Pondering Open Source

Gerald (Jerry) Carter
Hewlett-Packard
SAMBA Team

http://www.plainjoe.org/

jerry@samba.org

http://www.samba.org/~jerry/slides/ponder_open_source.pdf

Credentials

- Linux user for over 10 years
- Samba developer for 8 years
  - ❏ current acting release manager for Samba 3.0 series
- HP employee for 4 years
  - ❏ Member of HP's Open Source Review Board
    - ✔ internal group for responsible for reviewing license and IP issues regarding use of Open Source components in HP products
  - ❏ Developer on HP Print Server Appliance
    - ✔ Windows print server replacement
    - ✔ Samba + Apache + Linux
  - ❏ Extreme telecommuter
- Talented speculatist

What is Open Source?

- Open Source Initiative
  - http://www.opensource.org
- Requirements
  - Allows for free redistribution
  - Availability of source code
  - Allow modification and derived works
  - No restrictions on end user's rights
- OSI currently listed 58 approved OS licenses
  - Do we *really* need that many?
- Copyright and License are separate issues
  - OS License grants you the right to distribute copyrighted material

Types of OS Licenses

- BSD, MIT
  - do what you want
  - Free as in beer
- GNU GPL, LGPL
  - derivative works must be licensed under the same license as the original work
  - Free as in freedom
- Artistic License (Perl)
  - Retains some central control over the project
- Apache, Mozilla Public License, IBM Common Public License, etc....

Open Source Development

- Usually run by developers and not managers
- Resemble more lightweight development models such as eXtreme Programming (XP)
- Meritocracy
  - Based on technical merits, not on job title
  - Open Source Karma
    - Gained by either long term activity in a single project, or prolific activity in a variety of projects
    - Fades over time
- Leadership is granted based on karma
  - copyright and ownership are not necessarily the same thing wrt to open source licensing

Open Source Processes

- What is valued in code? Portability? Stability? Cleverness? New features?
- Specific to a project
  - processes are fluid and may changes over time
- Conformity to the ways of the clan
  - coding style, code comments, documentation, etc...
- Open design
- Code Review
- Community based QA
  - not always as good as you would hope
  - Vendors provide value add here

Incremental Development

- “Release early. Release often.” -- ESR
- Respond to community bug reports
  - ❑ sense of neglect will weaken the community bond
- Many projects provide immediate access to development code
  - ❑ Note that the GPL only requires making the source of modifications available if you redistribute the modified binaries
- People gain a sense of ownership in their code
  - ❑ long term, emotional investment
- Patch pressure vs. Rate of change
  - ❑ development branches vs. production branches

Documentation

- Generally the last thing to be done
- Usually written by someone other than the initial developer
  - ❑ Usually incomplete
- Irony is that users need documentation to use new features, but someone must learn the new feature to write the documentation

Tools

- Distributed Source Control Management systems
  - Bitkeeper, subversion, CVS, etc..
- Secure remote access (ssh)
- Communication
  - Email, IRC, etc...
- Bugtracking
- Sourceforge.net
  - Packages the necessary tools together in one place

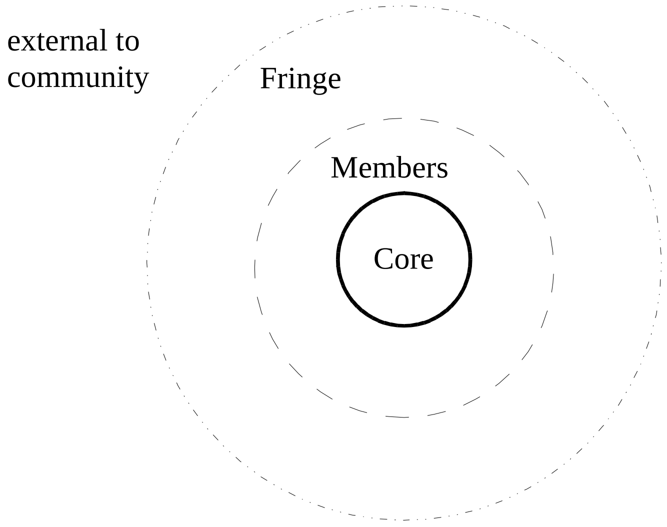
Big Balls of Gas

- The social dynamics of Open Source communities tend to act as equalizers
- “Samba is essentially a garage band that made it big” -- Jeremy Allison, Samba co-creator
- Open Source leaders as
  - Rock Stars?
  - Replaceable musicians?
- Open source solar theory
  - Large egos tend to want to blow the community up
  - Peer pressure acts as gravity to counteract egos

Open Source Communities

- Have a core set of leaders or members
- Public means of communication
  - ❑ Mailing lists, web sites, IRC, etc...
- Tools to support distributed nature of group dynamics
- Monolithic vs. Maintainer models
  - ❑ smaller teams tend to touch all code
  - ❑ larger teams tend to divide code in to smaller chunks with defined interfaces

Community



GNU [L]GPL

- The [L]GPL does
  - promote giving back to the community
  - can help to enforce cooperation
- The [L]GPL does not
  - require any relationships between members of the community
  - you can't spell “community” without “people” (or at least 'u' and 'i')

New Blood

- How are new developers fostered into the community?
  - Varying skill and experience
- New members take an active role in developing small features or bug fixes
  - Public code review and assimilation
- Communities can take
  - a “sink or swim” approach
  - an apprenticeship model
- Apprenticeship is great, but takes a great deal of time and investment

Google's Summer of Code

- Google spent $2 million funding students to work on open source during the summer of 2005
  - ❏ 417 student projects
  - ❏ http://code.google.com
- Student projects had to be done in public and associated with an established open source project acting as an adviser/mentor
- Purpose was to get new software engineers involved in open source
- Was it a success?
  - ❏ Samba had 6 students, 2 of which have done work post SoC

Project Forking

- Same ideas as “fork(); exec()” in Unix
- Sometimes communities split
  - ❏ Two projects can take on a life of their own, or
  - ❏ One may eventually get folded into the other
- Examples: gcc/egcs, emacs/xemacs, samba/samba-tng, etc...
- Generally forks, even good ones, take a toll on developer resources
- What about brand recognition?
  - ❏ Who gets the name?

Example Project:
Samba
http://www.samba.org/

Samba Team

- Made up of 8 – 12 active core developers
  - ❑ http://www.samba.org/samba/team/
  - ❑ Many are currently work full time on Samba
  - ❑ Split between two development branches
- Tools
  - ❑ SSH + Subversion
    - ✔ http://devel.samba.org/
  - ❑ Communication
    - ✔ https://lists.samba.org/
    - ✔ irc://irc.freenode.net/#samba-technical
    - ✔ http://news.samba.org/
  - ❑ Bugtracking
    - ✔ https://bugzilla.samba.org/

Project Goals

- Microsoft File and Print services
- Authentication service
- General Unix/Windows interoperability issues
- Windows is the definition of a bug
    - ❏ No spec for CIFS or related protocols
    - ❏ If Windows works and we don't, then we have a bug

La Biblioteca de Samba

- Microsoft has continually refused to share the IDL files for their DCE/RPC implementation
    - ❏ Printing, domain control, service control, remote registry access, etc....
- Everyone is doomed to reinvent the wheel
- Samba has developed IDL files for the known MS RPC pipes and has licensed those under a very liberal license (Samba is GPL'd)
    - ❏ Possible adoption by the Wine project, ReactOS, etc...
- Samba becomes the protocol spec based on network analysis

Questions to Consider

BSD vs. Linux

- Why has Linux gained ground in places where *BSD did not?
  - ❏ Technical superiority? probably not
  - ❏ Linus? probably not
  - ❏ Community? probably so
- The GPL guarantees that your code will always be open
  - ❏ MS initially used the BSD TCP/IP stack in Windows
  - ❏ What about Apple, OS X, Darwin?

Bitkeeper Fiasco

- /me steps onto soapbox
- The Bitkeeper fiasco with the Linux kernel community was destined to happen
- Linus choose to use a controversial tool that divided the community
  - Was it Linus' choice to make?

Tools & Work Flow

- To what degree does the adopted toolset define the work flow model of a project?
- Distributed SCM vs. Central Repository
  - Patch review
- IRC vs. Mailing lists
  - Timezone deltas

Can anyone create a masterpiece?

- "If you start off with a draft full of crap, all the editing in the world will do is to produce polished, shiny crap." -- Neil Stephenson (paraphrased)
- Can enough processes make bad coders good?
- Is process simply a floor on which new developers can stand? Or is it scaffolding?

Ecological Feedback Loops?

- Ecological feedback loops occur when one action feed into another
  - ❑ Water evaporates from ocean,
  - ❑ Evaporated water forms clouds,
  - ❑ Clouds rain onto land,
  - ❑ Runoff on land flows to rivers,
  - ❑ Rivers flow to ocean, ...
- Are there any ecological feedback loops in Open Source as a whole? Or is it more of a pyramid in nature?

Related Reading

- “Understanding Open Source & Free Software Licensing”, A. Laurent, O'Reilly
- “Open Source Licensing: Software Freedom and Intellectual Property Law”, L. Rosen, PTR
- “The Cathedral & The Bazaar”, E.S. Raymond, O'Reilly
- “Embracing Insanity”, R. Pavlicek, SAMS
- “Open Sources: Voices of the Open Source Revolution”, ed. C. DiBona, et. al., O'Reilly