



Open Source and IIS

AIRA Regional Meeting
San Diego, CA
October 16, 2015



Agenda

- Introduction and definitions
- Open Source “Community”
- SWOT for Open Source
- Examples from Outside of IIS
- Examples in IIS World
- Strategies for Success
- Resources



Definitions

“**Copyright** is a legal right created by the law of a country that **grants the creator of an original work exclusive rights to its use and distribution**, usually for a limited time. The exclusive rights are not absolute; they are limited by limitations and exceptions to copyright law, including fair use.”

<https://en.wikipedia.org/wiki/Copyright>

“**Software copyright** is the **extension of copyright law to machine-readable software**. While many of the legal principles and policy debates concerning software copyright have close parallels in other domains of copyright law, there are a number of distinctive issues that arise with software.”

https://en.wikipedia.org/wiki/Software_copyright



Definitions

“**Public domain software** is software that has been placed in the public domain, in other words **there is absolutely no ownership** such as copyright, trademark, or patent. Unlike other classes of licenses, there are **no restrictions as to what can be done with the software**. The software can be modified, distributed, or sold even without any attribution.”

https://en.wikipedia.org/wiki/Public_domain_software

Popular in the early days of computing but often a misused term. “Free” software may or may not actually be public domain. Public domain products have *no license*.



Definitions

“**Copyleft** (a play on the word copyright) is the practice of offering people the **right to freely distribute copies and modified versions** of a work with the stipulation that the same rights be preserved in derivative works down the line.”

<https://en.wikipedia.org/wiki/Copyleft>

Typically, this is used to make sure a modified piece of software is not then converted into a commercial product with restricted access or use. Pretty much all Copyleft products are Open Source, but not the other way around.



Definitions

“Generally, **open source** refers to a **computer program in which the source code is available to the general public for use and/or modification** from its original design.”

https://en.wikipedia.org/wiki/Open_source

Source code: What programmers write

Machine code: What computers understand

Source code is compiled (transformed) into machine code which users can then execute. Only machine (executable) code is typically available to end-users, but open source includes the human-readable *source* code.



Definitions

Open Source Initiative:

1. Free Redistribution – no restriction on selling or giving software away, and no fee
2. Source Code – must be included, as well as compiled form, without fee
3. Derived Works – must be allowed, with distribution under same terms
4. Integrity of The Author's Source Code – can require that modifications are distinguishable from the original (*e.g.*, different version number)
5. No Discrimination Against Persons or Groups
6. No Discrimination Against Fields of Endeavor (*e.g.*, business use, or research use)
7. Distribution of License – included with the software
8. License Must Not Be Specific to a Product – rights transfer even if software parsed or repackaged
9. License Must Not Restrict Other Software that might be distributed with it
10. License Must Be Technology-Neutral (*i.e.*, no particular technology dependence)

<http://opensource.org/docs/osd-annotated>



Definitions

- Over the years, open source license varieties began to proliferate
- OSI initiated an approval process to identify compliant licenses to try to reduce confusion
- Popular licenses:
 - Apache
 - GNU General Public License
 - BSD
 - MIT
 - Mozilla Public License
- Some licenses permit downstream commercial development (*e.g.*, BSD); some require contributions back to the originator (*e.g.*, GPL) - each has benefits and challenges.



Definitions

- Conclusions

- Open source is an easing of default copyright for software
- Open source concept is about right to *modify source code* as well as the right to *use* software
- Many variations and conditions possible
- Open source can promote sharing, but also inhibit sharing through potential loss of intellectual property rights
- Mixing open source and proprietary products can have important impacts on a software developer



Open Source Community

- Not all projects operate this way
- More common where the community of *users* is smaller, but there are exceptions (*e.g.*, Linux)
- Focus on collaboration
- Benefit from the knowledge of others: many project encourage others to “fork” the software and make their own modifications
- Most successful model offers a paradox: collaborative development but fierce control of the “production” source code



SWOT Analysis

Strengths

- No license fee to use
- No loss of access to source code if developer stops work
- Freedom to make/share changes
- Transparency in governance
- Enables modular IIS deployment

Weaknesses

- Risk of *detrimental* source code “forking”
- Burden of enhancements may fall to individual users/organizations
- Software support may be harder to secure

Opportunities

- “Joint development” can reduce cost of enhancements & support
- Commercial vendors often provide solid support
- More modular IIS might enable more Open Source component use

Threats

- IIS community will not financially support product development
- IIS community expects open source market to behave like commercial market
- Commercial vendor reactions



Examples from Outside of IIS

- Linux: Major operating system
- WordPress: Used for blogging
- FireFox, Thunderbird: Web & E-mail
- OpenOffice: Desktop productivity
- PostgreSQL: Relational Database Management System
- Moodle Virtual Learning Environment (VLE): Course management



Examples from IIS World

- CAT Quality Assurance Tool
- Choicemaker (patient matching)
- Data Quality Assurance Tool (DQA)
- FEBRL (patient matching)
- HAPI (HL7 Parser)
- Immunization Calculation Engine (ICE)
- Mirth (Interface Engine)
- Texas Children's Hospital Forecaster



Strategies for Success

- Begin to move IIS to modularity and SOA
- Leverage widely-used Open Source products *where feasible* (*e.g.*, Linux, PostgreSQL, HAPI, Mirth)
- Jointly develop/support more specialized products when necessary (*e.g.*, forecaster, QA tools)
- Look beyond IIS community for collaboration (*e.g.*, EHRs, PHRs)
- Encourage one organization to maintain stewardship over and support each product to prevent “detrimental” forking
- Recognize and manage any turbulence this may cause in the commercial product marketplace



Resources

- Open Source Initiative

<http://opensource.org/>

- Open Source Electronic Health Record Alliance

<http://opensource.org/>



Contact Information

Noam H. Arzt

President, HLN Consulting, LLC

858-538-2220 (Voice)

858-538-2209 (FAX)

arzt@hln.com

<http://www.hln.com/noam/>