

# Libre Software Communities: An Approach

## Master on Free Software

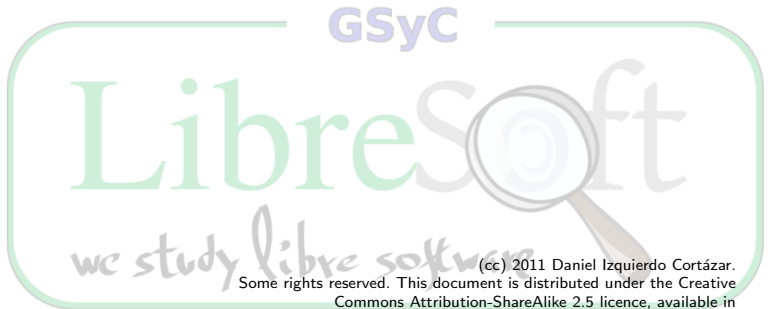
Daniel Izquierdo Cortázar

dizquierdo@gsyc.escet.urjc.es  
GSyC/Libresoft

14 de enero de 2011



Universidad  
Rey Juan Carlos



(cc) 2011 Daniel Izquierdo Cortázar.  
Some rights reserved. This document is distributed under the Creative Commons Attribution-ShareAlike 2.5 licence, available in <http://creativecommons.org/licenses/by-sa/2.5/>

# Index

## 1 Introduction

## 2 Legal Aspects

## 3 Image

- Web site
- Source code

## 4 Tools

## 5 Announcing

## 6 Conclusions



# What a project needs is:

- Communication among its members
- Skilled people on the communication technologies
- All the tools are needed because you are not working alone
- You have to be aware of the rest of the work
- You have to make the rest be aware of your work

## Take a look around you

- Is somebody doing the same work?
- Try to look for information at:  
Forges: SF - <http://sourceforge.net>, Berlios - <http://berlios.de>  
Project's indexes: Freshmeat <http://freshmeat.net>, FSF directory <http://directory.fsf.org>
- Is better to unify efforts?

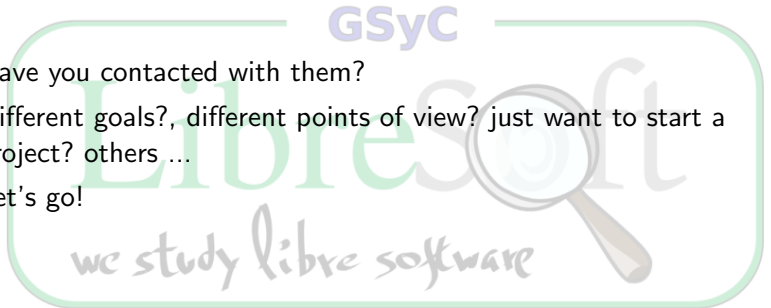
## To unify efforts or not

- GNOME - KDE
- Beryl - Compiz
- Several distributions (Debian, Suse, Fedora, Mandriva, Ubuntu, Linex, GuadaLinux, ...)



## Take a look around you

- Have you contacted with them?
- Different goals?, different points of view? just want to start a project? others ...
- Let's go!



# First steps

- Mission limits
  - What your product will do... (List of features)
  - What your product will not do
- Name
  - What's your name?*
  - Apache!*
  - Nice to meet you :)*
- Have a clear mission statement
- Alpha, Beta



# Index

- 1 Introduction
- 2 Legal Aspects
- 3 Image
  - Web site
  - Source code
- 4 Tools
- 5 Announcing
- 6 Conclusions

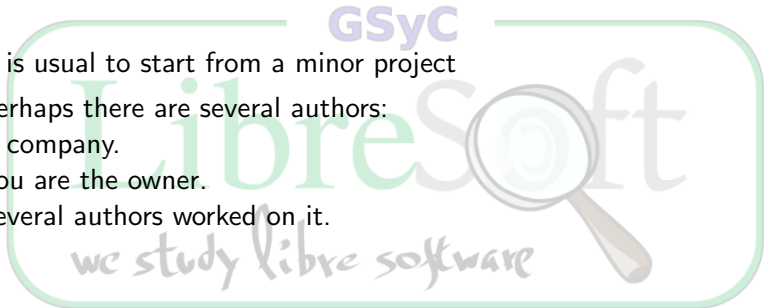


## License, what is the best?

- Source code (remember):  
GPL family (+ restrictive)  
BSD family (- restrictive)
- Documentation:  
Creative Commons?
- You could create your own license, however...

# License

- It is usual to start from a minor project
- Perhaps there are several authors:
  - A company.
  - You are the owner.
  - Several authors worked on it.



## Pieces of advice

- Your web site should contain a clear reference to the license you use.
- Source code:  
License could be included in each file.  
Every file could contain a reference to the license (COPYING or LICENSE)

# Index

- 1 Introduction
- 2 Legal Aspects
- 3 Image
  - Web site
  - Source code
- 4 Tools
- 5 Announcing
- 6 Conclusions



# Visibility

- Basic to:  
Attract new users, developers.  
More users = more bug reports = more activity = ...



# Structure

- Basic information (Remember! first steps!)
- Hot news
- Activity and maintenance in mailing lists, wiki, trac ...
- Related projects
- Projects which use our product

# Documentation

- User documentation
- Developer documentation: specific mailing lists, SCM URL, BTS, others
- Developer documentation: architecture, style, rules, others





# Examples

- Plone
- GNOME projects
- SourceForge



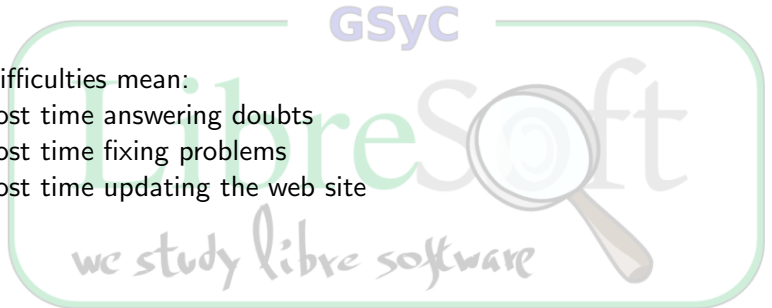
# Users

- Expert, medium and beginner
- Easy, clear and fast installation
- Good documentation



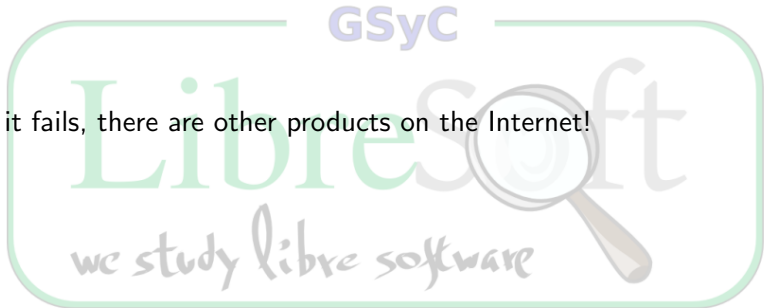
# Installation process

- Difficulties mean:
  - Lost time answering doubts
  - Lost time fixing problems
  - Lost time updating the web site



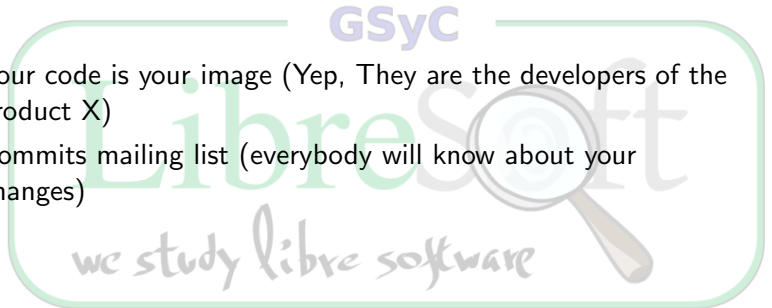
## Installation process

- If it fails, there are other products on the Internet!



# Cleaning code

- Your code is your image (Yep, They are the developers of the product X)
- Commits mailing list (everybody will know about your changes)

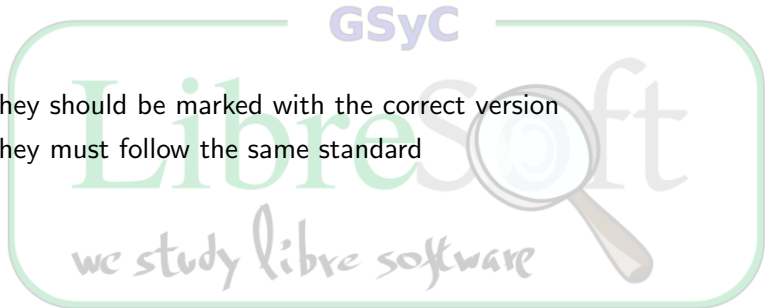


# Files

- Source code must have several “burocratic” files:
  - License (LICENSE or COPYING)
  - General Info (README)
  - Changes made (Changelog)
  - Installation process info (INSTALL)
  - Documentation (Users, developers)

## Packages or binaries

- They should be marked with the correct version
- They must follow the same standard



## Other recomendations

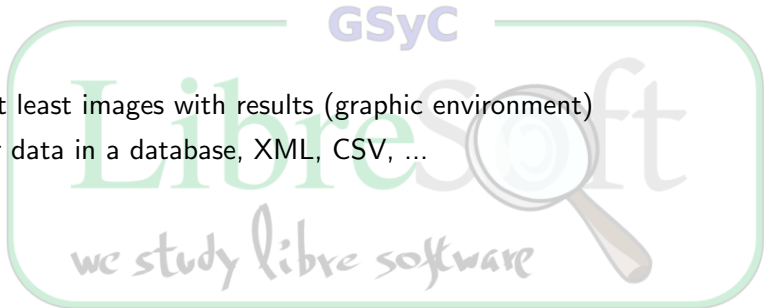
- Clean your code (use English)
  - Documentation (local language + English)
  - Try to prepare your software to be used by others
- Translators: \*/po/\*
- Developers: Clear architecture and comments

...



## Demos

- At least images with results (graphic environment)
- or data in a database, XML, CSV, ...



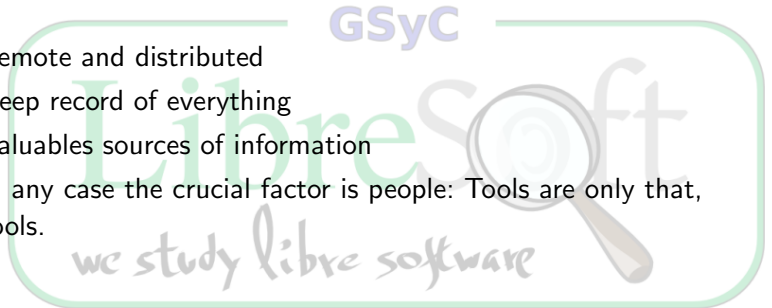
# Index

- 1 Introduction
- 2 Legal Aspects
- 3 Image
  - Web site
  - Source code
- 4 Tools
- 5 Announcing
- 6 Conclusions



# Tools

- Remote and distributed
- Keep record of everything
- Valuable sources of information
- In any case the crucial factor is people: Tools are only that, tools.



## Web site

- Collaborative development - Wiki
- Fast installation - Zope/Plone
- CMS in general



- Each project has its goals:
- SCM - SVN, CVS, Git ...
- BTS - Bugzilla, Gnat, Trac...
- Trac

GSyC

LibreSoft



*we study libre software*

## Examples: mailing lists

- List management software: i.e., Mailman
- Email and web (un)subscribing
- Digest and on-the-fly modes
- Moderation
- Administrative interface
- Archiving

# Netiquette in mailing lists

- Write intelligible messages
- Some acronyms are ok though:
  - IMHO (in my humble opinion)
  - YMMV (your milleage may vary)
  - BTW (by the way)
  - ROTFL (rolling on the floor laughing)
  - GISY (google is your friend)
- Some other acronyms are not ok (RTFM)

## Pasting style

- Top Posting
- Bottom Posting
- Inline Reply





# Index

- 1 Introduction
- 2 Legal Aspects
- 3 Image
  - Web site
  - Source code
- 4 Tools
- 5 Announcing**
- 6 Conclusions



## Ok, but where?

- SourceForge, Freshmeat, FSF directory
- Your own blog
- Basic scheme:

*What is it?*

*Where to get it?*

*Features*

*Requirements*

*Current version*

# Index

- 1 Introduction
- 2 Legal Aspects
- 3 Image
  - Web site
  - Source code
- 4 Tools
- 5 Announcing
- 6 Conclusions



# Conclusions

