

Open Source

The Provider's Perspective

Mark Hindess
Glyn Normington

IBM United Kingdom Ltd
Hursley Park
Winchester
Hants



Contents

- Introduction
- What Open Source Licence?
- Organisation of typical projects
- Why would a business contribute open source?
- Processes and tools
- Issues
- Conclusion

Introduction

- Who we are? We both work in IBM's Java Technology Centre:
 - Mark Hindess: unix sys. Admin and Perl guru
 - Glyn Normington: designer/developer working on J2SE
- Our Open Source background:
 - Mark contributed to some OS projects and instigator of many internal “bazaar” projects
 - Glyn was a committer on Apache Axis and created an internal “bazaar” project for CICS SOAP support
- We welcome comments, discussion, and questions throughout the presentation...

What Open Source Licence?

Freedoms – to be OS you must allow these:

- 1) Users are free to use the program for any purpose.
- 2) Users are free to examine the source code to see how it works.
- 3) Users are free to distribute the program to others.
- 4) Users are free to improve the program.

Restrictions – you have a choice about these:

- Distribution must remain “Free” - GPL, “viral”
- Distribution is unrestricted – BSD

Organisation? Example: Linux Kernel

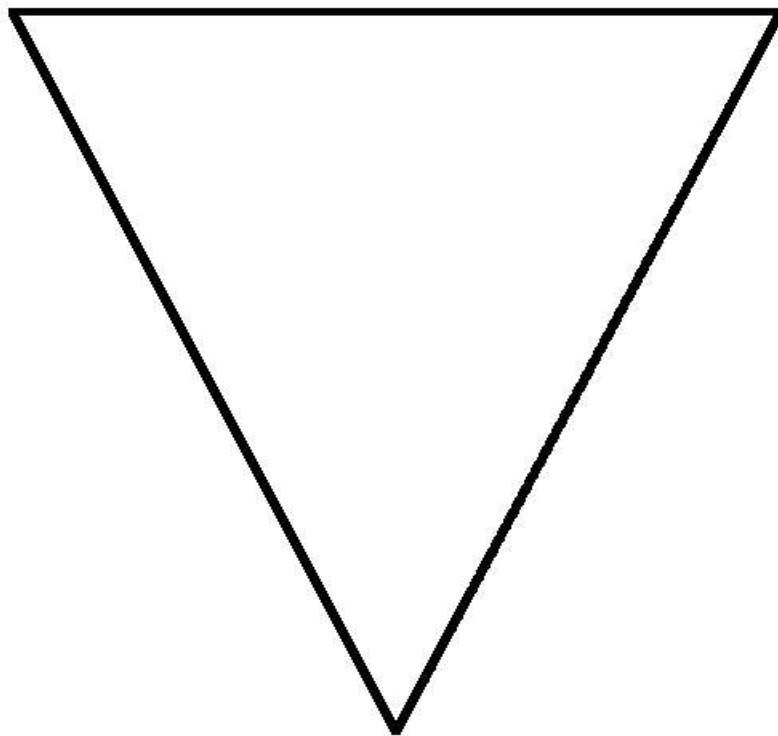
“I'm doing a (free) operating system (just a hobby, ...)” - Linus

- Benevolent dictator
- A few “lieutenants” and maintainers for some “components”
- Many “hackers” supplying patches
- Lots of bottlenecks
- Odd mechanisms – like the Trivial Patch Monkey – to try to help fix some problems



Organisation? Example : Eclipse

- Smalltalk/Pyramid:



Users

Configurers

Extenders

Contributers

Publishers

Committers

- Adhered to with employees too

Organisation? Example : Debian

- Social Contract & Debian Free Software Guidelines
- Release Managers & Maintainers – very committed due to “barriers to entry”:
 - Identity verification
 - Experience
 - Advocate
- But any one can create packages and make them available.

Organisation – Common Themes?

- Non-technical managers are rare.
- You prove yourself to technical peers not managers.



(Photo: Ronald Kohn)



- That means getting your hands dirty – not just talk.

Why Contribute? – 1



- Collaboration
 - Small organisations can contribute to a large project
 - Large organisations can avoid being proprietary
 - Encouragement of working with good people in other companies
- Alternative to open standards
 - 'Bazaar' rather than 'cathedral' (e.g. Java Community Process)
 - Focus on developing working software

Why Contribute? – 2



- Increase market and mind share
 - No proprietary tie in
 - Particularly important in public sector
 - Always an escape route if things go wrong
 - Early user involvement

Why Contribute? – 3



- Influence of Open Source and web communities
 - Current set of developers have grown up with the Internet
 - Open Source now acceptable in commercial environments
- Better quality
 - More eyes on the code, at least in principle
 - Common ownership encourages tooling

Tools : Code Repository

Currently, mostly CVS:

- ✓ Efficient remote working
- ✓ Conflict resolution and merge
- × Important operations aren't atomic
- × No versioning of metadata – permissions, character set, directories, etc.
- × Poor complex merging – e.g. between branches

Future – arch, bitkeeper, subversion, ...

Tools: Internet Relay Chat (IRC)

- Effect of ego is diminished using this medium.
- Those who might have the “loudest”, most forceful voice in face-to-face meetings don't necessarily have the same weight on IRC.
- This has interesting impact on decision making.

Issues – 1

- Project management
 - Decision making
 - “Boring but important” work items
 - Shipping on schedule
- Intellectual property
 - Certifying origin of code
 - Inclusion in commercial products
 - e.g. 'viral' effect of GNU Public Licence
 - Effects on patents and copyright

Issues – 2

- Support of Open Source tools and infrastructure
- Testing
 - But note open source security test project
- Documentation: user and internal design
- Service and code ownership

Conclusions

- Organisational structure has significant impact
- Solutions evolved are technically sound
- Solutions solve real customer problems
- Solutions increase visibility in marketplace

Some Resources

- <http://www.opensource.org/docs/definition.php>
- <http://www.opensource.org/licenses/>
- <http://www.eclipse.org/tools/eclipsetools-charter.html>
- <http://www.apache.org/foundation/roles.html>
- <http://www.osstmm.org/> (security testing)