

You get what you don't pay for.

Open Source, Debian and more

Dr Colin Turner

9th March 2006

Outline

- 1 Introduction
- 2 Free software
 - What is “free” software anyway?
- 3 GNU/Linux
 - FSF
- 4 Debian
 - What is unusual about Debian?
 - Installing Debian

This talk explores the issues of

- what “free” or “open source” software means;
- why it is important;
- a little about GNU/Linux or Linux;
- more specific information about GNU/Debian, a Linux distribution;

Free as in Freedom

The **Oxford English Dictionary** (Reference Version) has **19** definitions of the word **free**. Here are a few:

- unrestricted, unimpeded; not restrained or fixed;
- unconstrained;
- available without charge; costing nothing.

Most people think the last of these is what “free” software pertains to, but no, it is the former two which have a higher priority than the latter, although often strong restrictions on cost exist. As the FSF (<http://www.fsf.org>) say, it is **Free as in Freedom**.

Why is Freedom important?

It might seem that this is a fairly academic distinction. After all, if the software is free in price, why should we care? Software that is free in price but not in licence is usually that way for a reason.

- **Internet Explorer**, alleged to be free in a simple (successful) bid to crush Netscape,
- **Media Player**, alleged to be free to crush the opposition.

These products have tended to help the market position of MS Windows (understandably, to be fair to Microsoft that was they they were developed), but such commercial power has been the undoing of many competitors (e.g. OS/2). Is a mono-culture a healthy thing?

Why is Freedom important?

There are other reasons:

- Mozilla rose from the ashes of Netscape, and now Mozilla Firefox is a highly popular extendable browser with open source so that **it is available in almost identical form on many platforms.**
- In 2004 the developer of the PWC (Linux) webcam driver withdrew his proprietary driver (which was only available for the x86 platform) **and** his open source driver. It was judged by the kernel developers that hooks to closed source drivers could not be included for reasons of **maintenance** and **cross platform** support.

How much free software exists?

This talk was prepared **entirely** with free software:

- on a free operating system (**GNU/Linux (Debian)**);
- with a free editor (**Emacs**), using free extensions;
- compiled with a free document preparation system (**L^AT_EX**);
- with free extension packages (**beamer**);
- checked with a free PDF file viewer (**xpdf**);
- occasionally I looked up resources with a free browser (**Mozilla Firefox**).

And for almost all of these tasks I had several free alternatives. I could simply have used **OpenOffice**.

But, but I run Windows?

Really? Well, you are not alone...

- Special organisations do exist to help wean you off powerful dependencies to harmful addictions;
- you could always try one of the Linux distributions for those that like its strong similarity to Windows such as Red Hat or Mandrake;
- you could run **dual boot** with a free operating system on another partition of your hard disk;
- you could run a **live CD** of a free operating system, but long term it's no way to live;
- if you must, you must. But this is no excuse, since even if you must run Windows much of the above software is available for Windows, free.

The Free Software Foundation

In 1984 **Richard Stallman** (author of the emacs editor) launched the **Free Software Foundation** or **FSF** (<http://www.fsf.org>).

- Software produced by the FSF was often branded **GNU**. GNU stands for **Gnu's Not Unix** and was the first of many self-referential acronyms;
- The FSF intended to produce a complete **free** operating system — which was to be codenamed the **Hurd**. The intention was that this would be a free **Unix** like system.
- The software would be released under a licence known as the GNU General Public Licence or GPL (<http://www.gnu.org/copyleft/gpl.html>).

The Free Software Foundation

Currently, the version of the GPL in common usage is v2. It is **one** of the licences used for free software, though there are many others. See

<http://www.gnu.org/licenses/gpl-faq.html>.

- Not all GPL programs are GNU;
- Users are **free** to modify the GPL code as they wish;
- The licence ensures that any improved, or modified version of the code must also be released under the GPL (it cannot be closed);
- It **is** permissible to sell GPL programs;
- However, anyone with the software has the right to give it away.

GNU's not unix

- Years passed and Gnu produced a great deal of excellent and famed software, mostly under the GPL;
- Much of this software runs on Unix distributions of all makes;
- The **Gnu C Compiler (GCC)** becomes extremely popular (although now GCC stands for the **Gnu Compiler Collection** since it supports many languages);
- But the OS kernel does not appear in any meaningful way.

The rise of Linux

And then...

- In 1991, a Finnish student called **Linus Torvalds** began work on a new kernel to provide a freeware unix kernel for people to play with.
- It was (perhaps) based on an attempt at a free version of Minix, which was a x86 based kernel for students to explore, produced by **Andrew Tanenbaum**.
- Linus initially claimed that it would never be portable, but at the time of writing it is ported to many platforms and version 2.6.15 of the kernel is imminent. Linux is released under GPL v2.

This filled the missing gap in the movement.

Tensions in the Open Source / Free Software Community

Even this altruistic bunch of people have their own politics. At the front page of the GNU website (<http://www.gnu.org>) you will find a reference to why **Linux** Operating Systems should be known as **GNU/Linux** Operating Systems at (<http://www.gnu.org/gnu/linux-and-gnu.html>).

Richard Stallman on “Linux”

“If we tried to measure the GNU Project’s contribution in this way, what would we conclude? One CD-ROM vendor found that in their “Linux distribution”, GNU software was the largest single contingent, around 28% of the total source code, and this included some of the essential major components without which there could be no system. Linux itself was about 3%. So if you were going to pick a name for the system based on who wrote the programs in the system, the most appropriate single choice would be “GNU”.”

Linus Torvalds on “GNU/Linux”

“About the GNU/Linux argument; have you talked with Richard Stallman about this?”

*“**Linus**: rms asked me if I minded the name before starting to use it, and I said “go ahead”. I didn’t think it would explode into the large discussion it resulted in, and I also thought that rms would only use it for the specific release of Linux that the FSF was working on rather than “every” Linux system. I never felt that the naming issue was all that important, but I was obviously wrong judging by how many people felt very strongly about it. So these days I just tell people to call it just plain “Linux” and nothing more.”*

(<http://kde.sw.com.sg/food/linus.html>)

Free versus Open Source

Richard Stallman also draws a sharp distinction between **Free** and **Open Source** software:

“The fundamental difference between the two movements is in their values, their ways of looking at the world. For the Open Source movement, the issue of whether software should be open source is a practical question, not an ethical one. As one person put it, “Open source is a development methodology; free software is a social movement.” For the Open Source movement, non-free software is a suboptimal solution. For the Free Software movement, non-free software is a social problem and free software is the solution.”

More can be read at <http://www.gnu.org/philosophy/free-software-for-freedom.html>

DRM and GPL v3

The **TiVo** video system was built around linux, and was used to enforce (via a subscription fee) access to a service. This (and similar practices) have become known as **DRM - Digital Rights Management**.

As part of the drafting of version 3 of the GPL

<http://gplv3.fsf.org/draft> some restrictions on DRM are planned, to help prevent free software being used to restrict freedom of other kinds.

The wording is necessarily legalistic, and has been interpreted by Linus Torvalds in a particular way.

DRM and GPL v3

3. Digital Restrictions Management.

As a free software license, this License intrinsically disfavors technical attempts to restrict users' freedom to copy, modify, and share copyrighted works. Each of its provisions shall be interpreted in light of this specific declaration of the licensor's intent. Regardless of any other provision of this License, no permission is given to distribute covered works that illegally invade users' privacy, nor for modes of distribution that deny users that run covered works the full exercise of the legal rights granted by this License.

No covered work constitutes part of an effective technological protection measure: that is to say, distribution of a covered work as part of a system to generate or access certain data constitutes general permission at least for development, distribution and use, under this License, of other software capable of accessing the same data.

Linus Torvalds on DRM and GPL v3

“The difference? The hardware may only run signed kernels. The fact that the hardware is closed is a `_hardware_` license issue. Not a software license issue. I’d suggest you take it up with your hardware vendor, and quite possibly just decide to not buy the hardware. Vote with your feet. Join the OpenCores groups. Make your own FPGA’s.

[..]

Notice how the current GPLv3 draft pretty clearly says that Red Hat would have to distribute their private keys so that anybody sign their own versions of the modules they recompile, in order to re-create their own versions of the signed binaries that Red Hat creates. That’s INSANE.”

The FSF on Linux, DRM and GPL v3

Richard Fontana, a representative of the Free Software Foundation (FSF) stated that GPL 3.0 is intended to stop developers using GPL code to build DRM products, services or protected content. That means not allowing developers to build code that requires a DRM key to be unlocked and certainly not allowing developers to prosecute others for unlocking their code without the use of specially provided DRM keys - or, in other words, "hacking".

“Linus Torvalds has misread it... We require disclosure of the codes if it's necessary to make the software run.”

http://www.theregister.co.uk/2006/02/15/gpl_drm_license/

Debian GNU/Linux

Debian is one of the many distributions of Linux, although in fact Debian emphasises the fact that it is **GNU/Linux** (a bit of digging will reveal that the FSF sponsored its early development), and that it is being developed to run with other free kernels (such as the **Hurd**).

The Debian Social Contract

The **Debian** project is quite forceful in its support of the free software model, and has published the following **Social Contract** (http://www.debian.org/social_contract).

- **Debian Will Remain 100% Free Software**

We promise to keep the Debian GNU/Linux Distribution entirely free software. As there are many definitions of free software, we include the guidelines we use to determine if software is "free" below. We will support our users who develop and run non-free software on Debian, but we will never make the system depend on an item of non-free software.

The Debian Social Contract

- **We Will Give Back to the Free Software Community**
When we write new components of the Debian system, we will license them as free software. We will make the best system we can, so that free software will be widely distributed and used. We will feed back bug-fixes, improvements, user requests, etc. to the "upstream" authors of software included in our system.
- **We Won't Hide Problems**
We will keep our entire bug-report database open for public view at all times. Reports that users file on-line will immediately become visible to others.

The Debian Social Contract

- **Our Priorities are Our Users and Free Software**

We will be guided by the needs of our users and the free-software community. We will place their interests first in our priorities. We will support the needs of our users for operation in many different kinds of computing environment. We won't object to commercial software that is intended to run on Debian systems, and we'll allow others to create value-added distributions containing both Debian and commercial software, without any fee from us. To support these goals, we will provide an integrated system of high-quality, 100% free software, with no legal restrictions that would prevent these kinds of use.

The Debian Social Contract

- **Programs That Don't Meet Our Free-Software Standards**

We acknowledge that some of our users require the use of programs that don't conform to the Debian Free Software Guidelines. We have created "contrib" and "non-free" areas in our FTP archive for this software. The software in these directories is not part of the Debian system, although it has been configured for use with Debian. We encourage CD manufacturers to read the licenses of software packages in these directories and determine if they can distribute that software on their CDs. Thus, although non-free software isn't a part of Debian, we support its use, and we provide infrastructure (such as our bug-tracking system and mailing lists) for non-free software packages.

The Debian Free Software Guidelines

Debian produces a set of guidelines known as the **Debian Free Software Guidelines (DFSG)** which was subsequently adopted with some minor changes by the free software community as the **Open Source Definition**. The latter is produced here: Open source doesn't just mean access to the source code. The distribution terms of open-source software must comply with the following criteria:

- **Free Redistribution**

The license shall not restrict any party from selling or giving away the software as a component of an aggregate software distribution containing programs from several different sources. The license shall not require a royalty or other fee for such sale.

The Debian Free Software Guidelines

- **Source Code**

The program must include source code, and must allow distribution in source code as well as compiled form. Where some form of a product is not distributed with source code, there must be a well-publicized means of obtaining the source code for no more than a reasonable reproduction cost preferably, downloading via the Internet without charge. The source code must be the preferred form in which a programmer would modify the program. Deliberately obfuscated source code is not allowed. Intermediate forms such as the output of a preprocessor or translator are not allowed.

The Debian Free Software Guidelines

- **Derived Works**

The license must allow modifications and derived works, and must allow them to be distributed under the same terms as the license of the original software.

- **Integrity of The Author's Source Code**

The license may restrict source-code from being distributed in modified form only if the license allows the distribution of "patch files" with the source code for the purpose of modifying the program at build time. The license must explicitly permit distribution of software built from modified source code. The license may require derived works to carry a different name or version number from the original software.

The Debian Free Software Guidelines

- **No Discrimination Against Persons or Groups**
The license must not discriminate against any person or group of persons.
- **No Discrimination Against Fields of Endeavor**
The license must not restrict anyone from making use of the program in a specific field of endeavor. For example, it may not restrict the program from being used in a business, or from being used for genetic research.
- **Distribution of License**
The rights attached to the program must apply to all to whom the program is redistributed without the need for execution of an additional license by those parties.

The Debian Free Software Guidelines

- **License Must Not Be Specific to a Product**

The rights attached to the program must not depend on the program's being part of a particular software distribution. If the program is extracted from that distribution and used or distributed within the terms of the program's license, all parties to whom the program is redistributed should have the same rights as those that are granted in conjunction with the original software distribution.

The Debian Free Software Guidelines

- **License Must Not Restrict Other Software**
The license must not place restrictions on other software that is distributed along with the licensed software. For example, the license must not insist that all other programs distributed on the same medium must be open-source software.
- **License Must Be Technology-Neutral**
No provision of the license may be predicated on any individual technology or style of interface.

Source (http://www.opensource.org/docs/definition_plain.html).

Debian Pros & Cons

Debian as a whole has some of the following advantages:

- 1 strongly pro-free software, so an excellent choice for this priority with thousands of packages;
- 2 exceptional support of bug tracking, package management etc.;
- 3 excellent for servers, easy to maintain.

However, there are some downsides:

- 1 it is not the most “friendly” distribution to install;
- 2 makes little effort to provide the kinds of charged services that other distributions provide (e.g. printed manuals, telephone support).

Installing Debian

The Debian installer has come on enormously in recent years, and generally installing Debian is a snap.

- A lot of information can be found at `http://www.debian.org/distrib/`.
- You probably want either a net install, or multi-arch DVD.
- There is a link to a manual for any one of 11 architectures at `http://www.debian.org/releases/stable/installmanual`.
- Always take particular care when installing a dual operating system machine that you have fully read the instructions for installing the **boot loader** and **partitioning**.
- Be careful to pick the appropriate **distribution** (stable, testing, unstable, experimental).

Installing Debian from USB

An interesting approach is to install Debian using a USB stick. Here's how to prepare the stick, and you really won't need a large memory stick. First, get the files you will need.

```
1 wget ftp.uk.debian.org:/debian/dists/sid/\  
main/installer-i386/current/images/netboot/\  
debian-installer/i386/linux  
wget ftp.uk.debian.org:/debian/dists/sid/\  
main/installer-i386/current/images/netboot/\  
6 debian-installer/i386/initrd.gz  
wget ftp.uk.debian.org:/debian/dists/sid/\  
main/installer-i386/current/images/netboot/\  
mini.iso
```

If you want to install AMD64, substitute it for each i386 above.

Installing Debian from USB

Switch to a virtual terminal (Alt-F1) and put in the memory stick that you **will completely overwrite!**. The system console will probably tell you where the device is `/dev/sda` or similar. If the stick is automatically mounted, then unmount it using `umount /dev/sda` or similar.

If nothing appears on the system console, then try looking at the end of your `syslog` or `messages` log.

Installing Debian from USB

Switch to a virtual terminal (Alt-F1) and put in the memory stick that you **will completely overwrite!**. The system console will probably tell you where the device is `/dev/sda` or similar. If nothing appears on the system console, then try looking at the end of your `syslog` or `messages` log. If the stick is automatically mounted, then unmount it using `umount /dev/sda` or similar.

Installing Debian from USB

Try to install the following packages, here is the command for a Debian based system.

```
apt-get install mbr syslinux mtools
```

Now make the stick bootable

```
install-mbr /dev/sda  
syslinux /dev/sda1
```

Remember, you might need to use /dev/sdb or whatever instead.

Installing Debian from USB

Now mount the device.

```
mount /dev/sda1 /media/usb
```

you might need to create a /media/usb directory temporarily if it does not exist. You could also try to unplug and plug-in the device if your current OS will automount it.

Copy the linux kernel, initial ram disk, and iso image onto the device.

```
cp linux initrd.gz mini.iso /media/usb
```

Installing Debian from USB

Copy on a configuration file, and then unmount.

```
4 echo -e "default_linux\nappend_priority=low_\n  vga=normal_initrd=initrd.gz_\n  ramdisk_size=12000_root=/dev/ram_rw"\n  > /media/usb/syslinux.cfg\numount /dev/sda1
```

That's it, now just insert into the target machine and ensure its BIOS is configured to boot from USB.

Summary

- We have seen some of the motivation in the creation of open source and free software, and that there is a subtle distinction between the two.
- There are varying grades of militancy in the support of OSS, and this can even be seen in the various Linux / GNU/Linux distributions available.
- The free software movement is stronger than ever, with vast projects spanning continents, and frequently world leading products like Apache.