

(History of) **Free Software Foundation** **in a Nutshell**

it's all about Saints

Philosophy , Freedom, Human Rights

... and printers!!

First Edition October 4th 2012
Marco Massari Calderone
MarcoMC.com

This is 'Free Software Foundation in a Nutshell', inspired by **Free as in Freedom: Richard Stallman's Crusade for Free Software** and other articles found on <http://www.fsf.org>

Copyright© 2012 Marco Massari Calderone

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License."

Copy of this document is available on the website <http://marcomc.com>

[Introduction](#)

[RMS: It's not the Royal Meteorological Society](#)

["Prelude to Foundation": Of Hacking And Philosophy](#)

[1971: the Hackers community](#)

[1977: passwords](#)

["Forward the Foundation": For Want of A Printer](#)

[1979: Xerox](#)

[1982: Symbolics and Lisp Machines Inc.](#)

[1983: the GNU Project](#)

["Foundation": Copyright, Right-to-Copy... Copyleft!](#)

[1985: Free Software Foundation, the movement and GNU Manifesto](#)

[1989: GPL, EMACS, GCC, GDB](#)

["Foundation and Empire": March of the Penguins](#)

[1991: Linux](#)

[1993: Debian GNU/Linux](#)

["The Foundation's Edge": Shock To The System](#)

[1996: First Conference on Free Redistributable Software](#)

["Second Foundation": Free Software vs Open Source](#)

[1998: Netscape, Freeware Summit and Open Source Initiative](#)

["Foundation and The Empire": The Day After Tomorrow](#)

[2012: FSF Campaigns](#)

["Foundation and The Earth": about Licenses, Patents, Innovation and Human Rights](#)

[Quick Facts About Licenses](#)

[Overview on Patents](#)

Introduction

This document was born with the intention to give a summarization of what the Free Software Foundation is, its origins and achievements and of course about his founder Richard Stallman.

The content is inspiration by **Free as in Freedom: Richard Stallman's Crusade for Free Software** and other articles found on <http://www.fsf.org>, and is meant to be very short and absent of many details.

I do not have any intention to to rewrite a detailed description of the subjects that will be mentioned in this document, as the internet is full of a more accurate quantity of information.

My wish is that reading the following few pages you will have a glance of what the Free Software Foundation and the Free Software stand for, and you will find encouragement to dig yourself about the aspects that will interest you the most.

I apologize in advance if my summarisation will bring any mistake, please contact me via email at marco@marcomc.com to notify me, and I will take care amend the document accordingly.

My favorite quote from 'Free as in Freedom' is

"Few things irritate the hacker mind more than inefficiency"

I deeply hope that some of you will appreciate the various references to Asimov's publications.....

RMS: It's not the Royal Meteorological Society

RMS is the common abbreviation used to refer to Richard M. Stallman the creator of EMACS and founder of the Free Software Foundation and of the GNU project and inspirator of the Free Software movement.

He is the man that allegedly said *"Since I was driving, I was also losing time to answer my email, and that's a real pain since I can barely keep up anyway."*

Stallman's alter-ego is Saint IGNUcius a saint of the Church of Emacs, the credo of the Church of Emacs is: *There is no system but GNU, and Linux is one of its kernels.*

<http://stallman.org/saint.html>

He also recognises that the use of a free version of the editor "Vi" it is NOT a sin in the Church of Emacs, but it's a penance.

According to it, Vi is 'the editor of the beast', in fact Vi-Vi-Vi, read as Roman numbering, correspond to 6-6-6.

“Prelude to Foundation”: Of Hacking And Philosophy

1971: the Hackers community

In 1971 Stallman is a Student at Harvard university and when he discovers that the MIT is at walking distance from Harvard he joins the MIT hacker community and find the way to gain a job at the MIT Artificial Intelligence Laboratory.

Among the hacker community he embraces an hacker culture that is characterised by a rigorous moral sense, according to which an hacker would never damage an information system or a software application and have an instinctive sense of brotherhood, solidarity and equality.

He is strong on his believes, and he never accept compromises, and has a strong predilection about jokes and pranks, he almost get a snake elected at the head of his campus dormitory... doesn't he remind you about Jobs and Wozniak? Probably yes if you have read their biographies.

1977: passwords

The hacker culture is mostly about freedom,so far that hackers at the MIT would open professors' offices to “liberate” sequestered terminals, just because is amoral to prevent the use of a computer, (probably) consider a freedom right.

In 1977 the MIT Laboratory of Computer Sciences install a password control system, that makes Stallman infuriating because it consists in a violation of the freedom and sharing principles he strongly believes in.

When Stallman finds a way to decrypt the passwords he sends users messages containing their decoded password, with a suggestion to change it to the empty string re-enabling anonymous access to the systems.

Note: At my advice, here you can catch another parallelism to Steve Jobs who is pretty subversive on the matter of rules and traditional thinking. In fact Jobs uses the pirate flag to represent his working team at apple during the development of the first Mac to take distances to conducts of the rest of Apple.

Please don't be confused, I'm not implying that Stallman and Job are similar, I am just saying that the core of their values may have been similar, but developed differently, but this is another story I may write about another day.

“Forward the Foundation”: For Want of A Printer

1979: Xerox

It is 1979 the Artificial Intelligence Lab receives a new printer as gift by Xerox. Annoyingly the printer is frequently jamming and that is making people losing a lot of time because they would discover about the jam only after they go to collect the supposedly printed

paper.

Because of that inefficient behaviour of the printer, Stallman decides that he will fix the printer's software to send an alert whenever a jam would occur.

It has to be said that until then every device manufacturers is used to provide a copy of the source code of the software that operates the machinery, but this time Stallman discovers that no source code is shipped. He finally came to know that at the Carnegie Mellon university there is a copy of the sources, but when he shows up to reclaim a copy for his laboratory, with his horror, he is denied it because the possessor of such sources had signed a nondisclosure agreement with the company that developed the printer software (Xerox partner company).

That is the first time that Stallman perceives as harmful the application of nondisclosure agreements to a software and he considers it very dangerous for the people as much as it has been for himself, particularly because it isn't a denial just to his persona, but it is an 'impersonal refusal' against anybody without any reasonable excuse, preventing him or anybody else to benefit of any positive modification that anybody could have applied.

That violates the society principles of brotherhood and mutual help via the sharing practice.

1982: Symbolics and Lisp Machines Inc.

Not far from the episode of the Xerox printer, in 1982 the hacker community of the MIT disaggregates having the staff of the AI Lab to leave their jobs to join two new spin-off companies: LMI and Symbolics. Stallman is one of few that refuses to abandon the Lab.

The two companies provide softwares to the MIT and shortly they stop providing the source codes of their programs for fear that the concurrent company would take advantage of the reciprocal work. In a short period most of the machines of the laboratory will be running proprietary software that would make the MIT commercially dependant from the software producer.

1983: the GNU Project

With the disappearance the MIT's hackers community and all the above mentioned changes at the AI Lab, Stallman decides that it is no more a place for him.

He deeply refuses to join any proprietary software company because that would mean that one day he will end up developing software covered by nondisclosure agreements 'adding the pressure to other people to betray their fellows too'.

He already experienced on his own skin the pain of a refusal due to a non-disclosure agreement, and he feels that participating at that vicious system would made the world a worse place.

At my personal advice this feeling is crucial to understand the whole philosophy of Stallman and of his movement afterwards.

Isn't the 'society' trying make you feel guilty if you download a 'pirated' copy of a software today?. I believe that this is the burden he wanted to avoid to give people.

Until then there is no concept of Free Software, because there is no need for that as it is free 'by nature'. The software with no source code, came as an unnatural imposition by the software companies.

The concept of 'natural' is also important to understand Stallman future positions.

Stallman understands that to protect his freedom and everybody's freedom to use a computer without depending from any third-party embodiment, it is needed a complete operating system that should be accessible and developed by a community of hackers.

In 1993 Stallman takes the decision to step away from the MIT and leaves his position as a programmer at the AI Lab, and decides to dedicate his full time to a new project named GNU (GNU's Not Unix).

The Free Software is born. The Free Software movement will begin... and that's just the beginning.

Quote from a recent interview to Richard Stallman: [The Law of Success 2.0](#)

".. In regard to software, proprietary software does not respect users' freedom because the program controls the users.

If the users aren't free to change a program and do so either individually or in groups cooperating, then the program controls the users.

Now, with typical proprietary software there is even a licence that says what users are allowed to do with the program and what they're not allowed to do and it can be as restrictive as the developer chooses to make it.

For instance, there is a Microsoft program for managing webpages, websites, and its licence says it can't be used to publish anything that criticises Microsoft.

So here, nonfree software takes away your freedom of speech."

“Foundation”: Copyright, Right-to-Copy... Copyleft!

1985: Free Software Foundation, the movement and GNU Manifesto

Free Software stands for Free as in ‘freedom’, not for Free as in ‘free beer’.

The Free Software is not intended to be free of charge, in fact Stallman itself is selling tapes with copies of his Emacs editor to finance his job on the GNU project. The Free Software is intended to give (consolidate) the users with their [natural] rights to freely use the software without limitations.

Stallman decides to found the non-profit organisation called Free Software Foundation in 1985, with the intent to support the development of the GNU system, and to support as well the Free Software sustainers and their movement. At the same time the GNU Manifesto is also written to express to the community the commitments of the FSF about the project.

Definition of Free Software:

So the definition of free software is the four freedoms that are needed for the users to have control of their computing.

1. **Freedom zero** is the freedom to run the program.
2. **Freedom one** is the freedom to study the source code and change it so it does your computing as you wish.
3. **Freedom two** is the freedom to help others, which is the freedom to redistribute exact copies.
4. **And freedom three** is the freedom to contribute to your community, which is the freedom to distribute copies of your modified versions.

*So these four freedoms ensure that the users, both individually and collectively, control the program. **If the users don't control the program then the program controls the users. That's proprietary software and that is what makes it evil.***

1989: GPL, EMACS, GCC, GDB

Stallman is the creator of many tools and software and he realises that he need to protect the freedom of the users to use them. He need to prevent people to take his creations and redistribute them in a non-free form.

Because the Copyright is a lawful matters, and Stallman can not just ignore it although it is apparent that the restrictiona of a copyright agreement clash with his concept of freedom, he decided to react in a very hacker way: he bends the copyright to act against itself.

He finally create the **GNU Emacs General Public License**, a copyright license that define the Right-to-Copy, and it only restrict the non-free redistribution of the software.

This license introduces the innovative concept of the Viral Licensing, that oblige the redistribution of the software and its derivatives under the same version of the license or newer. This restriction guarantees the users that they will never will be denied their freedom of the software due to a change of license from third-party.

With the release of the GNU C Compiler (GCC) and the GNU Debugger (GDB) Stallman will modify his license to simply be the **GNU General Public License** and that license will be used for every new program create by the FSF and not only.

An exception to the GPL is the LGPL or Lesser GPL that allows the use of the GNU Libraries in non-free software, that explicitly explained as a strategic move intended to 'poison' the proprietary software to slowly move to free software, making it dependant to Free components. The concept is that if the proprietary software is allowed to use Free libraries than the proprietary software producers will not urge to develope they own proprietary versions of the libraries, weakening the proprietary software eco-system and depending more an more by Free components.

This new licensing principles will be defined as **Copyleft** because they turn the copyright principles on the opposite way, towards the users and not against them.

“Foundation and Empire”: March of the Penguins

1991: Linux

In 1991 the Finnish student Linus Torvalds develops the first version of the Linux kernel that very soon becomes the main kernel of the GNU system because the development of the official GNU Hurd kernel is behind schedule in it's development. GNU Hurd kernel will be released only more than 10 years later.

The version **0.99** of Linux is released under the GNU GPL license in the year 1992.

Linux mascot is TUX, a penguin!

1993: Debian GNU/Linux

Linux is just a system kernel, useless by itself, GNU is a system that need a kernel, and uses the Linux kernel on most of its instances.

When Linux and the GNU softwares are used together, they form an Operating System that is good convention to call as the 'GNU/Linux' Operating System.

Using the name 'GNU/Linux' is very appreciated in the Free Software community, and you will show-off if you use it, because you will show you 'know' what you're talking about!

Many companies start distributing and selling bundles of disks containing the 'GNU/Linux' operating system any many personalised flavours.

The principles and philosophy behind the organisation distributing GNU/Linux varies and many times differ from Stallman's visions of freedom and frequently they include in these bundles non-free or even proprietary software.

In 1993 the Free Software Foundation endorses Ian Murdock in the foundation of the non-profit Debian Project, an independent and decentralised organisation that is not backed by a company like Ubuntu, openSuse, Fedora or Mandriva are instead.

The project gives born to the **Debian GNU/Linux** operating system.

As per Wikipedia definition:

The Debian Project is governed by the Debian Constitution and the Social Contract (http://www.debian.org/social_contract), which set out the governance structure of the project and explicitly states that the goal of the project is the development of a free operating system.

The Debian Social Contract guarantee that Debian will always be Free.

Today the Free Software Foundation do not endorse any GNU/Linux distribution, not even Debian GNU/Linux because it host on the servers 'non-free' and 'contrib' repositories although they are not part of the system itself, and because some of the installation packages propose to the user to install non-free software, therefore encouraging the use of 'evil' programs.

“The Foundation’s Edge”: Shock To The System

1996: First Conference on Free Redistributable Software

In Cambridge, Massachusetts, year 1996, the Free Software Foundation sponsorises the first conference on Free Redistributable Software.

At the conference participates Linus Torvalds, the creator of the Linux kernel, who will ‘shock’ the public admitting being a fan of Microsoft’s Powerpoint. That draws immediately a line between the hackers community purists and those who believed that

“Being a hacker wasn’t about self-denial, it was about getting the job done, and the job, for them, was defined in practical terms”.

At the same conference participates Eric Raymond a developer quite known in the community and author of the ‘fetchmail’ program, that later will analyse and describe the reason behind the huge success of of Linus Torvald’s work defining the Linux Development Model as the ‘Bazaar’ model that differs from the ‘Cathedral’ model adopted by the rest of the world including Richard Stallman.

“Second Foundation”: Free Software vs Open Source

1998: Netscape, Freeware Summit and Open Source Initiative

After Raymond's publication of the book [The Cathedral and the Bazaar](#), In 1998 Netscape is convinced to release the source code of its popular -but declining- browser called Netscape. Today's inheritance of Netscape's choice, is the Mozilla Foundation and the browser Firefox.

With the goal to convince more companies to embrace Netscape's decision, the technology publisher Tim O'Reilly organises the 'Freeware Summit' in the same year, with the participation of Netscape representatives and Raymond (and others).

To this group of people, the term 'Free Software' sounded too scary and misleading for the opinions of the software producer companies, because of their money-driven nature they would probably misinterpret the word 'free' in the wrong exception of 'free-of-charge'.

Because of that, before the end of the summit, they define the new term '**Open Source**', based on values very different from the FSF ones, but that are more appealing to the industry of software: **powerful, reliable, cheap, advanced**.

The term 'Open Source' is rejected by Richard Stallman, because **he fears that the Open would prevail** over the Free Software Foundation values in the long run if the this name is used to identify the actual Free Software, and here is where the community will split between the Free Software movement and the Open Source movement.

The Open Source Initiative (<http://opensource.org/>), is created and Raymond will shortly become the spokesman of the Open Source community.

Stallman thinks that *“open source,” while helpful in communicating the technical advantages of free software also encouraged speakers to soft-pedal the issue of software freedom.*

Famous quote of Raymond relating to Stallman doctrine:

“Shut Up and show them the code!”

“Foundation and The Empire”: The Day After Tomorrow

2012: FSF Campaigns

- [Secure Boot vs Restricted Boot \(http://www.fsf.org/campaigns/\)](http://www.fsf.org/campaigns/)
It's about preventing hardware manufacturers to implement boot restrictions in a way that will prevent users from booting anything other than Windows.
- [Working together for free software](#)
- [The GNU Operating System](#)
- [DefectiveByDesign.org](#)
FSF anti-DRM campaign, where we mobilize large vocal communities to reject products from businesses that insist on using to DRM to control their customers.
- [Windows 7 Sins](#)
- [PlayOgg](#)
Promotes the use of free audio and video formats unencumbered by patent restrictions
- [End Software Patents](#)
FSF campaigns for formats that are free of software patents and the more fundamental task of ending software patents entirely, through legal and legislative action.
- [Campaign for OpenDocument](#)
Campaign that fights for the use of free formats in government documents, pushing governments to adopt policies requiring that all digital public documents and information be stored and distributed in formats that are standard, open, and royalty-free
- [RIAA Expert Witness Fund](#)
- [High Priority Free Software Projects](#)
- [LibrePlanet](#)
- [Campaign for Hardware that Supports Free Software](#)
- [Free BIOS Campaign](#)
- [Campaign against the ACTA](#)
Campaign against the ACTA (Anti-Counterfeiting Trade Agreement) and other international agreements that undermine people's right to control technology

“Foundation and The Earth”: about Licenses, Patents, Innovation and Human Rights

Quick Facts About Licenses

GPLv3 main differences with GPLv2 include:

- Anti-Tivoization

- DRM Regulations
- Patent Regulations that guarantee a not expiring permission to use the patented feature for code release under GPL
- Do not requires the disclosure of the source code as far as a copy of the software is not provided to the client.

The Linux Kernel still adpts GPLv2

GCC and SAMBA have moved to GPLv3 and that is why Apple has abandoned them.

Apple has his own FSF approved Free License, APSL (Apple Public Source License).

X11 License is too open that allows third parties to develop and distribute non-free copies of the software.

GNOME was born as counterpart of KDE that was base on the non-free QT libraries. That lead to the release of QT and KDE sources under a free license.

Some licenses may help to violate human rights, like forcing to declare your real full name in services like Google+ and Facebook.

Overview on Patents

“A software patent is a 20-year monopoly on the use of a feature in a computer program. When a company has a patent on nested menus, a video format, or pinch-to-zoom, then no one else can implement that feature for 20 years unless they get permission from the patent holder. With tens of thousands of software patents, the result is that software development is not safe for small- and medium-sized businesses, for non-commercial projects, and for individuals.”

In the USA the Patents System was born with the intention to protect the people (not the company) from the recycling of the ideas by concurrent parties, so to push the market to keep offering innovative products.

Unfortunately in the software-world the 20-years monopoly term is counter productive for the development of the technology, sometimes days are enough to make a software technology obsolete. But this is not the only bad reason to adopt software patents.

It's then the case where a law for the people turns against the people.

In Europe consider unlawful the use of Software Patents, with the exception of Germany.. in fact the Fraunhofer Institute request a royalty for every mp3 player implementation. Many hackers choose Europe to operate.

USA repeatedly tries to force Europe to embrace the use of Software Patents.

Paraphrasing Stallman's speeches:

“When USA has something, instead of fixing it, it tries to export it to other countries.”

"I bless your computer my child"