Free Software Development

Andrew Tridgell
tridge@osdl.org

(Please ask questions during the talk)

A little story ....

- The above 3 products might look similar, but are very different inside
- Two of them run Linux

What is Free Software?

- It's all about the freedoms it gives to the user
  - “free as in free-speech, not free as in free-beer”
- Have you ever been frustrated with a software package?
  - As a software engineer, I find a big frustration in seeing a bug and not being able to fix it
- Does everyone care about freedom? No.
  - Some people care more if it “works for them”. That's fine, as it so happens that free software can often (but not always) be high quality too

The development model

- Most free software projects are meritocracies, where (at least in theory), the best idea wins
  - you must use technical arguments to make progress
  - there is often a very low tolerance for “bad” code
- Developers as managers
  - In most projects, the lead developers are also the project leaders
  - this is a huge contrast from most proprietary software development, where complains about management are common

Code quality

- Untested code? no way
  - There is a common misconception that free software projects don't care about QA. In fact, the opposite is true.
  - Most projects I have worked on place a much higher emphasis on quality assurance and testing than the proprietary software development that I have seen
- build.samba.org
  - At the center of QA for projects hosted on samba.org is the “build farm”.
  - 41 machines with dozens of OS/architecture combinations, testing every code change as it is made

Project Styles

- Within the free software community there are a large variety of project styles
    - lots of tiny projects with one or two developers
    - some projects (such as the Linux kernel) use a central person who reviews all changes
    - some projects have a “core team” with commit access
    - some projects have formalised development processes, some are much less formal
- Dealing with copyright
    - Projects vary a lot in how copyright assignment is dealt with
    - gcc, Samba and the Linux kernel cover the full spectrum

The Samba project

- Samba is now a teenager
  - Started in Canberra in late 1991, the project has produced a file/print server used by millions of people
  - The Samba Team consists of around 30 developers from 10 countries
  - We average around 7000 code commits a year
  - Our annual budget is about $10k, mostly spent on conference travel
  - Samba has spawned over 100 ancillary projects, which build on the core code in some way

Samba4 rewrite

- Could this happen in the commercial world?
  - The “rewrite most of the code” effort for Samba4 has taken 2 years so far
  - We recognised that the core design of Samba was holding back the project. It needed to be thought through again from scratch
- Test driven development
  - A test is developed for every protocol element, with every field of every packet tested
  - The core infrastructure is being constantly updated to make programming and maintenance less error prone
  - Heavy emphasis on code generation techniques

Code generation

- Auto-generating code can lead to a huge reduction in complexity
  - 53% of code in Samba4 is now auto-generated.
  - I expect this to grow to 70% by the time 4.0 is released
- PIDL - a new IDL compiler
  - None of the existing IDL compilers were suitable, so we built our own.
  - This gave us the flexibility to extend the IDL language to support a much wider range of constructs

The Mindcraft saga

- Long long ago (in 1999) in a galaxy far away (called Redmond) a benchmark company called Mindcraft ran some performance tests ....
    - probably the first time Microsoft started taking Linux seriously as a threat
    - the contrast between the community reaction and the developer reaction was very noticeable
    - the solution? a new free software project!

Questions?

- These slides are available from http://samba.org/ftp/tridge/slides/pcug.pdf