Free Software - Open Source Business Models

Franco Gasperoni

gasperoni@act-europe.com

Legal Background: Copyright

Exclusive legal right of copyright holder to

- **Copy**

- **Distribute**

- **Modify**

- **Display**

- **Perform**

- **Exploit a work**

Software and Copyright

- **Both source and object code can be copyrighted**

- **Loading software onto a computer is considered copying**

Copyright & Licensing

- A copyrighted work cannot be copied unless…

- … the copyright holder grants you a license …

- … permitting copies under specified circumstances

Virtually all software today is sold with a license

The Free Software Movement (ca 1980)

- *Freedom* to run

- *Freedom* to redistribute copies

- *Freedom* to study and adapt

- *Freedom* to improve it and release the improvements

FREE AS IN FREEDOM

RICHARD STALLMAN &
THE FREE SOFTWARE FOUNDATION

SAM WILLIAMS
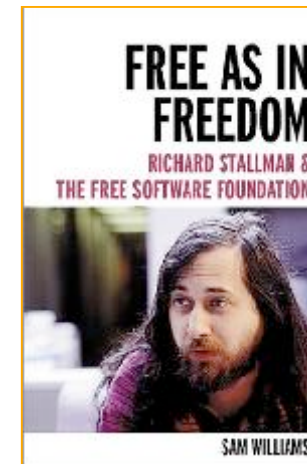
Free Software Is Copyrighted Software

To protect these freedoms *Free Software* comes with a license

- The General Public License (GPL)

From this point there is NO DIFFERENCE between

GPL: The License of *Free Software* (FS)

- **The GPL is written to favor FS users**

- **Specifically the GPL guarantees:**

  - *Freedom* to run

  - *Freedom* to redistribute copies

  - *Freedom* to study and adapt

  - *Freedom* to improve it and release the improvements

- **Examples of FS:**

  - *Emacs, GCC, GNAT Ada, GNU/Linux, …*

The Meaning of *Free* in FS

Freedom

- You can sell it

- You can make it available for free

FS is a matter of liberty not price

Open Source Software (OSS)

- **Providing the sources** *(under some license)*

+

- **Encouraging a wide community to participate in development**

There have been abuses in the licenses used

- **Open Source Initiative (OSI) created to**
  - Define what licenses qualify as "Open Source"

The OSS Movement

- **Attractive to major companies (e.g. IBM, SGI, HP, …)**

  - Can leverage on a larger developer community

- **Claims are made for better quality, better security etc.**

- **In practice:**

  - Some OSS projects work, some don't.

  - Some OSS software is high quality, some is not.

  - Some projects make sense as OSS some don't.

FS and OSS

- **One of the important freedoms for FS is**

  - The freedom to modify, which means that sources are available.

- **So it is often, but not always, the case that FS:**

  - Ends up with an open source community participating in development.

  - E.g. Linux

- **Not all OSS projects are FS because of the license**

FS/OSS and COTS

Commercial Off-The-Shelf (COTS)

- **Most people look for COTS software**

  - Economies of scale

  - Reduced Costs

  - Inexpensive way to stay with the state of the art in technology

  - User community

COTS and Closed-Source Software

- **Two big downsides**

- **Vendor lock in for support**

  - Only the vendor can provide support

  - This can be locked in with licenses etc

  - If the vendor goes bankrupt, too bad

  - Source escrows are not much help

- **Vendor lock in for modifications**

  - If the software does almost what you want, but not quite, you have to ask the vendor for changes

  - This can be arbitrarily expensive

Free Software Licensed COTS

- **Fixing the two big downsides of COTS**

- **NO vendor lock in for support**
  - Everyone has access to the sources
  - Anyone can provide support
  - You can even build your own support
  - If there is a demand other companies will compete

- **NO vendor lock in for modifications**
  - Everyone has access to the sources
  - Anyone can do modifications
  - You can do modifications yourself if you like

COTS + FS = COTS without the risks

Worrying about Licenses and Quality

FS, OSS, and Proprietary Software share 3 common truths

- **CHECK THE LICENSE**

  - Make sure it is suitable for your use

- **CHECK THE QUALITY**

  - No software license guarantees quality

  - Use your normal procedures to ensure that you choose quality software

  - Buy SW products whose business model aligns with your quality needs

"Quality" and
FS/OSS Business Models

FS/OSS and Dependable Systems

SW in a dependable system:

- Part of an auditable & repeatable process

- With stringent "quality" requirements

- What are the quality guarantees for FS/OSS ?

FS/OSS Product With No Support

- **Supplier sells FS/OSS applications**

  - Perhaps with some installation help

- **E.g. previously commercial GNU/Linux distributions**

- **Can check quality by inspecting the sources … ☺ ☹**

- **This is an advantage over conventional proprietary SW**

- **Not particularly attractive to developers of Dep. Sys.**

Dual License

- **Available to FS/OSS companies that own 100% copyright**

- **Whose products are included in the sw developer's code**

- **E.g MySQL, Cygwin**

- **Relies on vendor lock in**

- **No additional advantage over previous model**

Infrastructure Provider

- **OSS development website**
  - E.g. OSDN, SourceForge

- **Leverages on large developers community**

- **Free for basic services, fee for advanced web browsing**

- **Revenue from some advertising**

- **SourceForge Enterprise Edition**
  - To manage and execute offshore and distributed team development

- **Interesting for large/distributed teams**

Pure Service

- **E.g. Alcove, IBM Global Services for GNU/Linux**

Different from "traditional" service models in that:

- **Consultants have access to the sources**

- **Can contribute to OSS efforts**

- **Allows deeper level of consultants know-how**

Sell the Artifacts

- **SW in a dependable system:**

  - Part of an auditable & repeatable process

  - With stringent quality requirements

- **SW in a dep. sys. = sources + build scripts + artifacts**

- **Provide the artifacts for FS/OSS product and sell them**

  - Creation of artifacts is not the main focus of FS/OSS

Software Coops

- Coop to share resources and know how

- To develop artifacts for FS/OSS application

- More generally to guarantee FS/OSS quality

- For the members of the coop

Leveraged Service

- **FS/OSS product with expertise-based service**

- **Provided by the developers of the FS/OSS product**
  - E.g. AdaCore and GNAT Pro

- **Quality guaranteed by aligning interests with customer's**
  - Subscription-based model
  - Quality can be verified on an ongoing basis
  - Quality feedback loop in place
  - If poor quality/service subscription not renewed

Quality is an ongoing process

Conclusion

Common truths of FS, OSS, and Proprietary Software:

- **CHECK THE LICENSE**

- **CHECK THE QUALITY**

- **CHECK THE BUSINESS MODEL**

  - Make sure it is aligned with your interests