PC.

Under MorphOS, with SCSIConfig, you must create at least two partitions. One for Linux itself, the other one for the Swap partition :

- For the Linux partition, you need at least of 2 Gb, choose Custom fileSystem, and enter the type "0x4c4e5800".
- For the Swap partition, you need at least 200 Mb, chooses Custom fileSystem and enter the type "0x53575000".

Now, it is necessary to be able to identify your Linux partitions emplacement. This is very important for the installation. For more information about this, see the "Linux global information" chapter.

Examples :

Example : if you only have one hard disk and it is on ide.device, unit 0 like this :

DH0: FFS  
DH1: FFS  
DH2: SFS  
DH3: SFS  
DH4: SFS  
DH5: Linux  
DH6: Linux  
DH7: Swap

For Linux, the ext2fs filesystem if your hard disk is on unit 0 -> /dev/hda6.

Remarks :

- It is better to use two hard disks. One for MorphOS and the other one for the different Linux distributions. This avoid human errors like deleting partition tables accidentally.
- It is possible to create the partitions under Linux, but the better is to create them into MorphOS so that they can be recognized by both systems. You also have to delete the MBR (Master Boot Record) if the hard disk has already been used into a PC (or sometimes when it's new).

For that, you can use MorphOS. On the installation CD of MorphOS :

MorphOS < 1.4.x : `sys:tools/debug/hdwrite`

MorphOS >= 1.4.x : `MorphOSBoot:morphos/c/hdwrite`

`hdwrite ide.device hard_disk_number mossys:c/reboot 0`

example : `hdwrite ide.device 1 mossys:c/reboot 0` for disk 1

### 1.1.3.3 Information about "parted" tool usage

Parted is a partitioning command line tool used in several distribution. Using it is really easy. Here is list of most important commands. Beware, that everything is executed immediatelly, there is no *undo* function. This is just a short command summary, for a complete list, visit <http://www.gnu.org/software/parted/manual/>

First, a little reminder about partition naming :

To get started, one needs **parted /dev/hda** to edit partitions on your primary master IDE disk. Replace the device name, if an other hard drive is to be edited. When started, *parted* will drop you in some kind of command line. Anything you type there is executed immediatelly, so be careful!

**Help** will display a short help

**print** will list existing partitions, if there is any.

**mklabel amiga** creates an amiga partition table. **All previous data is lost!**

**mklabel msdos** creates an msdos partition table. **All previous data is lost!**

**mkpart type fs start end** creates a partition. It does not format the partition, but creates the correct entry in the partition table.

- **type** is always *primary* in the case of Amiga partition table, a **p** is enough
- **fs** in any of the following: *ext2*, *ext3*, *reiserfs*, *linux-swap*, etc.
- **start** and **end** represent the start and end point of a partition in mega or gigabyte. **M** is appended to the number to represent a megabyte, **G** stands for gigabyte.

Example:

```
mkpart p ext2 0M 100M
mkpart p linux-swap 100M 612M
mkpart p ext3 612M 10G
mkpart p ext3 10G 40G
```

**quit** quits the program. There is no need to save, everything is saved as soon, as you hit enter after a command.

### 1.1.3.4 Mounting partitions of other operating systems

*How to mount a FFS partition under Linux ?*

Firstly, only FFS partitions are supported. The command is simple, in root mode, type: "`mount /dev/xxxx /mnt/myamigapartition -t affs -o user,rw`" (to mount the partition "xxxx"). You can also add a line in /etc/fstab to mount your Amiga partition at each start :  
`/dev/xxxx /mnt/myamigapartition affs user,noauto 0 0`

*How to mount a Linux partition under MorphOS ?*

For that it is necessary to download the Ext2/Ext3 filesystem for MorphOS: <http://home.elka.pw.edu.pl/%7Emszyrow/programy/ext2filesystem/>.

The file `ext2fs_0.4_mos.lha` contains the necessary files to manage the EXT2 and EXT3 Linux filesystems. While following exactly the 'readme' file, it should not be a problem to install it and to have access to the Linux partition in a transparent way.

### 1.1.4 Multi-boot with bootcreator

This paragraph will be helpful after you will have installed more than one operating system on your Pegasos, because in this case, you will need a multi-boot tool.

For that, you can use "bootcreator" which is for example available in the Gentoo packages (or under MorphOS).

To install it from Linux Gentoo, just type "emerge bootcreator" from your Gentoo system.

To install it from another Linux distribution, get the sources and compile them (configure; make; make install).

Once installation is done, you need a small but empty partition which will be your boot partition with ext3 filesystem where you will put all your boot kernels. For that, you can use "parted" or "gparted" if it's installed.

Then, in all your Linux systems, you must:

- 1) Edit the /etc/fstab file, and add an entry such as the following one:

```
/dev/sda1 /boot ext3 defaults 0 0
```

Note : replace "/dev/sda1" by the adequate partition on your system.

- 2) Rename your current /boot partition to boot\_old (for example)
- 3) Mount the new /boot partition under each of them: "mount /boot".
- 4) Copy all your boot kernels in this partition.

Then to create the possible boot entries, copy the example file from the bootcreator.example file to /boot/bootmenu.txt and edit the file to your needs.

Then to create a boot menu file, type the following command:

```
bootcreator bootmenu.txt bootmenu
```

Finally, reset your Pegasos and press the "Esc" key when it boots.

Under OpenFirmware, set the following environment variables to automatically boot on your bootcreator's bootmenu:

```
setenv boot-file /boot/bootmenu
setenv boot-device hd:0
setenv auto-boot? true
```

Now if you reset your Pegasos again (or just type boot), you should see your bootmenu.

If you have input troubles, edit your menu again, set VERSION=0 and re-create the bootmenu file.

### 1.1.5 Using a Linux System from Another Linux System by Changing root

If you want to use a Linux system from another one without using a boot kernel you can use the chroot command.

For example if you are under Linux Debian and that you want to use your Linux Gentoo which is installed in another partition, let's say in /dev/hda4, just do the following from a terminal:

The first time:

```
mkdir /mnt/gentoo
mount /dev/hda4 /mnt/gentoo
```

Then to change root into it :

```
mount -o bind /dev /mnt/gentoo/dev
mount -o bind /proc /mnt/gentoo/proc
chroot /mnt/gentoo
```

After having typed those commands you are logged as root under your other Linux system which can be very convenient (for example if you have problems booting a fresh kernel).

### 1.1.6 NFS : Network File System and Remote Compiling

NFS makes it possible to mount directories from a server computer to a client computer as local files/folders.

NFS can be used to directly transfer files from a computer to another, or to install a Linux distribution, but combined with the "chroot" command explained above, it can also be used to run your applications and compilations on a less powerful "server" computer from a faster "client" computer.

Personally I use a Pegasos as NFS client with an Efika as a NFS server. This has the advantage to be PowerPC powered, so no cross compilation is needed in this case.

Note that to "chroot", on the Efika you must not use NFS on the target distribution on server side or you won't have access to the /dev and /proc directories on the NFS client. Instead, the NFS server must be run from another distribution under another partition. For example, I've used the NFS server under a Linux Debian distribution on my Efika, and the NFS client under a Linux Gentoo distribution on my Pegasos. The target distribution which I was doing "chroot" in, was a Linux Gentoo distribution installed on the Efika. This was a way to optimise compilation time. This is an example of what can be done with NFS. It's up to you to imagine other ways to use it.

Also note that if you just want to export an NFS partition from one machine (without the need to "chroot" on client side) then any client machine can import it without needing a second distribution to run on the server side.



There are many tutorials about NFS available on Internet. Here are those which helped me the most :

- Installation on Linux Gentoo :

[http://gentoo-wiki.com/HOWTO\\_Share\\_Directories\\_via\\_NFS](http://gentoo-wiki.com/HOWTO_Share_Directories_via_NFS)

- NFS Easy way (Ubuntu)

[http://doc.gwos.org/index.php/NFS\\_Easy\\_Way](http://doc.gwos.org/index.php/NFS_Easy_Way)

- Linux mechanics – NFS (French page – Use Google for translation)

<http://linuxmecanic.fr/linux.net/fr/nfs.html>

<http://translate.google.com/translate?u=http%3A%2F%2Flinuxmecanic.fr%2Ffr%2Fnfs.html&langpair=fr%7Cen&hl=fr&ie=UTF8>

In fact, the installation steps are the same on each distribution.

On the server :

- Add NFS server support to your kernel
- Install the NFS server packages
- Edit the /etc/export file on the server to configure the directories that can be mounted and the access rights. Then export it ("exportfs -a").
- Configure "portmap"
- Run services and daemons. Check it with ("rpcinfo -p")
- Configure host.deny and host.allow files

On client :

- Add NFS server support to your kernel
- Install the NFS server packages
- Mount the remote directory by hand via the /etc/fstab file

If it does not work on the first try, don't panic, check rights, ports and launched services.

## 1.2 Installation of Linux Debian on Pegasos I and II

Debian is a complete Linux distribution intended for the users who makes it possible to have an environment having all kinds of software. This distribution works both on Pegasos I and Pegasos II. However, if you have Pegasos II I advise you rather use Linux Ubuntu (based on Debian), Linux Suse or Linux YDL.

### 1.2.1 Installation types

Before installing Linux Debian, you should ensure that you have sufficient place on your hard disk (at least 3Gb). It is highly recommended to install Linux on a second disk if your first disk is for example dedicated to MorphOS.

There are several ways of installing Linux Debian on Pegasos, but in all the cases one needs an Internet connection.

- Netinstall : boot on a minimal kernel and installation by Internet.
- Full installation(22 CDs !)
- Netinstall (partial) by a minimal CD of installation (recommended method on a Pegasos I)
- boot and installation starting from a CD, completion of the installation after a reboot by Internet.

**Warning : there are specific kernels for Pegasos I which are different from the Pegasos II specific kernels.**

Pegasos II (and I?) : ISO Image of a Netinstall Linux PowerPC 4.0.r0 CD (April 2007 – 233Mo) :

[http://cdimage.debian.org/debian-cd/4.0\\_r0/powerpc/iso-cd/debian-40r0-powerpc-netinst.iso](http://cdimage.debian.org/debian-cd/4.0_r0/powerpc/iso-cd/debian-40r0-powerpc-netinst.iso)

Download the ISO image corresponding to your Pegasos's type, and burn it on a CDROM.

For the 22CDs installation (not recommended, unless you haven't a high speed access to Internet from your Pegasos) :

[http://cdimage.debian.org/debian-cd/4.0\\_r0/powerpc/iso-cd/](http://cdimage.debian.org/debian-cd/4.0_r0/powerpc/iso-cd/)

### 1.2.2 Installation

#### 1.2.2.1 First step

Insert the Debian CD in your CD drive and start again your Pegasos. Enter into OpenFirmware (press "ESC", if you are in auto-boot) and type the following command :

```
boot cd install/pegasos
```

A small menu then appears, in which you can choose between :

- 1) install
- 2) expert
- 3) rescue
- 4) return to OF prompt

Choose "1" to start install.

The installation then starts. You will have to answer a certain number of questions in the following order :

Choose language : "English"; Country: "UK"; Keyboard: ""

Network interface : "eth2 : VIA Technologies, VT6102 [Rhine II] "

The let DHCP decide which IP address to give to your computer, then continue.

Computer name : enter what you want, for example "pegasos"

Domain name : leave this field empty

Partition : select manual modification and choose the disk where you prepared the partitions for Linux. For example if you created two partitions named DH4 and DH5 on disk 1 (hdb):

DH7, "EXT3" filesystem ; mount point "/"; boot : enabled ; end of partition setup

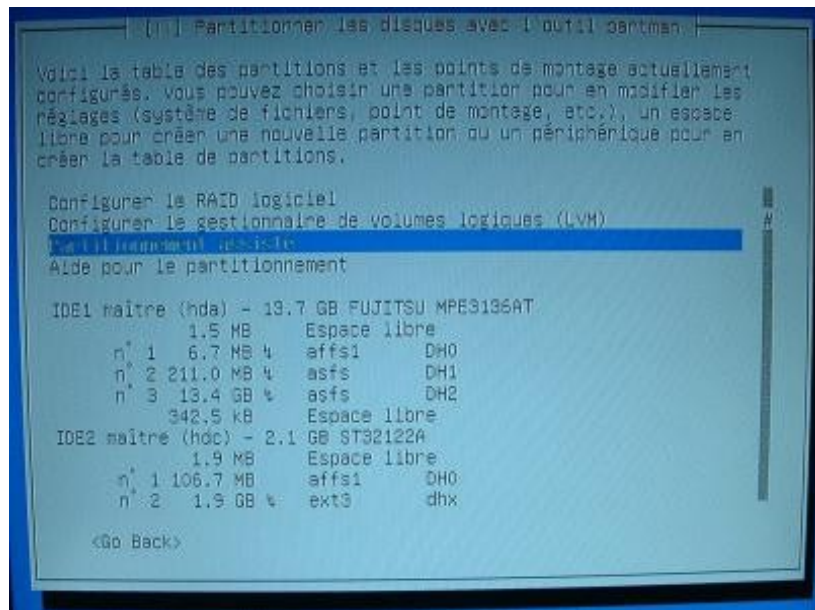
DH3, "swap" filesystem ; end of partition setup

UTC clock : yes

Then type your login and username

Choose a mirror for install, for example [ftp.fr.debian.org](http://ftp.fr.debian.org)

For software selection, leave the one enabled by default



The installation then really starts.

Let your computer work until it gives back control to you. At the end of the installation, the Debian installation software will indicate the commands to be used to start Linux from the firmware. Note this important command scrupulously !

Example:

```
setenv boot-device hd0:6
setenv boot-file boot/vmlinuz-2.6.18-4-powerpc root=/dev/hdb7
setenv auto-boot-timeout 5000
setenv auto-boot? true
```

Otherwise, just note the kernel name (boot-file) and the startup partition (boot-device). Then, to start directly from Open-firmware, you will only have to use a command like « boot <boot-device> <boot-file> ». For example, with the above information that would be :

```
boot hd0:6 boot/vmlinuz-2.6.18-4-powerpc root=/dev/hdb7
```

Let Linux start, then type your login/password like set before during install.

Install is now finished !

## 1.2.3 Tricks and hints

### 1.2.3.1 Installing / Uninstalling softwares

*Installing / Uninstalling a software (or package)*

To install a package easily : open a console and go to root mode. Type "apt-get install your\_package".

Examples : "`apt-get install tuxracer`", "`apt-get install kdevelop`"

To remove a package : "`apt-get remove the_package`".

#### *Installing a software from its sources*

Unpack sources archive : `tar -xzvf I_archive` (if it is a tar.gz) or `tar -xjvf I_archive` (if it is a "tar.bz2"). Go in the directory where the sources of the program are :

`"cd the_archive_directory"`.

Go to root mode and type :

`« ./configure && make && make install ».`

The program should now be installed.

#### *Installing the software from Internet*

To install the packages directly from a Debian mirror (for fast network connections).

Install netselect : `apt-get install netselect`, then `apt-get install netselect-apt`.

Let netselect work : `netselect-apt unstable`. Netselect creates a file named "sources.list", copy this file into `/etc/apt/`.

Update the packages database : `apt-get update`.

Update your system with the last packages : `apt-get upgrade`.

Now, you have an up to date system and any package installed with apt-get will be downloaded from the Web !

## 1.3 Installation of Linux Ubuntu on Pegasos II

Ubuntu is a version of Linux targeting "desktop" and based on the Debian distribution.

The "Tricks and hints" Debian Linux chapter can also be applied to Linux Ubuntu.

Since Ubuntu 6.06 "dapper", Pegasos II support is not available any more on the usual CDs (Desktop and alternate). For Ubuntu 6.10 "Edgy", an install DVD containing a bootable kernel for Pegasos has been released a few weeks after the distribution release. For "Feisty" this is not (yet) the case and the situation is getting complicated because the PowerPC versions are no more developed or maintained par official Ubuntu developers but by an independent team which does not have the same means.

We are now going to see in that chapter how to install the Ubuntu 7.04 version on Pegasos. Previous versions install (Dapper, Edgy and Breezy are detailed in addendum section).

### 1.3.1 Pre-requisite

Before starting install, we need to download a few resources.

The Ubuntu Feisty "alternate" installation CD for PowerPC :

<http://cdimage.ubuntu.com/ports/releases/7.04/release/>

Beware of choosing the right version :

<http://cdimage.ubuntu.com/ports/releases/7.04/release/ubuntu-7.04-alternate-powerpc.iso>

The kernel that will make it possible to start-up installation, generated by Peter "czp" Czanik :

[http://genesi.hu/ubuntu/feisty\\_inst](http://genesi.hu/ubuntu/feisty_inst)

The kernel that wil make it possible to boot Ubuntu Feisty once install is done :

[http://genesi.hu/ubuntu/feisty\\_boot](http://genesi.hu/ubuntu/feisty_boot)

The first step consists of burning the ISO image on a CD, then to copy the kernels "feisty\_inst" and "feisty\_boot" on a bootable partition of your Pegasos (affs or ext2/ext3).

### 1.3.2 Start-up and install

Switch on your Pegasos and insert the Feisty's installation CD in your CD drive. On OpenFirmware prompt, type a command line to start the "feisty\_inst" kernel.

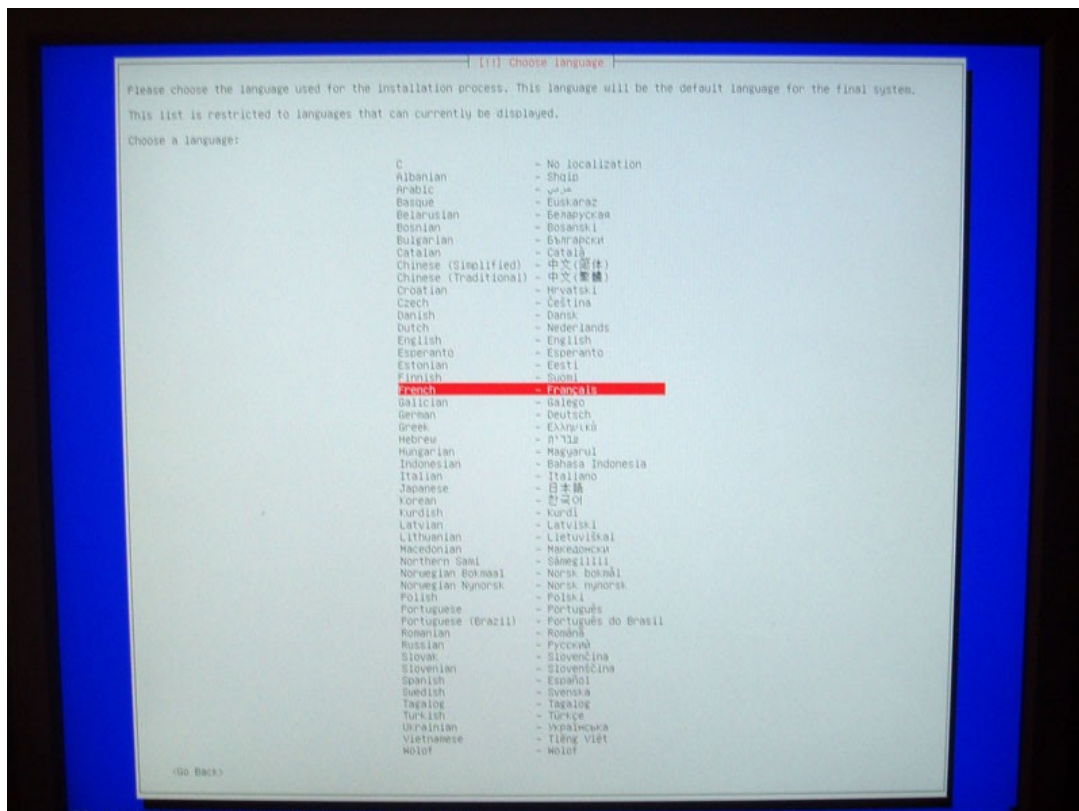
I've personally copied the kernel on the first partition of an IDE disk, which is the slave of a IDE primary controller, and this in the /boot directory ; which gives us:

```
boot /pci/ide/disk@0,1:0 boot/feisty_inst
```

Adapt the command line depending of your configuration. In case of doubt, consult the OpenFirmware manual for IDE peripheral naming conventions.



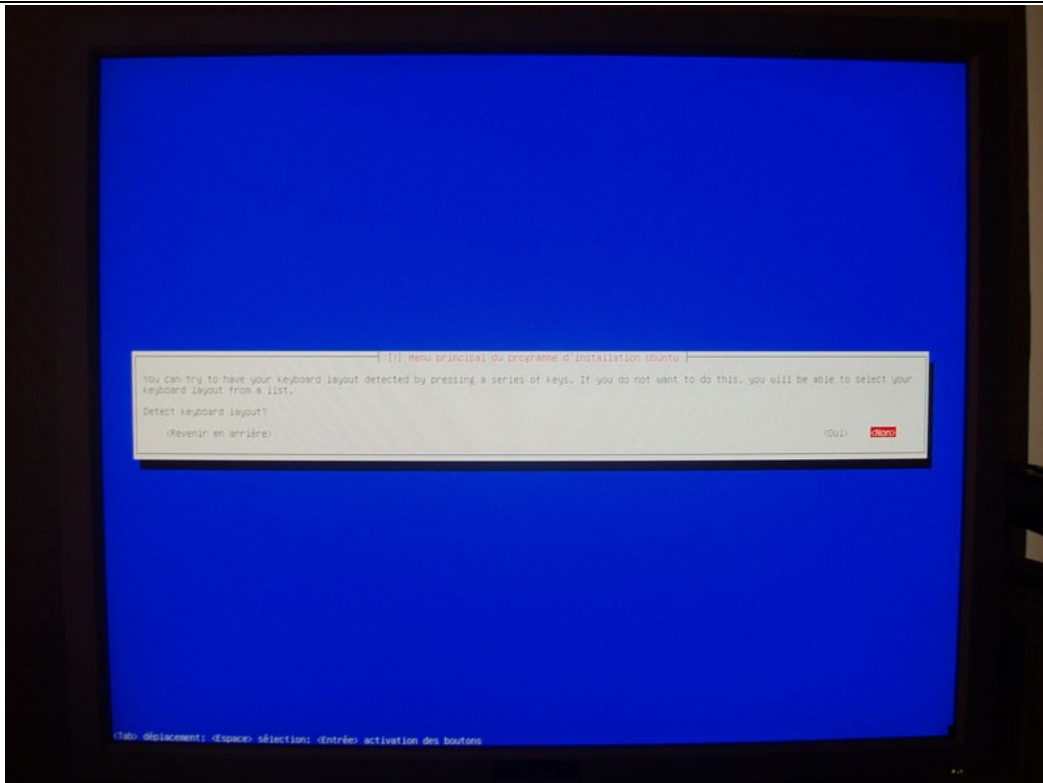
The Pegasos is now going to start in a few seconds and display the classic Alternate installer :



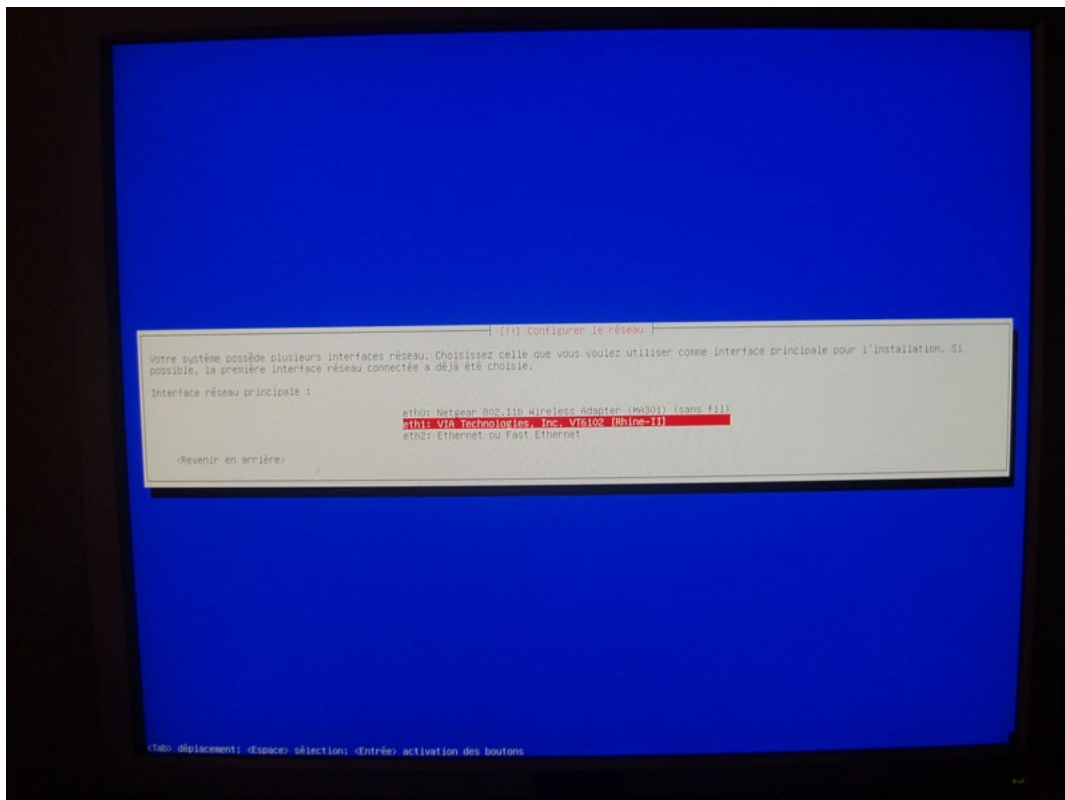
For the language, choose whatever you want "English", "French"...

Then answer all the questions, and when the installer asks you if it must automatically detect your keyboard, answer "no".





For the network interface, choose "VIA technologies, INC VT6102" if you use the "Ethernet 10/100" integrated port of the Pegasos.



The installer is now going to detect your disks and will give you the option to assist you for partitioning.

Do whatever you may like (please refer to above section for partitioning details).

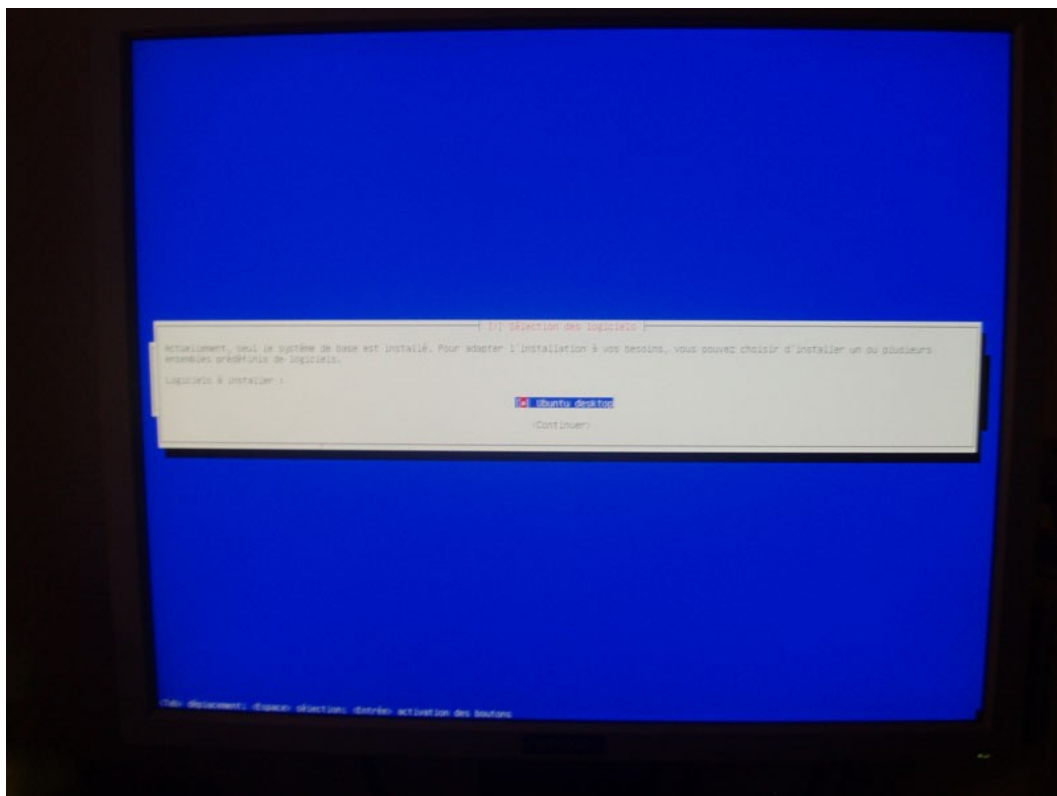
Once the installer asks if the system clock must be set to universal time (UTC), answer "no" if you have other operating systems installed on your Pegasos.

The next step consists of filling the fields with your name, your user-name and your password.

The real install can now start...



After some time, the installer will give you the option to install "Ubuntu Desktop" to have a the Gnome desktop manager and Xorg installed. Without that, only the basic system is installed. So check the "Ubuntu Desktop" install option and continue.



The installer then detects that it's running on a Pegasos and then gives the command lines needed to be able to boot from OpenFirmware once install is done. Do not take it in account, because we won't boot on the Ubuntu kernel installed right now.



Continue.

The install now ends. The CD is ejected and the system ready to be booted.

Re-boot your Pegasos.

### 1.3.3 Post-installation steps needed on a Pegasos

Our system has rebooted, and theoretically, our new Ubuntu Feisty system should be installed and working. But this is not completely true : the default installed kernel by Ubuntu installer won't boot under OpenFirmware. So we have to use the "feisty\_boot" generated by Peter Czanik.

We will need to call this kernel under OpenFirmware and we will also need to specify the partition which contains our Ubuntu Feisty install. In my case, I've copied "feisty\_boot" directly in the /boot directory of my Ubuntu Feisty partition by mounting it under another system (Debian), to be able to do a copy from it. You can also copy "feisty\_boot" on an alternate partition of your Ubuntu installation, under the condition that this partition is bootable under OpenFirmware.

I've used again the first IDE partition of the primary slave IDE controller. This is also the one where I've installed Ubuntu and copied "feisty\_boot" in the /boot directory. Which gives us the following boot command :

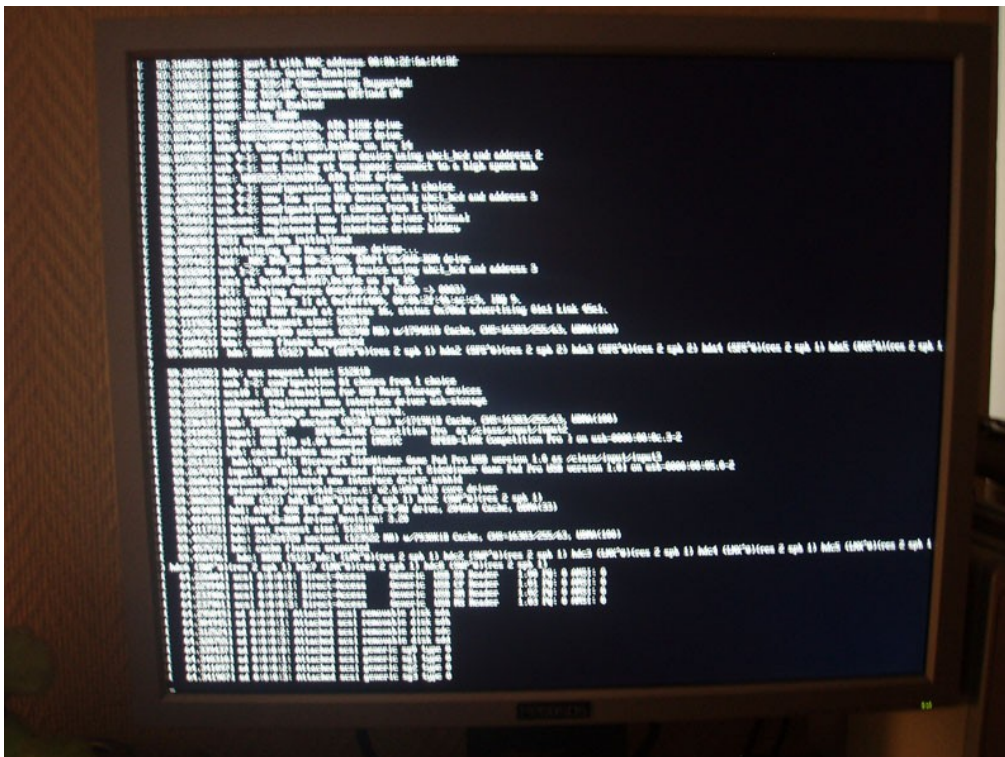
```
boot /pci/ide/disk@0,1:0 boot/feisty_boot root=/dev/hdb1
```

**Note :** /pci/ide/disk@0,1:0 and hdb1 are the same partition (slave on IDE1).

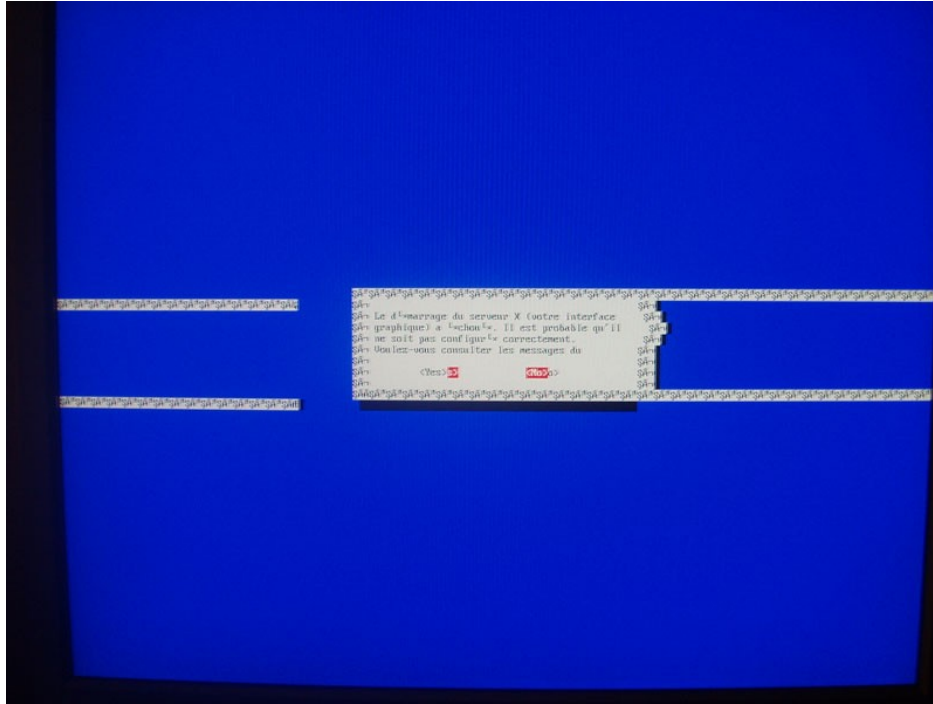
In case of doubt, consult the OpenFirmware manual for IDE peripheral naming conventions.

### 1.3.4 First boot and Xorg update

We now start our fresh Ubuntu Feisty install for the first time :



Unfortunately, the X server won't start and Ubuntu will inform you that there is an error about this.



Do not care of error log, and choose "no" when asked for it.

Again, we are under the command line. Log-in with the username and password selected during install.

We are now going to edit the "sources.list" file which contains the software repositories and updates available for Ubuntu Feisty. We are going to add a new repository to it, which contains a functional Xorg for Pegasos.

On command line, type the following command :

```
sudo nano -w /etc/apt/sources.list
```

Then, type your password when asked. The nano text-editor now opens and displays the sources.list content.

Add the following line (at the bottom) :

```
deb http://www.pegasos.org/ubuntu/feisty binary/
```

Now, save the file (Hit "CTRL and "X" simultaneously, then "Enter" to save the changes).

Once back on command line, type the following command (don't miss the "-" at the end of command line).

```
wget -O- http://kozz.org/files/keys/kozz-pubkey.gpg | sudo apt-key add -
```

This will download and install a public key required to access to the new repository we've just added to sources.list.

Now, update your local repositories database:

```
sudo apt-get update
```

Then update your Ubuntu system:

```
sudo apt-get dist-upgrade
```

Xorg should now be working. To be sure, start a X server with the following commands :

```
sudo /etc/init.d/gdm stop
```

```
sudo /etc/init.d/gdm start
```

And admire the Ubuntu 7.04 login screen !

## 1.4 Installation of Linux Suse on Pegasos II

Linux Suse 10.x is a rather complete distribution whose installation remains relatively easy and where it is possible to choose its window manager during the installation (KDE or Gnome).

In this chapter, we will detail the 10.1 version installation. The 10.3 alpha3 is now available and complementary information can be found at the following URL :

<http://www.powerdeveloper.org/forums/viewtopic.php?p=7972#7972>

Now, let's start 10.1 version installation :

### 1.4.1 Preparing

#### 1.4.1.1 Machine

This distribution only works today on Pegasos II.

#### 1.4.1.2 Graphic card

I do not know if this distribution works on graphic cards other than ATI RADEON.

#### 1.4.1.3 Memory

For a normal CD or DVD installation 256MB of RAM is enough. Installing 'factory' requires at least 384MB of RAM, as data for packages on all platforms is read by the package manager (20000+ packages instead of about 8000).

#### 1.4.1.4 Source and installation media

There are five CDs to download. Information is available at <http://en.opensuse.org/Download>. Actually, there is also a 6th CD, which can be used for network installation or rescue. Here are the possibilities:

- Download the 5 CD images. Burn them all on CDs and use them. This takes more time, but requires no additional machine.
- Download the 6th image as well, which is called 'mini'. You can use it to install from a partition or from over a network.

There are a lot of alternatives to burning the five CDs:

- Download only the 'mini' and use an installation source available over the Internet. The constantly developed version is called 'factory' and is a moving target. For the final 10.1 release it will be frozen. It has a lot more packages, than the CD release.
- Burn only the 'mini', and loop mount the images on another machine in a directory structure like this: CD1, CD2, ..., CD5 (`mount -o loop,ro SUSE...CD1.iso CD1`), and make it available through ftp, http, or nfs. (See the adminguide for more details)
- if you already have Linux and partitions on your Pegasos, you can loop mount the CD images, and copy the content to a directory structure, like already mentioned. From CD1/suseboot copy inst32 to an OpenFirmware readable partition (primary ext2 or ext3 partition) and boot it to start installation
- use of downloaded iso images from a partition should also be possible
- inst32 can also be booted from the network using tftp and can be combined with any network or partition based installation.

In a more direct way, use for example <ftp://suse.inode.at/> which offers a good download rate. Go then in the directory "[/opensuse/distribution/SL-10.1/iso](http://opensuse.org/distribution/SL-10.1/iso)" and get the following CDs ISO images :

- SUSE-Linux-10.1-GM-ppc-CD1.iso
- SUSE-Linux-10.1-GM-ppc-CD2.iso
- SUSE-Linux-10.1-GM-ppc-CD3.iso
- SUSE-Linux-10.1-GM-ppc-CD4.iso
- SUSE-Linux-10.1-GM-ppc-CD5.iso
- SUSE-Linux-10.1-GM-ppc-mini.iso

Note: In case of burning size problem of the 3<sup>rd</sup> CD, it is possible to reduce its size by removing some useless linguistic files. For that: Uncompress the ISO file.

In your burning software, select all the uncompressed files then remove useless files like those of "suse/noarch/myspell - \*\*" by keeping for examples only those of your favorite languages for example "american" and "british".

Then, burn the CD like a data CD (standard ISO 1) with the name "SUSE1000.003".

#### 1.4.1.5 Partitioning

Partitioning a hard drive for SUSE Linux 10.1 on Pegasos PPC needs special considerations. While Pegasos officially supports only the Amiga partition table format, YaST (the installation and configuration tool of SUSE) supports only DOS partition tables. In practice both have partial support for the other format.

For booting the Pegasos can use ext2 or ext3 partitions (or the Amiga FFS or SFS, if you have that one). Usually there is a small ext2 partition on /dev/hda1 which holds a boot menu and kernels. But it's possible to boot from almost everywhere on the hard drive.

First, let's see the Pegasos point of view about partition tables. Pegasos has its roots in the Amiga community. This means, that its firmware officially supports only the Amiga partition table. When using it, the firmware can boot the Linux kernel from any (ext2 or ext3) partition. There is some unofficial support for the DOS partition table in the firmware also. It can't read extended partitions, so only primary ones can be used for booting.

YaST has official support only for DOS partition tables on hard drives. What it means for the Pegasos, that only DOS partition tables can be modified by YaST. As the YaST partitioner is based on parted, which has extensive support for a lot of different partition tables, it can read Amiga also. YaST can format and assign a mount point to these partitions, so pre existing partitions can be used during installation.

This leaves us three possibilities:

- ODW machines (computers built around the Pegasos mainboard by Genesi) come with a couple of preinstalled Linuxes on a harddrive using the Amiga partition table. The first partition is prepared for the boot menu and kernels, the second is a swap partition, and the rest are preinstalled Linuxes. If you don't need one of these, you can install SUSE Linux over that partition, without the need to modify the partition table. You can select and format the partition in YaST.
- If you need to change partitions but also need to keep the Amiga partition table (for example to be able to boot MorphOS), then YaST is not much help at the beginning. One needs to use a rescue CD, like the one included on SUSE boot CDs or Mupper, which was created especially for Pegasos users in mind. You can use parted for partitioning, documentation is available at <http://www.gnu.org/software/parted/manual/> . There is also a really short command summary in the next chapter ( <http://en.opensuse.org/PegasosQuickStart#parted> ). It's also possible to use the install CD, just use the 'start\_shell' boot parameter. Just after everything is loaded, this options provides a shell. 'parted' is available there. **The machine must be rebooted after changing partitions.**

- If you use the machine for SUSE exclusively, you can safely use DOS partitioning on Pegasos. All you need to consider, is that there is a primary partition, which OpenFirmware can read (ext2 or ext3).

In any case it's wise to leave/let the first partition as a small ext2/ext3 partition and use 'menu'. This is the default setting in OpenFirmware. If it's already prepared (like an ODW machine), do not format it.

#### 1.4.1.6 Information about "parted" tool usage

See "Global information" chapter.

#### 1.4.1.7 Booting the installer

Once you have your installation media prepared, it's time to get started with installation. First, you need to escape to the OpenFirmware prompt. This can be done by hitting 'Esc' during the countdown, or if you have a boot menu, choose the appropriate entry. The next code sample shows, how to boot the installation system from a CD. It is the same for CD1 or the miniboot. The `linemode=0` parameter is there to provide an easy to use curses based menu interface to installation settings, instead of a line based, which is ideal for serial console environments.

The command to boot from CD is:

```
boot cd suseboot/inst32 linemode=0
```

The command to boot inst32 copied to the root of the third ext3 partition is:

```
boot hd:2 inst32 linemode=0
```

What happens next, depends on how installation of SUSE Linux 10.1 was started:

If you burned the five CDs, and booted from CD1, then you can sit back and relax for about two minutes, until you reach a graphical screen asking you which language to use during installation. You can safely skip the following chapter.

If there are troubles getting into the graphical installation screen (and you're welcomed with a text based one), you may want to try the following variant when booting the system:

```
boot cd suseboot/inst32 video=radeonfb:1024x768@75 linemode=0
```

### 1.4.2 Setting up the installation environment

If you don't have CD1 in your drive, you will be first presented with some questions before reaching the same stage of the installation. These are required to find the installation sources.

In the later case, first a blue screen appears, still in text mode. It will stay like this, until all information is gathered, and the installation software can be loaded.

The very first request is to put CD1 in your drive. This is not an error, the default installation mode is to use your CD/DVD drive. It is possible to choose other installation sources as boot parameters, but it's more convenient (less typing, and less documentation reading) to choose installation sources using a curses based menu system. If you don't have the CD in the drive you need to use the right arrow or tab to get to 'back' and hit 'enter' to continue.

Here the language of installation can be selected. All languages use their own character sets, some more exotic ones can not be read at all. For these English or an other language with 'Latin' type characters is suggested. This can be changed later on the graphical installation screen. Up/down arrows can be used to choose another language, and 'enter' to use that and continue with installation. The main menu of linuxrc (the tool used to set up the installation environment) is shown.

If network installation is to be used, the first menu to visit is 'Kernel Modules (Hardware Drivers)'. The menu works with up and down arrows and Enter/Escape. First enter the 'Kernel Modules' menu, and next choose 'Load ppc Modules'.

There are two ethernet ports on the Pegasos mother board. The left one is a gigabit ethernet interface, which requires the **mv643xx\_eth** driver to be loaded. The right one is a 10/100 ethernet interface, the **via-rhine** driver is needed by this one. Just hit 'Enter' to load them and follow on screen instructions. You don't need any other modules to be able to install SUSE Linux on an ODW over the network, so escape back to the main menu.

To get started with installation, choose 'Start Installation or System'. This brings to another menu. The first one is needed to start installation, but the other two are also important:

- 'Start Installation or Update' starts a new installation
- 'Boot installed system' allows booting an already installed system. It's needed in the second stage of installation, when the computer first reboots, as there is no boot loader or bootable kernel prepared. It has to be done by hand, when the second stage of installation is ready.
- 'Start Rescue System' starts a mini Linux with lot's of handy tools. Needed for partitioning as well, if you need to change an Amiga partition table (the default on ODW)

As installation is just started, the first menu entry is to be chosen. This shows another menu:

- CD-ROM: does not have much use here, except when drivers need to be loaded before CD based installation can go on.
- Network: FTP/HTTP/NFS/SMB
- Hard Disk: from a partition

If you choose 'Hard Disk', you are presented with a list of partitions on the system. First, you have to choose the partition, where installation sources are to be found, and then enter the path relative to that partition. For example, if you copied the installation CDs to your /home partition in directory 'suse101', and /home is /dev/hda5, then choose /dev/hda5 as partition and /suse101/CD1/ as source directory. The rest of the directories will be used automatically.

From the network install, FTP has the most questions. If you forgot to load the kernel module for networking, you will be presented with a big, red warning message. Other way, you are asked, if you want to use DHCP for network configuration. If not, be prepared to answer some questions about your network. The first is: the IP address of your machine, netmask follows (with defaults for a class C network filled in), IP of the gateway machine, IP of the name server, and finally IP of the ftp server. You can use the name of ftp server here, if there is a proper DNS entry for that machine. Then choose, if you need a username to reach the ftp site, or a HTTP proxy. At the end, you have to specify a source directory, including CD1 in the name, if you set up the installation source yourself from CDs, or 'inst-source', if you use the 'factory' distribution.

In a few seconds or minutes, depending on your network speed, you will get to the same graphical screen mentioned already at the beginning of the document.

### **1.4.3 YaST startup and starting of the graphical installation**

#### **1.4.3.1 Global settings**

The first menu appearing on the graphical screen is the 'language' menu. It looks much better as the previously mentioned 'language' menu, as this one has correctly displayed characters from many different languages. What you choose here, influences a lot of things. Not only the language of the installation program, YaST, but if a software package has language extensions, like KDE or OpenOffice.org, it will be automatically selected during software installation based on this choice. So, choose your language, and go to the next screen. You can check the integrity of your installation CDs, if you use them for installation. Other way, the next screen is about accepting the 'License Agreement' of SUSE Linux. (During beta testing, an extra window reminds, that it's a beta release, and there is no support at all. To continue, one has to accept both.)

Hardware detection comes next. This also means, that drivers are loaded, and if you don't use CD1 for installation, YaST asks for your permission in some cases. If you have a Hard Drive



with an Amiga partition table, you are warned, that the installer can't modify it. You can still use existing partitions.

It is possible, that you get another few kernel module related questions, and then you are presented with the 'Installation mode' menu. On first installation you need to use the default setting: 'New Installation'. 'Update' helps to update from a previous version of SUSE Linux (but it's not possible with the current beta (beta4)...). 'Other' has some important possibilities, well hidden. 'Boot Installed System' is need in the second stage of installation, if you use CD1 for booting. There you can choose which partition to boot.

#### **1.4.3.2 Starting a new installation**

As this is the first stage of a new SUSE Linux installation, 'New installation' is to be used. It must be mentioned here: no data is written to disc, until at the end of installation settings, you click the 'Accept' button and answer 'Yes' to the question. Until that you are free to abort the installation process, and no change is made to the system.

The first screen will allow to set the time and/or timezone. The next question is a bit more difficult: choosing a desktop environment. KDE was much better supported by SUSE for a long time, the difference is getting smaller now. Use 'other' if you don't want any of these large desktop environments, or text mode only.

If you are installing on a completely empty hard drive or have a DOS partition table, there will be some useful information on the 'Installation Settings' overview page. YaST has a suggestion for partitions, and this way it can also suggest software. But for the rest of us, who use the native Amiga partition table format on Pegasos, there is only a big red warning about partitioning and no real software suggestion.

In either case, to go on with installation, one has to choose the 'Expert' page. This has even more red warning messages. Installation settings can be changed by clicking on bold underlined texts looking like links on a web page.

#### **1.4.3.3 Partitioning**

The first thing is to fix partitioning. If you are using a DOS partition table, please use the installation manuals, the process is well described there. Just make sure, that you have at least one primary partition formatted to ext2 or ext3 file system.

Those who use an Amiga partition table need to choose 'Custom partitioning (for experts)'. It provides with a list of partitions, which you previously prepared, or came pre-installed with your ODW. Select the partition from the list, which you want to format and/or attach to a mount point, and click 'edit' to setup the partition. You need to have at least a root (/) and a swap partition. It's also advised to have a small ext2/ext3 partition at the beginning of your first hard drive to hold booting related files. Be aware, that SUSE Linux uses ReiserFS by default, so creating/using this small partition is a must, if you want an efficient filesystem for the rest of your partitions. More detailed description of menus is available in the installation manuals. At the end, click 'Finish'. Remember, your partitions are crated and formatted only after all settings are configured.

When you click 'Finish', you are presented with a warning message. Pegasos PPC does not need the extra boot partition mentioned here, so you can safely answer 'No' to this question.

#### **1.4.3.4 Software**

Software selection works the same way, as on any other platforms. Click on « change » then on « Packages sekection ». Then chose additional packqges that you want to add. For more information about that, have a look at the official manuals.

#### **1.4.3.5 Booting**

There is no big red warning message here - yet. So, let's change it, as Pegasos does not need a boot loader to be installed. Click on 'Bootting' and choose 'Boot loader Installation' on the top. Set the type of boot loader to 'Do Not Install Any Boot Loader' and choose continue and later 'Finish'.

#### **1.4.3.6 Committing installation**

Up until now, the hard drive was not modified, only information about installation was collected. When the 'Accept' button is clicked, there is a last chance to modify settings or abort the installation. When 'Install' is clicked, the installation is carried out according to settings made in the previous menus.

At first, partitions are formatted and mounted and next software packages are installed. It takes some time. Estimate of remaining time can be tracked on screen. After the content of CD1 is installed, the computer is rebooted. Unlike on x86 and other PPC computers, Pegasos can not boot from the hard drive at this stage. What comes next, depends on what we use for installation.

If CD1 was used, then follow previous instructions up until the 'Installation Mode' menu is reached. Click 'Other' from the bottom right corner of the screen and choose 'Boot installed system'. After clicking 'OK', the hard drive is searched for bootable Linux partitions. Choose the appropriate partition and click 'Boot'.

In any other cases follow instructions from section 'Setting up an installation environment' up until 'Boot installed system', and choose the partition to be booted. If you created many partitions, it's the root partition, marked with '/'.  
The remaining CDs are asked in order to finish up package installation.

#### **1.4.3.7 Configuring the system before first boot**

You then will answer other questions:

Choose the root password.

Network setup : automatic

Test Internet connexion: Yes

In case of failure : ignore

In case of success : the installation software will then propose online updates to you, do "Yes" (if you are not too in a hurry). Let then the machine work (download of modules from Internet is sometimes a very long operation...). Then click "Finish". YasT might be updated too. In this case validate the message of YasT restarting without any fear.

User authentication method : « local (/etc/passwd) ».

Select then a complete name, a user name and a password. For security, uncheck "automatic login".

Release notes : next

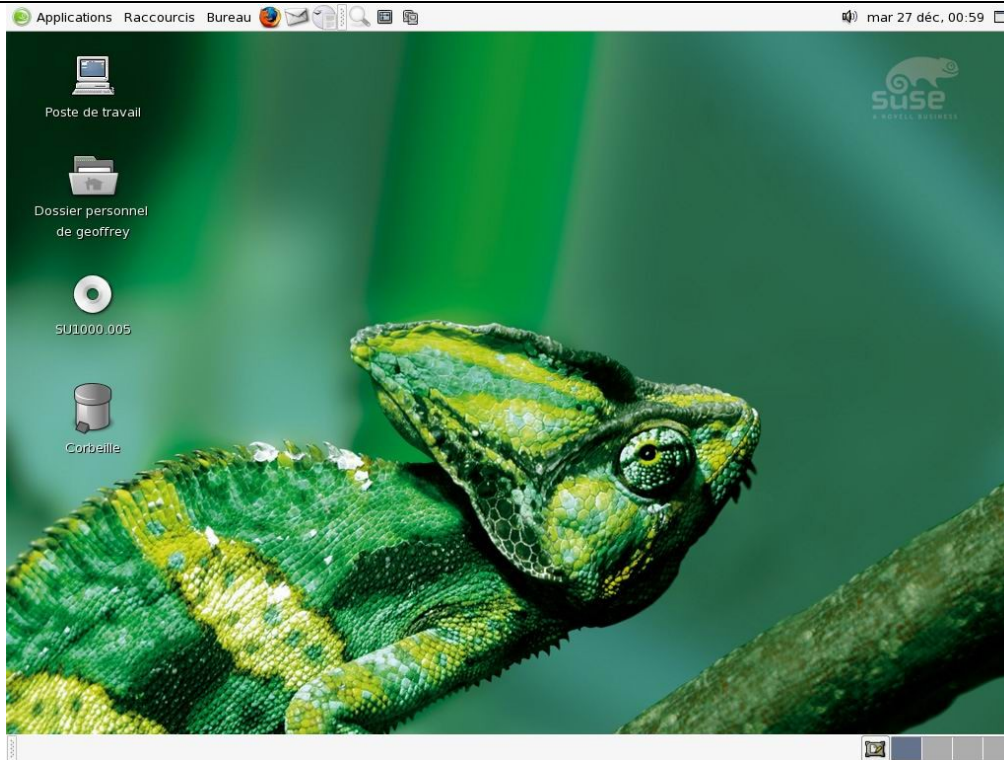
Hardware configuration : use the default detected configuration.

End of Installation.

Start the YaST control center : leave unchecked. Finish installation.

The installation software then ends, and the graphic interface is then launched after a short switch-over to a console mode.





#### 1.4.4 Making Linux SUSE bootable from hard disk

Booting Pegasos from a CD is not the most convenient thing possible, so it's highly recommended to make SUSE Linux boot from the hard drive. Pegasos does not rely on a boot manager. Instead of that, OpenFirmware loads the kernel itself. The kernel, as it's installed in directory /boot is not directly loadable by OpenFirmware.

To create an OF bootable kernel and put it where OF can read it, the user needs to be logged in as root (or use 'su -' to become super user). Use the following command to create a bootable kernel:

```
mkzimage -vmlinux /boot/vmlinux -initrd /boot/initrd -output /tmp/peg10lsuse
```

The last file name can be anything, just make sure, that you don't overwrite anything. The next step is to copy this file to an ext2/ext3 partition, where OF can read it. Check, if it's mounted by examining the output of the 'mount' command. If not, mount it. To mount /dev/hda1 to /mnt use the following command:

```
mount /dev/hda1 /mnt
```

You can now copy the previously created bootable kernel to this location. Make sure, that you don't accidentally overwrite anything by listing the directory first, and then copy the file:

```
ls /mnt
cp /tmp/peg10lsuse /mnt
```

**Warning:** these steps need to be repeated in case of a kernel update! Otherwise still the old kernel will be booted, but /lib/modules will include only kernel modules for the new kernel, which can lead to unexpected results.

You can now reboot your machine and use the new kernel for booting. If the kernel resides on /dev/hda1 and the root partition of SUSE Linux is /dev/hda6, you can use the following command in OpenFirmware to boot:

```
boot hd:0 peg10lsuse root=/dev/hda6
```

The number after 'hd:' is one less, than the number in the Linux device name (unless it's a late beta firmware and set to be chrp compliant). More information is available in the OpenFirmware manuals at: <http://www.pegasosppc.com/support.php>

Lastly, to avoid typing the command on each boot on your Pegasos, you can use Bootcreator. For the example above, it will be necessary to add the following lines to your configuration file:

```
[SECTION]
Linux Suse
hd0:4 /boot/pegasos_file
```

### 1.4.5 Troubleshooting

#### *PS/2 mouse not moving under XWindows*

This problem was occurring on Linux SUSE 10.0 when the kernel was updated during the installation. It should normally be corrected now.

- To resolve it for the current session:

Use Alt+ctrl+F1 to go to console mode and log as "root". Then enter the following commands:

```
modprobe psmouse
```

to solve it definitively: Edit the file "/etc/sysconfig/kernel", for example with "pico/etc/sysconfig/kernel". In the field MODULES\_LOADED\_ON\_BOOT, add "psmouse". Save, quit, then restart...

### 1.4.6 Update from a previous version

For an update from a previous version (for example from 10.0 to 10.1) you must use the mini-CD and factory installation. Restart this tutorial, then choose "Update" instead of "New install". Then ignore the "Start a new install" and "Partitionning" sections.

## 1.5 Media Center with Linux Gentoo on Pegasos II

### 1.5.1 Objective

In this tutorial, we will transform our Pegasos into a “Media Center” as was done for “the Genesi’s HMCS”, with a Pegasos II mother board, Linux Gentoo and the MythTV software.

This media center will make it possible to watch the TV on your Pegasos, to consult the TV schedules live and to record the programs using “time shifting”, to watch videos, images or to listen to music all of it controlled by a remote control. You will also be able to consult the latest news or the weather in your city from your sofa.

For that we will thus learn how to install and configure Linux Gentoo and all that is necessary to make MythTV and its plugins work.

Before continuing, please note that this tutorial is targeting people having a minimum experience with another Linux distribution on PPC or other. Indeed, the use of Linux Gentoo proves to be powerful but can appear rather complex at first.

Also note that the installation of Linux Gentoo is very long and will take several days. However, you will be able to stop it after a command is completed and resume it later by typing ten command lines from the install CD. The least courageous will be able to use the “EZPegTV” install CD which was created following this tutorial.

### 1.5.2 Presentation of Genesi’s HMCS



Genesi’s HMCS (Home Media and Communication System) is a multimedia system based on a Pegasos II mother board and using a Freescale G4 processor.

It is presented in a very aesthetic aluminium case, and it is equipped with a high end TV tuner card and a THX sound card.

The system does uses a Linux Gentoo distribution and uses the MythTV software among others.

The HMCS consists of :

- A Pegasos II mother board with a G4 processor (€415)
- An ATI RADEON 9250 graphic card (€35)
- A Hauppauge WinTV-PVR-500 MCE dual-TV tuner card (€175)
- A Creative THX Sound Blaster Audigy 2ZS sound card (€65)
- An HFX Mcubed aluminium case (€340)
- An IR receiver (Zapway then iMon VFD and IR (€15-90)
- Pegasos II compatible memory (512Mb ?)
- SATA hard(s) disk(s) (movie storage) and flash card (system storage)

This computer was before built on demand by Genesi for a cost of €2115.50.

It is now discontinued (mostly because of RoHS compliance manufacturing problems).

A few links :

Genesi HMCS : <http://www.pegasosppc.com/homemedia.php>

MythTV : <http://www.mythtv.org/>

Gentoo : <http://www.gentoo.org>

Mcubed : <http://www.mcubed-tech.com>

Zapway : [http://zapway.de\\*](http://zapway.de*)

IMon VFD and IR : [http://www.t-balancer.com/catalog/product\\_info.php?products\\_id=102](http://www.t-balancer.com/catalog/product_info.php?products_id=102)

### 1.5.3 Necessary hardware

We thus want to build our own multi-media system at the lowest possible price using a Pegasos II mother board.

Important : the setup of the system will take several days because you're going to build your entire system from scratch.



The Hauppauge PVR-500 is, in fact, a bi-tuner card made of two Hauppauge PVR-150. Instead of using the PVR-500, we are going to use a PVR-350 which is less expensive but better than a single PVR-150. We will use a more ordinary case, the integrated sound chip of the Pegasos II's mother board and a basic IDE hard disk.

Our media center will then consist of :

- A Pegasos II mother board with a G4 processor (€415)
- An ATI RADEON 9250 graphic card (€35)
- A Hauppauge WinTV-PVR-350 TV tuner card with remote control (€175)
- The Pegasos II's integrated sound chip (VIA VT82C686 AC97 - Apollo Super AC97 Audio)
- A Textorm SLM-17 - 250W alim - aluminium case (€70 - optional : any box will do it)
- 512Mb of Pegasos II compatible memory (€60)
- A 40Gb IDE hard disk (€30)

The overall sum : only €75 to €120 if you already have a Pegasos II or €680 to €725 for a complete multimedia system.

Of course, you can also build your own HMCS for a hardware price of approx €1050 using almost the same components as Genesis's.

Install all the hardware in your Pegasos before continuing this tutorial.

Textorm : <http://www.textorm.com/>

### 1.5.4 General information on Gentoo

Before starting installation, you must first get used to the Linux Gentoo specificities which use "Portage" as the package management core. Knowledge of Portage and mastery of the "emerge" tool are necessary to install and use Gentoo.

#### 1.5.4.1 Portage and emerge

Portage is the package distribution system under Linux Gentoo. It is expected to completely remove the need of "release" versions, by building a local database that the user can update when he wants, just by synchronising it with the Gentoo knowledge database. The main difference with other distributions come from the fact that each package is compiled before being installed.

=>Pro : the installed package are always optimized for your architecture.

=>Cons : installation is much longer than on other distributions because each package is compiled before install

The database content is located in the /usr/portage directory. On the server's side it is updated each 20 minutes. Avoid to update your database more than once a day. The applications are classified by themes and it is thus easy to explore the various directories to see which are the apps names also as the available versions (indeed by default the unstable versions are masked). In each applicative directory, a "Changelog" file makes it possible to have a short information about the available versions.

The /etc/make.conf file is the heart of the system. It make it possible among others to define the compilation flags and the supported architecture type (CFAGS, CHOST, KERNEL\_ARCH ...), the keywords used for compiling the various modules (USE flags), also as the directory that must be protected on updates (CONFIG\_PROTECT\_MASK).

In the /etc/portage directory, the user can define the package.keywords, package.unmask, and package.use files to respectively define the keywords to be enabled for a specific package, and the packages to unmask or use. In fact, with those files you will be able to use packages which are not totally validated or officially supported. This is sometime essential as we will see thereafter.

The "emerge" tool completes "portage". In fact, it is the command line of the Portage system. Here are some of the most useful commands :

« emerge -sync » : synchronizes with Gentoo's knowledge database. Must be done frequently (but not more than once a day to respect the "net-iquette")

« emerge mypackage » : compiles and installs mypackage and any package that it depends on the computer

« emerge -f mypackage » : fetch command that makes it possible to download a package before installing it. Very usefull when you have inoportune disconnections. You can for example do a « emerge -f mypackage » command and launch it several times until every package is completly downloaded without any errors (don't worry, once a package is downloaded the command is uneffective, so you won't download twice the same file). Then do a "emerge mypackage" to install (emerge will use directly the downloaded package)

« emerge -vp mypackage » : displays the version of mypackage to be installed also as a few indications with color codes :

N = new, (not yet installed) – green

S = new SLOT (side to side version) – green

U = updating, (changing version) – yellow

D = downgrade, (the best version seems to be inferior) – red

R = replacing, (re-emerge the same version – useless except in case of problem) – yellow

F = fetch restricted, (manual install, for example IBM Java) – red

f = fetch restricted, (already downloaded) – red

B = blocked, (blocked by another package) – red

A list of "USE flags" is also displayed. In red, the flags that will be used, in blue the flags that won't be used, in yellow the flags that were not used and that will be used this time.

« emerge -Na world » : displays the list of all that needs to be updated after a synchronisation. Must be done frequently (at least each time after a synchronisation).

« emerge --unmerge myallrottenpackage » : deletes myallrottenpackage from the system

« emerge --resume mypackage » : resumes the last interrupted emerge where it was. Very useful if you need to do a emerge command before continuing a longer emerge.

« emerge --skipfirst » : continues emerge ignoring the first one of the list (that may have failed).

For more information about portage and emerge, once the CD booted, in a console, type « man portage » for portage and « man emerge » for emerge.

#### 1.5.4.2 Bugzilla and Gentoo forum

It can happen sometimes that some packages do not compile or that you may have problems with some features. If it's the case, try to do a "emerge --na world" then try a recompilation of the package having a problem. If this still does not work, you can then go on the Gentoo's bugs knowledge database : (<http://bugs.gentoo.org/>).

For a fast answer, you can also ask help on the various Gentoo forums on <http://forums.gentoo.org/>.

Choose an adequate topic in consequence and post your message after having registered.

If you think that you have identified a bug, then prefer Bugzilla. You will then have a GUI into which you will be able to do a research to see if your problem hasn't been identified and resolved yet. If you don't find anything, don't hesitate to create a new entry explaining exactly what is your problem. You will quickly have an answer about what to do to try to correct your bug or on how to give more details about it to help the developers to correct it for a next release.

#### 1.5.5 Easy install with "EZPegTV" CD

Before going further, you can download an install CD containing the result of the installation described in this tutorial : this is "EZPegTV".

If you only wish to try MythTV and use already compiled binaries for a Pegasos II G4 1GHz, this will avoid you more than a week install (in computer time). But if you'd rather like a customized install, go directly to next chapter.

EZPegTV : <http://pegasos3d.free.fr/ezpegv/>

Download the various files, then rebuild the ISO by launching buildISO.sh script. Then burn the CD and insert it into your drive.

On Pegasos boot, press the 'ESC' key.

On OpenFirmware, type the following command to boot the CD :

```
boot cd boot/menu
```

Then select your video setup. If you don't know which one to choose, just press the Enter key (default choice : 1, automatic setup).

If needed, start by selecting your favorite keyboard layout, for example in France :

```
loadkeys fr-latin1
```

Then launch the install script « /mnt/cdrom/install.sh » contained on the CD for installing all the files on your hard disk. The default root password is "root".

See the HOWTOINSTALL.TXT file contained on the CD for more information in english.

To re-start a new install, press the "reset" button of your Pegasos, or type the "reboot" command.

On Pegasos boot, press the 'ESC' key.

To start from OpenFirmware, enter the following command, for a partition on /dev/hdb5 (to be adapted depending of the partition you used) :

```
boot hd0:4 pegasos video=radeonfb:1024x768@75 root=/dev/hdb5
```

If you can, add this command line to your favorite boot program (for example GRUB2 or bootcreator, see previous chapters).

After the first start, do « `emerge -sync` » to retrieve the portage database (there is no database on the install CD).

Then do « `emerge -Na world` » to update the already installed packages.

## 1.5.6 Customized install, step by step

### 1.5.6.1 Creation of a boot CD

If you have chosen the customized install, start by downloading a Gentoo's boot CD, for example on the following website : <http://ftp.club-internet.fr>, and more particularly on the following URL : <http://ftp.club-internet.fr/pub/mirrors/gentoo/releases/ppc>.

=> Either with the mini CD :

Mini CD :

<http://ftp.club-internet.fr/pub/mirrors/gentoo/releases/ppc/2006.0/ppc32/installcd/install-ppc-minimal-2006.0.iso> (54Mb)

Stage 1 (minimal system) :

<http://ftp.club-internet.fr/pub/mirrors/gentoo/releases/ppc/2006.0/ppc32/stages/stage1-ppc-2006.0.tar.bz2> (20Mb)

Stage 3 (complete system) :

<http://ftp.club-internet.fr/pub/mirrors/gentoo/releases/ppc/2006.0/ppc32/stages/stage3-ppc-2006.0.tar.bz2> (104Mb)

Note : you can choose stage1-g4 or stage3-g4 if you have a G4 to optimize the performances. Recommended for a dedicated minimal operating system (stage1).

=> Or with the Universal CD :

<http://ftp.club-internet.fr/pub/mirrors/gentoo/releases/ppc/2006.0/ppc32/installcd/install-ppc-universal-2006.0.iso> (605Mo)

Note : this CD contains all the "stages".

If you have chosen to use the mini CD with "stage1" or "stage3" file (recommended), create a new ISO file with the same directory tree then add a "stage" file at its root. Then burn it as a ISO CD.

Note : on the time while those lines are written, version 2006.1 is available. It integrates however a 2.6.17 kernel in which the drivers of the PVR tuner card are not yet available (in a working state) on PowerPC.

Note #2 : I haven't tested it by myself, but now, it seems that the ivtv v0.8.1 drivers are working with a 2.6.18.2 kernel (the firmware line must be commented in /etc/udev/rules.d/50-dev.rules to make it work).

Prefer 2006.0 version if it's still available, or use 2006.1 and after first install step, get 2.6.15 sources of kernel, build it, and then use that kernel in that tutorial (if you don't know how, use 2006.0 !).

Personally I had problems with the clock with 2006.1 CD which prevented me from booting from mini or universal CDs.



Optional : you may also need to get Java from IBM's website (for that, create a free account at IBM) [http://www-128.ibm.com/developerworks/java/jdk/linux/older\\_download.html](http://www-128.ibm.com/developerworks/java/jdk/linux/older_download.html).

Select the following line : "32-bit iSeries/pSeries", Column "Java 1.4.2", then "SR4".

Files to download :

- IBMJava2-JRE-142.ppc.tgz
- IBMJava2-SDK-142.ppc.tgz

You can add those two files to your install CD (in a « java » directory or at root) or install them later.

### 1.5.6.2 Base install of Linux Gentoo

Linux Gentoo install is well described at the following URL :

<http://www.gentoo.org/doc/en/handbook/2006.1/handbook-ppc.xml>.

We will however see all the needed steps for Pegasos in this document.

On Pegasos boot, press the 'ESC' key.

On OpenFirmware, type the following command to boot the CD :

```
boot cd boot/menu
```

Then select your video setup. If you don't know which one to choose, just press the Enter key (default choice : 1, automatic setup).

If needed, start by selecting your favorite keyboard layout, for example in France :

```
loadkeys fr-latin1
```

#### 1.5.6.2.1 Partitioning the hard disk

Use the "parted" tool to create three or two partitions on one of your disks ("boot", "swap", and "root"). In parted, type "h" to have the commands list (or see previous chapter about parted in Linux SUSE section). Select a size of 32 to 512Mb for the boot partition (this partition is optional as you can also use a simple directory for that in your root "/" partition), and 1 to 2Gb for your swap. Gentoo system is quite big, so select a 4 to 15Mb partition for system also as a few Gb more for your videos and multimedia files.

#### 1.5.6.2.2 Filesystems

Create "ext3" filesystems on the different partitions :

boot partition (optional, personally I did not created one) : ext3 filesystem (for example on /dev/hdb7 partition):

```
mkfs ext3 /dev/hdb7
```

root partition (« / ») : ext3 filesystem (for example on /dev/hdb5 partition):

```
mkfs ext3 /dev/hdb5
```

Enable swap (for example on /dev/hdb2 partition)

```
mkswap /dev/hdb2
```

```
swapon /dev/hdb2
```

Mount root filesystem:

```
mkdir -p /mnt/gentoo/
```

```
mount /dev/hdb7 /mnt/gentoo/
```

#### 1.5.6.2.3 Create a minimal filesystem :

We are now going to populate our root filesystem using a minimal OS with minimum applications to be able to compile, to update and to boot.

For this, we will use a « stage1 » install.

If you wish, you can also install your system from a "stage3" (stage1+precompiled apps) which will be a way to save time compared to a fully customized install.

For a "stage1" install (mini or Universal CD needed) :

```
cd /mnt/gentoo
```



`tar jxvpf /mnt/cdrom/stages/stage1-xxx` (with <xxx> any string, press <TAB> for automatic completion)

For a "stage1" install (Universal CD needed) :

`cd /mnt/gentoo`

`tar jxvpf /mnt/cdrom/stages/stage3-xxx` (avec <xxx> une chaine quelconque (appuyez sur <TAB> pour la completion automatique)

For example, uncompress « stage1 » archive from your CD :

`cd /mnt/gentoo`

`tar jxvpf /mnt/cdrom/stages/stage1-stage1-ppc-2006.0.tar.bz2`

Copy config.gz (this file is the one that was used to make the kernel, it will be usefull later to make a new kernel) :

`cp /prog/config.gz /mnt/gentoo`

Mount

`mount -o bind /dev /mnt/gentoo/dev`

`mount -o bind /proc /mnt/gentoo/proc`

Copy resolv.conf file (contains resolved IP adress of the computer)

`cp /etc/resolv.conf /mnt/gentoo/etc/resolv.conf`

Change root to use your new system as if we would have booted on it:

`chroot /mnt/gentoo`

Note :

If you need to switch off your Pegasos or to interrupt temporarily the install, you can recover your work later by typing some of the previous commands. For that, type the following (to be adapted with the partition names that you are using).

Beware however to do those commands while using the same starting kernel :

`loadkeys fr-latin1`

`swapon /dev/hdb2`

`mkdir -p /mnt/gentoo`

`mount /dev/hdb7 /mnt/gentoo`

`mount -o bind /dev /mnt/gentoo/dev`

`mount -o bind /proc /mnt/gentoo/proc`

`chroot /mnt/gentoo`

#### 1.5.6.2.4 Synchronising with Gentoo

It's now time to synchronise with Gentoo :

`emerge -sync`

On the first time, this procedure can be quite long depending of the number of months elapsed since the version used and the current date.

This is why, after install, you will have to pay attention to do this procedure quite often, if you need to install additional software or update your system.

This procedure will update your /usr/portage directory tree.

You may need to update portage at this moment. For that :

`emerge portage`

You may also need to update some files. If this is the cas, Portage will tell you so. Do "`emerge --help config`" for additional information. To find the files that need to be updated, do : "

`find /etc -iname '*.cfg????_*' "`

Then to take the changes in account (or not), you can use the "`etc-update`" tool. To have colors in etc-update, do `emerge colordiff` , then in the /etc/etc-update.conf file change the line "`diff_command="diff -uN %file1 %file2" "` by "`diff_command="colordiff -uN %file1 %file2" "`

#### 1.5.6.2.5 Manual update of configuration files

Host name update (/etc/hostname or /etc/conf.d/hostname) :

To define a name for your computer (in the following example « pegasos »), do the following command :

```
echo "pegasos" > /etc/hostname
```

Configuration and compilation file update (/etc/make.conf) :

We have seen previously that this file is the heart of portage. To configure it, edit it with nano on the following way :

```
nano /etc/make.conf
```

Then, update the fields on the following way :

```
CFLAGS="-O3 -mcpu="G4" -mtune=G4 -mabi=altivec -fno-strict-aliasing -pipe"
USE="ppc altivec nptl nptlonly"
FEATURES="buildpkg sandbox"
ACCEPT_KEYWORDS="ppc"
```

A few explanations :

In the CFLAGS, we enable the compilation flags for altivec et and e600 optimisations.

In "USE", we enable the compilation flags or specific options of the various packages : the "ppc" (PowerPC) option, "altivec" and "nptl" (Native Posix Thread Library) which replaces the original glib implementation for thread management.

In "FEATURES", we enable "buildpkg" which means to emerge that it must create a package for each compiled application.

In « ACCEPT\_KEYWORDS », we enable the ebuild type that we want to install. An ebuild is a metadata about a specific version of an application: "ppc" means that by default, we want to use stable packages contrary to ~ppc which is used for unstable applications (not recommended unless you know what you are doing).

#### 1.5.6.2.6 Complete rebuild of the tools chain (bootstrapping) :

You must now rebuild the complete tools chain to obtain a MPC7447 optimized mini-operating system

processor. For that, launch the following script :

```
/usr/portage/scripts/bootstap.sh
```

This procedure is quite long and takes several hours.

#### 1.5.6.2.7 Getting the kernel sources

To unmask the latest 2.6.15 kernel version, type the following command :

```
echo "=sys-kernel/gentoo-sources-2.6.15* ~ppc" >> /etc/portage/package.keywords
```

This will create a package.keywords file if it does not exists yet (which is normally the case for a new install). It will also add the line "=sys-kernel/gentoo-sources-2.6.15\* ~ppc" at the end of the file, indicating that we want to unmask all the unstable gentoo-sources 2.6.15 ppc versions.

Then, do the following command to display the sources version to install:

```
emerge -vp =sys-kernel/gentoo-sources-2.6.15*
```

And to install it for real:

```
emerge -vp =sys-kernel/gentoo-sources-2.6.15*
```

Note : the use of the /etc/portage/package.keywords file is very powerfull : this will let you apply a particular keyword for a particular package (or even to one or several particular packages), contrary to the make.conf file which applies a keyword to all packages.

At last, get the default configuration file of the kernel which is on your CD-ROM (if you have used the 2006.0 CD, it's the 2.6.15-r5 kernel configuration file) :

```
zcat /proc/config.gz > /usr/src/linux/.config
```

This will create a « .config » file that will allow the applications to know how the kernel is configured. This step is important because the emerge command of some packages need that file.

Remarks :

- To know the current kernel version, type the « uname -r » command.

- In our case, we must not use the `emerge gentoo-sources` command which would have as effect to install the latest kernel which is not good because of uncompatibilities with the latest `ivtv` drivers (for PVR-350 TV card in particular).
- The sources are installed in the `/usr/src`. A "linux" link is normally created on the latest installed version.
- If you need to manage several kernel versions, think to modify this link and make it point on the version which you want to use.

#### 1.5.6.2.8 Basic system

Start by installing/update everything about the system :

```
emerge -vp system
```

Note : « system » is a shortcut for a set of packages and applications listed in `/usr/portage/profiles/base/packages`.

The « -vp » option (standing for "verbose pretend") makes it possible to display all the enabled and available keywords. You can modify your `/etc/make.conf` doing :

```
nano /etc/make.conf
```

Add for example the following USE flags :

```
USE="ppc altivec nptl nptlonly unicode nls python ipv6 ldap perl tcpd bzip2 gtk"
```

If you now do an « `emerge -vp system` », you will note that the software list to install has grown.

Now, do a « `emerge -e system` » to launch the compiling and package installation. This procedure is quite long and takes several hours.

At the end of this `emerge`, you may have many files to rename/change.

For that use the "etc-update" command has explained previously.

You can now add your favorite applications, using the same way of doing things.

For example :

```
emerge -vp portage-utils dhcpcd metalog cron
```

The « portage-utils » package will install the « `qlop` » program which is very useful to know the `emerge` list being done.

Do « `qlop -p` » for an overview.

#### 1.5.6.2.9 Desktop installation

For the desktop, we will choose `gnome`. Note that we could also have chosen `kde`, `xfce` or any other. This is an arbitrary choice like any one.

Start by doing the following command to see the long package list which will be installed :

```
emerge -vp gnome
```

Before doing installation, choose the USE flags that you will need. Do "emerge -vp gnome| more" to see the entire list (but without the colors).

Add for example the following additional USE flags to the previous list:

```
"zlib jpeg jpeg2k tiff png gif mpeg mp3 vorbis truetype sasl ssl samba nfs gs firefox"
```

This includes compression modules, images and sounds types support, fonts, Ghostscript support, and various network protocols.

Also add the line `VIDEO_CARDS="radeon"` if you have a RADEON or `VIDEO_CARDS="tdfx"` if you have a 3DFX : this line will be used for the "mesa" package installation.

Once finished, launch install :

```
emerge gnome
```

This procedure is once again very very long (it takes almost 2 full compilation days – yes you have well read !).

#### 1.5.6.2.10 Additional applications install

You can benefit from it to install a few additional applications. For each application there after, think to do a "emerge -vp myapp" to check the USE flags, then do a "emerge -f myapp" to download the packages and then a "emerge myapp" to install "myapp" for real.

- profuse : this is a USE flags edition software. This will be useful later...
- ftp : this is always useful...
- gentoolkit : installs the additional tools like equery which can handle « ebuild » packages. For example, the "equery list" command display all the packages installed in your distribution.
- Xorgautoconfig : automatic configuration of your video card for Xorg. This application will detect your video card and will configure automatically the /etc/xorg.conf configuration file. Note : note that the X is a capital letter. Then launch Xorgautoconfig to configure Xorg.

#### 1.5.6.2.11 System configuration for a first boot

We now have a first complete system and it's now time to configure it to be able to do a first boot on the hard disk.

##### 1.5.6.2.11.1 Configuring DHCP network

Edit the /etc/conf.d/net file « `nano /etc/conf.d/net` » and add the following line :

```
iface_eth0="dhcp"
```

This will configure eth0 (the Ethernet 10/100 port of the Pegasos) with a dynamic IP at startup. If you'd rather use the second Gigabit port, add the `iface_eth1="dhcp"` line instead.

For more details about network configuration, see the Gentoo's website.

Also edit the /etc/conf.d/domainname with nano ("`nano /etc/conf.d/domainname`") and set the OVERRIDE flag to 0 if you don't use a domain name.

##### 1.5.6.2.11.2 Configuring the timezone

Do the following commands :

```
rm -rf /etc/localtime
```

```
ln -s /usr/share/zoneinfo/Europe/Paris /etc/localtime
```

Note : of course, change « Europe/Paris » depending of your country. Use the <TAB> key for automatic key completion.

##### 1.5.6.2.11.3 Configuring the mount points

Edit the /etc/fstab file with nano :

```
nano /etc/fstab
```

Replace "ROOT" with the partition name onto which you have installed your system, for example "/dev/hdb5".

Also replace "SWAP" by the partition name that you have selected for swap, for example "/dev/hdb2".

Comment the « BOOT » line by adding a # at the beginning of the line.

##### 1.5.6.2.11.4 Selecting a password

To choose a password (necessary to start with a login manager such as « gdm » which is installed with GNOME), do:

```
passwd
```

Then type your password, and confirm it by typing it again.

#### 1.5.6.2.11.5 Adding automatic startup of services

If you have installed other applications and that they must be started automatically, you will only need to do the « `rc-update add myservice default` » command. The available services can be found in `/etc/init.d`.

Metalog : "`rc-update add metalog default`". Metalog is the « logger » that we have installed at the beginning of this tutorial. If a system needs to send a message to a log file, it does contact metalog which handles the writing into the good file. Metalog is ideal for a multimedia system because it avoids writing too many messages at the same time which could slow down the system.

Network : "`rc-update add net.eth0 default`".

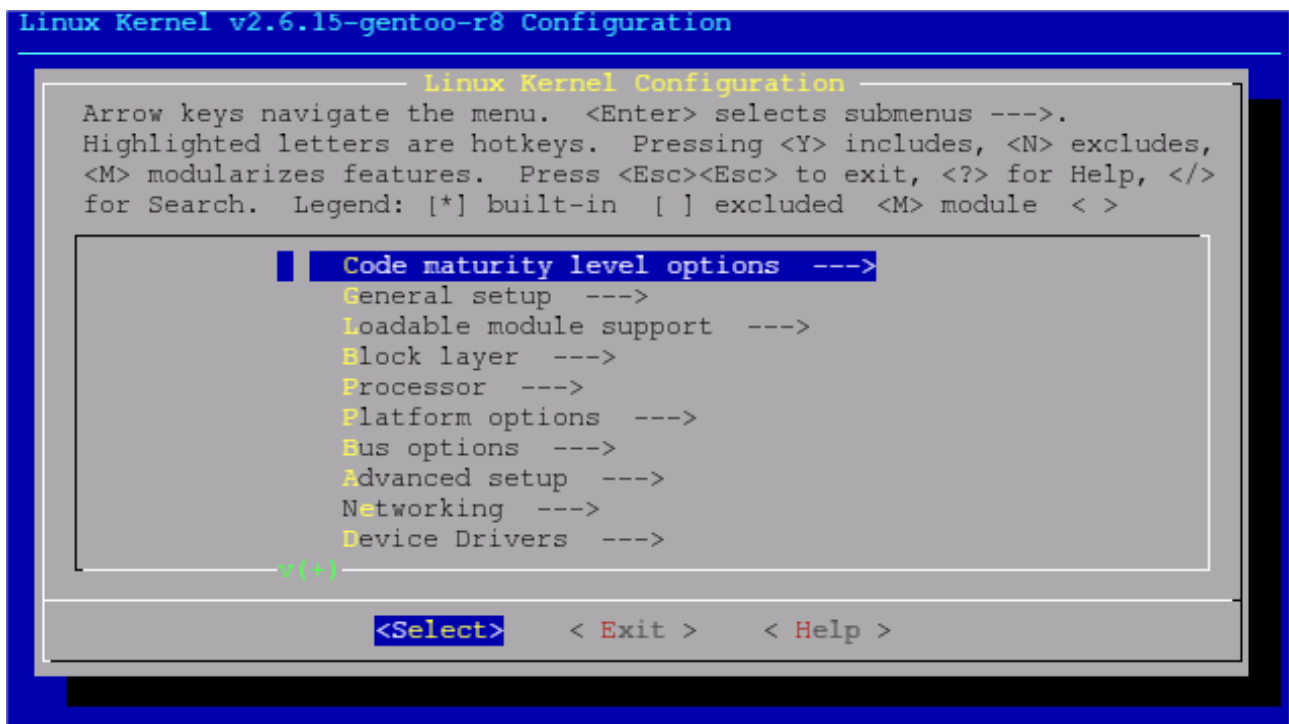
#### 1.5.6.2.12 Building a new kernel

We are now going to build a new kernel that we make it possible to boot from the hard disk. We have installed the sources in `/usr/src/linux` also as the default configuration file in `/usr/src/linux/.config`.

Just in case, do a backup of that file under another name, for example, « `.config.backup` ». That may be useful in case of mistake...

Then, type the following commands :

```
cd /usr/src/linux
make menuconfig
```



A graphic user interface appears into which you will be able to select what you want to enable or not into your kernel. With the use, you will see that its use is not too complex (knowing that the `.config` file is already quite well configured for a Pegasos).

From the starting configuration :

- Processor / Thermal support : to be disabled
- Platform options / High memory support : to be enabled if you have much memory (it allows you to use more than 750Mb)
- Platform options / Default bootloader arguments : to be disabled (useless on Pegasos because we use OpenFirmware for the arguments)

This is enough for a first kernel, we will enable other options later after having tested the first boot.

Quit and save the file, you are now ready to compile a first version of the kernel.

For that, do : `"make all modules_install"` and wait until completion (15 to 30 minutes).

Then copy the kernel into your boot directory like this :

```
cp /usr/src/linux/arch/ppc/boot/images/zImage.chrp /boot/pegasos-2.6.15-r8_base
```

Use the name you want unless you're not lost with it, then create a link on the kernel so that you can always boot on it with a simple name (for example "pegasos"):

```
ln -s /boot/pegasos-2.6.15-r8_base /boot/pegasos
```

You are now ready for a first startup. Press the "reset" button of your Pegasos, or enter the "reboot" command.

On your Pegasos boot, press the "ESC" key.

To start, from OpenFirmware, enter the following command, for a partition on /dev/hdb5 :

```
boot hd0:4 pegasos video=radeonfb:1024x768@75 root=/dev/hdb5
```

Change "pegasos" depending of the name you have chosen for your kernel link.

If you can, add this line now to your favorite boot manager (for example GRUB2 or bootcreator).

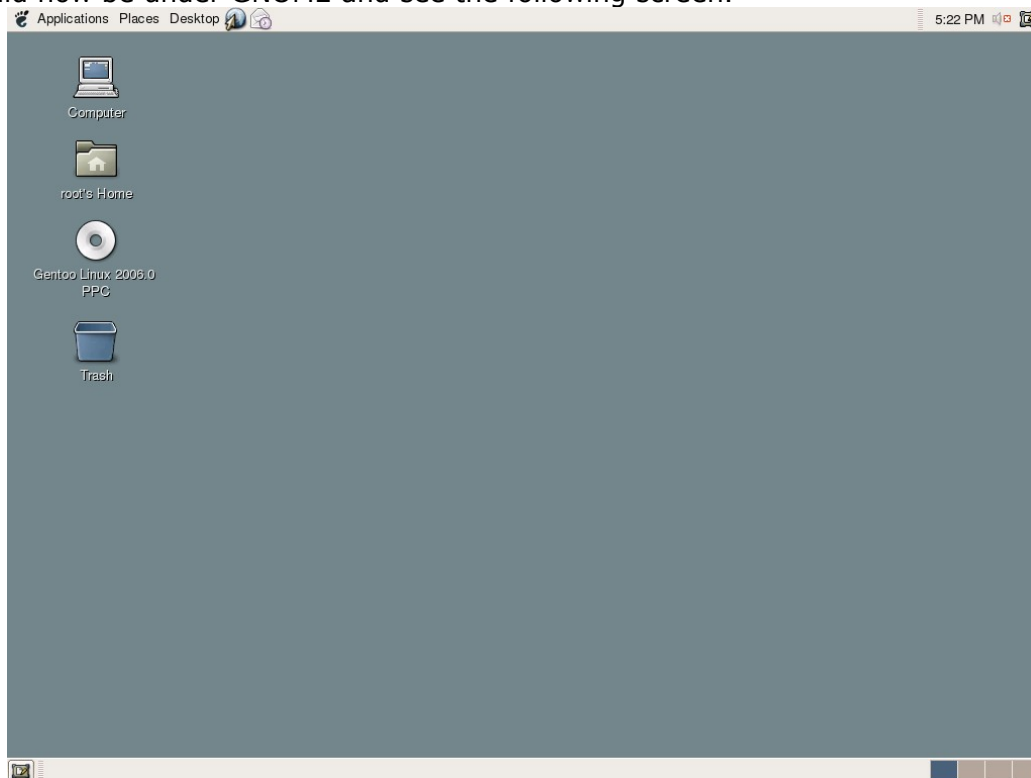
#### 1.5.6.2.13 First startup

On first startup, you are still in text mode. Log as « root » and enter the password you have chosen before (beware that you are again in QWERTY layout).

Then load your favorite layout :

Example for french "AZERTY" : `"loadkeys fr-latin9"`

Now, do the `"gdm"` command to launch the Gnome Desktop Manager. This should normally launch a graphical window under Xorg. Enter your login and password again into this window. You should now be under GNOME and see the following screen.



Select "Application/Accessories/Terminal" to launch a console and check that everything is OK. If everything works correctly, you can now make gdm be launched automatically on each boot.

For that, start by editing the /etc/rc.conf file with nano : `"nano /etc/rc.conf"`.

Uncomment the DISPLAYMANAGER line by adding a "#" at the beginning and then replace "xdm" by "gdm".

Also uncomment the XSESSION line at the end of the file, by removing the "#" at the beginning of the line.

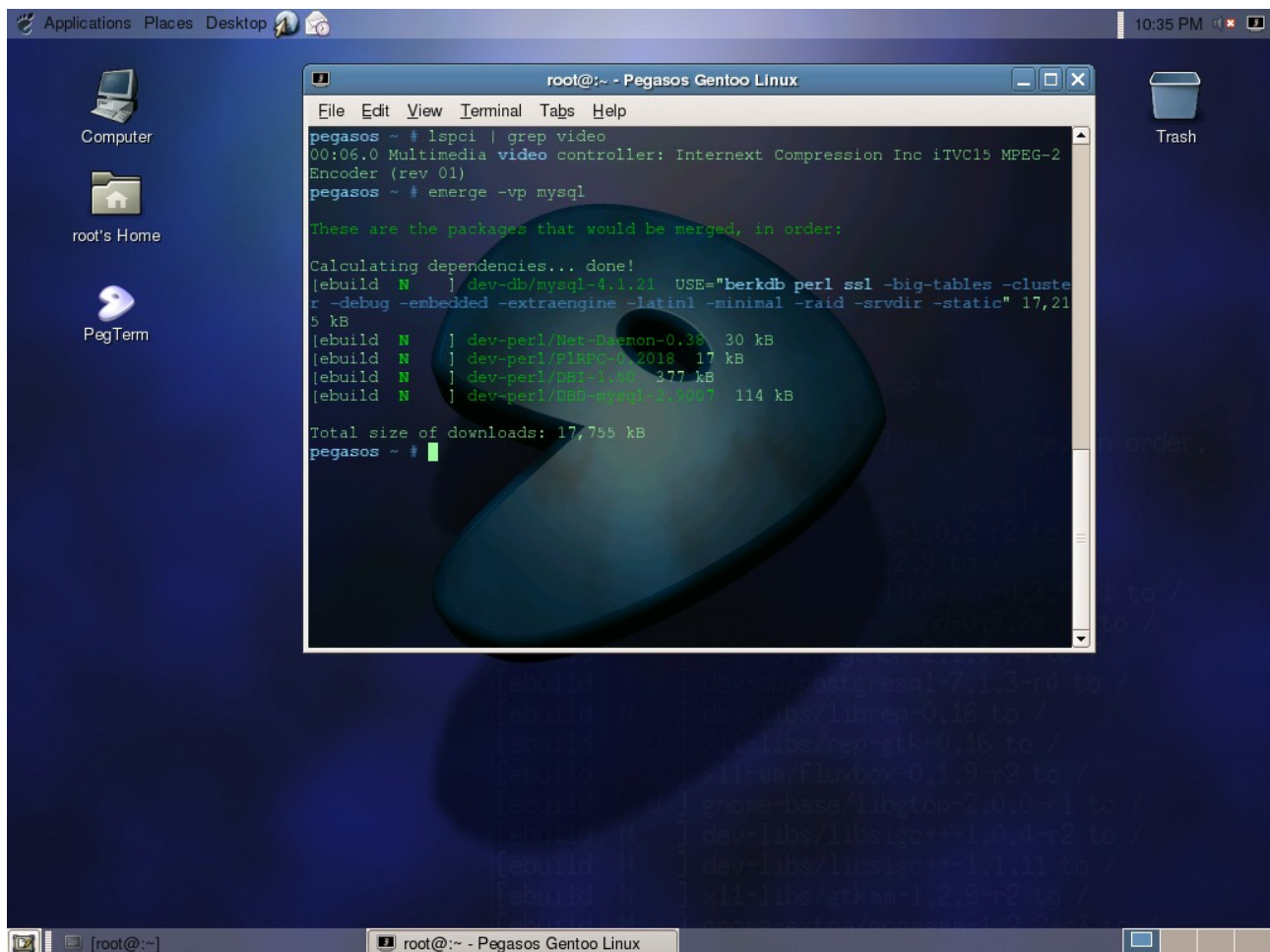


Then do the command "`rc-update xdm default`" to launch the `/etc/init.d/xdm` script on each boot (which will have for effect to launch gdm automatically).

#### 1.5.6.2.14 Customizing the desktop

Before going further, we are going to customize the graphical interface of our desktop.

- Keyboard layout : to enable your favorite keyboard under Gnome, go into the "Preferences\Keyboard" then on "Layout" tab. Click on "Add" then select your country, for example "France" and close the window. Open a new terminal. This should normally work with the keyboard you've just set.
- Styles : background, terminal, login window... Everything is customizable.
- Terminal windows : Applications/Accessories/Terminal, then "Edit/Profiles" in the terminal menu. You can edit different profiles to have windows with different styles.
- Background : Desktop/Preferences/Desktop background
- Login window : Desktop/Administration/Login window



### 1.5.6.3 Installing drivers

#### 1.5.6.3.1 ATI drivers and Xorg server

To install xorg, start by enabling the following options into your `/etc/make.conf` file :

```
VIDEO_CARDS="fbdev radeon vesa"
```

The « dri » USE flag must also be enabled, but it is normally already the case in `/etc/make.profile/make.defaults`.

Do the following commands to install ATI drivers then Xorg. As previously, one with "-vp" to see what USE flags must be added, then "-f" to download the files, then without any option to compile and install.

```
emerge ati-drivers mesa
emerge xorg-x11
```

Restart your computer now (or 3D won't work).

To check that 3D works after having rebooted, do the following command :

```
glxinfo | grep rendering
```

The Direct rendering answer must be « Yes » if it works. If it doesn't, check your /etc/X11/xorg.conf file (the "glx" module must be loaded and the used driver must be "radeon" and "BusType PCI" option enabled).

Extract from a working /etc/X11/xorg.conf file:

```
Section "module"
Load "glx"
...
Section "Device"
Option "BusType" "PCI"
Driver "radeon"
```

#### 1.5.6.3.2 « ivtv » TV tuner driver for Hauppauge PVR 350 TV tuner card

Install of ivtv drivers is quite delicate because the only version which seems to be working well is the 0.4.2 version.

Strangely, it is not available under ebuild form anymore (maybe it's too old ?) and it only works with 2.6.15 kernel, from where our choice of the beginning of the tutorial.

To get the sources, go to <http://ivtvdriver.org> and select 0.4.2 version in "Download" section and then "Previous releases".

Direct link: <http://dl.ivtvdriver.org/ivtv/archive/0.4.x/ivtv-0.4.2.tar.gz>.

You must also get the firmware which can be found here :

<http://dl.ivtvdriver.org/ivtv/firmware/firmware.tar.gz>.

Then create an IVTV directory for example in /usr/src and put the two archives here.

Recompile the kernel :

Before going further we have to change a few settings into our kernel.

Go into « /usr/src/linux » and do "menuconfig" to launch the configurator.

In "Device drivers" then in "Multimedia devices", enable « Video for Linux » as built-in « <\*> ». In the sub-menu, enable « BT848 Video for Linux », « Add support for additional audio chipsets », and « Add support for additional video chipsets » options:

```
Device Drivers
>
  Multimedia devices
  >
    <*> Video For Linux
      <*> BT848 Vidéo for Linux
      <*> Add support for additional audio chipsets
      <*> Add support for additional video chipsets
```

Then recompile the kernel as explained previously (but choose a new kernel name).

```
make all modules_install
cp /usr/src/linux/arch/ppc/boot/images/zImage.chrp /boot/pegasos-2.6.15-r8_ivtv
rm /boot/pegasos
ln -s /boot/pegasos-2.6.15-r8_ivtv /boot/pegasos
```

Reboot from OpenFirmware, still with the same command line or from your bootcreator or GRUB menu:

```
boot hd0:4 pegasos video=radeonfb:1024x768@75 root=/dev/hdb5
```

After reboot, install the firware and ivtv drivers as follows. firmware et les pilotes ivtv.

Installing the firmwares:

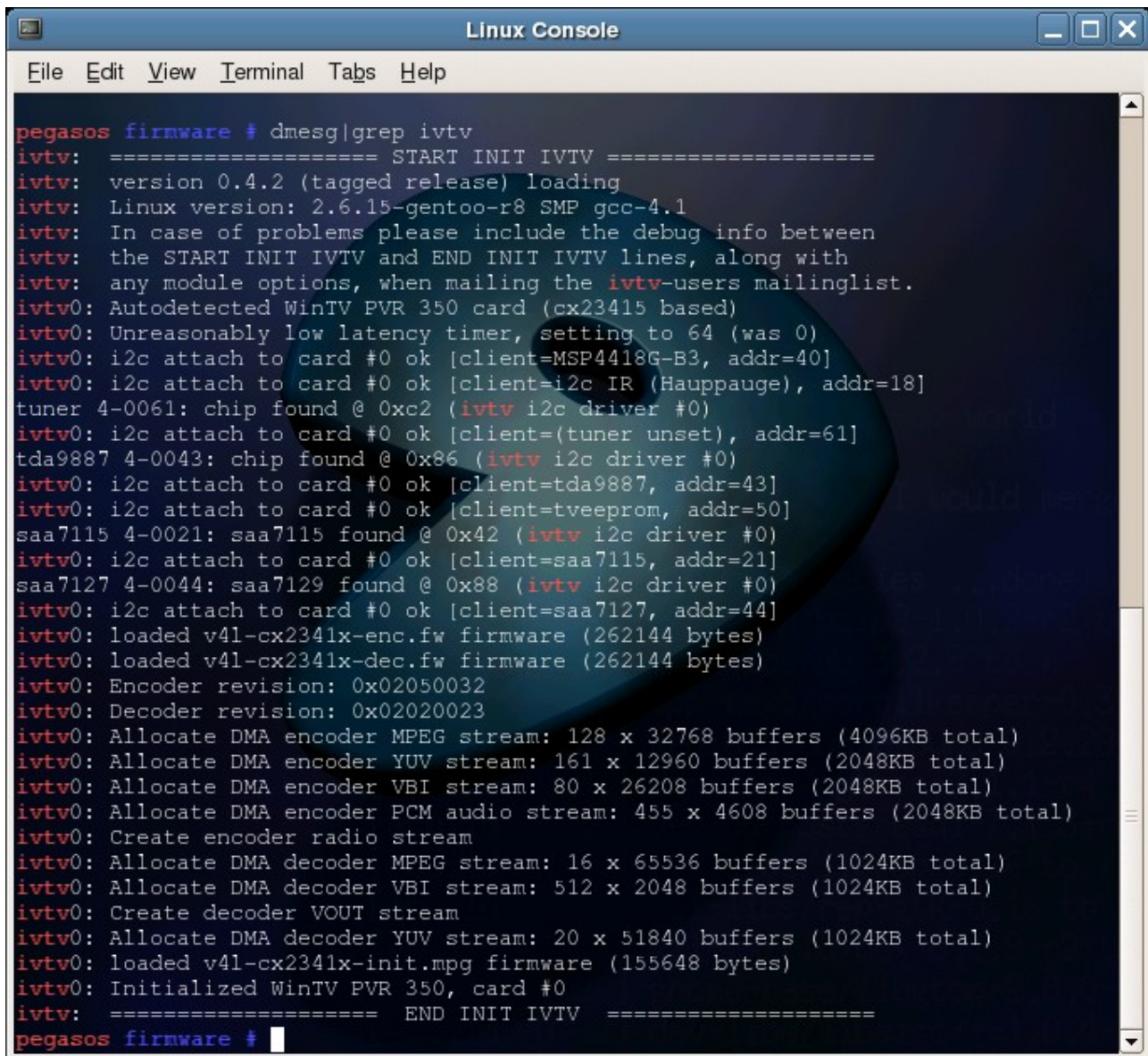
```
cd /usr/src/IVTV
tar zxvf firmware.tar.gz -C /usr/lib/htplog/firmware
```

Compiling and installing the ivtv 0.4.2 drivers:

```
cd /usr/src/IVTV
tar zxvf ivtv-0.4.2.tar.gz
cd ivtv-0.4.2
```



```
make
make install
depmod
modprobe ivtv
```



```
Linux Console
File Edit View Terminal Tabs Help

pegasos firmware # dmesg|grep ivtv
ivtv: ===== START INIT IVTV =====
ivtv: version 0.4.2 (tagged release) loading
ivtv: Linux version: 2.6.15-gentoo-r8 SMP gcc-4.1
ivtv: In case of problems please include the debug info between
ivtv: the START INIT IVTV and END INIT IVTV lines, along with
ivtv: any module options, when mailing the ivtv-users mailinglist.
ivtv0: Autodetected WinTV PVR 350 card (cx23415 based)
ivtv0: Unreasonably low latency timer, setting to 64 (was 0)
ivtv0: i2c attach to card #0 ok [client=MSP4418G-B3, addr=40]
ivtv0: i2c attach to card #0 ok [client=i2c IR (Hauppauge), addr=18]
tuner 4-0061: chip found @ 0xc2 (ivtv i2c driver #0)
ivtv0: i2c attach to card #0 ok [client=(tuner unset), addr=61]
tda9887 4-0043: chip found @ 0x86 (ivtv i2c driver #0)
ivtv0: i2c attach to card #0 ok [client=tda9887, addr=43]
ivtv0: i2c attach to card #0 ok [client=tveeprom, addr=50]
saa7115 4-0021: saa7115 found @ 0x42 (ivtv i2c driver #0)
ivtv0: i2c attach to card #0 ok [client=saa7115, addr=21]
saa7127 4-0044: saa7127 found @ 0x88 (ivtv i2c driver #0)
ivtv0: i2c attach to card #0 ok [client=saa7127, addr=44]
ivtv0: loaded v4l-cx2341x-enc.fw firmware (262144 bytes)
ivtv0: loaded v4l-cx2341x-dec.fw firmware (262144 bytes)
ivtv0: Encoder revision: 0x02050032
ivtv0: Decoder revision: 0x02020023
ivtv0: Allocate DMA encoder MPEG stream: 128 x 32768 buffers (4096KB total)
ivtv0: Allocate DMA encoder YUV stream: 161 x 12960 buffers (2048KB total)
ivtv0: Allocate DMA encoder VBI stream: 80 x 26208 buffers (2048KB total)
ivtv0: Allocate DMA encoder PCM audio stream: 455 x 4608 buffers (2048KB total)
ivtv0: Create encoder radio stream
ivtv0: Allocate DMA decoder MPEG stream: 16 x 65536 buffers (1024KB total)
ivtv0: Allocate DMA decoder VBI stream: 512 x 2048 buffers (1024KB total)
ivtv0: Create decoder VOUT stream
ivtv0: Allocate DMA decoder YUV stream: 20 x 51840 buffers (1024KB total)
ivtv0: loaded v4l-cx2341x-init.mpg firmware (155648 bytes)
ivtv0: Initialized WinTV PVR 350, card #0
ivtv: ===== END INIT IVTV =====
pegasos firmware #
```

To check if the driver works, do the command `"dmesg|grep ivtv"`. There should be no error message in the resulting log (see image below).

You can also try the following command to test the video tuner:

```
cat /dev/video0 > /root/testivtv.mpg
```

Note that you will need additional software to check that you have recorded the TV without any error (or snow if you haven't plugged your TV cable yet).

If this works, edit the file `« /etc/modules.autoload.d/kernel2.6 »` and add the line `« ivtv »` to make ivtv loaded on each boot.

```
nano /etc/modules.autoload.d/kernel2.6
```

### 1.5.6.3.3 ALSA sound driver

To know the name of your sound chip, do :

```
lspci -v | grep -i audio
```

On Pegasos, the sound chip is located on the mother board and should be:

```
00:0c.5 Multimedia audio controller: VIA Technologies, Inc. VT82C686 AC97 Audio
Controller (rev 40)
```

We are now going to enable the necessary modules in the kernel.

```
cd /usr/src/linux
make menuconfig
```

Then, enable the following options:

```
Device Drivers --->
    Sound --->
```

(This module must be enabled)

```
<M> Sound card support
```

(Make sure that OSS is disabled)

```
Open Sound System --->
    < > Open Sound System (DEPRECATED)
```

(Go back to the previous menu and go into ALSA)

```
Advanced Linux Sound Architecture --->
    <M> Advanced Linux Sound Architecture
```

(Select the following option if you need MIDI support)

```
<M> Sequencer support
```

(Support of old /dev/mixer\* and /dev/dsp\*. Recommended.)

```
<M> OSS Mixer API
```

```
<M> OSS PCM (digital audio) API
```

(Now enable the Pegasos's sound chip module)

```
PCI devices
>
<M> VIA 82C686A/B, 8233/8235 AC97 Controller
```

Quit and save.

Then recompile the kernel as explained before (still with a new name each time).

```
make all modules_install
cp /usr/src/linux/arch/ppc/boot/images/zImage.chrp /boot/pegasos-2.6.15-r8_ivtv_alsa
rm /boot/pegasos
ln -s /boot/pegasos-2.6.15-r8_ivtv_alsa /boot/pegasos
```

Reboot from OpenFirmware, still with the same command line or from your bootcreator or GRUB menu:

```
boot hd0:4 pegasos video=radeonfb:1024x768@75 root=/dev/hdb5
```

Now, open a new terminal and do the following commands:

```
emerge alsa-utils
alsaconf
```

The second command will open a small graphical interface which will automatically try to detect your peripherals (depending of the modules enabled in your kernel). On the question about /etc/modules.d/alsa update, answer "yes" to let "alsaconf" configure it by itself. The sound volume parameters will be adjusted at their optimal levels, the "module-update" command will be done and the /etc/init.d/alsasound service will be started, all automatically.

Now, add the "alsasound" to boot level.

```
rc-update add alsasound boot
```

If you haven't forgotten anything, you should see the following messages :

```
* alsasound added to runlevel boot
* rc-update complete.
```

Check the /etc/conf.d/alsasound file and ensure that SAVE\_ON\_STOP is set to "yes". This field enables audio parameters saving on shutdown.

The users having access to the sound must be added to the audio group. This must also be done for "root" user:

```
gpasswd -a root audio
```

Don't forget to execute this command for any new user, changing « root » by the user login name.

To test if sound works, we are now going to add a few audio applications. For example "madplay", "mpg123", "vorbis-tools" for ogg :

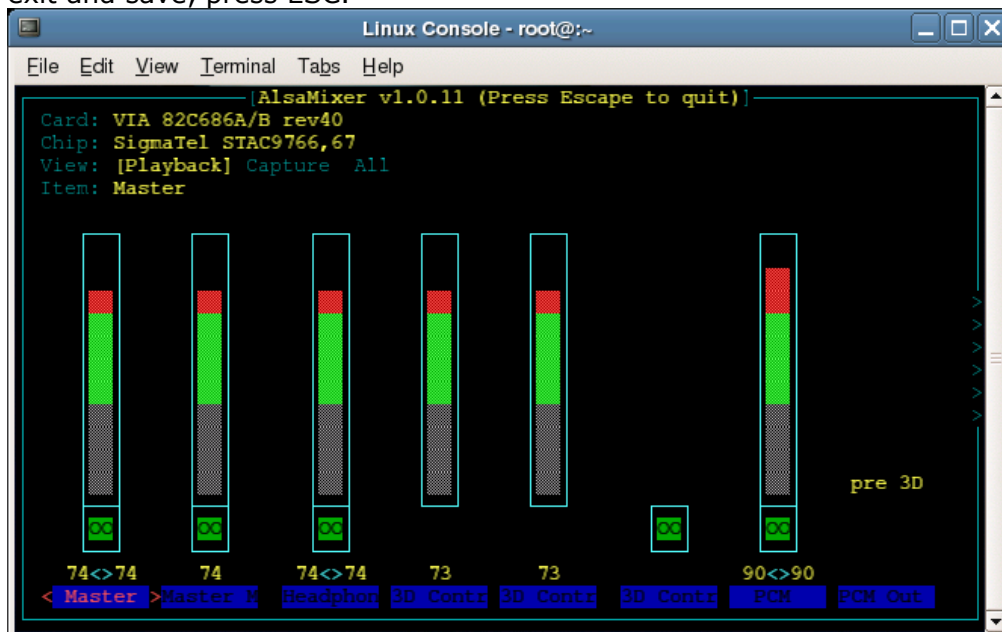
```
emerge madplay mpg123 xmms
emerge vorbis-tools
```

Now, in /etc/make.conf, do not forget to add the "alsa" USE flag to enable the sound for all applications supporting it.

Then to test sound, do one of the following commands with a musical file of your choice:

```
madplay -v mymusic.mp3
mpg123 mymusic.mp3
ogg mymusic.ogg
```

Note : to set the sound volume, use "alsamixer" from a terminal. In alsamixer, enable/disable a sound source with the M key (mute), and sound level with the up and down arrow keys. To exit and save, press ESC.



Then, execute "alsactl store" to save the changes.

#### 1.5.6.3.4 lirc driver : infra-red remote control

Edit the make.conf file:

```
nano /etc/make.conf
```

For a Hauppauge remote control, add the following line at the end of the file:

```
LIRC_DEVICES="hauppauge"
```

The « `emerge -vp lirc` » command should now advise you that the "hauppauge" driver will be enabled on compilation.

Do "emerge lirc". After compilation, install should have created a /etc/lircd.conf file containing the Hauppauge remote controllers configurations.

Do "modprobe lirc\_i2c" to load the module, then "irw" to test your remote control.

To load the module on each boot, add it to /etc/modules.autoload.d/kernel-2.6 file :

```
echo "lirc_i2c" >> /etc/modules.autoload.d/kernel-2.6
```

Then launch the service doing a "/etc/init.d/lircd start" and a "/etc/init.d/lircmd start" command, then to test that it works to the command "irw".

```
rc-update add lircd default
rc-update add lircmd default
irw
```

#### 1.5.6.4 Various multimedia applications

Before installing mysql and mythtv, we are going to install mplayer, which is a video player. We will benefit from it to activate a good number of USE flags to avoid recompiling multiple packages. For that, we are going to use "-vp" options to have a list of the modules that can be enabled or are already enabled:

```
emerge -vp mplayer
emerge -vp mysql
emerge -vp mythtv
```

This time, we will use « profuse » in another terminal to enable interesting USE flags:

```
profuse
```

At the end, in /etc/make.conf file, the following USE flags have been enabled:

```
USE=" aac aalib bzip2 cdparanoia dga directfb dts dv dvb dvdread \
fbcon fftw firefox ggi gs ieee1394 jack jpeg2k libcaca lirc \
lzo musepack mysql nas nfs nptlonly openal oss samba sasl \
sndfile speex tga theora tiff toolbar v4l v4l2 x264 xanim \
Xaw3d xvid"
```

Now, we must do a « `emerge -Na world` » to update the modules which are already installed depending of the USE flags modifications:

Now do a first « `emerge -Naf world` » ("Yes") for downloading the different packages to compile. Then, do a « `emerge -Na world` » to update the system for real. This procedure is quite long.

Now we install mplayer:

```
emerge mplayer
```

#### 1.5.6.5 Installing MySQL database manager

To install MySQL, type successively the following commands:

```
emerge mysql
emerge --config mysql (emerge --config =dev-db/mysql-4.1.21)
/etc/init.d/mysql start
rc-update add mysql default
```

#### 1.5.6.6 Installing and configuring MythTV

Our system is now ready for a MythTV installation.

##### 1.5.6.6.1 Installation

The "`emerge --search mythtv`" command will give us a list of all available MythTV plugins. Some of them are masked. We then have to unmask them in the /etc/portage/package.keywords with nano or gedit : (`nano /etc/portage/package.keywords`)

```
media-tv/mythtv -* ~ppc
media-plugins/mythbrowser -* ~ppc
media-plugins/mythcontrols -* ~ppc
media-plugins/mythdvd -* ~ppc
media-plugins/mythflix -* ~ppc
media-plugins/mythgallery -* ~ppc
media-plugins/mythgame -* ~ppc
media-plugins/mythmusic -* ~ppc
media-plugins/mythnews -* ~ppc
```

```
media-plugins/mythphone -* ~ppc
media-plugins/mythvideo -* ~ppc
media-plugins/mythweather -* ~ppc
www-apps/mythweb -* ~ppc
x11-themes/mythtv-themes -* ~ppc
```

Check once again that the USE flags are enabled:

```
emerge -vp mythtv mythbrowser mythcontrols mythdvd mythflix mythgallery mythgame
mythmusic mythnews mythphone mythvideo mythweather mythweb mythtv-thèmes
```

Download the files to be compiled :

```
emerge -f mythtv mythbrowser mythcontrols mythdvd mythflix mythgallery mythgame mythmusic
mythnews mythphone mythvideo mythweather mythweb mythtv-themes
```

Then, compile and install :

```
emerge mythtv mythbrowser mythcontrols mythdvd mythflix mythgallery mythgame mythmusic
mythnews mythphone mythvideo mythweather mythweb mythtv-themes
```

#### 1.5.6.6.2 Initial configuration of MythTV

To configure MythTV database, do the following commands:

```
mysql -u root -p < /usr/share/mythtv/database/mc.sql
```

Type your password.

Then create the following directories that will be used by MythTV and its plugins:

```
mkdir /mnt/store : TV recordings
mkdir /mnt/store/dvd : dvd
mkdir /mnt/store/dvdpix : dvd pix
mkdir /mnt/store/pix : images
mkdir /mnt/store/music : musics
```

To configure MythTV (backend), do the command:

```
mythtv-setup
```

This command launches a graphical interface in which you can configure MythTV.

After having chosen your language, you have access to five menus:

- 1.General
- 2.Capture cards
- 3.Vidéos sources
- 4.Input Connections
- 5.Channel editor





You have to configure MythTV from steps 1 to 5 :

General :

- \* General – Host address backend setup

Leave the default values

- \* General – Host-specific backend setup

Select a directory name to hold your recordings. By default /mnt/store.

- \* General – Global backend setup

Select here your TV format and country settings. For example for France:

TV format: SECAM,

VBI format: PAL Teletext

Channel frequency table: France

Leave the default values for other settings.

Then leave the default values in the following menus.

Capture card :

- \* Capture card - Capture card

Create a new capture card

- \* Capture card setup

Select "MPEG-2 encoder standard (PVR-x50, PVR-500)

Video device : /dev/v4l/video0 (valeur par défaut)

Default input : Tuner0

- \* Capture card - Capture card

Go back to main menu (ESC key).

Video sources :

- \* Video sources – Video sources

Create a new video source

- \* Video sources – Video source setup

Example of settings for France cable :

Video source name: Cable

XMLTV listings grabber: France

Channel frequency table: France

Input connections :

- \* Input connections

Choose Tuner 0

- \* Input connections – Connect source to input

Capture device : Cable

Click on Scan for channels (after having connected your TV cable to your PVR card).

- \* Scan type

Video source: Cable

Capture card: [MPEG:/dev/v4l/video0]

Scan type: full scan

Then let it scan, and finish.

Channel editor :

Select a channel, then change its name and its number. Restart for all other channels.

Once all the settings done, quit mythtv-setup by pressing the ESC from main menu.

Then do the following command to update the database.

```
mythfilldatabase
```

#### 1.5.6.6.3 Launching MythTV

In a console do : "mythbackend"

In another console do : "mythfrontend"

Note : you can add mythbackend to the boot script:

```
/etc/init.d/mythbackend start
```

```
rc-update add mythbackend default
```

#### 1.5.6.6.4 Configuring MythTV (frontend) and its plugins

News :

Edit the /usr/share/mythtv/mythnews/news-sites.xml file with nano to include your favorite websites.

```
nano /usr/share/mythtv/mythnews/news-sites.xml
```

Add for example the following lines after the « <sites> » tag:

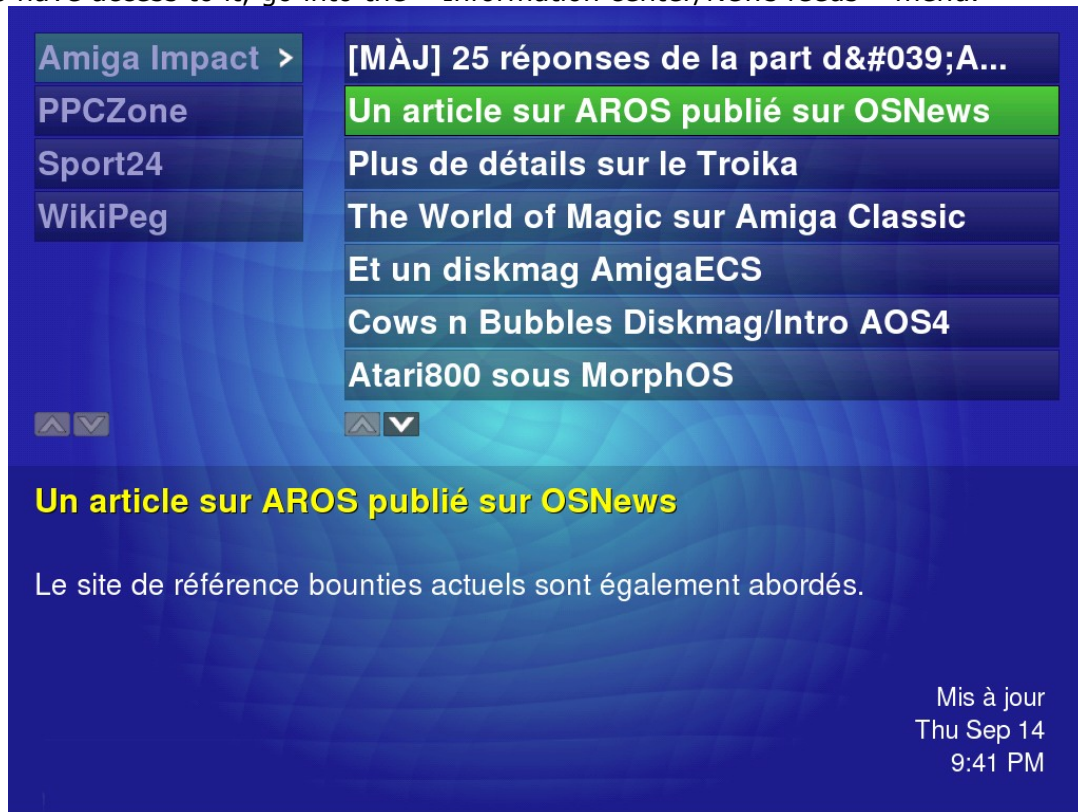
```
<category name="Sites français">
  <item>
    <title>Amiga Impact</title>
    <url>http://www.amigaimpact.org/backend.php>
    <ico>www.amigaimpact.org/favicon.ico</ico>
  </item>
  <item>
    <title>Sport24</title>
    <url>http://www.sport24.com/rss/index.xml>
    <ico>http://www.wikipeg.org/favicon.ico>
  </item>
</category>
```

Then launch mythfrontend and go into « Utilities setup/Setup/Info center settings/News settings » menu.



Select the website you want to add to the list.

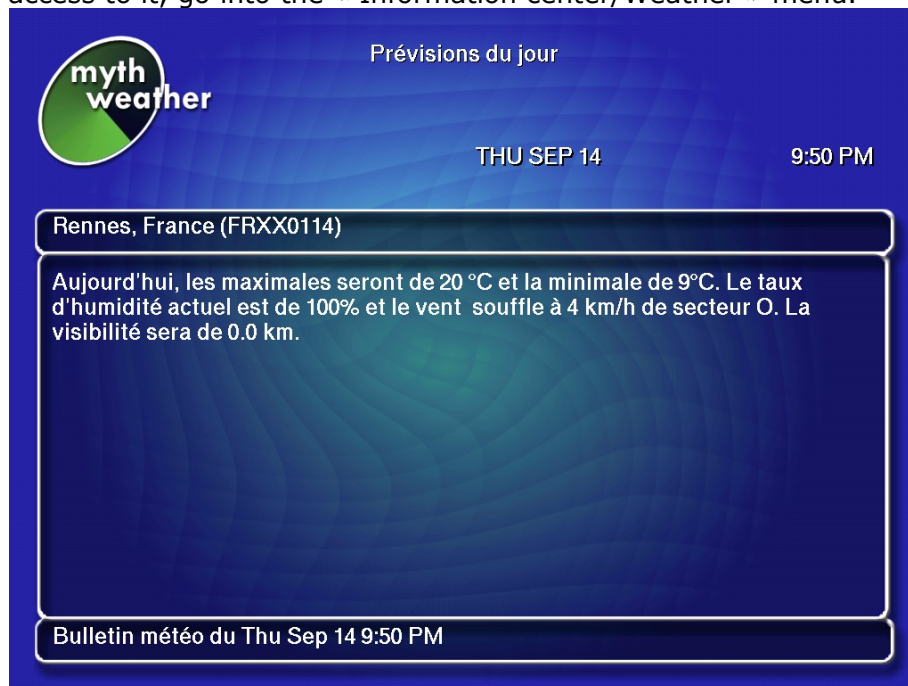
Then to have access to it, go into the « Information center/News feeds » menu.

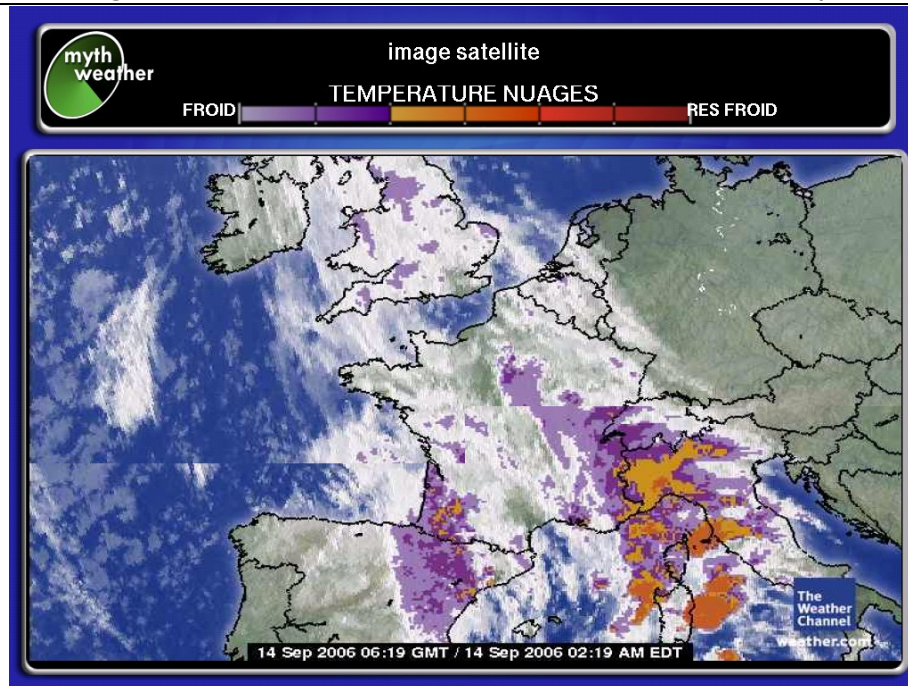


#### Weather:

To configure the weather plugins, in «Utilities setup/Setup/Info center settings/Weather settings», simply choose your choisisez simplement your weather units system and your city.

Then to have access to it, go into the « Information center/Weather » menu.





myth weather

Conditions actuelles

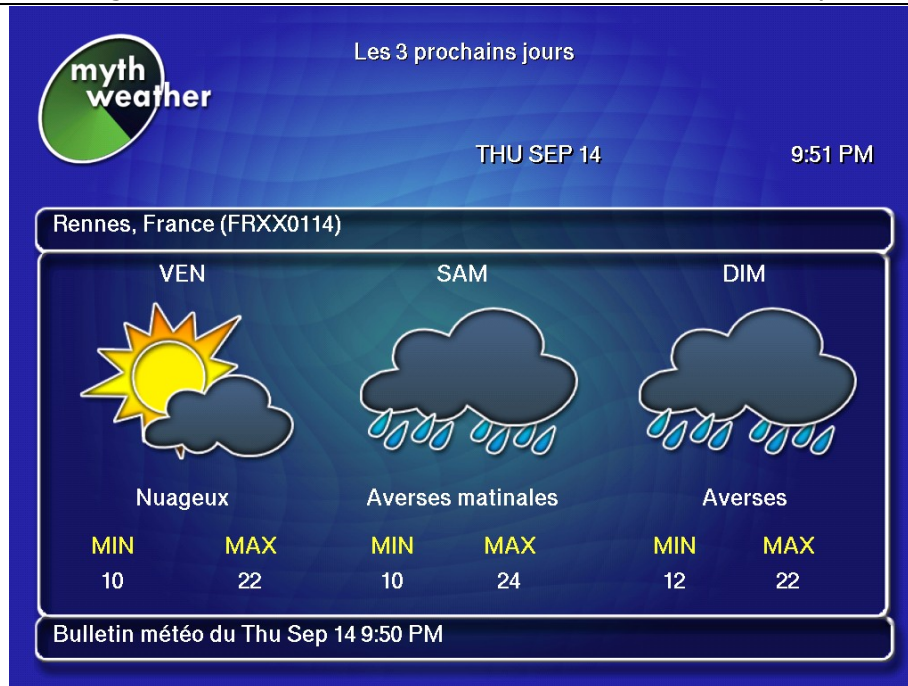
THU SEP 14 9:51 PM

Rennes, France (FRXX0114)

Humidité	100%
Pression	100.3 kPa
Vent	0 à 4 Km/h
Visibilité	0.0 km
Sensation	16 °C
Indice UV	0 (minimal)

Couvert 16°C

Bulletin météo du Thu Sep 14 9:50 PM



### Games

The MythTV's game plugins is mostly targeting emulation. You will have to specify an emulator name with ROMs directory (which does unfortunately prevent a standard Linux game usage, because this directory argument is mandatory).

To install xname (Linux Multi Arcade Machine Emulator) :

```
emerge xname gxname
mkdir -p /usr/share/xname/{highscore,roms,snaps}
ln -s /usr/share/games/xname /usr/local/share/xname
```

You will then need additional ROMS (or games). You can already download two legal ROMS at the following URLs:

<http://www.mame.net/roms/gridlee.zip>

<http://www.mame.net/roms/robby.zip>

Copy the two files without uncompressing them in /usr/local/share/xname/roms.

```
cp /root/Desktop/Downloads/gridlee.zip /usr/local/share/xname/roms
cp /root/Desktop/Downloads/robby.zip /usr/local/share/xname/roms
```

Add the "root" user in the games group (mandatory to use the games) :

```
gpasswd -a root games
```

Edit the /root/.bash\_profile file and add the following lines:

```
PATH=$PATH:/usr/games/bin
export PATH
```

Then configure the xname profile:

```
nano /root/.xname/xnamerc
```

And add the following lines:

```
rompath /usr/local/share/xname/roms
snapshot_directory /usr/local/share/xname/snaps
cheat_file /usr/local/share/xname/cheat.dat
hiscore_file /usr/local/share/xname/hiscore.dat
hiscore_directory /usr/local/share/xname/hiscore
history_file /usr/local/share/xname/history.dat
mameinfo_file /usr/local/share/xname/mameinfo.dat
```

Then do "source /root/.bash\_profile" to take it in account immediately.

To launch mame, do "gxname" for the graphical version or just « xname mygamename » for the command line version.



On first launch of gxgame, the application will create a small database from the missing/present games of your disk.

Example of game launching with xmame:

`xmame gridlee`

Launch MyhTV, then go into the menu « Utilities setup/Setup/Media settings/Game settings ». Then go to the "Game Player" menu, create a new "Xmame" entry and fill in the fields with the following:

```
Player name : Xmame
Type : MAME
Command : /usr/games/bin/xmame -fullscreen
Rom path : /usr/local/share/xmame/roms
Screenshots : /usr/local/share/xmame/snaps
Working Directory: (laisser vide)
File Extensions: (laisser vide)
[] Allow games to span multiple roms/disks (laisser décoché)
```

Quit with ESC, then ESC and click on "scan for games".

Then go back to main menu (ESC, ESC, ESC), and in « Media library/Play games » to enjoy MAME usage.

xmltv:

xmltv is a software that grabs TV schedules. Unfortunately the french version is not working (the website where it search is not found), so I could not use it.

To install it, first do the following to check the flags:

```
emerge -vp xmltv
```

Then to unmask and install it for real, do:

```
echo "=media-tv/xmltv-0.5.44 ~ppc" >> /etc/portage/package.keywords
Activation des flag "fr" (pour la France) et "tvcheck" (GUI) pour le package xmltv seul :
echo "media-tv/xmltv fr" >> /etc/portage/package.use
emerge xmltv
/usr/bin/tv_grab_fr -configure
mythfilldatabase
```

Note : « tv\_grab\_fr » is the french version. To be adapted consequently.

#### **1.5.6.6.4.1 MythVideo and Allocine**

Go into « Utilities setup/Setup/Media settings/Video settings/General settings »

In General settings, enter the directory name you have created before :

Directory that holds videos:

```
/mnt/store/dvd
```

Directory that holds movie posters:

```
/mnt/store/dvdpix
```

Then, do « next » twice, and fill in the fields with the following:

Commande to search for movies listings:

```
/usr/share/mythtv/mythvideo/scripts/allocine.pl -M
```

Command to search for movie posters:

```
/usr/share/mythtv/mythvideo/scripts/allocine.pl -P
```

Command to extract data from movies:

```
/usr/share/mythtv/mythvideo/scripts/allocine.pl -D
```

#### **1.5.6.6.4.2 Music settings**

Go into « Utilities setup/Setup/Media settings/Music settings/General settings »

In General settings, enter the directory name you have created before :

Directory to hold music:

[/mnt/store/music](#)

### **1.5.7 Backup of the system**

A script is available that can backup your system. A modified version of that script was used to create the backup script of the EZPegTV CD.

For more information, go to the following URL:

[http://blinkeye.ch/mediawiki/index.php/GNU/Linux\\_System\\_Backup\\_Script\\_%28stage4%29](http://blinkeye.ch/mediawiki/index.php/GNU/Linux_System_Backup_Script_%28stage4%29)

## 1.6 Installation of Linux Fedora Core on Pegasos II

### 1.6.1 Preparing

This chapter contains the information needed to install Linux Fedora Core 6 on a Pegasos II. As you will see thereafter, this distribution is quite easy to install. It was tested on a Pegasos II with an ATI RADEON 8500 graphic card.

Additional information can be found into the documentation provided on the Fedora web site at <http://fedora.redhat.com/docs/install-guide/fc6/en/>.

Now, go to <http://fedora.redhat.com/Download/>, download the ppc ISO DVD of Fedora Core 6 and burn it on a DVD.

### 1.6.2 Installation

To start the installation, insert your DVD, and on OpenFirmware enter the following command (after having pressed the ESC key to interrupt the boot sequence):

```
boot cd images/netboot/ppc.img
```

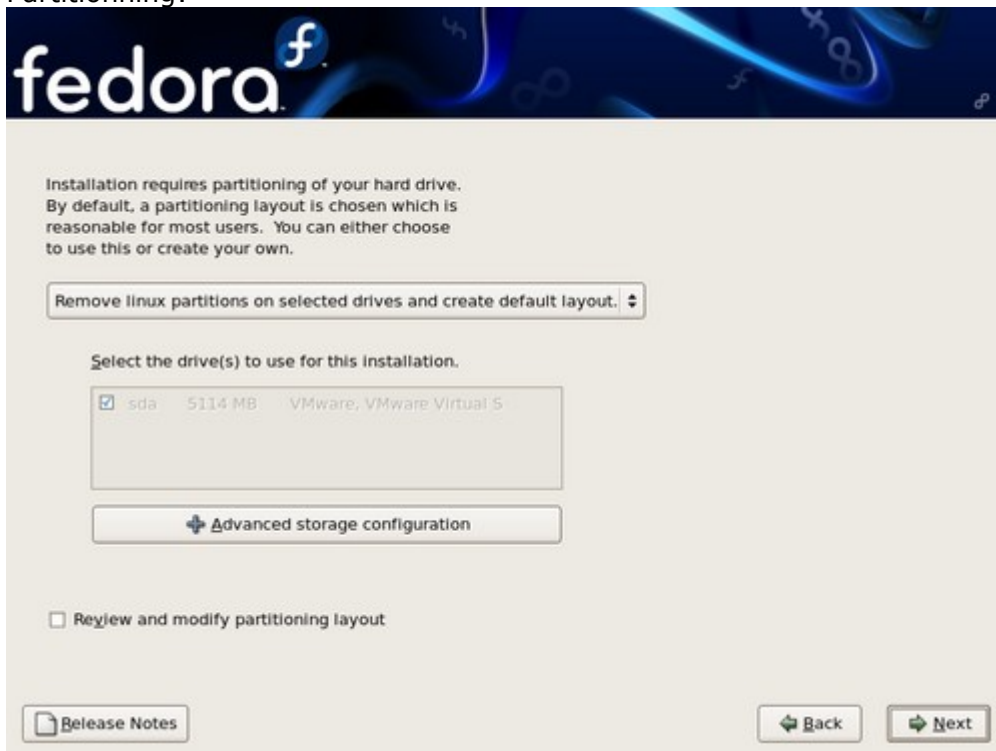
You will then have to select a few settings into the following screens.

Test CD media: on first install, it is recommended to test your media, but you can skip this step if you are sure of your DVD. After this step, the installation program will launch "Anaconda", the installation utility of Fedora Core.

Language selection: select your language.

Keyboard: select your keyboard.

Partitionning:



Here you have 4 choice :

- Use all space on selected drives and create default layout
- Remove Linux partitions on selected drives and create default layout (default choice)
- Use free space on selected drives and create default layout
- Create custom layout

On the "Advanced storage configuration" button, you can also add a SCSI or raid layout. The best is to select the custom layout (choice 4) and then click "Next". Choose a partition, click "Edit", format it in "ext3" and select the "/" mount point. Click "Next" again, then on "Format".

Network device:

Leave default setting (eth0, DHCP)

Hostname:

Select "manually" radio button and fill in a name for your computer. For example "pegasos".

Miscellaneous settings:

Leave default (grayed)

Timezone & clock:

Adapt those settings to your preferences.

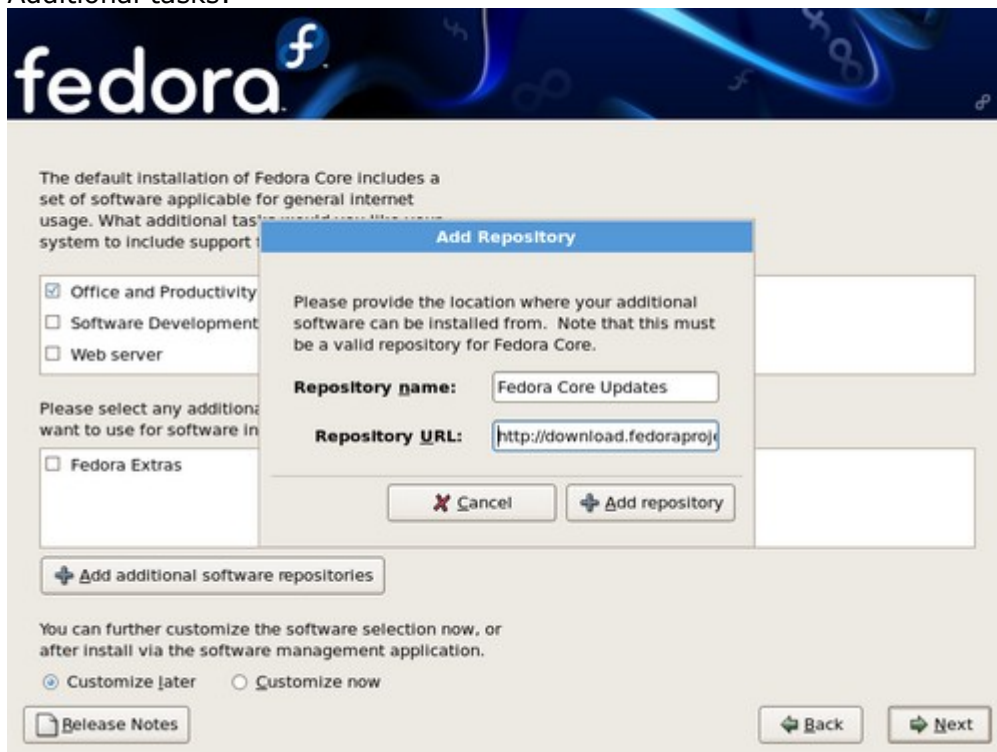
Root password:

Select a password and confirm it.

Retrieving installation information...

Just wait !

Additional tasks:



Select what you want. For example "Office and productivity". You can also check the "Fedora extras" option or add repositories.

Software installation:

Select "Customize later" (recommended) or "Customize now" if you want to use KDE instead of GNOME.

Click next twice to start installation.

The installation program will now install all the needed packages.

Once this step is finished you will have to reboot your Pegasos after having removed the installation DVD from the DVD drive.

On OpenFirmware, type the following commands, for example for /dev/hdb4:



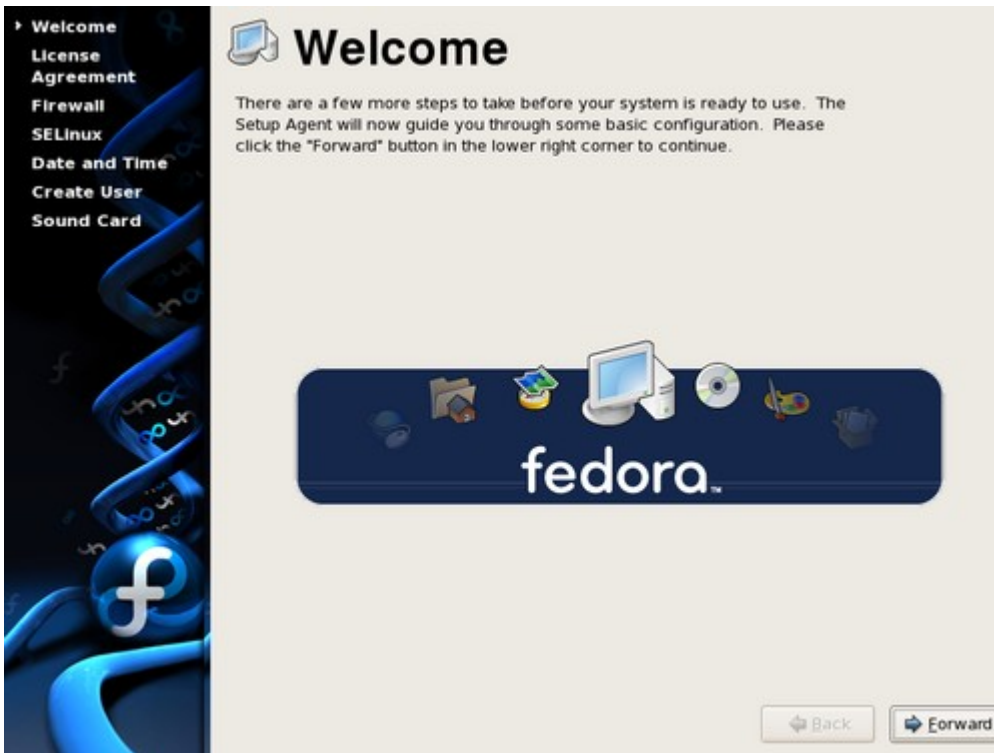
```
boot hd0:3 /boot/yaboot/yaboot conf=hd0:3,/yaboot/yaboot.conf
```

or for /dev/hda1 :

```
boot hd:0 /boot/yaboot/yaboot conf=hd:1,/yaboot/yaboot.conf
```

Note that the best is to add this command into your favorite boot loader (BootCreator or GRUB).

After the first boot, you should see the following screen:



You will have to change a few settings in the following screens. Go from a screen to another with the Back and Forward buttons.

Welcome : just click Forward

License : agree and click Forward

Firewall : here you can enable a software firewall and open/close the ports as you want. If you have a hardware firewall (for example a router/firewall), just click Forward.

SELinux : Security control : select a security level between "Enforcing", "Permissive" or "Disabled". You can also customize it to your needs by clicking on the "Modify SELinux Policy" arrow.

Date & time: Set your date and time manually, or use the NTP (Network Time Servers) to maintain an accurate clock using your Internet connexion.

Create user: create a user filling a username and a password. To use network services for authentication or user information, click on the "Use Network Login" button (NIS, LDAP, Winbind...).

Soundcard: the Pegasos's VIA VT82C686 AC97 onboard sound chip should be detected automatically. You can test it using the "sound test buttons".

You can then click the "Finish" button and Fedora should start for the first time: the following screen should appear.



### 1.6.3 Update

To update your system with the latest packages manually, use the Software Updater:

- Choose Applications / System Tools / Software Updater
- When prompted, enter the root password
- Review the list of updated packages. The package list displays a double arrow next to any updates that require a system reboot to take effect.
- Click Apply Updates to begin the update process.
- If one or more updates require a system reboot, the update process displays a dialog with the option to Reboot Now. Either select this option to reboot the system immediately, or Cancel it and reboot the system at a more convenient time.

## 1.7 Installation of Linux Yellow Dog on Pegasos II

Yellow Dog 4.0 is a very simple distribution to install and it includes a good number of essential tools for Linux. This distribution only works on Pegasos II with a RADEON graphic card.

### 1.7.1 Preparing

Download and burn Yellow Dog 4.0 (YDL) installation CD for Pegasos :

<http://stage.terraplex.com/~bill/freescale/ydl4-install.iso>

Prepare a partition for Linux Yellow Dog under MorphOS as explained in the Linux Debian part. You will however be able to do it during the installation of YDL by means of the "parted" tool.

### 1.7.2 Installation

Under OpenFirmware, press 'ESC' et enter the following command line at prompt :

```
boot cd ydl-install root=/dev/ram0 ramdisk=22000 rw
```

After kernel's load, the installation software asks you if you want to restore or install a system.

Press 1 to install Yellow Dog Linux 4

Press 2 to open a Rescue shell

Enter choice « 1 ».

Then, the software asks which is the peripheral on which the installation CD is.

/dev/hda = IDE 1, master

/dev/hdb = IDE 1, slave

/dev/hdc = IDE 2, master

/dev/hdd = IDE 2, slave

Example : `/dev/hdc`, if it's on the CD drive which is on master slot of second IDE connector.

In the same way, the software asks where is the disk on which you want to install Yellow Dog.

Example : `/dev/hdb`

The installation software, will then ask whether you wish to create partitions.

Would you like to partition `/dev/hdb`?

Press 1 to start partition tool (parted).

Press 2 to skip partitioning

If you already made this operation under MorphOS, press "2" to pass this stage, if not "1". In the case "1", to have the assistance on "parted" enter "h". "p" command makes it possible to display the current partitions table. Note your partitions must be formatted in EXT3. If they're not, do it now with the partition tool.

If you need a partition press "1". Even if you did this poeration in MorphOS, press "1" to enter in "parted" tool to format your partitions in EXT2 or EXT3 and in swap.

Partitioning a disk can cause loss of data, type "y" at the question :

All data on \$HARDDISK will be erased.

Continues (y/n)?

Once in "parted", press "h" to have the list of all the commands. The "p" command makes it possible to display the current partition table : it is of course empty if your hard disk is empty.

If you created your partitions under MorphOS, there's only need to format it in EXT2 or EXT3 and swap. Type "mkfs", then type the number of this partition (see the previous partition table help), input for example "1". Finally type "ext2" (or "ext3") for the filesystem. We also need

to format the swap partition : type "mkfs", then the number of this partition (for example "2", and type "linux-swap ". Now, your partitions are formatted.

If you did not create a partition yet, it is necessary to create at least one (the "swap" partition is indeed optional). Type "mkpartfs", then "primary", then "ext2" (or "ext3"). When it is necessary to enter the values for "start" and "end", check that these values do not overlap those shown in your partition table. Normally, if your hard disk is empty, you can enter values like "4" for "start, and" 20000 "for" end "in order to create a partition of a size of 20000 Mb (20 Gb). If you wish to create a partition of exchange (swap), type "mkpartfs", then "primary", then "linux-swap", then "20001", then "20500".

You can then type "p" to check the state of your partitions in order to know if they were created correctly and if they are formatted with the good filesystem.

Once the partitions are created, the installation software, will then ask on which partition number you want to install YDL (numberer beginning at 1).

Example « 1 » for « /dev/hdb1 ».

Before starting installation, the software will remind you all your previous choices :

```
Preparing to install
Source = /dev/hdc
Target = /dev/hdb1
Is this correct? (y/n)
```

Enter « y » to launch the installation on the disk.

At the end of installation, press on your box's reset button to restart.

To boot on your Yellow Dog distribution, press « ESC »

```
boot /pci/ide/disk@0,1:0 boot/2.6.8.ydl-freescale root=/dev/hdb1
```

Note : change /dev/hdb1 and disk@0,1:0 depending on your disk (IDE 0 or 1 interface, master : 0 or slave : 1, number of partition starting at 0).

You will also note that Linux and SmartFirmware do not address the numbers of partitions in the same way. For "hdb1", the digit at the end is the number of partition on the basis of 1. For "disk@0,1:0", the figure at the end is the number of partition on the basis of 0 ; the first two digits are the number of IDE interface (0 or 1) and a number telling if it's a master (0) or a slave (1). Here are some equivalences between SmartFirmware and Linux addresses:

```
/pci/ide/disk@0,0:0 -> /dev/hda1
/pci/ide/disk@0,0:1 -> /dev/hda2
/pci/ide/disk@0,1:0 -> /dev/hdb1
/pci/ide/disk@0,1:4 -> /dev/hdb5
/pci/ide/disk@1,0:0 -> /dev/hdc1
/pci/ide/disk@1,1:0 -> /dev/hdd1
```

The first starting on YDL will launch the utility "Kudzu" which is a utility launched at each start, and which makes it possible to automatically detect the physical changes on your computer (graphics board, keyboard, mouse or network interface...). In most of the cases validate the Kudzu choices. You can modify the choices by using the arrows, tabulation and enter keys.

You will then end your system setup with a succession of screen under X-Windows. Go from a screen to another by clicking on "Next" or "Previous" (return to previous screen).

Screen « Language ». Select english which is the only choice.

Screen « Welcome ».

Screen « License agreement ». Validate.

Screen « Keyboard ». Select keyboard of your choice.

Screen « Mouse configuration ». Select your mouse (PS/2, USB...).

Screen « Root password ». Enter a password for administrator.

Screen « Network setup ». If you use DHCP, leave the default choices.

Screen « Security Level ». Configuration of a firewall and its access services if needed. If you use a router/firewall, ignore this step.

Screen « Timezone ». Timezone selection.

Screen « Date and time ». Date and hour.

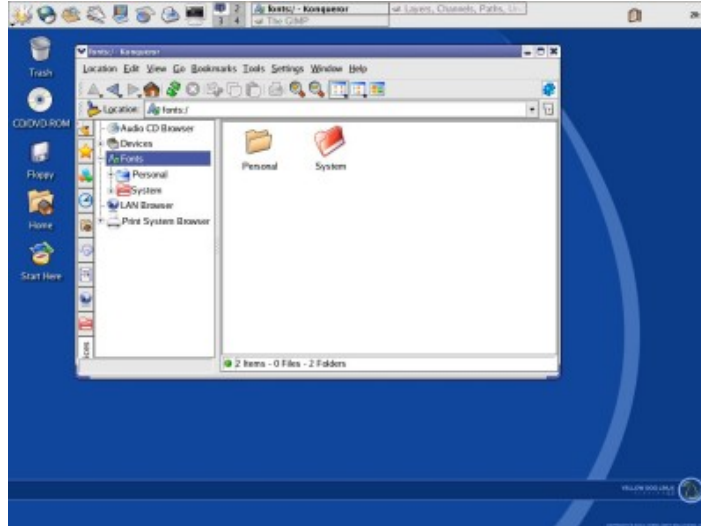
Screen « Display ». Select X-Windows screen resolution.

Screen « User account ». This screen makes it possible to create users by specifying a name of user, a complete name and a password.

Screen « Sound card detection ». Sound card detection. Leave the default choice if you do not have a PCI card.

Screen « Additional CDs ». Install additional CDs if you have any.

Screen « Finish setup ». End of installation.



Note : don't hesitate to use BootCreator (see. previously) to avoid having to enter this command at each boot on your Pegasos.

Example : note that it's almost the same command but with a "/" before the "boot" directory.

[SECTION]

Linux - YDL

`/pci/ide/disk@0,1:0 /boot/2.6.8.ydl-freescale root=/dev/hdb1.`

## 1.8 Installation of Linux Crux on Pegasos II

Crux PPC 2.0 is a Linux distribution targeting experimented users. The user has a total control among installation because almost nothing is automated. He must edit all the different configuration files and also compile the kernel to make this distribution works.

The goal of this distribution is to have a minimal set of tools in a recent and controlled version.

This distribution spirit is on the opposite side of Linux Ubuntu, Linux Suse or Linux Yellow Dog, in which a maximum of things are automated for an easy installation with a graphic interface.

This chapter will not enter into details and is targeting users with Linux knowledge.

### 1.8.1 Preparing

Download the ISO image of CRUX PPC 2.0 (crux-ppc-2.0.iso) for example <http://crux-fr.org/iso/crux-2.0/contrib/crux-ppc-2.0.iso> and burn it on a CDROM.

Under OpenFirmware, press on 'ESC' and type the following command on prompt : `boot cd boot/zImage-Pegasos root=/dev/cdroms/cdromX`, where X is the CD-ROM number where the CRUX PPC CD is inserted (0 ou 1).

Example : `boot cd boot/zImage-Pegasos root=/dev/cdroms/cdrom0`

After the kernel's load, log as root (without password).

Create if necessary partitions for CRUX PPC with « parted » tool (But the simplest is to do it under MorphOS as explained in Linux Debian installation chapter).

`parted /dev/discs/discX`

Example :

`parted /dev/discs/disc1`

Note : reserve at least 1Gb of disk space for CRUX system partition (the installation needs between 200 and 500Mb but it is always better to schedule a bit more for installing additional softwares).

Create then the EXT3 filesystems and the swap with the following commands :

`mke2fs /dev/discs/discX/partY`

`mkswap /dev/discs/discX/partS`

Where X is the number of the disk (0 or 1), and Y and S respectively the numbers of system and swap partitions (numbered from 1).

Example :

`mke2fs /dev/discs/disc1/part1`

`mkswap /dev/discs/disc1/part2`

Mount the partition where you wish to install the distribution :

`mount /dev/discs/discX/partY /mnt`

Example : `mount /dev/discs/disc1/part1 /mnt`

In case you would like to use other partitions (than swap), for example for mounting a partition in "var", do :

`mkdir /mnt/var`

`mount /dev/discs/disc1/part3 /mnt/var`

Enable then your swap partition :

`swapon /dev/discs/discX/partS`

Example : `swapon /dev/discs/disc1/part2`

## 1.8.2 Installation of packages

Type then the command « `/usr/bin/setup` » for starting the installation.

This command launches a graphic tool which will let you install the additional packages you may want. Choose at least all the “base” packages. If you are not sure, select all the packages, because in this distribution there is no dependencies checking between packages.

## 1.8.3 System configuration

Enter the following command which will let you set your new partition as a system partition to continue the installation process :

```
mount -t devfs devfs /mnt/dev
mount -t proc proc /mnt/proc
chroot /mnt
```

Enter a root password with the command « `passwd` ».

With “pico” editor, edit the file “/etc/fstab” to configure your file system (« `pico /etc/fstab` »).

If you have a system and a swap partition, you will only have to uncomment a few lines and change them by the right disk name :

Example :

#<filesystem>	<dir>	<type>	<options>	<dump>	<pass>
/dev/hdb1	/	ext3	defaults	0	1
/dev/hdb2	swap	swap	defaults	0	0
/dev/cdroms/cdrom0	/cdrom	iso9660	ro,user,noauto,unhide	0	0

Now, edit the file /etc/rc.conf in order to configure the keyboard type, the timezone and the language. Change for example HOSTNAME to “Pegasos”.

Edit the files « /etc/rc.d/net », « /etc/hosts » and « /etc/resolv.conf » to configure your network settings if you wish to.

The file /etc/hosts give the computer lists known by your Pegasos. Each line is with the pattern « <IP address> myhost.mydomain.org aliases »

Example :

```
127.0.0.1 localhost.localdomain localhost
192.168.0.1 www.my-domain.org myrouter
```

The file /etc/resolv.conf must contain your name server address (if you use any) and your domain. Enter your IP address with network mask in this file.

## 1.8.4 Compilation of the kernel

For compiling the kernel, enter the following commands :

```
cd /usr/src/linux-2.6.8.1
patch -p1 -i ../pegasos-2.6.8.1-cumulative-27092004.patch
make menuconfig
```

Menuconfig is a graphic tool where you will be able to configure all your kernel settings.

By default you can use a Pegasos dedicated file by doing “Load” and by typing « `../linux-2.6.8.1.config-Pegasos` » in the file load path.

Then launch the compilation in the following manner :

```
make all && make modules_install
```

Let the compilation be processed (long operation) and then copy the resulting files at the right place :



```
cp arch/ppc/boot/images/zImage.chrp /boot/vmlinuz
cp System.map /boot/System.map
```

### 1.8.5 Starting Crux PPC

To launch the distribution, enter the command : « boot disk@I,X:Y boot/vmlinuz root=/dev/hdZY+1 », where I is your IDE port number, X the disk number, Y the partition number starting from 0 and Z the letter corresponding to your disk (a=1, b=2 ...).

Example :

```
boot /pci/ide/disk@0,1:0 boot/vmlinuz root=/dev/hdb1
```

You are now under a very minimalistic distribution. To install new packages or update, you will need the « pkgadd » command.

Do a « `man pkgadd` » to know more about this command.

Warning, I recall that the dependencies between packages are not managed...

On the opposite, the packages removal is realized by the « pkgrm » command.

The « pkginfo » command, makes it possible to query the packages installation database.

At last, the « pkgmk » command makes it possible to create new packages.

## **2 MacOS X**

### **2.1 Running principles on Pegasos**

It is not possible to install MacOSX directly on a Pegasos as one can do it for MorphOS or Linux. There is however a way to install it via the application "Mac-On-Linux" or "MOL".

Mac-on-Linux is not an emulator to be strictly accurate but rather a virtual machine. Indeed, the microprocessor is not emulated and it's the machine one which is used natively. Only certain hardware specificities of Macintosh are emulated like audio interface, network interface, the firmware, etc... MOL is not besides specifically a virtual machine intended to run MacOS, but rather a generic virtual machine for PowerPC platform. A system of plugins/drivers then makes it possible to emulate specificities of a platform and a particular OS. Thus, one can theoretically make run any OS designed for a PowerPC platform supposed that the driver exists. The drivers existing are for the moment :

- MacOS (versions of MacOS up to system 9).
- MacOSX (versions of MacOS starting to 10).
- Linux.

You will have understood it, MOL will thus only work on a machine equipped with a PowerPC processor. Before the arrival of Pegasos and AmigaOne, there was hardly but only the Macintosh itself that could run MOL. That made it possible not to reboot under MacOS, when one was under Linux, for launching Mac applications. On our preferred machines, MOL takes it all its dimension.

For the experts, one can compare MOL with VMWare on PC which, following the example MOL on PowerPC, makes it possible to launch virtual X86 machines in order to make run any X86 operating system.

## 2.2 Installation of PegXMac on Pegasos II

PegXMac is a Live-CD created by David Benthams. Version 1.4.4 is now free (without support) and can be downloaded at <http://pegxmac.zftp.com/>. In 1.4.4 version, PegXMac contains Mac-On-Linux 0.9.70-17, a MacOS emulator for Linux which makes it possible to install MacOS 9 or MacOS X.

Version 2.0, which can be bought, is now based on Ubuntu and is not explained in this document. For a Mac-On-Linux install from Ubuntu, see the corresponding chapter.

Install of version 1.4.4 described there after was done on a Pegasos II but this operation is very similar on Pegasos I.

### 2.2.1 Pre-requisite

We need the following items :

- A Pegasos I or II, with G3 or G4.
- A least, 128 Mb of memory.
- A least a hard disk space of 6 Gb
- A Voodoo or Radeon graphic card.
- An IDE or SCSI CD or DVD drive.

### 2.2.2 Starting PegXMac

Once that you've downloaded the ISO of PegXMac, burn it with your favorite. On this CD, some significant files for this first stage are present : bootusb, bootcd, boothd and bootscsi. Copy these files in the boot partition of your MorphOS system (where your file boot.img is present). This partition must be in FFS.

Start again the computer and stop in OpenFirmware. Insert PegXMac CD in the CD-drive and type the following command :

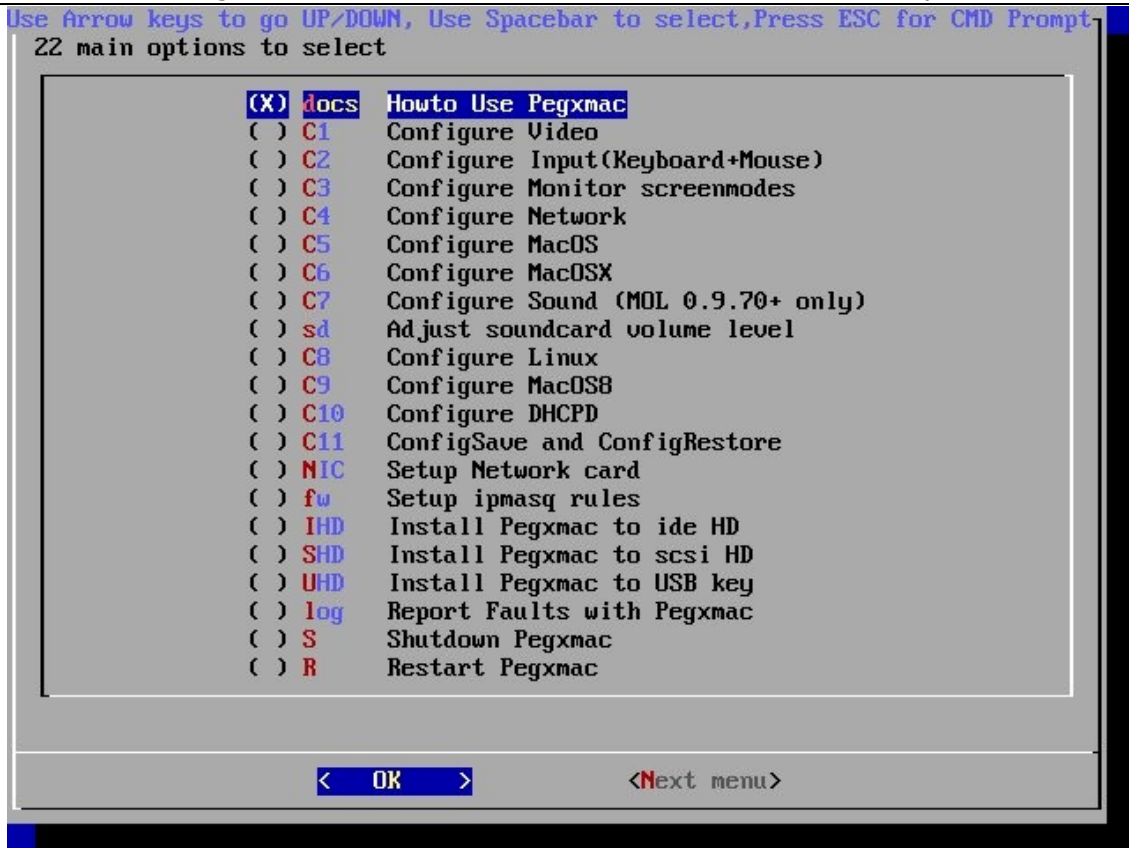
```
boot /pci/ide/cdrom bootcd root=/dev/ram rw init=/linuxrc video=tdfxfb:1024x768-8@60 lang=uk
```

The argument for the video must be changed according to your video card. If you have a Radeon (any model), replace "tdfx" by "radeon" in the command line. "1024x768" is the screen resolution. "60" at the end of the line, is the refresh rate (frequency) : it can be replaced by "75" for a better visual comfort. If you have a Gigabyte of memory or more, please add "mem=768M" at the end of the line because this quantity of RAM currently can make PegXMac crash. It should be noted that another argument is possible here, for example "lang=fr" will put the interface of Mac-On-Linux in French. Let us keep for the moment "lang=uk" as the other languages translation may be corrupted. Lastly, the owners of Pegasos I can add the argument "ide=nodma" if they encounter problems of corrupted data.

To summarize, here a command line example for a Pegasos II user with 1 Gb of memory, Radeon and for a display in 1024x768 to 75 Hz :

```
boot /pci/ide/cdrom bootcd root=/dev/ram rw init=/linuxrc video=radeonfb:1024x768-8@75 mem=768M lang=uk
```

Once that you typed your command line, press on "Enter" to validate. A minimalistic Linux is launched then. It displays a few text lines of configuration and recognition of peripherals. At the end of about thirty seconds, it arrives at MOL-Menu, the small interface to configure Mac-On-Linux.



### 2.2.3 Configuring Mac-On-Linux

Here are a few information on navigating in MOL-Menu : move with the arrows, select with the space bar and validate with the "Enter" key.

Now, it is necessary to copy PegXMac somewhere : IDE or SCSI hard disk or even a USB key. We will choose the option to copy from an IDE hard disk. Select the "Install PegXMac to HD" option and validate. You are now in the hard disk installation tool.

For recall, Linux recognizes IDE peripherals in the "HDxy" form, where "x" is the number on the IDE chain (a, b, c or d) and "y" is the possible number of the partition on a hard disk. For example, the 3rd partition of my main hard disk on first IDE port will be recognized as "hda3".

Click on "OK" to continue. Select a hard disk on which you have more than 6 Gb free. In my case, I have two hard disks and I wish to make the installation on my slave disk (recommended if possible), I thus put "hdb". It then shows to you, the total capacity of the disk, press on "OK".

You are now in Amiga-FDisk, a Linux tool of hard disk partitioning which recognizes the Amiga file systems. Press on "m" to see the possible commands. We will create, in order, 4 partitions : one for Linux, one for the swap, one for the MacOS system and another one for the Mac applications :

Type "n" then "1", low cylinder to "2", high cylinder to "300".

Type "n" then "2", low cylinder to "301", high cylinder to "450".

Type "n" then "3", low cylinder to "451", high cylinder at "1700".

Type "n" then "4", low cylinder at "1701", high cylinder to "8000".

1, 2, 3 and 4 indicate the numbers of partition. As I am on the slave hard disk, the partitions will thus be recognized by Linux under the names : hdb1, hdb2, hdb3 and hdb4. The "cylinders" determine the size of the partitions. On my configuration, that gives respectively 1.4 Gb, 0.7 Gb, 5.9 Gb and 30 Gb. The size of the cylinders depends on the hard disks models, in all the cases, ensure that the first partition (Linux) makes more than 1.1 Gb, the second partition (Swap) makes more than 128 Mb and that the cumulated size of the three first does

not exceed 8 Gb (the first MacOSX partition must be in the first 8Gb). The size of the 4th partition has only little importance : the larger it is, the better it is. Lastly, you have to know that you can, according to your needs, add other partitions but that is not useful in our example.

It is now necessary to give a file system to each partition :

Type "t" then "1" to select the first partition, then "11" to put it at the Linux format.

Type "t" then "2" to select the second partition, then "12" to put it at the Swap format.

Type "t" then "3" to select the third partition, then "11" to put it at the Linux format.

Type "t" then "4" to select the fourth partition, then "11" to put it at the Linux format.

Your 4 partitions are then created with the adequate file system. Press on "w" to save and then return to the procedure of installation of PegXMac.

It is asked to you then if you want a swap partition. Select "No" if you do not want any or "Yes" in the contrary case. For my part, with 512 Mb of memory, I do not put any. If you want such a partition, select your partition of swap then.

Then, select the root partition. In my case, it is "hdb1".

With the "EXT3" file system.

Answer "Yes" when asked to format in EXT3.

The copy will start.

Press on "OK".

After a few moments, one asks you to choose the partition of boot.

In my case it is "hdb1".

Then "EXT3".

Then "hdb1" (in order to copy the "boothd" file).

Press "OK".

Then "Yes".

Leave the name "PegXMac".

For the network, press "No", leave the configuration network by default.

Press on "OK" in the window "Finished".

You can start again PegXMac, but this time, it is on hard disk. In OpenFirmware, type the following command line which will make it possible to launch PegXMac from your hard disk :

```
boot boothd root=/dev/hdb1 video=tdfxfb:1024x768-8@75 lang=uk
```

Refer to the paragraph "Start PegXMac" to read again the possible arguments according to your configuration.

## **2.2.4 MOL-Menu : additional configuration**

Arrived in MOL-Menu, we will continue to make some adjustments for the memory and the recognition of the hard disks.

In "Configure screenmodes", it's wise to test the various video modes supported by your video card and your monitor. This to avoid to be "off frequency" if one the mode is not supported by your hardware. Enter this menu, and test the various proposed modes, this will exclude unsupported modes from your settings.

In "Configure MacOSX".

Go on "Configure MacOSX". You arrive in Nano, a nice text editor. Seek the line "ram\_size" in order to change the value "128" if you have more than 128 Mb of memory. This change is advised because the use of MacOS X will be more comfortable. For my part, having 512 Mb on Pegasos, I allocate 384 Mb for MacOS X. Of course, if you have only 128 Mb on your computer, it will be necessary to lower this value to 64 or 96 Mb (what is really limiting under MacOS X !).

Always in Nano, seek the line "`blkdev: /dev/cdrom`". Uncomment this line. Then, below this line, add "`blkdev: /dev/hdb3 -rw -force -boot`" (without the quotation marks). That will make it possible for Mac-On-Linux to mount the partition hdb3 which we created a few time ago.

Seek the line "`blkdev: /dev/hda6`" and replace "hda6" by "hdb4". That makes it possible to mount the partition "hdb4" which we also created a few moments ago. It should be noted that, by default PegXMac 1.4.3 proposes a partition "hda6" in the MacOSX configuration. That can be dangerous if this partition exists already (example: if it is one of your partitions under MorphOS) because you can lose your data. Here, we replaced it by another partition which, is dedicated to Mac (hdb4).

Lastly, if you have a second CD/DVD reader (as a slave on second IDE slot), add the line "`blkdev: /dev/hdd`", there still below "`blkdev: /dev/hdd`".

Save your modifications while pressing on the keys "Ctrl-o" then "Enter" and finally "Ctrl-x" to leave the editor.

In "Configure Linux":

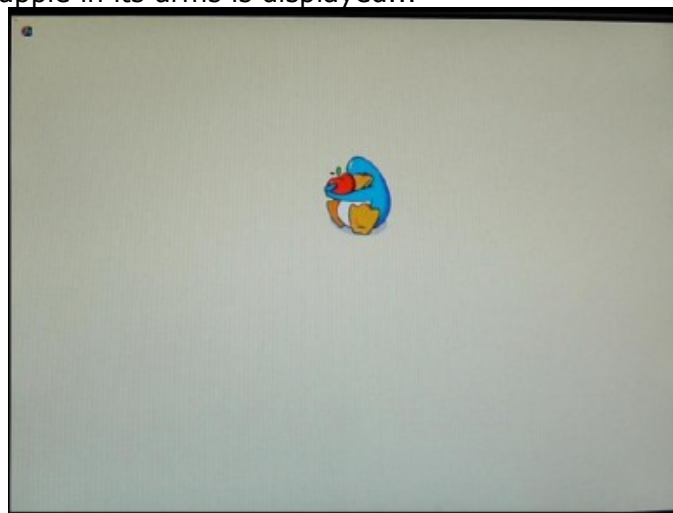
Go in "Configure Linux". Here, search the line "ram\_size". The value beside, must be equal or lower than the remaining memory that than you entered for MacOS. With 512 Mb on the whole, less the 384 Mb of MacOS, it remains to me 128 Mb for Linux. I can thus put between 0 and 128 Mo, I choose 128. If you have 128 Mb on the whole, put 32 or 64 Mb. Save your modifications while pressing on the keys "Ctrl-o" then "Enter" and finally "Ctrl-x" to leave the editor.

## 2.2.5 Installation of MacOS X

In MOL-Menu, go on "Next Menu". Insert CD #1 of MacOS X then select "Start MacOSX install" in order to start the installation.



A penguin holding an apple in its arms is displayed...



At the installation page of MacOS X, select your language then, in the menus, choose "Disk utility". This is mandatory to make MacOS recognize the two partitions which we created for it (hdb3 of 5.9 Gb and hdb4 of 30 Gb). In this disk utility, the two partitions should be formatted. Select one of the partitions and click on the function "delete". Give it the name that you want (for example "machd0" and "machd1"). It should be noted that you can format



starting from the option "Partitioning": create then only one partition in each partition and the disk utility will format them.

Leave the disk tool once the formatting finished. Leave also the MacOSX installation so that the formatted partitions are recognized.

You are again in the MOL-Menu. Do "Next Menu", then "MacOSX Install" to start again the installation process. This time we are going to install MacOSX for real.

Click on your language.

Press on « Continue » on each page.

For the choice of the installation partition, select the 5.9 Gb disk (no other partition is possible but MacOS X wants absolutely a partition created in the the first 8 Gb of the hard disk).

Before launching the installation, you can proceed to a personalized installation of the system. That is useful to gain some little space. Click on "Customize" and uncheck unwanted languages and printer drivers.

Click on "Continue": the installation starts.

MacOS X starts again at the end of the first CD.

You are again in MOL-Menu, insert CD #2 of MacOS X.

Start again the installation ("Next Menu", then "Start MacOSX Install"). It is possible that the 3rd CD may be necessary. In this case, start again the installation ("Next Menu ", then "Start MacOSX Install"). In such case, it is necessary to start again MacOS X a last time.

At the end of the installation, a few questions will be asked to you. Just answer to them.

We are now under MacOS X Note that on the desk, a "MOL-Install" icon is present. It is the last step of the installation, that will initialize the sound and the network under MacOS X. Double-click on this icon then on "MOL Package.pkg".

Installation is done !



Now, for launching MacOS X, it is enough to enter the following command line from OpenFirmware :

```
boot boothd root=/dev/hdb1 video=tdxfb:1024x768-8@75 hdd=scsi hdc=scsi lang=uk
startmacosx
```

Defer you to the paragraph "Start PegXMac" to read again the possible arguments according to your configuration. The "startmacosx" at the end will enable you to launch MacOS X directly, without having to touch the menu of Mac-On-Linux. If you withdraw this argument, you will go to MOL-Menu quite simply. Such a command line to enter each time is unpleasant, it is for that that I advise you to use the BootCreator tool which will create you very easily to use multiboot menu.

## **2.2.6 Troubleshooting**

### **2.2.6.1 CD / DVD problems**

*The PegXMac CD does not function or has read errors.*

Download the ISO again and burn it on a good quality CD.

*My second CD/DVD drive is not recognized.*

Add the line "`blkdev: /dev/hdd`" according to the position of your drive in the "molrc.osx" file which is accessible in MOL-Menu by the "configure" option.

*Under MacOS X, CD/DVD ejection is not possible.*

Add to your command line the argument "`hdx=scsi`" (where "x" represents a, b, c or d according to your IDE configuration).

### **2.2.6.2 Partition problems**

*My hard disk partitions are not recognized.*

Add "`-force`" and "`-rw`" to the blkdev line in the "molrc.osx" configuration file which is accessible in MOL-Menu by the option "Configure MacOSX". That will force the detection of your partitions.

### **2.2.6.3 Error message**

*I have an error of type « SQUASHFS error: etc. ».*

Use only one memory bar and lower or equal to 512 Mb. Put also the argument `mem=512M`.

### **2.2.6.4 Mouse problem**

*The wheel does not work.*

Add to your command line the argument "`wheelmouse`". This argument does not work with everyone's mouse.

*My PS/2 mouse does not work.*

Add to your command line the argument "`rpci=irqmask=0x0e98`". If that does not still work, connect a USB mouse.

### **2.2.6.5 Network problem**

*I do not have the network.*

If you use MacOS X, take care that the line "`devnet: tun${session} -tun`" is present in your molrc.net file which is accessible via the menu "Configure Network" in MOL-Menu. Erase or comment on the three lines with Sheep (If etc. until the closing bracket to the 3rd line). If you would use MacOS 9, rather put the line "`devnet: eth0 -sheep`", and delete/comment the line with Tun.

Note: this problem is rare and is corrected by version 1.4.3 of PegXMac. The configuration of the network is simplified and most of the time it does not require any adjustment.

### **2.2.6.6 Graphic problem**

*The colors are completely weird.*

This is a MOL known problem of « endianness ».

It occurs with 3DFX Voodoo 5 graphics boards in 32bits or "thousands of colors" and with ATI RADEON having a GPU RV280 in "thousands of colors". There is nothing to do from your point of view. In the case of the RADEON it is not too annoying and in the case of Voodoo 5, if you have a Pegasos II, I recommend to you to change it for an ATI RADEON instead.

## 2.3 Installation of MacOS X on Pegasos II from a minimal Linux Debian



### 2.3.1 Introduction

This chapter proposes a complete installation of Mac On Linux without PegXMac thanks to a minimum Linux Debian installation. This chapter was validated on Pegasos II but should suite to a Pegasos I (> April1, however you have to apply the previous trick to use a valid MAC address).

### 2.3.2 Installation of a minimal Linux distribution

See the previous Installation of Linux Debian chapter, with the following alternatives :

- only use a partition of 1Gb (what is definitely sufficient for a minimal installation)
- choose ftp method for "apt" (on basic installation)
- do not install any additional modules with "Aptitude" and just quit this utility ("q").

The installation of Linux should now be finished. On "login" prompt, identify yourself as "root" with the required password which you created previously.

### 2.3.3 Installation of Mac-On-Linux

Before beginning the installation of Mac-On-Linux, we will add new repositories sources in the description file of the available sources of programs, if not we would not have access to the specific MOL drivers for MacOS-X.

Type de following command :

```
pico /etc/apt/sources.list
```

This opens a text editor. At the end of the second line - that which starts normally with "deb ftp..." - add "non-free contrib".

Type Ctrl-x to leave, "O" to confirm and "Enter" to validate.

It is now necessary to inform the Debian's installation system of the presence of new downloading sources by typing the command :

```
apt-get update
```

After a certain time, the system will give back control to you. We are now ready to install MOL. First of all, you have to know that it is possible to run MOL under X-Windows, the graphic environment of Linux, in windowed mode. This operating mode would be much more complicated to install and would not bring much to you if you are not much interested by Linux. In our case, MOL will be launched in "Frame-Buffer" mode in full screen.

To download MOL, just type :

```
apt-get install mol
```

Then, it should be known that to work, MOL requires a "module" whose version must match the version of the Linux kernel used. You can see the precompiled modules at your disposal thanks to this command :

```
apt-cache search mol |grep modules
```

For example, the linux kernel installed might be the 2.6.8 (you can check the version of your kernel by typing "uname -a"). You must install the module corresponding to your kernel version by typing a command as follows :

```
apt-get install mol-modules-2.6.8-powerpc
```

If no precompiled version matches your kernel, you will have either to compile the module by yourself (for that get the sources with apt-cache search sources, apt-get install sources), change the kernel, or find the precompiled module for your kernel on Internet. This last solution will undoubtedly be the simplest.

The last thing to be recovered is the specific drivers for OSX. To see the list of the precompiled drivers available, just type :

```
apt-cache search mol |grep drivers
```

The one we are interested in is the MacOSX one. Let's type the command :

```
apt-get install mol-drivers-macosx
```

Before being able to even launch MOL itself, there remain two or three small things to be made. Initially, it is necessary to create a disk which will be used for the operating system MacOS. There are three solutions for that :

- creating a file which will be used as virtual disk for Mac. We will not argue about this solution whose performances are generally bad.
- creating a partition under Linux which will be used as a disk for Mac if you haven't done it before
- allocating a complete physical disk for Mac.

If you choose the second solution, and if you did not already created it at the Linux installation time, it will be necessary to create a partition under Linux of an unspecified type (EXT3 for example) thanks to the tool "cfdisk" provided with the distribution. Another tool making it possible to manage partitions with the Mac format directly is available : "mac-fdisk", but I did not test it. "cfdisk" takes as argument the device corresponding to the disk which one wants to manage. Under Linux, the main disk of first IDE slot is named /dev/hda and so (/dev/hdb, /dev/hdc and finally /dev/hdd for the slave disk of the second IDE port). If you wish to create a partition on the main disk of second IDE slot, you will thus type:

```
cfdisk /dev/hdc
```

If you chose the third option, then there is no partition to create on your dedicated disk to MacOS for the moment.

We now will setup MOL configuration files.

Start by editing the general setup file for MacOS-X:

```
pico /etc/mol/molrc.osx
```

Insert on the line "ram\_size" the quantity of memory which you wish to allocate to MacOS X. Per default this value is set to 96, but it is too low for MacOS X. Set at least 128 or 256. Approximately 128Mb should be left with Linux if one does not want it to "swap". Put then in comments the line "blkdev: -cd \${cdboot }" by adding "#" in front. Then, just after this line, add :

```
« blkdev: /dev/hdxy -rw -force -boot », if you chose to use a partition as a Mac disk  
Replace "x" in the "hdxy" string by your disc (a, b, c or d) and "y" by the sequential number of your partition (1, 2, 3...).
```

or

```
"blkdev: /dev/hdx -rw -force -boot -whole" if you chose to use a whole disc for Mac. Replace  
"x" in the "hdx" string by your disk (a, b, c or d).
```

As previously, leave by Ctrl-x then "Y" then "Enter".

We will make now MOL correctly recognize the CD drive by activating SCSI emulation of Linux. For that, edit the file `"/etc/modules"` with `"pico"`, and to add the line `"ide-scsi"`. As previously, leave by `Ctrl-x` then `"O"` then `"Enter"`.

Let's now setup the network by editing the file `"/etc/mol/molrc.net"` :

```
pico /etc/mol/molrc.net
```

There are three types different of network drivers for MOL. The simplest solution to is to enable the `"sheep"` driver. For that, add the following line in the file:

```
netdev: eth0 -sheep
```

As previously, leave by `Ctrl-x` then `"O"` then `"Enter"`.

To make the sound work, we now need to install `"ALSA"` drivers. Type in the command :

```
apt-get install alsa-bases
```

then

```
alsaconf
```

That makes it possible to launch the ALSA setup. Let you be guided by its interface and select the sound card you wish to use in the list of the recognized cards. You can then edit the file `"/etc/mol/molrc.sound"` to adjust settings there, but normally that should be working without modifying it.

A last thing has to be done before finally being able to boot on MacOS's CD : you have to setup the video modes supported by your monitor. For that, it is necessary to launch the tool :

```
molvconfig
```

Let you be guided by the questions. Press on a key to validate each mode which is displayed correctly (gradient colour). Avoid testing the modes for which you know that your monitor will not function.

We are now going to proceed to a last reboot so that MOL can `"see"` its newly installed modules and add a parameter on Linux's launching command line. Type :

```
reboot
```

Start Linux again with the command line from the firmware but by adding `"hdx=scsi"` (to be adapted to `"x"` value of your CD drive). If you have several CD drives, burners etc... add as many `"hdx=scsi"` as necessary. Re-log as `"root"`.

## 2.3.4 Installation of MacOS X

Insert the first MacOSX's CD in your Pegasos CD driver. You must, I suppose, legally have a Mac to have the right to do this operation. You can for example use the `"MacOS X 10.3 - Panther"` version composed of 3 CDs.

Launch the following command :

```
startmol -X -cdboot
```

If everything is all right, you should soon see appearing a small Penguin lovingly tightening an apple in its arms. A few moments later, you will reach the first screen of the installation program of MacOS X, while the system continues to read your CD. Let you be guided through the screens (which I will not detail) until the program requires that you select the `"installation volume"`. No volume must appear in the list except if you perhaps used `"mac-fdisk"` for partitioning previously. To launch the MacOS's partition tool, in the `"Install"` menu, select `"Open tool disk"`. Select in the left-hand column the volume which corresponds to the disk that you allocated for the Mac and click on `"Partition"`. Be careful, the MacOSX's boot partition must be in the 8 first Gb of the disk ! If needed, create 2 partitions. Leave the partition tool. It is necessary to start again the MacOSX installation so that the disk is definitively recognized, which you can do with the option `"Quit installation"` of the `"Install"` menu.

Start again MOL with the same command line as before.



When you are under MacOS, you can always switch to the Linux console from where MOL was launched, thanks to the combination of key "Ctrl-Alt-f1" or on another console thanks to "Ctrl-Alt-f2", etc. To go back to MacOS, type "Alt-f7". When you are on the console which launched MOL, you can kill MOL processes by typing twice "Ctrl-c" keys.

## 2.4 Installation of MacOS X on a Pegasos II from Linux Ubuntu

### 2.4.1 Preparing the system

Install Linux Ubuntu distribution Linux if it's not already done (see previously) and log under Ubuntu.

To install Mac-On-Linux, you will have to enable Mac-on-Linux "repositories" which are "universe" and "multiverse".

You will then have to compile mol.ko kernel by yourself because there are no binary packages of these modules available for Ubuntu.

To enable the required repositories; edit the `/etc/apt/sources.list` file with "pico" editor by typing the following command line :

```
sudo pico /etc/apt/sources.list
```

Note : if you are logged as root, use of « sudo » command is not necessary.

In this file, change the following lines :

```
#deb http://archive.ubuntu.com/ubuntu warty main restricted universe
#deb-src http://archive.ubuntu.com/ubuntu warty main restricted universe
```

as follows :

```
deb http://archive.ubuntu.com/ubuntu warty main restricted universe multiverse
deb-src http://archive.ubuntu.com/ubuntu warty main restricted universe multiverse
```

Quit and save your file ("Ctrl+x", "o").

After editing, you'll need to resynchronise your package index files with the command :

```
sudo apt-get update
```

Now you can install all the needed packages with the command :

```
sudo apt-get install mol-modules-source build-essential linux-headers-$(uname -r)
```

Here is a short overview of the necessary packages:

mol-modules-source : This package contains the source for the mol kernel modules

build-essential : a metapackage that depends on the applications and tools necessary to build Ubuntu packages.

linux-headers : Linux headers

Type "`uname -r`" and it will let you check if your kernel and headers versions match.

### 2.4.2 Compilation of MOL modules

Before building the package you need to make sure that "debhelper" and "m4" packages are installed. If you don't the build will fail.

So, to be sure type the following command :

```
sudo apt-get install debhelper m4
```

The package mol-modules-source adds a mol-modules.tar.gz file in the directory `/usr/src`. Unpack the file with the commands :

```
cd /usr/src
sudo tar xzvf mol-modules.tar.gz
```

Then, some environment variables must be set. These are needed for the compilation.

```
export KVERS="$(uname -r)"
export KSRC="/usr/src/linux-headers-$(uname -r)"
export KEMAIL="your@email.address"
export KMAINT="Your Name"
export KDREV="ubuntu0"
```

You should, of course, replace [your@email.address](#) and Your Name with your actual email address and your name.

Use the following commands to actually build the modules:

```
cd /usr/src/modules/mol
sudo debian/rules build
```

The building is then executed. Once done, you have to create a Debian package of the modules with the command :

```
sudo debian/rules binary-mol-modules
```

This will result in creating a debian package `mol-modules-2.6.12-9-powerpc_0.9.70+ubuntu0_powerpc.deb` in the directory `/usr/src`.

### 2.4.3 Installation of MOL

The first package you need to install are the kernel modules you just built. This is because they provide the virtual package `mol-modules` on which the `mol` package depends.

```
sudo dpkg -i /usr/src/mol-modules-2.6.12-9-powerpc_0.9.70+ubuntu0_powerpc.deb
```

Warning : the specified name in this command must be the filename you just created. This name matches the one of December,1st 2005.

Now you can install the other MOL packages :

```
sudo apt-get install mol mol-drivers-linux mol-drivers-macos mol-drivers-macosx
```

Here is a brief overview of the necessary packages :

`mol` : the actual Mac-on-Linux emulator.

`mol-drivers-linux` : drivers needed to run Linux within the emulator.

`mol-drivers-macos` : drivers needed to run Mac OS 9 and earlier within the emulator.

`mol-drivers-macosx` : drivers needed to run Mac OS X within the emulator

### 2.4.4 Running Mac-On-Linux as a normal user

By default, Mac-on-Linux requires root privileges to run. The following command makes sure you can run Mac-on-Linux as a normal user.

```
sudo dpkg-statoverride --update --add root root 4755 /usr/lib/mol/bin/mol
```

## 2.5 Configuring and running Mac-on-Linux

This assumes that you have a working install of Mac OS X or 9, on a partition on your harddrive somewhere.

There are a few settings that must be modified before launching MOL.

#### 2.5.1.1 RAM memory

The default is 96mb, which is barely enough to boot Mac OS X (although ok for Mac OS 9 if you aren't planning to do much).

Edit the file `/etc/mol/molrc.osx` with `pico` (« `pico /etc/mol/molrc.osx` ») and find and replace "ram\_size: 96" by at least "ram\_size: 128" if you have 256Mb of RAM or even by "ram\_size: 384" if you have 512Mb of RAM.

It's possible to give mol more RAM than you physically have, thanks to swap but is really not recommended as it will slow down the system and make it unstable.

#### 2.5.1.2 Video modes

Run `molvconfig` :

```
sudo molvconfig
```

In this tool, you basically have to type 'y' to try out a configuration. You will then see a nice gradient on the screen. The red screen comes out as a stripes, but it seems to work anyway. If a mode does not suit then press 'n'.

### 2.5.1.3 Launch of MacOSX and end of install

To launch MacOSX, type "startmol -osx" or "starmol -X" and for MacOS9 "startmol". You will only need to install the MOL package that you find on the desktop. This provides sound and network drivers.

To exit mol, choose "shutdown" from the Apple menu.

## 2.5.2 Configuring Network

You will need to install additional package to have the network under MOL. For that, you will have to install "ipmasq", "dnsmasq" and "dhcpcd" packages using synaptic.

For that, type the following command :

```
sudo apt-get install ipmasq dnsmasq dhcpcd
```

Then enable dhcp for the tun0 interface that mol uses, in /etc/default/dhcp, put the line. For that type :

```
sudo pico /etc/default/dhcp
```

And in the edited file (which might be a new one), add « `INTERFACES="tun0"` ».

The same way, with pico, in /etc/mol/molrc.net add the line :

```
netdev: tun0 -tun
```

and in /etc/mol/tunconfig at the bottom just before "exit 0" add the lines :

```
/etc/init.d/ipmasq restart
```

```
/etc/init.d/dnsmasq restart
```

Now run "startmol -X". It is now possible to configure the network on the Mac OS X side. MOL creates a network tunnel called tun0 at the linux end, and en3 on the Mac OS X end.

Once you have logged into to Mac OS X open up the "system preferences" and make sure that "en3" is set to configure with DHCP.

Now open up Safari or Firefox, and you should have a web connection.

## 2.5.3 FAQ

If you get « mol module not found" when starting MOL, and you've just made a build and an install following the previous steps, try uninstalling it by hand with the following command line:

```
insmod /lib/modules/2.6.(your kernel version)-powerpc/misc/mol.ko
```

Example :

```
insmod /lib/modules/2.6.12-9-powerpc/misc/mol.ko
```

If OSX doesn't start, edit the blkdev settings in /etc/mol/molrc.osx :

Your OSX boot partition must not be mounted in Linux and must be declared as follows :

```
blkdev /dev/hdb1 -boot -rw
```

Of course, adapt "hdb1" to your MacOSX partition name.

## 2.6 Installation of MacOS X with MolK

This chapter details the installation of MacOS X with MolK by Marcel Unbehaun (aka Frostwork). It is equally likely, that MacOS 9 can be installed by following this howto guide.

MolK represents a free and easy to use alternative.

This LiveCD will allow you to launch "MOL" giving support for CD/DVD-RW-RO, USB, and AC97 in MacOSX Tiger .

### 2.6.1 Installation

The required files can be downloaded from <http://obligement.free.fr/files/mol.k.lha> or the torrent for this file can be found at <http://www.pegasos.org/downloads/torrents/mol.k.lha.torrent> (about 7MB).

Once extracted, copy all of the contents of the archive in your folder containing the boot image (boot.img, etc.). By past experience, it is preferable that this boot image be the in the first partition of your hard disc (hda1 or hdb1) and that it is formatted as FFS. The files will take up a maximum of 14 MB.

MolK is now installed and ready to be used. Upon a reboot, a simple command "boot molk" in the openfirmware interface suffices to start up under MolK. You will be able to install MacOS after some necessary configuration of MolK.

### 2.6.2 MolK setup

#### 1. Preparation of a dedicated partition for MacOS X

Once again under MorphOS, prepare a partition in the first 8 GB of your harddrive. For simplicity's sake, I chose to install MacOS X on the secondary harddrive (hdb). Once again, it is necessary to think about installing a boot partition in first 8 GB of the physical volume. The simplest option is to therefore create a partition of 8 GB on the primary harddrive.

#### 2. Presentation of the configuration files of MolK

In the partition where you copied the content of the archive of MolK, you doubtless have noticed a folder named "mol". As in Linux, this folder contains all the necessary configuration files to produce a functioning install of MacOS. But this time, you will be able to modify them with your favourite text editor.

#### 3. General configuration of MolK OSX (file; "molrc.osx")

The first file to edit, but which is also the most important is "molrc.osx". Commence forthwith by modifying the quantity of memory available to MacOS X. For example for 512 MB, one can allocate 384 MB by default to MacOSX. It also will function if 448 MB is allocated without any problems.

A little lower in the file, you will be able to see a series of lines starting with "blkdev: hdx". My dedicated installation partition being in second partition on my second hard disc, I have therefore modified the file to have this line:

```
blkdev: /dev/hdb2 -rw -boot -force
```

For "hdb2", "b" signifies the second harddrive and "2" signifies the second partition. I would expect users to adapt this line to their hardware configuration. The parameters "-boot" and "-force" allow the user to define the partition of MacOS X as bootable and force the system to use this partition. This file is almost completely configured. As you can see on the screen capture, I have also used another partition to store my data. As this partition is FAT32, it allows me to exchange data with the other Operating Systems installed on the Pegasos.

```

-- perl -- (for emacs syntax coloring)
#
# /etc/mol/molrc.osx
#
# Mac-on-Linux master configuration file for MacOS X booting
#

-----
# Parse other config files (these files are in /etc/mol/)
#
-----
include      ${etc}/molrc.video    # video configuration
include      ${etc}/molrc.input    # mouse and keyboard
include      ${etc}/molrc.net      # networking
include      ${etc}/molrc.sound    # sound

-----
# Misc Settings
#
-----
ram_size:          384
disable_altivec:   no

-----
# OS X Volumes
#
-----
#
# Volumes to be mounted in MacOS X are specified through the
# blkdev keyword:
#
#      blkdev:      <device>  [flags]
#
# where <device> can be a complete disk (/dev/hda), a single
# partition (/dev/hda?) or a disk image.
#
# The following flags are supported
#
#      -rw/-ro      read-write / read-only [default]
#      -force       force usage of device (be careful!)
#      -boot        boot from this disk
#      -boot1       boot from this disk (ignore other -boot flags)
#      -whole       export the entire device (including
#                  any non-HFS partitions). BE CAREFUL!
#      -cd          CDR/DVD
#
# MOL will boot from CD if it invoked through 'startmol -X --cdboot'.
#
blkdev:            /dev/cdrom      -cd ${cdboot}

ifempty ${altconfig} {
# # default configuration
#   blkdev:            /dev/hda      -rw
#   blkdev:            /dev/hdb2     -rw -boot -force
#   blkdev:            /dev/hdb3     -rw -force
#   blkdev:            /dev/sda      -rw
#   blkdev:            /dev/sdb      -rw
#   blkdev:            /dev/hdc      -cd
#   blkdev:            /dev/hdd      -cd
} else {
# # alternate configuration
#   blkdev:            /tmp/osx.image -rw
#   blkdev:            /dev/loop0    -rw
}

-----
# SCSI configuration (also relevant for IDE CDs and other SCSI-like devices)
#
-----
# By default, MOL automatically scans for non-harddisk type SCSI devices.
# If only specified SCSI devices are to be used by mol, change the
# the autoprobe setting below.
# Note: IDE, firewire and USB devices typically show up as SCSI devices.

autoprobe_scsi:    yes

# If SCSI autoprobng is disabled then devices must be exported manually
#scsi_dev:         "0:0:0"          # host/channel/id
#scsi_dev:         "1:0:0"          #

```

#### 4. Network Configuration

For network configuration, first edit the file "molrc.net", comment out lines with the symbol "#" for all lines concerning the "Sheep" driver, and add this line to the part concerning the Tun driver:

```
netdev: tun0 -tun
```

Next, edit the file "start-misc" in mol/script/ and after "ifconfig" modify your IP address and your subnet mask according to your network. For example, I did not touch the default IP address (192.168.0.25) that was present, but my router was broadcasting the address 192.168.0.254, as a result, I modified the address afterwards with "gw". Here for the part



concerning "Linux" network configuration. Concerning the network configuration of MacOS X refer to the article concerning Freebox and edit these to have exactly the same IP address.

```
# -*- perl -*- (for emacs syntax coloring)
#
# Mac-on-Linux mouse and keyboard configuration
#
#-----
# SheepNet Driver (recommended for AppleTalk only)
#-----
#
# MOL uses a special kernel module driver (sheep.o) to share the
# network device with linux.
#
#     netdev:      eth0 -sheep
#
# - Linux and MOL must use different TCP/IP addresses (residing on the
#   same subnet). This is a problem if you use DHCP...
#
# Note: It is possible to configure up to three netdevices
# simultaneously. Thus the SheepNet driver could be used
# for AppleTalk and the tun driver for TCP/IP...
#
# ifeq ${boot_type} newworld oldworld {      # Configure for MacOS only
#     netdev: eth0 -sheep
# }
#-----
# Tun Driver (recommended for TCP/IP)
#-----
#
# A virtual network is created. The linux interface (e.g. tun0) is
# configured from the /etc/mol/tunconfig script which is
# invoked when MOL starts.
#
#     netdev:      tun0 -tun
#
# The default tunconfig script masquerades the network (thus, linux
# and MacOS can share a single IP address). A local dhcp server
# is also started (if available).
#
# - AppleTalk will not work over the tun device.
# - Make sure /dev/net/tun exists (mknod /dev/net/tun c 10 200).
# - The tunconfig script contains further information/documentation
# - Configure a suitable nameserver in /etc/mol/dhcpd-mol.conf.
#
#     tunconfig:    alternate_tunconfig_script
#
# netdev:      tun0 -tun
#
#-----
# Tap Driver (more or less obsolete)
#-----
```

## 5. Audio Configuration

Unfortunately, I did not succeed in getting the smallest least sound out of my Soundblaster Live. If someone manages to configure this, I would be happy to receive information on how they did it.

### 2.6.3 Installation of MacOS X

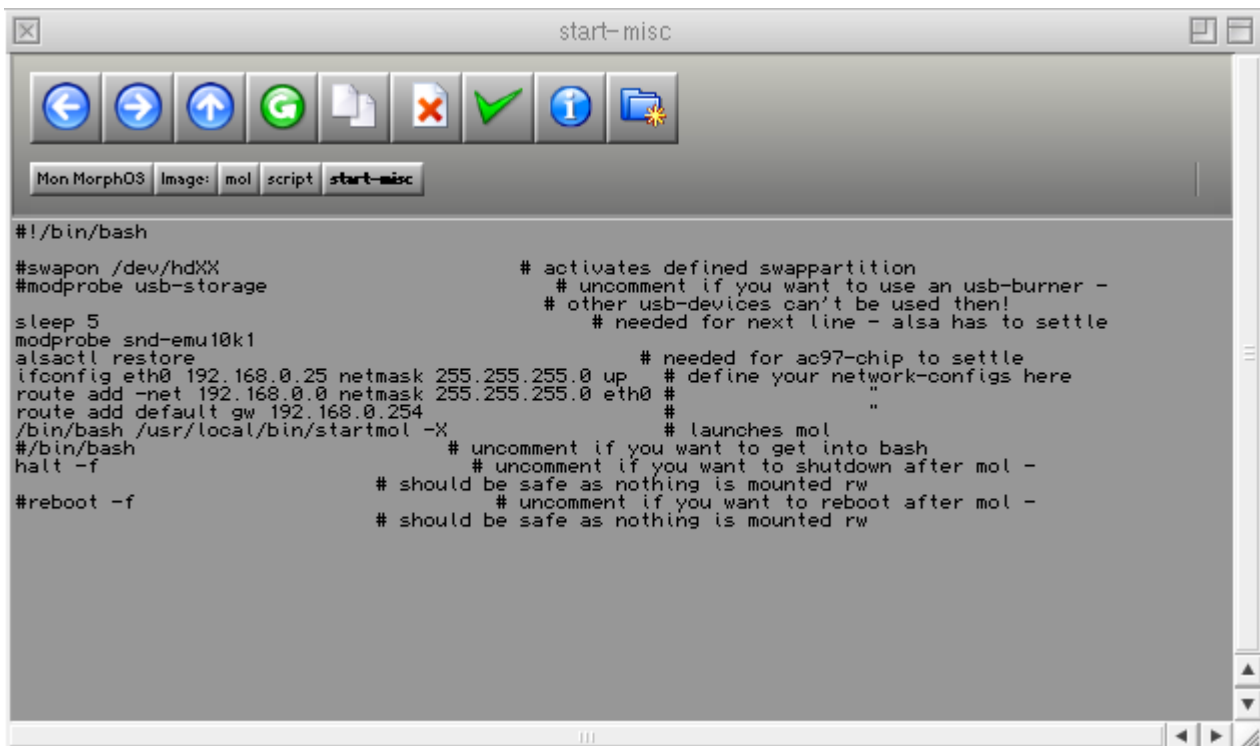
Now you are ready to install MacOS X, but a last complication imposes before you can do so. Still in the file "start-misc", you want to comment out with the symbol "#" the line "/bin/bash".

Next boot the Pegasos under MolK (with the command "boot molk" in the openfirmware interface). To the reader, type this command after you have inserted the CD in the working reader (if you have two and the disc does not work with the first reader, then try the second, that often works; -D) :

`startmol -X -cdboot`

Normally, you can now begin the installation of MacOS X. It is of course possible to use a previous installation of MacOS X done with PegXMac or a basic installation of Linux.

Once the installation has finished, reboot again under MorphOS to again modify "start-misc" to uncomment /bin/bash. This action will allow you to start up directly under MacOS X upon your next "boot molk" command issued.



MolK is just another method to install MacOSX on Pegasos by using MorphOS alone.

## 2.7 Configuring a Freebox with Mac-On-Linux (MacOS X 10.3)

This paragraph is only available in French version.

## 3 Other systems with LiveCD

Several "LiveCD" are now available on Pegasos in order to test an operating system or a succession of pre-configured software, without having to install anything on your hard disk.

The specialist on the matter is Marcel Unbehaun alias "Frostwork" who recently made a certain number of various LiveCDs.

### 3.1 AROS-Live

Download (torrent link – 267Mb) :

<http://www.pegasos.org/downloads/torrents/aros-live.torrent>

This LiveCD, based on a Gentoo distribution, will let you try the AROS operating system. Unfortunately this last is still a little unstable on PowerPC platform. That will however allow you see this new Amiga-like operating system on your Pegasos.

After having burn the ISO image, for launching it under Openfirmware do:

`boot cd laros`

For a hard disk installation, do :

`boot cd hdaros`

Then let you be guided throught the installation process on disk (see Linux installations chapter for the partitions names). It will then be necessary for you to copy the boot file "laros" from the CD to your boot partition as well as the file "hdaros". In this text file replace "/dev/hdXX "by the name of the partition on which you installed AROS.

For more information on this LiveCD, have a look at the « README-live » on the CD's root.

For more information on AROS see « <http://www.aros.org> ».

### 3.2 Freevo-Live



Download (torrent link – 236Mb) :

<http://www.pegasos.org/downloads/torrents/freevo-live.torrent>

This LiveCD, based on a Gentoo distribution, will let you to transform your Pegasos II into a home cinema system. You will be able in particular to view DVD, to listen to CDs and to see images by means of a simple and intuitive menu.

After having burn the ISO image, for launching it under Openfirmware do:

```
boot cd freevo
```

For a hard disk installation, do :

```
boot cd hdfreevo
```

Then let you be guided throught the installation process on disk (see Linux installations chapter for the partitions names). It will then be necessary for you to copy the boot file "freevo" from the CD to your boot partition as well as the file " hdfreevo". In this text file replace "/dev/hdXX "by the name of the partition on which you installed Freevo.

For more information on this LiveCD, have a look at the « README\_L. » on the CD's root.

For more information on Freevo see « <http://freevo.sourceforge.net> ».

### 3.3 CDGA (CD Game Arcade)

Download (torrent link – 656Mb) :

<http://www.pegasos.org/downloads/torrents/cdga.torrent>

This LiveCD, based on a Gentoo distribution, will turn your Pegasos II into game console. Into the 110 games available on the CD, you can find "Armagetronad" (tron), "Chromium" (shoot them up), "Foobillard" (US billiard), "Neverputt", "Neverplay", "Pinball", "SuperTux" (platform), "Trackballs" (marble madness clone), "Trigger" (car rally), also as Doom1, Doom2, Hexen and Heretic.

Before burning the CD, you must know that to be able to play to Doom1, Doom2, Hexen and Heretic, you need to get the "WADS" files of the original games.

It is then necessary to put these files either directly on the CD (by modifying the ISO file with your burning software and by putting them in the "wads" directory), or by using a removable USB media (USB key or hard disk).

In both cases, the files must be renamed the following way :

DOOM.WAD -> jdoom1.wad (10 Mb)

DOOM2.WAD -> jdoom2.wad (14 Mb)

HERETIC.WAD-> jheretic.wad (10 Mb)

HEXEN.WAD -> jhexen.wad (19 Mb)

Tu use those files on a USB media (normally formatted as standard VFAT) :

- 1) Uncompress the « cfg.tar.gz » file which is in the ISO file onto the USB media ;
- 2) Copy the various wad files into the « wads » directory onto the USB media ;
- 3) Renamed the wad files as explained before.

After having burnt the ISO image, to launch CDGA, under Openfirmware, just type :

```
boot cd ga
```

If you chose the "removable USB media" option, it is necessary to plug the media to the first USB port (/dev/sda1 under Linux) and to mount it from the CDGA menu ("mount" which is the first line of the menu).

### 3.4 AmigaOS/CD32-Live

Download (torrent link – 7Mo) :

<http://www.pegasos.org/downloads/torrents/cd32-kernel.torrent>

This LiveCD will let you use an AmigaOS/CD-32 environnement. You will need to add a ROM file from an Amiga machine.

I didn't have time to try it yet...

### 3.5 Others LiveCD

Here are a few others LiveCDs, most of them based on Linux Gentoo :

- Mupper rescue CD (tools suite for restoring your system) : <http://www.mupper.org/>
- Gentoo X Luminicity - <http://www.ppczone.org/downloads/ppc-livedvd.torrent> (broken)
- TORCS – cars game - <http://www.ppczone.org/downloads/torcs.torrent>
- Trigger – cars game - <http://www.ppczone.org/downloads/trigger.torrent>
- Cube – FPS game- <http://mirror.ovh.net/gentoo-distfiles/experimental/ppc/livecd/cube-gamecd-ppc-20050130.iso>

## 4 Tricks and hints under Linux with a Pegasos

### 4.1 Use a Nintendo Wiimote with a Pegasos

This section provides information to use a Nintendo Wiimote controller on a Pegasos. The Nintendo controller has motion sensing capability. It can give a totally new and attractive gameplay to any game.



The Wiimote communicates with the Wii via a Bluetooth wireless link. The Bluetooth controller is a Broadcom 2042 chip, which is designed to be used with devices which follow the Bluetooth Human Interface Device (HID) standard, such as keyboards and mice. The Bluetooth HID is directly based upon the USB HID standard, and much of the same documentation applies.

For more information about it, have a look at the following URL :

<http://www.wiili.org/index.php/Wiimote>

A Wiimote is shipped with every Wii console sold, but it is also possible to buy a single one for a price between €40 and €60. We will also be using a Belkin/Sitecom CN-520 Bluetooth USB device like the one on the picture below (about €15) :



The whole will be running on a Linux Gentoo system with a kernel version higher or equal to 2.6.19. For more information about Linux Gentoo installation, see the Linux Gentoo chapter before.

#### 4.1.1 Cwiid Driver Installation

First of all, we need to get the "CWiid" drivers. CWiid is a collection of Linux open source tools written in C for interfacing your Linux system with a Nintendo Wiimote.

Start by downloading the "Cwiid" driver sources from <http://abstrakraft.org/cwiid/>

Download the latest version (I personally did use the 0.5.02 version available at the following URL : <http://abstrakraft.org/cwiid/downloads/cwiid-0.5.02.tgz>).

Now, we must enable the "User lever driver" support into the kernel (or "uinput").



In /usr/src/linux ; do "make menuconfig" to setup your kernel.

Device Drivers -> Input Device support -> Miscellaneous devices -> <M> User level driver support

Quit and save. Check that the "grep UINPUT .config" commands displays the following result :  
"CONFIG\_INPUT\_UINPUT=m".

Copy your kernel at the right place and reboot (for more information about kernel configuration, check the Linux Gentoo chapter or/and the Linux Gentoo wiki).

After reboot, launch the module with the "modprobe uinput" command. To have it launched on each boot, add the "uinput" line at the end of the /etc/modules.autoload.d/kernel-2.6 file. For that type the following command :

```
echo "uinput" >> /etc/modules.autoload.d/kernel-2.6
```

Now we are ready to compile the Cwiid programs. Go into the Cwiid source directory and type the following commands :

```
./configure
make
make install
ldconfig
```

In case of any problem, see the associated README file.

Note : "configure" may fail if some packages are not installed on your system. For Bluetooth files, if you use gnome, the simplest is to emerge "emerge gnome-bluetooth" which will install all what is needed :

```
Mon Mar 26 13:49:13 2007 >>> dev-libs/openobex-1.3
Mon Mar 26 13:52:14 2007 >>> net-wireless/bluez-utils-2.25-r1
Mon Mar 26 13:56:02 2007 >>> net-wireless/libbtctl-0.6.0-r1
Mon Mar 26 14:09:09 2007 >>> net-wireless/gnome-bluetooth-0.7.0-r1
```

### 4.1.2 Bluetooth Configuration

Now, it's time to install and configure Bluetooth. For that, follow the Linux Gentoo Bluetooth tutorial available at <http://www.gentoo.org/doc/en/bluetooth-guide.xml> up to step 5 ("Detecting and Connecting to Remote Devices").

The Belkin/Sitcom CN-502 Bluetooth USB key is not listed in the peripherals supported by "bluez", but it works. Here is the console output for this device :

```
pegasos linux # lsusb
Bus 002 Device 002: ID 0a12:0001 Cambridge Silicon Radio, Ltd Bluetooth Dongle (HCI mode)
Bus 002 Device 001: ID 0000:0000
Bus 001 Device 001: ID 0000:0000
```

```
pegasos linux # hciconfig -a
hci0: Type: USB
BD Address: 00:10:60:A7:B1:CF ACL MTU: 192:8 SCO MTU: 64:8
UP RUNNING PSCAN ISCAN
RX bytes:396684 acl:24160 sco:0 events:288 errors:0
TX bytes:3593 acl:150 sco:0 commands:71 errors:0
Features: 0xff 0xff 0x0f 0x00 0x00 0x00 0x00 0x00
Packet type: DM1 DM3 DM5 DH1 DH3 DH5 HV1 HV2 HV3
Link policy: RSWITCH HOLD SNIFF PARK
Link mode: SLAVE ACCEPT
Name: 'BlueZ at pegasos (0)'
Class: 0x3e0100
Service Classes: Networking, Rendering, Capturing
```

Device Class: Computer, Uncategorized

HCI Ver: 1.1 (0x1) HCI Rev: 0x20d LMP Ver: 1.1 (0x1) LMP Subver: 0x20d

Manufacturer: Cambridge Silicon Radio (10)

In case of a problem : if the "lsusb" command displays the Bluetooth key but the hciconfig displays nothing, just unplug the USB key, check that the Bluetooth service is launched, then replug the USB key. It should now be working.

Note that some peripherals may be recognised in HID mode instead of HCI. To change this, just type the "hid2hci" command.

### 4.1.3 Recognising the Wiimote

Press the "1" and "2" buttons of your wiimote simultaneously. The Wiimote's LED should now be blinking. Then type "hcitool scan" command to display the Wiimote's MAC address.

Example:

```
pegasos linux # hcitool scan
```

```
Scanning ...
```

```
00:17:AB:2B:CD:81 Nintendo RVL-CNT-01
```

### 4.1.4 wminput, wmgui and wmdemo

Now, it your time to play : wmgui and wmdemo will help you check that everything works. To make a more interesting test it is possible to use wminput with the neverball configuration file. So you will be able to control the game with the Wiimote's sensor. Great fun !

### 4.1.5 And now what can we do with it ?

Use your imagination! This motion sensitive controller can be used with any application thanks to the configuration files. For example, you can set up UAE Amiga emulator and run your old games using the wiimote.

## 4.2 Using a cellular as a modem

I haven't tested this hint, but it might be very useful.

Test configuration :

- Pegasos II/G4
- Nokia 6630
- USB phone->PC cable
- Linux Ubuntu Breezy (5.10)

1) Create a file called "wvdial.conf" in /etc/ with a plain text editor (gedit did the job for me) with the following content:

```
[Dialer Defaults]
Modem = /dev/ttyUSB0
Baud = 460800
Init1 = ATZ
Init2 = AT+CGDCONT=1,"IP","internet.wind",,0,0
ISDN = 0
Modem Type = Analog Modem
Carrier Check = no
Phone = *99#
Username = ''
Password = ''
```

In the Init2 string change "internet.wind" to the access point string of your mobile operator.

2) Remember to configure your mobile for GPRS/UMTS services (follow their instructions) and to activate the service itself.

3) If you have connected in the past to ADSL from your Pegasos using Linux remember to do following:

a) open the Terminal;

b) type "network-admin" and give your root password when asked;

c) disable ethernet connection or you won't be able to go online using your mobile as a modem;

4) open the Terminal and type:

```
wvdial
```

Now you should be online !

## 5 Addendum

### 5.1 Complementary information on Linux Debian for Pegasos I and II

#### 5.1.1 Installation of Linux Debian on Pegasos I

Pegasos I image : ISO image of Linux Debian Sarge 2.4.27 CD (October 2004 - 305Mb) :  
<http://people.debian.org/%7Eluther/pegasos/sarge-pegasos1-netinst-2004.10.25.iso>

Complementary information for Pegasos I users are available on the <http://pegasos.powerpc.ch> website. They are more detailed than into this document (because of a lack of test hardware).

#### 5.1.2 Networks problems with some Pegasos I

On Pegasos I (April 1), the MAC address seems to be not always valid.

One finds oneself for example with a MAC address FF:FF:FF:FF:FF.

There is a way of avoid this problem by changing the MAC address under Linux. On the other hand this change is lost when one reboots (it can thus be interesting to save those instructions in the file ".bshrc" or in a script file launched manually).

First of all it is necessary to know the good MAC address of the network interface.

Let's say for example that your MAC address under MorphOS is 0F:0F:0F:0F:0F (for that seek in your TCP/IP stack, for example Miami).

Unmount Ethernet 0 interface :

```
ifdown eth0 down
```

Set the right MAC address (the one you have under MorphOS) :

```
ifconfig eth0 hw ether 0F:0F:0F:0F:0F
```

Mount up the whole :

```
ifdown eth0;ifup eth0
```

Now, for those commands to be done automatically on each boot, edit the /etc/network/interface file :

`pico /etc/network/interface`

Type in the previous instructions, then CTRL+x to finish, 'Y' to save.

Note: this method is a skirting. The best way to solve the MAC address problem on Pegasos I April 1, is in the long term to use another network interface.

### 5.1.3 Loss of network support (Pegasos I or II)

During Debian's installation, you may lost the network support depending on the peripheral network which you use, of your kernel version or maybe your hardware. Indeed, recent kernels recognize all the Pegasos interfaces (Ethernet 10/100, gigabit, firewire, PCI network interface if any) and it may not use the right one by default. To check that, open a console with the combination of keys "Alt-f2". Log as root (there is not yet any password).

Try for example to "ping" Google with following command line : `"ping www.google.fr"`. If it works (type Ctrl-c for to stop it), you can then go back to the installation console with the keys combination "Alt-f1" and then continue in the following paragraph. In the contrary case, you must find which is the interface which you use in the kernel's log thanks to the command `"dmesg|grep eth"`. If you do not use a PCI network adaptater, it should be : `eth0=gigabit`, `eth1=firewire`, `eth2=ethernet 10/100`. You have to edit the file `/etc/network/interfaces` with the command `"pico/etc/network/interfaces"`. In this file, replace all the "ethx" by your interface. Type Ctrl-x to leave, "O" to confirm and "Enter" to validate. It is also necessary for you to start again the network layer with the command `"/etc/init.d/networking/restart"` before being able to do the ping test again. You can then go back again to the installation console with the combination of keys "Alt-f1".

## 5.2 Complementary information about Linux Ubuntu for Pegasos II

This chapter contains complementary information about Linux Ubuntu.

- Base install of Edgy 6.10 version
- Base install of Dapper 6.06 version
- Base install of Breezy 5.10 version (the last version where the install was still easy)
- Installing EasyUbuntu
- Installing KDE desktop manager
- Installing Xfce desktop manager
- Installing e17 desktop manager
- Installing MacOSX via MacOnLinux

### 5.2.1 Base install of Edgy 6.10 version (with GNOME window manager)

I haven't tested Ubuntu Edgy installation by myself and those information are taken from pegasos.org's website.. Indeed, the problems encountered in the Dapper 6.06 version (see next paragraph) are not totally resolved, but this short howto may help you to install Ubuntu.

The boot CD does not work on Pegasos but hopefully the DVD does.

The Ubuntu Edgy DVD should preferably be downloaded with bittorrent.

To boot the DVD from OpenFirmware, type the following:

`boot cd install/pegasos`

For detailed installing instructions, you may have some additional information with the "Installing Dapper step by step" paragraph thereafter.

The bad news is that X won't start after a default installation. It's a rather easy to fix by simply forcing use of proc instead of sysfs when X scans for pci devices, Ubuntu added a patch that made scanpci in X use sysfs instead by default which currently causes problems on Pegasos.

So, Pegasos.org prepared a repository with a fix for the affected package. To use it add the line to your `/etc/apt/sources.list` file:

```
deb http://www.pegasos.org/ubuntu/edgy binary/
```

Also install the public key for authentication of the packages (not required).

```
wget -O- http://kozz.org/files/keys/kozz-pubkey.gpg | sudo apt-key add -
```

Then run the following command to install the new packages:

```
apt-get update && apt-get upgrade
```

Then it should be as simple as running the following command to make X start.

```
/etc/init.d/gdm restart
```

The source is available at:

```
deb-src http://www.pegasos.org/ubuntu/edgy source/
```

Note that two working kernels are available from <http://www.powerdeveloper.org/> :

Edgy (6.10) Desktop : <http://www.powerdeveloper.org/downloads/vmlinuz-6.10-desktop>

Edgy (6.10) Alternate : <http://www.powerdeveloper.org/downloads/vmlinuz-6.10-alternative>

## 5.2.2 Base install of Dapper 6.06 version (with GNOME window manager)

Skip this chapter if you have a working Ubuntu Edgy 6.10 version.

Installing Ubuntu 6.06 Dapper is not as easy as before (Breezy 5.10) concerning boot from install CD.

Ubuntu Dapper uses a 2.6.15 kernel with some 2.6.16 modules and « mkvmlinuz » shipped on the various CD does not work with last kernel. This essential tool is used to generate a file making it possible to boot on Pegasos system.

Ubuntu and most of the 'binary' distributions (Suse, YDL...) ship a small kernel and have initrd. To be loadable by OpenFirmware these must be either combined into one big file by mkvmlinuz (or mkzimage on SUSE), or one needs to use a bootloader like yaboot or GRUB2 to load these files. The problem comes from the fact that mkvmlinuz does not work in dapper version and that current firmware version does not support yaboot, but according to Sven Luther, firmware 1.3 should make it possible to use yaboot. The only solution then is to make mkvmlinuz work again to generate a boot file outside of CD or use GRUB2.

According to that, there are currently several way for installing Ubuntu 6.06 Dapper

- Installing it with « czp » boot files (made with a home made mkvmlinuz under SUSE)
- Installing it with the GRUB2 script from "Gunne"
- Installing it with PegXMac 2.0 LiveCD from David Benthams "pegxlindev"
- Installing Ubuntu Breezy 5.10 then upgrading it to Ubuntu Dapper 6.06

In case of trouble with your install, see 1.3.3 section.

### 5.2.2.1 Avoiding Dapper's boot problem on Pegasos

#### 5.2.2.1.1 Installing Ubuntu 6.06 with "czp"'s boot files

This solution is probably the easiest way to get started with Ubuntu Dapper. When everything is up and running, you can start experimenting with Gunne's GRUB2 solution (see next chapter), as that handles kernel updates automatically, while without a working 'mkvmlinuz' for Ubuntu, you can not create an updated kernel yourself. But let's hope, that mkvmlinuz will be fixed soon, to simplify things.

Note : "czp"'s images were prepared with the SUSE 'mkzimage' tool, so the first word appearing on your screen will be 'SUSE', but don't be afraid, it will quickly disappear.

« czip » has prepared files that make possible to install the system and to boot it from the three Ubuntu install types.

- Server
- Desktop (version which is the most popular)
- Alternate

It's up to you to choose which one is your preferred install : download it and burn the Ubuntu PowerPC CD you want : <http://www.ubuntu.com/download/>

#### **5.2.2.1.1.1 Server installation CD**

[http://genesi.hu/ubuntu/ubuntu\\_s\\_inst](http://genesi.hu/ubuntu/ubuntu_s_inst)

[http://genesi.hu/ubuntu/ubuntu\\_s\\_boot](http://genesi.hu/ubuntu/ubuntu_s_boot)

Burn your server CD and download the two czip boot files to somewhere, where the Firmware can find it, on the network or on an ext2/ext3 partition. In the following example we will use /dev/hda1, or hd:0 in Firmware terms. Load the CD. Enter the following at the Firmware prompt to start installation:

```
boot hd:0 ubuntu_s_inst
```

This will start the installer. Just follow the instructions. At the end don't worry if the boot loader installation fails. Choose 'Finish installation' from the menu, and when the popup window appears, choose 'Go back' to finish installation. Remember, which is your boot partition, you will need that !

When your Peg restarts, enter the following at the Firmware prompt:

```
boot hd:0 ubuntu_s_boot root=/dev/hdb6
```

Replace /dev/hdb6 with your actual root partition! The best is to put this line in your boot menu file.

#### **5.2.2.1.1.2 Desktop installation CD**

Proceed as for the Server installation CD, but with the following files :

[http://genesi.hu/ubuntu/ubuntu\\_d\\_inst](http://genesi.hu/ubuntu/ubuntu_d_inst)

[http://genesi.hu/ubuntu/ubuntu\\_d\\_boot](http://genesi.hu/ubuntu/ubuntu_d_boot)

Install :

```
boot hd:0 ubuntu_d_inst splash boot=casper
```

Boot of installed system :

```
boot hd:0 ubuntu_d_boot root=/dev/hdb6
```

Replace /dev/hdb6 with your actual root partition! The best is to put this line in your boot menu file.

#### **5.2.2.1.1.3 Alternate installation CD**

Proceed as for the Server installation CD, but with the following files :

[http://genesi.hu/ubuntu/ubuntu\\_a\\_boot](http://genesi.hu/ubuntu/ubuntu_a_boot)

[http://genesi.hu/ubuntu/ubuntu\\_a\\_inst](http://genesi.hu/ubuntu/ubuntu_a_inst)

Install :

```
boot hd:0 ubuntu_a_inst
```

Boot :

```
boot hd:0 ubuntu_a_boot splash root=/dev/hdb8
```

Replace /dev/hdb6 with your actual root partition! The best is to put this line in your boot menu file.

#### **5.2.2.1.2 Installing Ubuntu 6.06 with Gunne's GRUB2 script**



Here is now a Grub2 script realized by « Gunne » which makes possible to launch the Desktop install CD for Ubuntu, Ubuntu, Xubuntu ou Kubuntu on Pegasos II as the system once installed.

First, get the archive :

<http://tiw-pro.web.internet.telia.com/%7E1698137/dl/grub2dapper.tar.bz2>

The archive is compressed with bzip2, and contains the script to load from the firmware (or to load from Smart Firmware forth script).

Extract the archive on Your first partition on the harddrive. this partition must be EXT2 or EXT3 as grub2 not yet supports affs filesystem, neither does it support the iso file system yet, why the vmlinux + initrd.gz files must resist on this partition (as in the archive) as well.

To avoid a too big size of the archive, Gunne only included initrd.gz and vmlinux for ubuntu. If you would like to boot the xubuntu or kubuntu Desktop live CD instead, you need to copy vmlinux and initrd.gz from those Desktop live CD into the drawers he left empty (xubuntu and kubuntu).

If You would like to put the script and files on some other partition instead of the first partition, You need to edit the grub.cfg so it points to the right partition, else it wont work.

The same if You would like to load the installed system from another partition then /dev/hda5 as in the script.

Also note, that the script will only work with The PowerPC Desktop CD. Not with the server CD, or alternate CD. For those the script have to be modified a little further.

Download the Desktop PowerPC CD ISO, burn the cd, put it in the CD-Drive and execute the grub2 script from the firmware prompt.

Launch the grub2 script from the Smart Firmware prompt using this command string:

```
boot hd:0 grub/grubof.modules prefix=(hd,0)/grub
```

Then follow the install instructions.

#### *5.2.2.1.3 Installing Ubuntu 6.06 with PegXMac*

PegXMac's V2.0 is based on Ubuntu 6.06. This LiveCD is not free (but with support). It seems that there's everything needed on it to install Ubuntu without pain.

### **5.2.2.2 Installing Dapper step by step**

Start the installation software, by choosing one of the previous starting methods (cdp files/GRUB2 script).

#### *5.2.2.2.1 Language, timezone, keyboard*

On the first welcome screen, you will be asked to choose a language.

Use the arrows to choose your language then press Enter.

You will then be asked to choose your country including those into which your language is spoken. If your country does not appear in the list, choose "other" to obtain a list of all available countries.

The software will then give you a default keyboard, accept it or use the menu to choose a different keyboard into the list.

Then wait that the installation software detects your CD drive, test Ubuntu CD's integrity and installs additional components (for example partman).

#### 5.2.2.2.2 Network

On next step, the installation software will try to detect your network hardware and configure it by DHCP. If you are not connected to a network or if you don't have DHCP, you can specify manually your network configuration or report this to later.

To finish the network step, you must give a name to your computer (by default : ubuntu, choose for example « pegasos »).

Choosing a mirror : on this step, you will be able to select a mirror to retrieve/download software. Select a mirror from your country on <https://wiki.ubuntu.com/archive>, for example **fr.archive.ubuntu.com** for France. The installation software will download the package from this server.

#### 5.2.2.2.3 Disks partitionning

Ubuntu's installation software will then offer the possibility to use all the capacity of your hard disk to install your operating system. The advantage of this method, for a Linux beginner, is that the creation of essential partitions to make the system work will be done transparently.

However, it has the disadvantage to delete all the data on the hard disk. This operation is always risked, whether you choose an automatic partitioning or a manual partitioning. If you have important data, it is highly recommended to save it on an external support (DVD, CD-ROM or other...).

Choose the automatic partitioning for deleting all on your hard disks and restart from beginning. The essential partitions for the system will be created automatically. You will be proposed two automatic modes: IDE and LVM.

IDE : standard mode (by default).

LVM : mode more flexible for partitioning management.

Choose manual partitioning if you want to keep one or several existing partitions or if you want to customize the number, size and filesystem of your partitions. In all cases, do not forget to schedule at least one primary boot partition in ext2/ext3 on one of your disks to be able to start Ubuntu.

##### 5.2.2.2.3.1 Automatic partitioning

This option makes the installation software analyse your computer configuration and propose a classical partitioning. The changes proposed not being applied automatically (they require a validation from you), you may want to change it later by modifying manually your disks partitioning.

##### 5.2.2.2.3.2 Manual partitioning

The manual partitioning will let to choose which

Partitions must be created, reused, formatted or deleted.

Ubuntu can technically be installed on a single partition (« / » root partition) ; for a better fonctionnement and for more security, it is advised to create at least the three following partitions :

A root partition [/ mount point] in which the utilities and services will be installed, so as programs.

A swap partition [no mount point], which is used as a memory extension for your computer. It is suggested that this partition has a size 1.5 or twice the RAM capacity of your computer (ex: if your computer has 512Mb of RAM, then the swap shall have a size between 768 Mb and 1024 Mb) ;

A users partition [/home mount point], into which will be the users files (text docs, films, audio files, etc.) also as users personal parameters.

There is no filesystem to select for the swap partition ; the most commonly used filesystems for / and /home partitions are ext3 and reiserfs. The second one is more efficient but it is not recognized by OpenFirmware. If you want to use reiserfs, think about creating at least an ext3 partition for boot.

In all case, schedule if it is not already done, a boot partition of about a hundred Mb to install your favorite boot loader (GRUB2, BootCreator...) on your first disk.

#### **5.2.2.2.3.3 Finishing partitionning**

The changes in partitionning are not applied systematically, whether on an automated partitionning or a manual partitionning. A confirmation by user is required to apply the changes.

Once you have finished the partition settings, selecte "Finish partitioning and apply changes".

#### *5.2.2.2.4 Basic installation*

After confirmation, the installation software will format your partitions and will start the installation of the basic Ubuntu system (this process can take a while). Then, it will do the kernel installation and the copy of the remaining CD packages on your hard disk.

Once the basic system is installed, the installation software will ask you if you want to download the missing languages support packages from Internet. If you did not succeeded in configuring your network connexion (no DHCP, web proxy, password for connexion...), answer no to that question. If you answer yes, the system will try to download and install the missing packages.

Once the packages installation finished, you will have to choose a timezone depending of the country you have selected before.

#### *5.2.2.2.5 User account creation*

Enter your full name, then choose a pseudo (for example your surname) for your user account. This will be the default account.

Choose a password. By default, the "root" administrator account is disabled. You can access it by using the "sudo" program from the user account previously created.

#### *5.2.2.2.6 Installing the boot-loader*

The boot-loader is a menu which is displayed on the computer's boot and which gives you the possibility to choose an operating system which you would like to use. The boot-loader used by Ubuntu is called GRUB.

On x86, GRUB2 is installed in the MBR (on the first disk). To avoid any troubles, the best is not to install a boot loader for the moment.

The installation software, will then tell you when the first step is over. Write carefully the boot command which will be necessary for the first boot.

#### *5.2.2.2.7 Reboot and end of install*

Restart on your new Ubuntu install depending of your boot method choice (czp's files or GRUB2 menu).

After the kernel's load, the Pegasos will continue the install alone, downloading additionnal modules if necessary from Internet. Be patient, and let this operation go to its end.

On the end of installing, the only thing you have to do now is to get connected to Ubuntu with the user name you gave during the first step of the installation process.

Once connected, the first thing to do, if you could not do it during the installation process, is to setup your network connexion to install any missing packages and do updates to your system (do not update your kernel for the moment).

### 5.2.2.3 Getting a working « mkvmlinuz »

Go to <http://packages.ubuntu.com>, and in the search field, input « mkvmlinuz » and choose « edgy » distribution, then click on “Search”.

## Search

---

### Search package directories

---

Keyword:

Search on: ☒ Package names only ☐ Descriptions ☐ Source package names

Allow searching on subwords: ☒

Search case sensitive: ☐

Distribution:  Section:

There are shortcuts for some searches available:

- ♦ <http://packages.ubuntu.com/name> for the search on package names.
- ♦ <http://packages.ubuntu.com/src:name> for the search on source package names.

Then choose version 23 of mkvmlinuz (which is normally the only one), then download it. Then open a xterm terminal, go into the directory where you've downloaded the file, then input the following command to install it :

```
sudo dpkg -i mkvmlinuz_23ubuntu2_powerpc.deb
```

Fill in you password if it's your first sudo command in that xterm.

Now, you need to update your kernel, do the following command :

```
sudo mkvmlinuz
```

The Pegasos II CHRP-kernel will be generated, and a symlink **vmlinuz** pointing to the **recent kernel** installed will also be generated.

This link can then be used to start directly from OpenFirmware.

For example :

```
boot hd0:4 boot/vmlinuz root=/dev/hdb5
```

Now, after every kernel update, this symlink will point on the lastest generated kernel and its usage will be done transparently after each reboot.

### 5.2.3 Base install of Breezy 5.10 version (with GNOME window manager)

Ubuntu Breezy is the previous Ubuntu version. If you have installed Dapper without any problem, then just ignore this section.

This version is easier to install as there is no extra file to download somewhere else neither any hack to do to make the CD boot.

5.10 version includes Gnome 2.12.1 and OpenOffice 2.

#### 5.2.3.1 Preparation

First, download and burn Ubuntu installation CD for PowerPC

<http://www.ubuntu.com/download/>

Ex for PPC version on a french mirror :

<ftp://ftp.proxad.net/mirrors/ftp.ubuntu.com/releases/5.10/ubuntu-5.10-install-powerpc.iso>

### 5.2.3.2 Installation

Under OpenFirmware enter the following command to start the installation :

```
boot cd install/powerpc/vmlinuz-chrp.initrd root=/dev/ram
```

Follow the various installation steps, step by step. Select in particular the adequate keyboard, the good time zone, the name of user and the password associated by default.

When the installation indicates the procedure for booting under the openfirmware, do not validate but press on keys ALT and F2 to open a command line session. Then, type the following commands :

```
chroot /target  
mkvmlinuz -a chrp
```

Then, go back to installation by typing on ALT and F1.

Before finishing, note the boot commands proposed by the installation software (scroll down the page for it).

Ex : `boot hd0:4 /boot/vmlinuz-2.6.12-9-powerpc /dev/hdb5`

Finish finally the installation. Pegasos reboots by itself.

In OpenFirmware, input the commands that were proposed by the installation software just before :

```
Ex : boot hd0:4 vmlinuz-2.6.12-9-powerpc root=/dev/hdb5
```

hd:x is your boot partition (for example for the first master hard disk on IDE1 numbered from 0, hd0:0), and « hdb5 » being your root partition for linux (for example the fifth partition of slave hard disk on IDE1 is hdb5).

Note #1: in case of display problem on the Pegasos's boot with a RADEON graphic card, replace the previous command by :

```
boot hd0:4 vmlinuz-2.6.12-9-powerpc root=/dev/hdb5 video=radeonfb:1024x768@75
```

Note #2: If you have chosen an incompatible filesystem with OpenFirmware (ex: ReiserFS), you can copy your core (/boot/vmlinuz-2.6.12-9-powerpc) in the partition of boot created for MorphOS and boot by typing for example: "boot hd0:0 vmlinuz-2.6.12-9-powerpc root=/dev/hdb5" (if your boot partition is on the first IDE master disk).

Note #3: You should also know that Ubuntu is also available in LiveCD (bootable CD on Ubuntu without anything to install). That is sometimes useful if you can't boot any more to start on your hard disk. You can then start on CD, and you have all the required tools to make repairs to your hard disk installation.

The Pegasos will finish the installation by itself, while recovering if necessary complementary modules on Internet. Be patient and let this operation be held automatically until the end.

End of installation. You only have to log under Ubuntu with the name of user and the password typed during the first phase of installation.



You will be able to then update your version with information “bubble” proposed in top on the right of your Ubuntu screen.

In case of problem with your install, see section 1.3.3.

### 5.2.3.3 Update to Dapper 6.06 version

To update your Linux Breezy distro into Linux Dapper, start first by checking that you have enough space on your disk (several Gb). If you think that it's a bit short, or that you don't have enough space, it's always possible to get some space on an other partition by mounting it through the fstab file.

Also think to save your important files before update like for example the /etc/gdm/gdm.conf file (one is never enough wise).

Be aware that an update can always make your system unstable and that your current system can lead to malfunction of some programs. It is really recommended to save your data and your most important programs/sources somewhere before starting an update.

If you are sure you want to update, in a console input the following command :

```
gksudo "update-manager -d"
```

Then click on « update » and let the install go on (about 1h30).

Don't forget to use the new installed kernel, then to be sure to have an up to date system, input the following commands :

```
sudo apt-get update  
sudo apt-get dist-upgrade
```

In case of sound problem after your update, do a little « `sudo modprobe snd-via82xx` ». It this fixes the problem, in /etc/modules file, add the following line :

```
snd-via82xx
```

### 5.2.3.4 Troubleshooting (Dapper or Breezy)

*Malfunction of the network adaptater under MorphOS after using Linux Ubuntu*

After use of Ubuntu with the standard network adaptater of Pegasos (Via-Rhine), while returning to MorphOS the network might not work very well.

For that there are two solutions, you may cut off the power supply of Pegasos during 30 seconds what can be quickly a little painful or get the small utility “Via-

Wakeup" (<http://fabportnawak.free.fr/via-wakeup.lha>) which "awakes" Via-rhine. You only have to launch it starting from your file "S:user-startup" before the TCP/IP stack starts.

#### *Administator "root" account*

The root account is disabled when you first install Ubuntu. The first user created during the installation has administrative rights on the system, and can run programs as root with `sudo`, using only their normal user password.

For example: "`sudo apt-get update`".

If you wish to use the root account in more traditional UNIX fashion, you can set the root password by typing "`sudo passwd root`". This will allow you to use `su` or login as root on the console.

If you need a shell with root privileges, run "`sudo -s`".

All uses of `sudo` will require the user's password.

#### *Mount of MorphOS partitions under Linux Ubuntu*

For that it is necessary to edit the file `/etc/fstab` under Linux

```
/dev/hdxy /mnt/PARTITIONNAME asfs user,rw 0 0
```

Where x is the unit and y the partition number (ex : `hda1`).

To be able to write on your partition, log as root ("`su`" command) and type the following commands in the shell:

```
chmod -R a+w .
```

#### *3D hardware acceleration not recognized (on a few ATI RADEON 9200 & 9250)*

Start by checking with the tool "`glxinfo`" if the option "Direct rendering" is not already activated by typing the following command: "`glxinfo | grep rendering`".

If the answer is "no", it is disabled and you then have to edit the `/etc/X11/xorg.conf` file.

In "device" section, add :

```
"Option "BusType" "PCI"".
```

For the changes to be taken in account, you have to restart your Pegasos.

Once that you will have started again under Linux Ubuntu, always check with the `glxinfo` command, that "Direct rendering" is now set to "Yes".

#### *Additional monitors resolutions*

To have additional monitors resolutions, still in the file `/etc/X11/xorg.conf` ", you just have to add the additional resolutions in the various "Modes" headings of the "Display" sub-section of the "Monitor" section.

Example :

```
SubSection "Display"
Depth 24
Modes "1280x1024" "1024x768" "800x600"
EndSubSection
```

It is also necessary to extend the the frequencies ranges of your monitor:

Example :

```
Section "Monitor"
Identifier "Generic Monitor"
Option "DPMS"
HorizSync 28-120
VertRefresh 43-85
EndSection
```

In case of problem with the "UseFBDev" option, in the "Device" section, set the option to false or comment the entire line.

Example (comment) :

```
#Option "UseFBDev" "true"
```

Lastly, so that the change is taken into account it is necessary to start again the X-server.

Note: "`dpkg-reconfigure xserver-xorg`" also allows to carry out a total reconfiguration of your file `xorg.conf` in a more graphic way. You can also restore the original file while making "`dpkg-reconfigures -phigh xserver-xorg`".

#### *Sound problem*

In `/etc/modules` file, add the following lines :

```
snd-mixer-oss
snd-pcm-oss
```



It is also necessary to go in menu "multimedia system\Preferences\system selection". In the Audio menu, select "OSS - Open Sound System" for "Sink by default" and "Source by default".

*The session finishes immediately after the opening of session*

It can happen after many operations that you can't open your session any more. If you have a message related to a problem with the "Xauthority" file, it may be that you accidentally created this file with the "root" rights and that there is no way to log in anymore because of wrong rights. To solve this problem, it is enough to log as "root" by using the ("CTRL "+" ALT "+" F1") keys then to remove the file "~/.Xauthority" before going again in X mode ("CTRL "+" ALT "+" F5 or F9").

## 5.2.4 Installing EasyUbuntu

EasyUbuntu is a script that makes it easy to install the most commonly requested apps, codecs, and tweaks that are not found in the base distribution (apps and multimedia codecs, web, archives, system, voice on IP). Note however that some of those only work with the latest Ubuntu (Dapper) version or even only on x86 architecture.

To install it, input the following commands from a console :

```
wget http://easyubuntu.freecontrib.org/files/easyubuntu-3.022.tar.gz
tar -zxvf easyubuntu-3.022.tar.gz
cd easyubuntu
sudo python easyubuntu.in
```

## 5.2.5 Installing KDE

To install the KDE window manager in addition to the default one (GNOME), it is necessary to start by adding a source in your file "sources.list". For that, do the following command : `sudo pico/etc/apt/sources.list`.

Then, add the following line at the end of the file:

```
deb http://fr.archive.ubuntu.com/ubuntu breezy universe
```

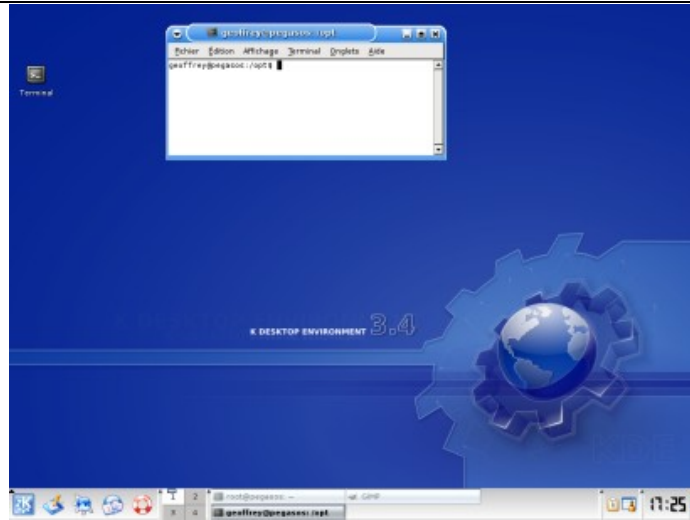
Once the file is saved (CTRL+X then "Y"), it is then enough to enter the following commands :

```
apt-get update
apt-get install kde
or
apt-get update
apt-get install kubuntu-desktop
```

That will install many modules and additional applications on your disk. Be patient!

Then, during the packages configuration, you will have to choose your session manager (launching interface). This interface makes it possible to choose your login and password and to choose the type of session which you wish to start (KDE, GNOME, Xfce, Enlightenment...). KDM is the system delivered by default with KDE and GDM with GNOME. It is up to you to choose your favorite one.

After restarting the computer, you will then have the choice between Gnome and KDE: for that at the time of the logging ((in the GNOME Display Manager "gdm" or in the KDE Display Manager or "kdm"), click on "Session" where you can now choose "KDE". GDM/KDM will ask you then if you want to choose KDE "right for this session" or "define it as default". In a first time, just click on "Just for this session". You will be able to change it later at the same place.



### 5.2.6 Troubleshooting

Opening session problem with kdm or gdm

If you can't open a kdm or gdm session anymore, don't panic : for that open a session in "console" mode and under a shell, type :

```
sudo dpkg-reconfigure gdm
```

or

```
sudo dpkg-reconfigure kdm
```

Changing language

If on KDE's startup language is set to english and you want to change it, you can change it in the menu "K -> System Settings -> Regional & Accessibility -> Country/Region & Language".

In "Country", choose for example "Europe, Western -> France". Apply the changes that will be taken in account on next start. You can also change the keyboard layout in "Keyboard Layout" on the left, then on "Enable keyboard layouts".

For example, in the "Available layouts" window, click on France, then "Add >>" then on the right window, click on "U.S. English (us)" and then "<< Remove". In "Keyboard model", choose you keyboard. If you have a standard keyboard you can select "Generic 105-key (Intl) PC".

You then only have to restart KDE (end this session) for the changes to be applied.

Note : if KDE is already well setup on first start, you can do those change to be sure to select your preferred units (euro, A4, metric system...).

### 5.2.7 Installation of Xfce

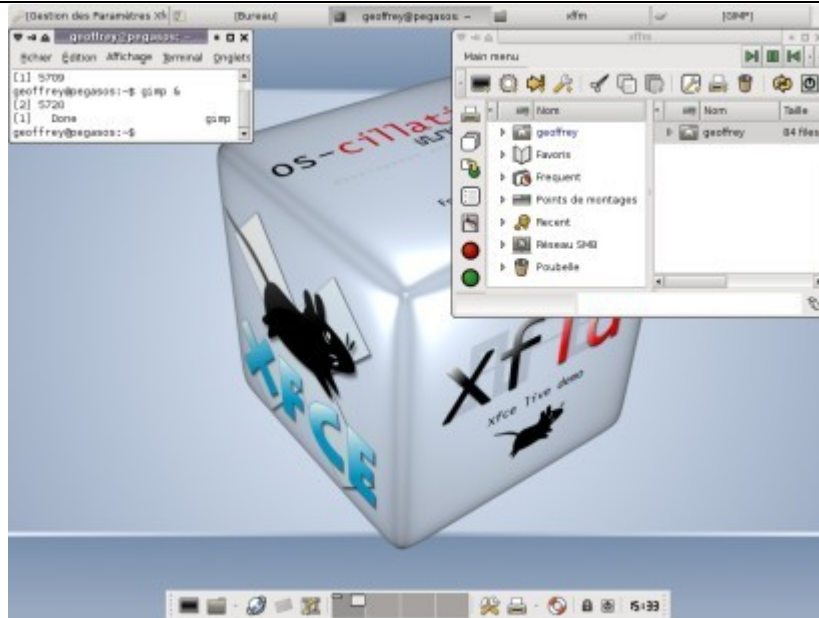
Xfce is another window manager.

To install it, you just have to execute the following command :

```
sudo apt-get install xfce4
```

After closing the session, you will then have the choice between Gnome, KDE (if you've installed it) and Xfce.

As for KDE, at the time of the logging (in the GNOME Display Manager "gdm" or in the KDE Display Manager or "kdm"), click on "Session" where you can now choose "Xfce". The user interface will ask you then if you want to choose Xfce "right for this session" or "define it as default".



## 5.2.8 Installation of Enlightenment e17

### 5.2.8.1 Introduction

Enlightenment is a windows manager as well as Gnome or KDE.

The e17 version (also known as DR17, or e16.999) It is still under development but already usable and appears rather promising with the level of the visual effects.

There is not official version compiled for Pegasos PPC. You will thus have either to download the packages of "Pegubuntu" having been compiled for Pegasos and available on WikiPeg's site, or compile it yourself in order to be able to use it in its last version.

Official site : <http://enlightenment.org/>

### 5.2.8.2 Installation of Enlightenment from "Pegubuntu" WikiPeg's Debian repository

A version of Enlightenment was compiled and put in a Debian packages repository on the site of WikiPeg. It includes Pegasos PowerPC binaries of CVS version of January 14th, 2006. If you prefer to compile the sources by yourself, you may jump directly to the following paragraph. Please however note that this repository is still experimental and it is better to download all the libraries so that all works correctly to avoid dependencies problems. Today, only the basic libraries (EFL), Enlightenment and the e17genmenu tool are available. The other applications were not compiled yet under packages form.

To install Enlightenment on your Ubuntu distribution (tested) or Debian (not tested) it is very simple.

With an editor like pico, edit your file /etc/apt/sources.list :

```
sudo pico /etc/apt/sources.list
```

Add the end of the file :

```
deb http://wikipeg.free.fr/pegubuntu pegubuntu main
```

Save ("CTRL"+"X", then "Y" to confirm).

Then do an update:

```
sudo apt-get update
```

Start by installing the following packages (48Mb) :

```
sudo apt-get install libedbl
```

```
sudo apt-get install enlightenment
```

Close your current session. In the session list, "Enlightenment" should now have appeared.

Start Enlightenment and enjoy the graphical effects and the “splash screen” !

Now, install all the Enlightenment libraries available on “pegubuntu” :

```
sudo apt-get install libe libe-dev libecore0 libedbl libedbl-dev libedje0 libedje0-dev
libee0 libee0-dev libemotion0 libemotion0-dev libengrave0 libengrave0-dev libepg0
libepg0-dev libepsilon0 libepsilon0-dev libesmart0-dev libevas0 libevas0-dev libewl0
libewl0-dev imlib2-loaders ewl0-bin esmart0-bin epsilon0-bin epeg0-bin edje0-bin edb-
tools
```

Then get all your application menus from KDE and GNOME :

```
sudo apt-get install e17-pegasos
sudo e17genmenu
```

Let e17genmenu work and display its logs. Now you should see appearing all the shortcuts towards your applications in the Enlightenment menus (left click).

### 5.2.8.3 Installation of Enlightenment by compiling the CVS sources

The sources compilation will let you have the latest version of Enlightenment.

#### 5.2.8.3.1 Installing the required packages for compiling

Start by executing the following commands in order to get the tools which will be necessary for you to compile e17.

```
apt-get install cvs autoconf automake1.7 gettext libtool libfreetype6-dev libjpeg62-dev
libpng3-dev libtiff4-dev libungif4-dev libbz2-dev libltdl3-dev pkg-config libxine-dev
libgtk1.2-dev build-essential flex bison byacc libxcursor-dev libcurl3-dev
```

All the sources under development are accessible in CVS on sourceforge.net. To recover them, carry out the following commands in normal user mode:

```
cd ~/
mkdir cvs
cd cvs
cvs -d:pserver:anonymous:@cvs.sourceforge.net:/cvsroot/enlightenment login
cvs -z3 -d:pserver:anonymous:@cvs.sourceforge.net:/cvsroot/enlightenment co e17 misc
```

This downloads all the sources necessary to compile the libraries, like e17 and some applications like “entice”, “entrance”, etc...

#### 5.2.8.3.2 Changing gcc version

If you try to compile the sources directly with the GCC version installed by default (4.0) you will immediately encounter problems with messages such as “undefined reference to memcpy@@GLIBC\_2.0”. To solve this problem it is necessary to use an older version of GCC.

So start by removing gcc-4.0 :

```
apt-get remove gcc-4.0
```

This will also remove the “g++”, “g++-4.0”, “gcc”, “gcc-4.0”, and “libstdc++-6-4.0-dev” modules.

Then, install gcc-3.3 :

```
apt-get install libstdc++5-3.3-dev
```

This will install “cpp-3.3”, “g++-3.3”, “gcc-3.3”, et “libstdc++5”.

You will have to then do some links by yourself :

```
cd /usr/bin
ln -s cpp-3.3 cpp
ln -s g++-3.3 g++
ln -s gcc-3.3 gcc
```

```
cd /usr/lib
ln -s /usr/bin/cpp-3.3 cpp
```

Note: there's maybe a simpler way to install GCC-3.3 without making the links by hand, but I did not find how...

#### 5.2.8.3.3 Compiling the EFL (Enlightenment Foundation Libraries)

Before starting to compile, you need to add a line in the **/etc/ld.so.conf** file. This line will allow the programs to access the newly installed library (which are now in /usr/local/lib) when running the **ldconfig** command. To add this line, just execute the following command :

```
echo "/usr/local/lib" >> /etc/ld.so.conf
```

Now you just have to compile every library in the order indicated on the official website. For each of the them, the procedure is always the same :

```
cd ~/cvs/e17/libs/<library_name>
./autogen.sh
make
sudo make install
sudo ldconfig
```

Compile the libraries in the following order : (Be careful : do not forget to execute **sudo ldconfig** after each library !) :

```
/cvs/e17/libs/edb
/cvs/e17/libs/eet
/cvs/e17/libs/imlib2
/cvs/e17/libs/imlib2_loaders
/cvs/e17/libs/evas
/cvs/e17/libs/ecore
/cvs/e17/libs/epex
/cvs/e17/libs/embryo
/cvs/e17/libs/edje
/cvs/e17/libs/epsilon
/cvs/e17/libs/esmart
/cvs/e17/libs/emotion
/cvs/e17/libs/etox
/cvs/e17/libs/ewl
/cvs/e17/libs/engrave
```

In some cases, a small program allows to check that everything went ok. For example, ecore comes with an executable called **ecore\_evas\_test** that shows samples of the powerful graphic libraries :

#### 5.2.8.3.4 Compiling and installing e17

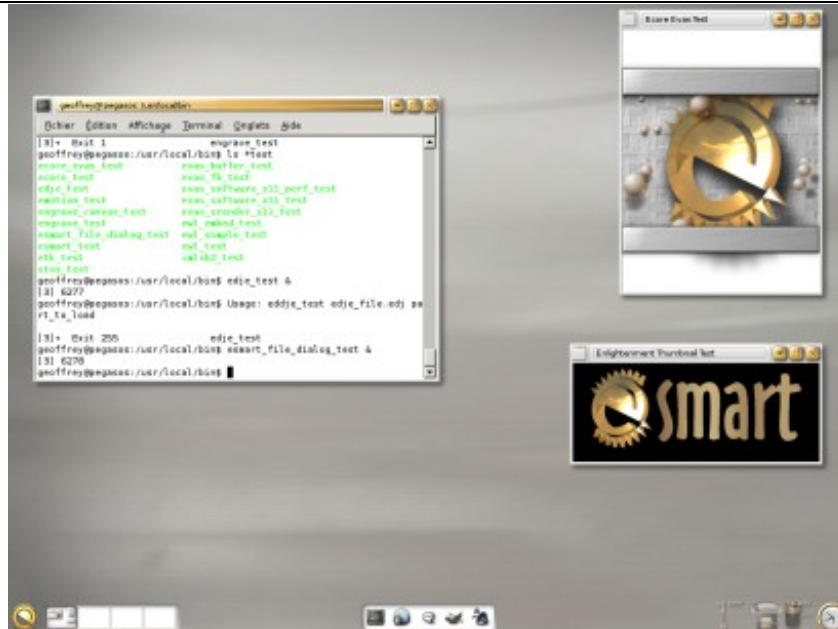
The E17 compiling is done as before :

```
cd ~/cvs/e17/apps/e
./autogen.sh
make
sudo make install
sudo ldconfig
```

To make a first fast test you can kill your current X-server (CTRL+ALT+BACKSPACE), then log as root. Create then a file ~/.xinitrc with the following contents:

```
exec /usr/local/bin/enlightenment
```

If you now do a « startx », e17 will be launched.



You will be able to benefit from it to start again the test applications like "ecore\_evas\_test" (in top on the right) or "esmart\_test" (in bottom on the right).

#### 5.2.8.3.5 Adding e17 to the sessions manager

Open a console and go to the directory « /usr/share/xsessions ».

With an editor like « pico », create a file « e17.desktop » with the following content :

```
[Desktop Entry]
Encoding=UTF-8
Name=Enlightenment-17
Comment=Enlightenment e17
Type=XSession
Exec=/usr/local/bin/enlightenment
TryExec=/usr/local/bin/enlightenment
```

Then restart the computer.

Now from GDM session manager menu, you should be able to launch either e17, KDE or GNOME.

#### 5.2.8.3.6 Compiling additional applications

You can now add some applications to e17. The sources are in several directories, in particular in /cvs/e17/apps and /cvs/misc. These two directories contain a great number of applications but they all are not all functional.

Among those which I could make work, you will find in particular:

/cvs/e17/apps/entice : image viewer for E17

/cvs/e17/apps/entrance : login manager (like gdm or kdm)

/cvs/e17/apps/examine : configuration tool for applications

Now, it's your turn to play !

#### 5.2.8.3.7 Adding application menus from Gnome et de KDE

To add the menus of Gnome and KDE in e17, it is enough to use the tool « e17genmenu » :  
<http://sourceforge.net/projects/e17genmenu>

Download the archive named « e17genmenu-4.2.0-1.fc4.src.rpm ».

Then install the Red Hat Package Manager (RPM) the following way :

```
apt-get update
apt-get install rpm
```

Uncompress the archive :

```
rpm -i e17genmenu-4.2.0-1.fc4.src.rpm
```

That will create a file of "tar.gz" type in which the source files in "/usr/src/rpm/SOURCES" will be created. Type then the following commands :

```
cd /usr/src/SOURCES
gzip -d e17genmenu.tar.gz
tar xvf e17genmenu.tar
```

Compilation :

```
cd e17genmenu
./autogen.sh
make
sudo make install
sudo ldconfig
```

To launch it, it is enough to type the "e17genmenu" command. Wait a while until all the menus are created. Once finished, you should have new menus (left click).

## 5.2.9 Installation of MacOSX

See chapter « Installation of MacOSX via Mac-On-Linux on Linux Ubuntu ».