How Can Public Health Benefit From Being an Open Source Community?

2016 Public Health Informatics Conference

Atlanta, GA

August 22, 2016

Noam H. Arzt, PhD, FHIMSS

President, HLN Consulting, LLC

Agenda

- Introduction and definitions
- Open Source "Community"
- SWOT for Open Source and Proprietary
- Examples from Outside of Public Health
- Public Health Examples: IIS World
- Strategies for Success
- Case Study: Open Source Governance
- Resources

Why is this an issue?

- Continued funding uncertainty
- Increased CDC interest
- Increased Federal interest overall
- Ongoing development of tools and products
- Confusion over the topic – need guidance on licensing, governance, fair use, and cost implications

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/osd-annotated

Definitions

“**Proprietary software** is licensed under legal right of the copyright holder, with the intent that the licensee is given the right to use the software only under certain conditions, and restricted from other uses, such as modification, sharing, studying, redistribution, or reverse engineering.”

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

Generally source code is not made available. Can also be referred to as “closed source.”

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: Proprietary

Strengths • Source code remains unified • Vendor bears the burden of enhancements • Software support usually easy to secure • Software is relatively mature and well tested	**Weaknesses** • License fee to use • Only vendor can make changes • Potential loss of access to source code if developer stops work • Users may or may not get the enhancements they want • May or may not enable modular system deployment
Opportunities • CDC IDIQs/BPAs may fund and facilitate product-specific enhancements • Vendors of these products may more readily support external hosting	**Threats** • Small PH software market may see fewer vendors over time • Vendor reaction to encroachment of Open Source is hard to predict • Agency funding continues to be constrained

SWOT Analysis: Open Source

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 system 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 systems might enable more Open Source component use	**Threats** • PH community will not financially support product development • PH community expects open source market to behave like commercial market • Commercial vendor reactions

Examples from Outside of PH

- 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

PH Examples: 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

Policy Models

Source Code Freely Available

	No	Yes
Yes	Managed Open Source	Fully Open Source
No	Proprietary	Opportunistic

Source Code Maintained by Multiple Parties

Strategies for Success

- Begin to move PH systems 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 public health community for collaboration (*e.g.,* EHRs, PHRs)
- Encourage one organization to maintain stewardship over and support each product to prevent "detrimental" forking (Managed Open Source)
- Recognize and manage any turbulence this may cause in the commercial product marketplace

Case Study: Open Source Governance

Background

- HLN has been working on CDS products for a number of years
- Original products begun under “work for hire” agreements with NYC
- NYC agreed to release products into Open Source community under GNU Lesser General Public License (LGPL) version 3

CDS Products

- ICE Web Service
  - Evaluates history, makes recommendations
  - Deployed within OpenCDS
  - Available and in use
  - The “rules” are the “data” and not covered by Open Source License, but freely shared
  - Multiple rule sets supported

CDS Products

- CDS Administration Tool (CAT)
  - Web-based built in Java
  - Used by SMEs to
    - Manage/author rules
    - Maintain code sets/terminology
    - Manage and execute test cases
  - Will be released module by module
  - 2,500+ test cases are the “data” and not covered by the Open Source license

CDS Knowledge Engineering

- ACIP defines and publishes recommended schedules as "best practice"; not computable
- CDC CDSi project developed and maintains a consensus logic specification often considered "gold standard"
- ICE rules developed by SME panel (HLN, NYC, ADPH) which determined default set of rules and in some cases core ICE functionality
- ICE has some documented differences with CDC CDSi

Governance Challenges

- ICE is neither commercial (responsive to the market) nor custom developed (responsive to its funders).
- Without rigorous control over software development and implementation there is potential for confusion and error, *though users are free to "fork" the product within license terms*.
- Even with consensus over the CDS rules some users might not agree nor accept consensus decisions. *ICE supports multiple rule sets.*
- Some users may not want to participate in a consensus process, especially if they are paying for support.
- Some users may be interest in a subset of the rules (for instance, adult or childhood) to the exclusion of others.
- Management and coordination of governance process requires purposeful effort and funding to be sustainable.
- Equity issue: who pays for enhancements since *everyone* benefits in the end?

Governance Principles

- Changes to the Open Source software should be available to all users.
- HLN (or anyone else) may create products with “enhanced features” that must comply with the Open Source license but might not be freely available.
- A base set of rules developed by consensus should be maintained and be freely available to all users.
- Alternate rule sets may or may not be freely available at the discretion of the organizations that create them or sponsor their creation.
- Resources and activities should be leveraged across participants as much as possible.

KM Governance: Option 1

- Expand current SME WG and rename it the ICE Rules Work Group. Allow others to participate, but HLN reserves the right to limit its size.
- Continue to post openly on the wiki the outcome of consensus decisions while striving to maintain a “core” or “base” rule set. Variations are inevitable.
- Convene an additional CDS Product Work Group to discuss and help prioritize product feature enhancements for ICE and CAT, with the goal of developing a “product roadmap” at least two versions ahead of what is available for production use. *Users from other CDS domains may be involved in this WG.*
- HLN will continue to control the actual contents of the software distributed and maintained by this collaborative effort.

KM Governance: Option 2

- Create a new ICE Rules Work Group made up of independent individuals. Work group functions as an “expert panel” to ensure that ICE implements an appropriate interpretation of ACIP rules. Expert panel independently funded. Selection process for membership to be determined.
- *Continue to post openly on the wiki the outcome of consensus decisions while striving to maintain a “core” or “base” rule set. Variations are inevitable.*
- *Convene an additional CDS Product Work Group to discuss and help prioritize product feature enhancements for ICE and CAT, with the goal of developing a “product roadmap” at least two versions ahead of what is available for production use. Users from other CDS domains may be involved in this WG.*
- *HLN will continue to control the actual contents of the software distributed and maintained by this collaborative effort.*

“Features” Governance

- Some features may be provided without external investment
- Funded “support” may provide features
- Several models
  - First one who needs a feature pays
  - Negotiate collaborative funding
  - If funded support is sufficient use that pool of dollars for enhancements

Resources

- Open Source Initiative
  http://opensource.org/
- Open Source Electronic Health Record Alliance (OSEHRA)
  https://www.osehra.org
- OSEHRA License Terms Document
  https://www.osehra.org/sites/default/files/osehra_licensing_terms_v.1.0.pdf
- Draft Federal Source Code Policy
  https://sourcecode.cio.gov/

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/