

# Free Software Development from the IT Industry

VI Jornades de Programari Lliure  
Campus de Vilanova i la Geltrú, UPC



8 July 2005

*Sergio Talens-Oliag*  
**sergio.talens@t-systems.es**

# Introduction

- IT related companies develop software that can be published using proprietary or free licenses:
  - Should IT companies cooperate with free software projects?
  - Should IT companies publish its own developments under free software licenses?
- What has to do an IT company to publish the code of it's own projects?
- How would a company keep a free project alive?

# IT business models

- Sell use licenses (own programs of third party),
- Personalization of the software for the client,
- Systems integration, which usually implies development,
- Software maintenance,
- Software use training,
- Technical support,
- Hardware support.

# Why use free software?

- Personal reasons:
  - It is Fun!
  - It is the right thing to do, ideologically
- Technical reasons:
  - Sometimes it is the best tool for the task
  - If it breaks or needs more functionality can be fixed by any skilled person.
- Economical reasons:
  - Usually it has zero or very low cost

# Why develop free software? (1)

- Personal reasons:
  - It is Fun!
  - It is the right thing to do, ideologically
  - Ego
- Technical reasons:
  - Problem not solved by existing software
  - To learn how to do something

# Why develop free software? (2)

- Economical reasons:
  - Free Software development is part of the work (client requirement or company strategy)
  - Publishing source code can be good for the CV, i.e. undergraduate students
  - Sometimes it is much cheaper to fix or improve an existing project than start a new one or buy commercial software that needs to be customized

# Reasons to keep code secret (1)

- Other developers might copy some of the code and use it in other programs
- Hackers will find vulnerabilities in the code that will enable them to to develop viruses, spy ware or other malicious code for it
- Public disclosure of the source code could expose its developers to charges that some of the code was plagiarized from other programs

# Reasons to keep code secret (2)

- Customers will try to modify the source code, resulting in new problems that could be difficult for either the customers or developer to correct
- The source code could be used as evidence in legal proceedings, particularly those related to whether the developer has been complying with certain legislation or court decisions
- The source code could contain unfavorable comments inserted by programmers about their employer, customers or competitors



# Reasons to publish source code (1)

- As a general rule, when the money is not on the code:
  - *Hardware vendors*: on the drivers case, the software is tied to specific hardware. Publishing the code allows free systems to support it and makes the potential customer base bigger
  - *Service providers*: can reduce maintenance costs (users and external developers can help on fixing bugs or with improvements), for example if the program has to be ported to different systems not available for the original developers

# Reasons to publish source code (2)

- When it can be used as publicity, mainly between free software users and developers. Usually done by companies that sell services and hardware, but sometimes also by software vendors
- To reduce maintenance costs and improve its quality and functionality, publishing it allows a much larger and more diverse set of qualified people examine the source code, thus resulting in the discovery of more bugs and providing more and better suggestions for improvements and extensions

# Reasons to publish source code (3)

- Note that publishing code not always means giving it away for free (as in free beer), different licenses can be used.

# User and developer communities (1)

- The power of the free software development model resides on the participation of many individuals and groups in the development, testing, quality assurance and advocacy of the projects they work on
- The distributed development is made possible by the availability of source code and the use of licenses that don't limit the redistribution of sources in original or modified form

# User and developer communities (2)

- The developer community will only be formed and maintained if the people has a reason to belong to it; the community members can be paid workers or volunteers, the first group can belong to one or multiple companies interested on the project and will usually keep working on a project as their daily work, but in a lot of cases the work is done by volunteers on their spare time, so they have to feel that their work is taken into account by the project leaders

# Free projects from IT companies (1)

- If it is already done or is being developed, audit the code; to be able to give away source code a company or institution has to make sure who is the copyright owner of it and if it can be distributed under a free license
- If it is not ready, look for similar projects and see if you can cooperate on them and reuse as much as possible; usually that will be faster and cheaper and, if the license allows it, you will be able to use your modifications or fork the original project if needed

# Free projects from IT companies

## (2)

- Avoid the vaporware effect; don't announce or publish projects until there is something to play with, a bad first impression can keep a lot of people away from a project for a long time
- Provide human resources: if the software is going to be used by the company there has to be someone in charge of the project who is able to review and integrate patches and handle releases (or at least keep the company's branch under control)

# Free projects from IT companies

## (3)

- Provide infrastructure for the project:
  - Set up a website to put the project's home page and give pointers to other project resources
  - Use a public revision control system (like cvs, svn, arch,...) and give write access to developers working on the code
  - Provide communication channels for developers and users: mailing lists, irc channels, forums, etc.
  - Install a ticketing or bug tracking systems to provide user support and control bugs



# Free projects from IT companies

## (4)

- Keep the interest; once the project is open and you have a community try to follow the release early, release often principle. Try to work incrementally, usually is better to have small improvements in short periods than having long delays between big changes
- Try to provide documentation; cooperation is always easier when the project has good documentation for users AND developers

# Example projects

- *Free software projects*: GNU Software, Apache, GNOME, Debian, ...
- *Mixed projects*: Ghostscript, MySQL, Wine, KDE/Qt, ...
- *Open sourced software*: Mozilla (Netscape), OpenOffice.org (Sun), Hula (Novell), Blender, ...
- *Failures*: thousands of projects at sourceforge without enough developers or user base