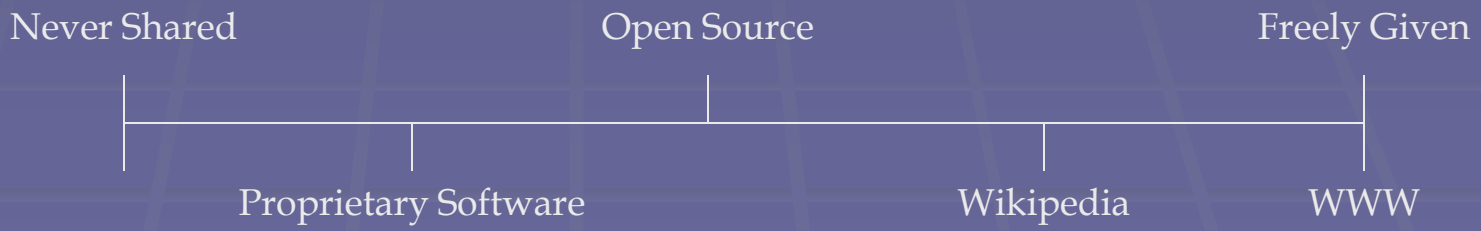
Open Source

What it is
What it is not
Why it is
Where to go with it

What is Openness?

- Openness is a characteristic based on accessibility and responsiveness
- Most products, services, or processes are neither open nor closed, but can be placed on a continuum of openness
- Moving towards openness means increasing accessibility and responsiveness
- The degree of openness required depends on the purpose of the activity and the need to exercise judgment and control



New Architecture Enabling Participation by Openness

- Linux
- Wikipedia
- SETI—world's fastest supercomputer
- Napster
- Flickr
- Podcasting, blogs
- Second Life
- E-Bay listings, ratings
- Amazon reviews, recommendations
- YouTube

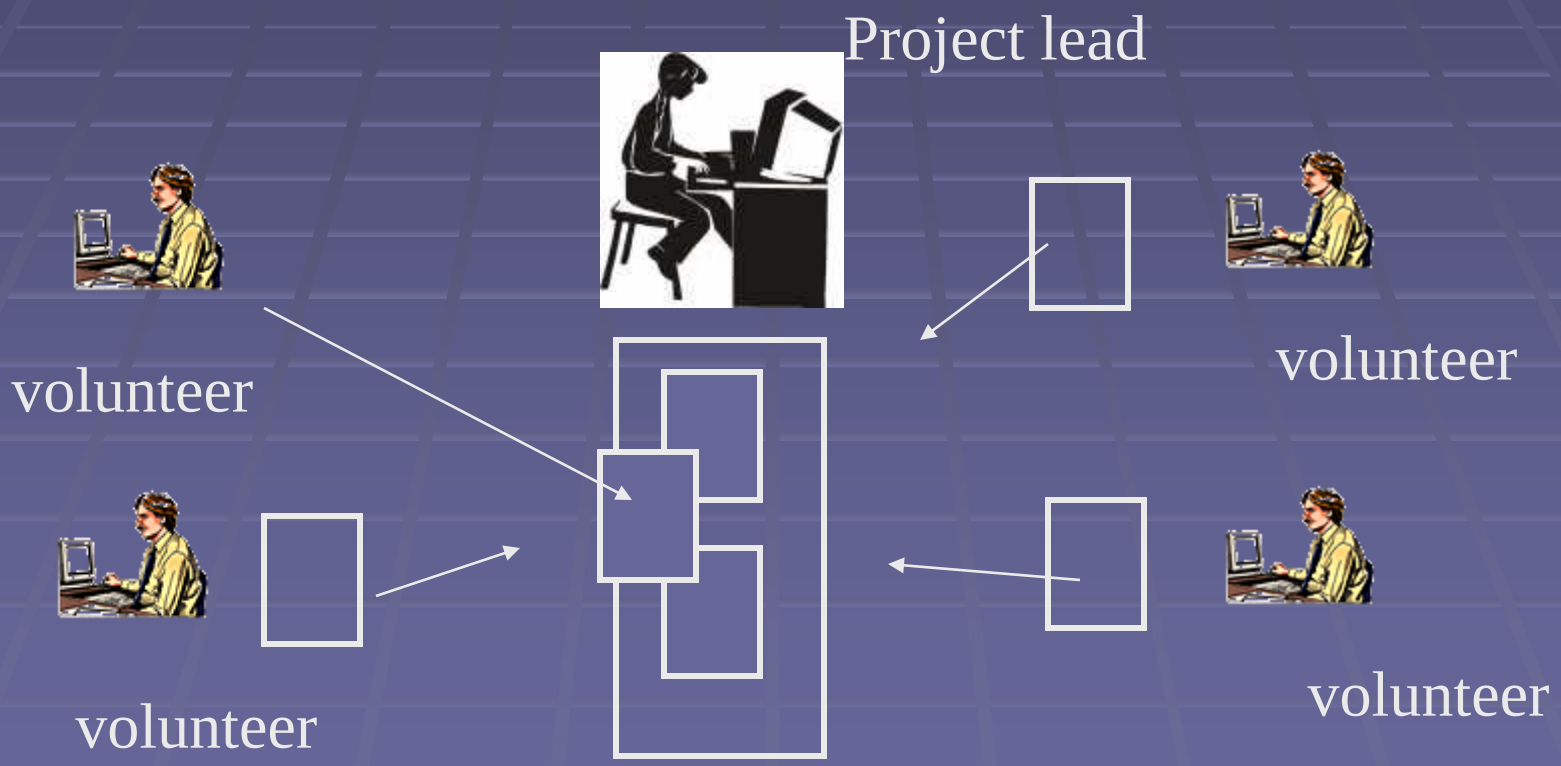
OSS Penetration

- Big share of cyberinfrastructure:
  - web servers (70%: apache)
  - mail servers (about 50%: sendmail, exim, postfix)
  - scripting languages (perl, php)
  - domain name system
- User Side
  - web browsers (Firefox)
  - office applications (OpenOffice)

Debunking Open Source Myths

- Free Lunch
  - Free Beer
- Free Speech
- Free of Bill
- Free of America
- Free to Do what I want any ol' time
- Debunking Urban Myths
  - Open Source is just a way to publish -- No
  - Open Source is Public Domain -- No
  - Open Source is Viral – Not Necessarily
  - Open Source is Immune from Patent Rights – No

Open Source Development Model



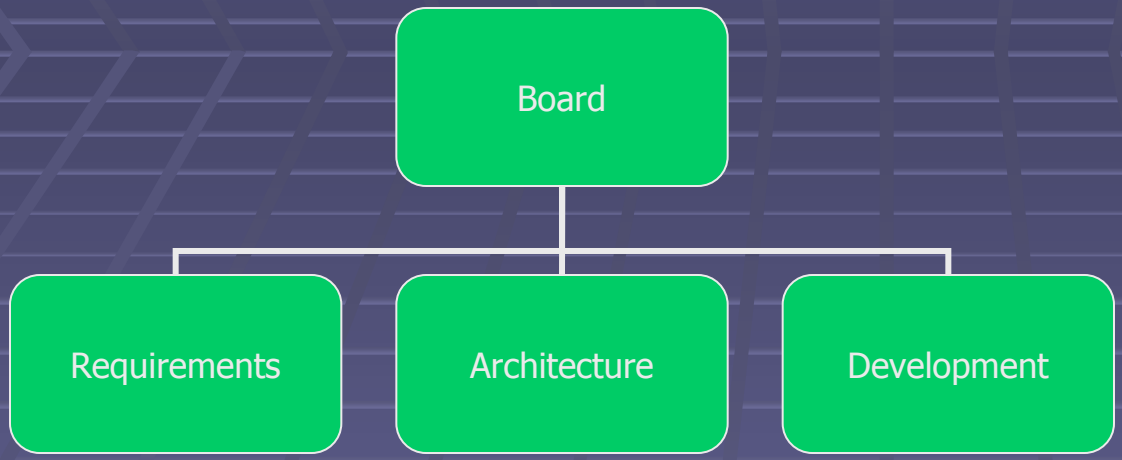
OSS as Composite of IPs

- Copyright
- Trade Secrets
- Patents
- Technology Transfer
  - Licenses are Contracts too!
- Reverse Engineer Rights
  - How are these Ltd?
    - DMCA standard technical protections
    - EULA decompilation restriction
    - Unlicensed “practicing”
- OSS as Overcoming the IP StraightJacket

Central vs. Distributed Control

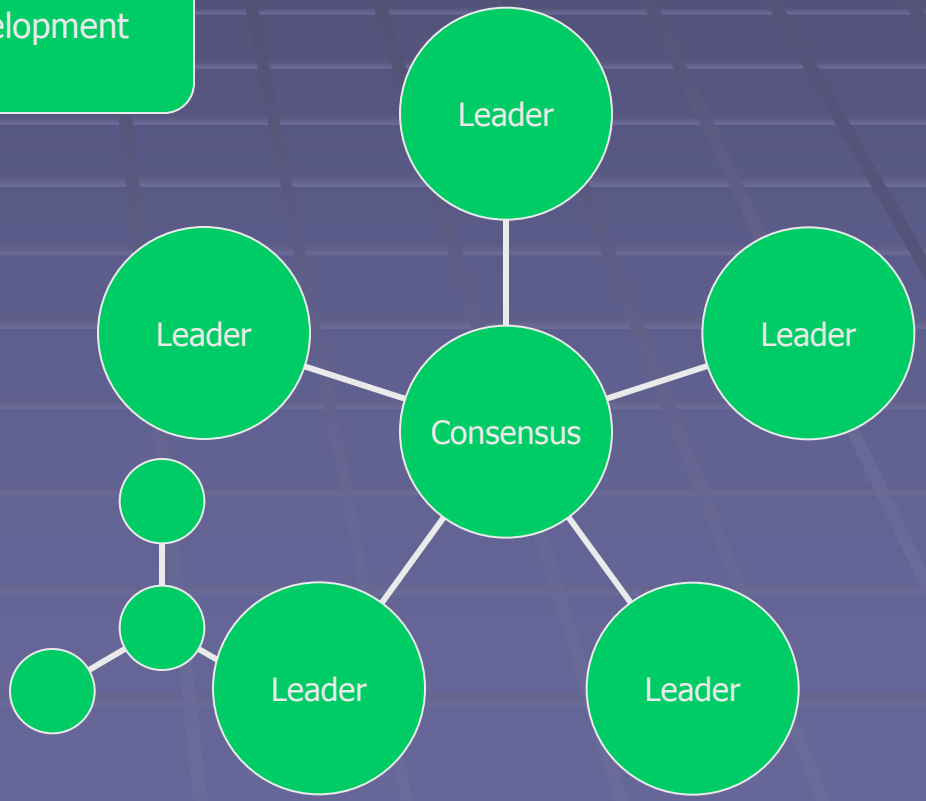
- Design
- Organization
- Ownership
- Control
- Benefits: metering, value capture

Project (product?) Governance?



Governance:

Specifying the decision rights and accountability framework to encourage desirable behavior to achieve an objective



Adapted from *IT Governance*, Weill & Ross, 2004

Proprietary vs. Public Goods

- Incentives
- Benefit Capture
- Control
- Access

Some Immutables

- IP
  - Royalty Rates & Business Models
  - Infringement
- Industrial Organization
  - Antitrust
  - Viscerality of Feudalism
    - Royalty again
- Communities are fun
- Software is a “Build Thing”

1st Creators, Follow-On Innovators & IP

- Every innovation has a first creator and potential follow-on innovators
- Creating or restricting rights of first creators or follow-on innovators through IP rights are just like taxing one or the other
  - Too high a tax on either group results in under-production of innovation
- Every first creator is “Standing on the shoulders of giants” (Newton)
  - There are far more follow-on innovators than first creators and they may bring new insights
- The key task of IP Policy balancing to produce the greatest possible innovation

Copyright

- Originality
  - Min. personal creativity/intellect required
- Works of Authorship
  - General categories under §102(a)
- Fixation in Tangible Medium
- Expression
- Generally Facts/Data Not Copyrightable

Bundle of Rights under Copyright

- Reproduce
- Derivative Works
- Distribution
- Performance
- Display
- License
- Assignment

What IP is in Open Source?

- Copyright Still Exists in Software
  - And the Open Source Development Model is Premised on That
  - Copyright is an intangible right – it exists independent of the code
- Copyright Attaches On Creation of Original Code
  - Copyright Notice and Registration Not Required
  - Ownership Initially Vests in Authors or Institution

Trade Secrets

- **Information** (formula, pattern, compilation, program, device, method, technique or process)
- Derives **independent economic value** from **secrecy** or by proper means discovery by potential competitors, and
- **Subject of efforts, reasonable** under the circumstances, to **maintain secrecy**

Misappropriation is Wrongful

- Acquired through **improper means**
- Acquired from another **knowing** improper means was used
- Use or disclosure knowing **violates duty of silence**
- Use or disclosure knowing **acquired by improper means**
- Use or disclosure knowing **acquired through fiduciary breach**

Proper Means of Discovery

- Reverse Engineering
  - Exceptions: DMCA, EULA restriction
- Independent Invention
- Exposure from Public View or Display
- Discovered from Published Literature
- Discovered from License, Unless Confidentiality Term Otherwise

Patent

- Patentable Subject Matter
  - Process (BMP, S/W)
  - Machine
  - Manufacture
  - Composition of Matter (bio-tech)
- Novelty, Non-Obviousness, Utility
  - New, Useful & Human-made

Trademark

- Word, Name, Symbol, Device
  - Recently: color, sound, smell
- Identifies Source of Goods or Services
- Distinguishes from competitors using Spectrum of Distinctiveness
  - Generic
  - Descriptive
  - Suggestive
  - Arbitrary or fanciful

Legal Tools for Technology Transfer

- Assignments
- Licensing:
  - software “sales,” website EULA
  - scope,
  - duration,
  - fields of use,
  - compensation,
  - geographic limits
- Shop Rights
- Work Made for Hire; Hired to Invent; M-S & S/E
- Confidentiality Duties: Non Disclosure Agreements (NDA)
- Non-Competition Agreements (non-competes)
  - Emerging Inevitable Disclosure Doctrine
- Leasing, Franchising

License Pricing

- Fixed price for deliverable irrespective of development time or expenses
- Metering: pay/use/time/MIPS
- Delivery, Installation, Prove working, Periodic installments, Upgrades
- Hybrid of compensations
- Application Service Providers

So . . . Which do you use: Proprietary v Free v Open?

- Proprietary Software
- Open Source Software
- Freeware Software

Advantages: Proprietary Software

- Indemnification for title & infringement ;
- Maintenance and support;
- Licensee doesn't have to have open source savvy staff;
- Licensees' rights if:
  - media is defective;
  - software contains viruses, backdoors, etc.;
  - product fails to meet written technical/business specifications.

Disadvantages: Proprietary Software

- COST!
  - License fee
  - Product bundling—example: Microsoft office.
- Licensee cannot modify or enhance the code;
- Often not built to open standards, leading to interoperability problems;
- Shut off from continuing development and information sharing in open source community;
- Some proprietary code is not as good as its open source counterparts.

Advantages: Open Source License

- PRICE: Generally no or low license fees;
- Availability of source code coupled with permission to make modifications;
- Access open source development community, which may be very active with respect to code used. Continuing improvement; outstanding development;
- More likely to be built to open standards, so interoperable with other open standards systems.

Open source licensing

- The **licence** is what determines whether software is open source
- The licence must be approved by the Open Source Initiative (www.opensource.org)
- All approved licences meet their Open Source Definition (www.opensource.org/docs/definition.php)
- Approved licences >50
  - EXs: GPL, LGPL, MPL, BSD
  - http://www.opensource.org/licenses

Source Code Escrow

- Client or 3d party retains source code to prevent competitor use
- Seller or 3d party retains source code to assure payments
  - Network effect: lock in leverage
- Independent 3d party source code firms
  - Escrow agreement defines rightful release

Open Source Software – Main Features

- Non-proprietary software which may or may not be used commercially;
- Typically licensed under an Open Source license (not given away)
  - License terms differ from proprietary software license terms
- Source code is generally made available
  - Legal restriction on reverse engineering (under the DMCA) do not apply.

Typical Proprietary Software License

- Fairly standard terms
- Source code availability
  - Source code not provided - trying to figure out inner workings of software through reverse engineering or decompiling of operating mode is forbidden; OR
  - Source code provided - may or may not include permission to create modifications and enhancements

Proprietary Software License terms - Licensees

- Restrictions on dissemination. Licensee and users strictly defined. Licensee has no right to share with those not defined as licensee users in license;
- Licensor indemnifies licensees against third party infringement claims;
- Often, have to sign a new license each time new licensee obtains the code.

Proprietary Software License terms – Warranty and Support

- Warranties provided:
  - Defects in media and existence of viruses, Trojan Horses, backdoors, etc;
  - Can negotiate for warranties re: meet specifications in product documentation
- Maintenance and support terms included (although may be in separate document).

Open Source Software – Main Features

- Non-proprietary software which may or may not be used commercially;
- Typically licensed under an Open Source license (not given away)
  - License terms differ from proprietary software license terms
- Source code is generally made available
  - Legal restriction on reverse engineering (under the DMCA) do not apply.

Open Source Software License - Licensees

- Original software owner or developer chooses to limit the rights that he asserts over licensees
- Licensees, subject to license terms, can:
  - make and distribute copies of software;
  - build upon software to create modifications or other works.

Open Source Software Licenses - Source Code

- Source code to original product always provided;
- Licensee can modify or enhance source code (create “derivative works”) or include source code with other license types (create “larger works”);
- Licensee may be required to share modifications with the world (in source and/or binary form), but not necessarily;
- Licensee may be prohibited from charging royalties for derivative and larger works, but not necessarily.

Open Source Software License – Warranties, Support

- Generally, software provided “AS-IS” with no warranties, warranties excluded;
- No indemnification;
- No maintenance or support.

The GNU “General Public License” (GPL)

- No standard open source license, but GPL most widely used (roughly 85% of open source software);
- Terms include:
  - User freedom to distribute and/or modify;
  - Requirement that original and modified source code be always made available to the world under the terms of the original license;
  - Must retain copyright notices and warranty disclaimers;
  - Does not include grant of patent licenses;
  - Extremely viral license

Common Open Source Models

- GNU General Public License ("GPL")
  - Grants right to copy, modify and distribute
  - Requires that source code be made available to future licensees
  - Generally Seen as "Viral"
    - Applies to separate works that are combined with distributed code
    - Effect may depend on how code linked
  - Disclaims Warranties
  - May blow-up in face of patent assertion
  - Proprietary distribution models difficult

Common Open Source Models

- GNU Lesser General Public License ("LGPL"):
  - Similar to GPL
  - Somewhat easier for licensees to combine the LGPL code with a separate program and distribute the combination under separate licenses
  - Often used with Open Source Libraries that are compiled into an application program

Common Open Source Models

- BSD/MIT/Apache Style License:
  - More permissive licenses
  - Generally allow freer distribution, modifying, and license change; much like public domain software
    - No future open source requirement
  - May require attribution
  - Variants may include non-standard restrictions
    - E.g., no military use – but not OSI-compliant
  - Disclaims Warranties
  - Subject to third-party patent claims

Common Open Source Models

- Mozilla/IBM/Apple Style Licenses
  - Combine facets of both the GPL and BSD style licenses:
    - Distribution of original code (and for some, modifications) include access to source code.
    - Not viral in reach.
  - Explicitly contemplate patent licenses.
  - Some provide backwards indemnification.

The Mozilla Public License

- Developed by Netscape for the Mozilla browser
- Terms include:
  - Very similar to the GPL but,
  - Can charge royalties for modified versions;
  - Can include source code within larger works licensed under different license types, thus license does not ‘infect’ all downstream projects;
  - Must retain copyright notices and warranty disclaimers;
  - May provide additional warranties to downstream users but may have to indemnify original developer for any claims arising as a result;
  - Includes grant patent licenses;
  - Less viral than the GPL.

The IBM Public License

- Terms include:
  - User freedom to distribute and/or modify;
  - No requirement for source code availability in downstream distribution;
  - The program can be distributed in executable form thus allowing downstream users to develop, sell, and install customized software packages without having to make all customizations available to the world;
  - Must retain all copyright notices and warranty disclaimers;
  - Includes grant of patent licenses.

Open Software License

- Terms include:
  - User freedom to distribute and/or modify;
  - Viral license, source code is always made available to the world;
  - Must retain copyright notices and warranty disclaimers;
  - Requires indemnification for attorney's fees incurred as a result of potential claims or litigation.

The Apache Software License

- Governs the Apache web-server software.
- Terms include:
  - User freedom to distribute and/or modify;
  - No requirement for source code to be made available to the world in downstream distribution;
  - Must retain all copyright notices and warranty disclaimers;
  - Not a viral license.

The FreeBSD License

- Unrestrictive license:
  - Only requires preservation of copyright notices and warranty disclaimers.

IBM vs. SCO

- Linux kernal's genealogy
- Emphasizes IBM's role as risk underwriter
  - What are IBM's incentives to do so?
- Emphasizes Fundamental OSS Risks
  - Composite only as robust as its weakest component
  - Pervasive Ignorance of Property Rights, Infringement, Permission & Fair Use among OSS community
- http://en.wikipedia.org/wiki/SCO_v._IBM_Linux_lawsuit