

GPL Project Management

Nicholas Mc Guire
der.herr@hofr.at
FSMLabs Data GmbH
Hollabrunn - AUSTRIA

Project setup

Initial concepts are often published on the internet

- Mailing list of related topics
- Announcement news-groups
- Published on the internet on your home page

The problem with initiating projects is targeting the correct group in the right form.

Slide 1

Slide 2

Problem statement

- Developers are distributed over the internet
- Fluktuation of participating developers
- Software and Concept synchronization
- Project Splitting up

So how do you stabilize the project development ?

Slide 3

Stablizing

A key issue of GPL projects is the project management facilities

- Central resources
- Communication channels and documentation rules
- White Papers and/or draft papers "Mission statement"
- Let a project split if it has developed two flavors but keep the community together !
- Let the community present the project(S) in public

A stabil GPL project requires the ability to grant the project a life of its own. A split in the project is not bad - a split in the community is a killer

Slide 4

Project Hierarchy

Suprise,suprise ! GPL project need a strict hirarchy

- The hirarchy is NOT bound to specific people
- Flexibility of the hirarchy is a key issue
- The hirarchy must be transparent to developers but clear to the cumunity in case of problems
- It must be clear to developers and users who is the responsible maintainer

Hirarchy as a functional model is ok - as a means to "constrol" a GPL project it is a wast of time.

Slide 5

Internet resources

GPL projects live on the Internet

- Mailing list systems like majordomo and pipemail
- Web-server and support software (databases etc.)
- Source tree managment tools (CVS BK etc.)
- Newsgroups and public forums (www.linuxdevices.com etc.)

A GPL project must take these resources up from the very beginning if it wants to be successfull.

Slide 6

Documentation

The big dark hole in GPL software is documentation

- Atleast try to provide documentation from the beginning on
- Check available documentation standards (gnu documentation standard)
- Check the licensing issues for documents !
- It is better to provide NO document than wrong documents (or out of date documents)

The developers chanel to the user are the documents.

Slide 7

Documentation Guidelines

So what does your GPL project need to provide aside from bug-free and perfect code ?

- Project description - what does this software want to provide
- Developers documentation - your API, filestructure, releas policy, etc.
- Install Guide - how to install it step-by-step
- User-Guide - this needs to explain how to use your package and should assume NO programing know-how.

One must clearly distinguish programers and users - the comunity is comprised of both - if you only service one group your software will not be accepted.

Slide 8

Public presentation

Presenting your project in public - many good ideas die due to them never getting out of the hard-drive from some "GNU-freak".

- USENIX
- Linux days
- Linux User Groups world wide
- Foundations to promote certain topics (Embedded Linux Foundation, Reallinux Foundation etc.)

Most projects need support to be able to present results - the linux community has developed such resources - use them !

Slide 9

Stabilizing software

"Open Source software is instable and unreliable."

"That right ! - All software starts as instable and unreliable chunk of coded brainstorms . We need some rules to get to stability"

- For larger project split into a development and stable tree
- Tag releases as stable and beta
- Feed user comments and patches back into a STABLE tree and the DOCUMENTATION not only into the development tree
- Don't go too fast with stable releases
- Ensure the communication between users and developers

Slide 10

Commercial Distribution

Linux has been promoted by many professional releases - these distributions are always looking for new software to integrate

- If your software is stable contact Linux distribution providers
- put your software into package formats fit for standard distributions (.rpm,.deb,etc.)
- Check your software on standard distributions - if your ideas only run in your environment then they truly are ONLY for you.
- Provide USER documentation not only developers documentation if you want to go for a standard distribution.
-

Many good projects never made it to the user !

Slide 11

Sponsoring

An important resource for GPL projects

- Get a clear "Mission Statement Out"
- Present the project on the Internet in a clear and not only technical manner
- Place your software into the big picture of GPL'ed and GNU projects

Not every project will find a sponsor but often ideas can be partially funded or hardware/internet resources provided by commercial entities.