How are FOSS projects governed?

COMP8440: FOSSD
Lecture 7

What is project governance?

- Projects often need to make decisions
  - Development decisions
  - Legal decisions
  - Tools decisions
  - Project aims
  - and many, many others

What project structures allow decisions to be made?

Key governance features

- How governance is distinguished
  - Formal vs informal
  - Degree of project funding
  - Is there an identifiable 'project team'?
  - Does the project have a legal structure?
  - Does the project have a BDFL ?
    - (benevolent dictator for life)
  - Is it a 'hobby' project?
  - Is there an agreed voting system?
  - Does the project have a copyright policy?

Project Roles

- Many projects have designated roles
  - Can be formal or informal
- Common roles
  - Project leader
  - Release manager
  - Bug master
  - Documentation master
  - Web master
  - Security officer
  - PR officer
  - Mailing list manager
  - Legal officer
- Not unique
  - Often one person will take several roles
  - Roles may rotate between people

Project Funding

- Funding varies a lot between projects
  - Some projects have corporate sponsorship
    - eg. opensolaris.org
  - Some projects rely on donations
    - eg. freebsd.org
  - But most are purely volunteer efforts
    - eg. most sourceforge projects
- Give more than they cost?
  - Various estimates put the cost of development of a major Linux distribution at several billion US$
  - Project funding is a tiny fraction of that

Legal Structure

- Some projects have a legal structure
    - Mostly for larger, older projects
- Many are non-profits or charities
    - Perhaps the most common is a US 501(c)3
- Project groups
    - Many projects form groups for their legal structure
        - Apache projects
        - GNU projects
        - KDE projects
        - Gnome projects
        - Software Freedom Conservancy

Voting Systems

- Many projects have adopted a voting system
    - Most decisions are not voted on
    - Often used for election of project roles
    - Sometimes used for patches
- The Apache voting system
    - Variants are used by many projects for patch voting
    - Voters can respond with
        - +1 : agree
        - =0 : abstain
        - -1 : veto
    - A veto cannot be ignored and must come with an explanation

Copyright Policies

- Why does copyright matter?
  - It's free, so who cares?
    - Copyright gives the 'right to sue'
    - and also the right to re-license
- Common policies
  - Copyright assignment: eg GNU project
  - Developer sign off: eg Linux kernel
  - Personal copyright: eg. Samba project
  - Anything goes: most projects!
- Reaction to legal challenge
  - Copyright policies are often see as creating friction
  - Often they are adopted only after some defining event in the project history

Committers

- Direct commit access
  - Most FOSS projects have a list of people who have commit access
- Adding Committers
  - Usually involves an invitation from existing committers
  - The key is good judgement
- Should inactive committers be removed?
  - If they still have good judgement, maybe not
  - Is there a security risk in old commit access?
- Committers with distributed SCM
  - May not have the notion of a committer role
  - 'Integrators' play a similar role

Project Leaders

- Varying approaches to leadership
  - Some projects have a default project leader, often the project founder
  - Some projects use a steering commitee or board
  - Some projects vote on the leader each year
  - Some projects have no easily identifiable leader
- What the leader does
  - Help to decide on project goals
  - Help to resolve disputes
  - May set the standards for code, testing and other development tasks
  - Often acts as project spokesperson

Release Manager

- A key role
  - Putting a release together can be a lot of work
  - Needs a lot of coordination of development activity
  - What goes into a release?
    - Some projects base it solely on time
    - Some projects vote on what is required to make something release ready
- Stable/unstable releases
  - Many projects have separate stable/unstable releases
  - Standards may be higher for the stable releases
  - Odd/even release numbering has become common

Projects Evolve

- Starting simple
  - Most projects start out with very little governance
  - Simple governance makes for fast early development
- Complexity added as its needed
  - As issues and disputes arrise, new governance systems are added
  - Often solutions are adopted from other projects