Making Compliance Easy: Filling in the Missing Pieces

Kate Stewart, Sr. Director of Strategic Projects
Feb 15, 2017
Product Distribution

Requires:

● Provide licenses of involved open source software

● Provide copyright statements of involved authors

● Provide disclaimers, etc.
Why is License Compliance still a problem?

- Sharing source code between projects is needed for rapid development of new features.
- Scale of open source software available!
- Product companies may have different focus than open source code developers.
- Focus on licensing after development done.
Identifying Licenses: Software Archaeology!

- License text at project level may not apply to all files in project.
- Written text found “explaining” licensing
- License relevant statements unclear
Open Source Compliance: The Challenge

Companies combine Open Source Software with other software

Creating an accurate bill of materials and notices requires effort & research
Open Source Compliance: The Challenge

Supplier 1
Supplier 2

Customers

The effort is repeated at each step in the supply chain
“Open Source”-scape

Upstream Projects

Useful “Collections” of Open Source

Added-value Software

Products
Open Source Upstream Projects

> 52,000,000+ repositories on github
> 430,000+ projects on sourceforge
> 200,000+ repositories on gitlab

… and so on
Open Source Software Licensing

Copyleft

Permissive
Open Source Communities Governance
Open Source Distributions & Packaging

- Debian / Ubuntu
- Fedora / Red Hat / CentOS
- Android / Chrome
- Open SUSE / SLES
- FreeBSD / NetBSD
- Yocto / Open Embedded
- ...

Many different policies and practices on packaging open source projects and how licensing information is expressed.
Code Repositories and Package Managers

What can you make with 350,000 building blocks?
The npm registry hosts over a quarter million packages of reusable code — the largest code registry in the world.

- **Find**: Popular libraries like jQuery, Bootstrap, React, and Angular, and components from frameworks including Polymer.
- **Discover**: Packages for mobile, IoT, front end, back end, robotics... everything you need to start building amazing things.
- **Build**: Assemble packages like building blocks to quickly develop awesome new projects.

PyPI - the Python Package Index
The Python Package Index is a repository of software for the Python programming language. There are currently 92741 packages here.

... and others per language
Everyone does things differently....

.... how can we automate?

One Step at a Time...

1) Common language to communicate licensing data
2) Open Source tools to generate licensing data summaries
3) Keep licensing data current with every source change
4) Transparency on software’s licensing data
5) Common processes to pass licensing data with software
6) Adoption in key projects, distributions, repositories, ...
# Ecosystem Automation Scorecard

<table>
<thead>
<tr>
<th>Status</th>
<th>Goal</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Common language to communicate licensing data</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Open Source tools to generate licensing data summaries</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Keep licensing data current with every source change</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Transparency of software’s licensing data</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Common processes to pass licensing data with software</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Adoption by key projects, distributions, repositories</td>
<td></td>
</tr>
</tbody>
</table>
The Need

Our suppliers aren’t giving us complete licensing information for open source packages.

I don’t mind vetting our code, but I’m sure this imported package has been analyzed a dozen times before.

Every customer wants a bill of materials in a different form.
Software Package Data eXchange

Open Standard:
• A standard format for communicating the licenses and copyrights associated with software packages

Vision:
• To help reduce redundant work in determining software license information and facilitate compliance

Guiding principles:
• Human and machine readable
• Focus on capturing facts; avoid interpretations
SPDX 2.1

Latest version published 10/2016, addresses all original use-cases.

▪ Use SPDX License List short identifiers to refer to common licenses found in Open Source efficiently
▪ Tag source files with SPDX license list short identifiers
▪ Provide an SPDX document to summarize the licenses in any software you distribute
# 2016 Ecosystem Automation

<table>
<thead>
<tr>
<th>Status</th>
<th>Goal</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Common language to communicate licensing data</td>
<td>SPDX</td>
</tr>
<tr>
<td></td>
<td>Open Source tools to generate licensing data summaries</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Keep licensing data current with every source change</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Transparency of software’s licensing data</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Common processes to pass licensing data with software</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Adoption by key projects, distributions, repositories</td>
<td></td>
</tr>
</tbody>
</table>
Open Source SPDX Document Creation

- **SPDX-Tools:**
  - [https://github.com/spdx/tools](https://github.com/spdx/tools)

- **FOSSology**
  - [https://github.com/fossology/fossology](https://github.com/fossology/fossology)

- **DoSOCSv2**
  - [https://github.com/DoSOCSv2/DoSOCSv2](https://github.com/DoSOCSv2/DoSOCSv2)
3.1 release generates: both SPDX tag:value & SPDX RDF documents.
Open Source tools for Summarizing Licensing

- Auditing existing code and generating SPDX document
  - FOSSology 3.1

- Command line generate SPDX with build scripts
  - DoSOCSv2 project used with Yocto
  - Prototype FOSSology with ELBE with Debian
  - LiD (announced this week).

- Dependency tracking in repositories
  - Maven POM, Eclipse Plugin prototypes
### 2016 Ecosystem Automation

<table>
<thead>
<tr>
<th>Status</th>
<th>Goal</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Common language to communicate licensing data</td>
<td>SPDX</td>
</tr>
<tr>
<td></td>
<td>Open Source tools to generate licensing data summaries</td>
<td>FOSSology, SPDX-tools...</td>
</tr>
<tr>
<td></td>
<td>Keep licensing data current with every source change</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Transparency of software’s licensing data</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Common processes to pass licensing data with software</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Adoption by key projects, distributions, repositories</td>
<td></td>
</tr>
</tbody>
</table>
Keep licensing data current with every change

- Command line tools able to generate SPDX documents
  - For upstream project to use for releases
  - For inclusion in check-patch utilities (stop garbage in)
  - For packaging & build scripts to run
  - For code composers from building blocks and libraries.

Some starting points exist but need to make robust.
Command Line SPDX Tools...

- DoSOCSv2
  - [https://github.com/DoSOCSv2/DoSOCSv2](https://github.com/DoSOCSv2/DoSOCSv2)

- **Coming Soon**: FOSSology command line improvements & wrapper scripts with ELBE

- **Coming Soon**: LiD code License Scanner
  - [https://www.codeaurora.org/qualcomm-ostg-lid](https://www.codeaurora.org/qualcomm-ostg-lid)

- **Coming Soon**: ScanCode is having SPDX added
  - [https://github.com/nexB/scancode-toolkit](https://github.com/nexB/scancode-toolkit)
# 2016 Ecosystem Automation

<table>
<thead>
<tr>
<th>Status</th>
<th>Goal</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Common language to communicate licensing data</td>
<td>SPDXX</td>
</tr>
<tr>
<td></td>
<td>Open Source tools to generate licensing data summaries</td>
<td>FOSSology, SPDX-tools</td>
</tr>
<tr>
<td></td>
<td>Keep licensing data current with every source change</td>
<td>DoSOCSv2, LiD, ...</td>
</tr>
<tr>
<td></td>
<td>Transparency of software's licensing data</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Common processes to pass licensing data with software</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Adoption by key projects, distributions, repositories...</td>
<td></td>
</tr>
</tbody>
</table>
Evolving From Package to Source File Licensing

• With widespread sharing of source files (composable repositories, etc.), the package level license may not be complete.

• The licenses of source files need to be reviewed for distribution obligations.
Establishing a License Coverage “Grade”

- Grade = % of copyrightable source files with clear licensing terms contained within the file.
- A license notice per file should be standard header (if it exists) but can be as simple as: “SPDX-License-Identifier: GPL-2.0”
- Grade bump from F to D if LICENSE.txt exists
- Although a top level license often exists, the emphasis is on individual source file licenses

<table>
<thead>
<tr>
<th>Grade</th>
<th>% Licenses Detected</th>
</tr>
</thead>
<tbody>
<tr>
<td>A+</td>
<td>97 - 100</td>
</tr>
<tr>
<td>A</td>
<td>93 - 96</td>
</tr>
<tr>
<td>A-</td>
<td>90 - 92</td>
</tr>
<tr>
<td>B+</td>
<td>85 - 89</td>
</tr>
<tr>
<td>B</td>
<td>80 - 84</td>
</tr>
<tr>
<td>B-</td>
<td>75 - 79</td>
</tr>
<tr>
<td>C+</td>
<td>70 - 74</td>
</tr>
<tr>
<td>C</td>
<td>65 - 69</td>
</tr>
<tr>
<td>C-</td>
<td>55 - 64</td>
</tr>
<tr>
<td>D+</td>
<td>50 - 54</td>
</tr>
<tr>
<td>D</td>
<td>40 - 49</td>
</tr>
<tr>
<td>D-</td>
<td>30 - 39</td>
</tr>
<tr>
<td>F</td>
<td>&lt;= 29</td>
</tr>
</tbody>
</table>

*based on idea from: Mark Gisi, Windriver
Example: OpenStack Packages*

- ironic-sysinv-1.0.tar.bz2
- novnc-0.4.tar.bz2
- keyring-3.2.zip

*source: Mark Gisi, Windriver
http://spdx.windriver.com/pkg_upload.aspx
Building on Best Practices in Communities

<table>
<thead>
<tr>
<th>Package</th>
<th>Foundation</th>
<th>Grade</th>
<th>% Detected</th>
</tr>
</thead>
<tbody>
<tr>
<td>apache-tomcat-7.0.47-src.tar.gz</td>
<td>Apache</td>
<td>A+</td>
<td>99.5</td>
</tr>
<tr>
<td>httpd-2.4.6.tar.bz2</td>
<td>Apache</td>
<td>A+</td>
<td>98.4</td>
</tr>
<tr>
<td>autoconf-2.69.tar.gz</td>
<td>FSF</td>
<td>A+</td>
<td>100</td>
</tr>
<tr>
<td>grep-2.20.tar.xz</td>
<td>FSF</td>
<td>A+</td>
<td>94</td>
</tr>
<tr>
<td>org.eclipse.datatools.sqltools-master.tar.gz</td>
<td>Eclipse</td>
<td>A-</td>
<td>92.3</td>
</tr>
<tr>
<td>org.eclipse.dltk.core-master.tar.gz</td>
<td>Eclipse</td>
<td>A-</td>
<td>90.4</td>
</tr>
</tbody>
</table>

- It is a best practice to include a license notice in every file.
- Apache & FSF packages are generally getting it right.
  - Key is stopping problems at the source! :-)

*table from: Mark Gisi, Windriver*
Projects tracking license at file level?

Dependent on Community Governance

- Apache Software Foundation
- Free Software Foundation
- The Linux Foundation
- Eclipse Foundation
- OpenStack Foundation
- ...

Also depends on Distribution’s project packaging.
Goal:
Get same level of *automatically* detectable information through entire ecosystem by increasing transparency on licensing.
Transparency of Software’s Licensing Data

Standard method for summarizing licensing at file level

- Need simple “License Coverage Grade” per project
  - provides grading A+-F based on transparent heuristics.
  - Simple to generate from SPDX document & sources.
- Need to develop open source command line tool to implement.
- Work with projects to “self score” (code authors).
- Work with foundations and distributions to adopt as part of packaging and distribution.
## 2016 Ecosystem Automation

<table>
<thead>
<tr>
<th>Status</th>
<th>Goal</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Common language to communicate licensing data</td>
<td>SPDX</td>
</tr>
<tr>
<td></td>
<td>Open Source tools to generate licensing data summaries</td>
<td>FOSSology, SPDX-tools</td>
</tr>
<tr>
<td></td>
<td>Keep licensing data current with every source change</td>
<td>DoSOCSv2, LiD, ...</td>
</tr>
<tr>
<td></td>
<td>Transparency of software’s licensing data</td>
<td>?</td>
</tr>
<tr>
<td></td>
<td>Common processes to pass licensing data with software</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Adoption by key projects, distributions, repositories...</td>
<td></td>
</tr>
</tbody>
</table>

*2016 Ecosystem Automation is a project initiated by the Linux Foundation to improve the compliance and licensing process within the open source community. The goals outlined in the table aim to standardize and automate the process of managing and communicating software licenses.*
Software Supply Chain Information Needs

Products today are built on many, many layers of software packages interacting together.

Product creators need to:

▪ understand which security vulnerabilities may be relevant
▪ understand who may be able to fix them
▪ understand distribution obligations associated with software’s licensing terms
Supporting Supply Chain Requirements

For each package:

▪ understand which security vulnerabilities may be relevant
  ⇒ link it to NIST Common Platform Enumeration (CPE), which will permit lookups to CVEs & CWEs as they change, via NISTs databases.

▪ understand who may be able to fix them
  ⇒ who are the copyright holders of all the files?

▪ understand distribution obligations
  ⇒ what are all the licenses in use for the package?

SPDX 2.1:

▪ supports licensing & copyright at file level
▪ support summaries at package level and links to NIST CPE
www.openchainproject.org

OPENCHAIN PROJECT
Identifying common best practices in compliance programs that should be applied across a supply chain for efficient and effective compliance with open source licenses.
OpenChain: Building the Business Processes

• Identification of the origin and license of FOSS software
• Tracking FOSS software within the development process
• Performing FOSS review and identifying license obligations
• Fulfillment of license obligations when product ships
• Oversight for Open Source Compliance Program, creation of policy, and compliance decisions
• Training
Common processes to pass licensing data

Supply chain Processes: OpenChain Project

- Curriculum (lead: Shane Coughlan):
  - 175 contributed slides from ARM, Qualcomm, Philips, Samsung
  - Curated down to 75 slides in 7 sections
- Conformance (lead: Miriam Ballhausen): Self-Conformance to online Questionnaire is first phase.

Community Project Processes: varied, based on community

- Need to interface community practices (1 license file per package) better with supply chain needs (licensing information at source file level).
## 2016 Ecosystem Automation

<table>
<thead>
<tr>
<th>Status</th>
<th>Goal</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Common language to communicate licensing data</td>
<td>SPDX</td>
</tr>
<tr>
<td></td>
<td>Open Source tools to generate licensing data summaries</td>
<td>FOSSology, SPDX-tools</td>
</tr>
<tr>
<td></td>
<td>Keep licensing data current with every source change</td>
<td>DoSOCSv2, LiD, ...</td>
</tr>
<tr>
<td></td>
<td>Transparency of software’s licensing data</td>
<td>?</td>
</tr>
<tr>
<td></td>
<td>Common processes to pass licensing data with software</td>
<td>OpenChain</td>
</tr>
<tr>
<td></td>
<td>Adoption by key projects, distributions, repositories...</td>
<td></td>
</tr>
</tbody>
</table>
Adoption in Ecosystem

- **Adoption SPDX License Identifiers:**
  - Debian recognized since DEP5 adopted, Fedora transitioning.
  - Linux Foundation transitioning, Eclipse considering.
  - New project in Package Manager Repositories adopting
  - Github adopted for projects in September 2016 (see Licenses API)!
Adoption in Ecosystem

- **Use of SPDX License Tags in Source Files:**
  - Developer initiated in U-Boot in 2013 for efficiency and to help with automatic processing.
  - Selective upstream projects adopt based on developer preferences.
  - Linux Foundation projects adopting: started adding to Linux in November.
  - “Open Government Partnership” created a [best practices template](#) for Open Source Policy that includes SPDX-License-Identifiers in December, France adopting “as is”.

Examples:

```plaintext
Original software

<AUTHORS>

Copyright © 2013-2015 ENTITY_NAME

SPDX-License-Identifier: GPL-3.0

v1.0
```
Adoption in Ecosystem

- **SPDX Specification:**
  - Windriver releases with all their products with SPDX documents. Hosts free service to generate documents.
  - Companies able to use commercial tools able to generate documents (BlackDuck, Palamida, SourceAuditor, etc.) as well as open source tools (FOSSology, homegrown, etc.).
  - Upstream projects need open-source based command line tools to integrate into CI loops (DoSOCSv2, LiD, ...)
  - Used to structure internal databases in large companies (Samsung, TI, ARM, Intel, Siemens ... )
# 2016 Ecosystem Automation

<table>
<thead>
<tr>
<th>Status</th>
<th>Goal</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Common language to communicate licensing data</td>
<td>SPDX</td>
</tr>
<tr>
<td></td>
<td>Open Source tools to generate licensing data summaries</td>
<td>FOSSology, SPDX-tools</td>
</tr>
<tr>
<td></td>
<td>Keep licensing data current with every source change</td>
<td>DoSOCSv2, LiD, ...</td>
</tr>
<tr>
<td></td>
<td>Transparency of software’s licensing data</td>
<td>?</td>
</tr>
<tr>
<td></td>
<td>Common processes to pass licensing data with software</td>
<td>OpenChain</td>
</tr>
<tr>
<td></td>
<td>Adoption by key projects, distributions, repositories...</td>
<td>Github, Debian, ....</td>
</tr>
</tbody>
</table>
How Can You Help?

- Add “SPDX-License-Identifier” tags to open source files where you have commit rights if they do not already have standard licenses.
  - If the license is common, but not on the SPDX license list, ask to be added.

- Participate (develop, test, report bugs, document) FOSSology creating command line interfaces to generate SPDX files and incorporate better agents.

- Generate SPDX documents for the projects you participate in
  - Make sure licenses are consistent ;-)

- Participate in defining policies and open source tools for industry wide standard on a “License Coverage Grade” based on analyzing SPDX documents and source code for projects in 2017.
Closing thoughts...

If everyone does a bit,

▪ we can make easy to understand which license apply for products,
▪ we can respect the open source developers intent when they contributed code

Step by step, together we can get this automated!

Source: https://catalog.archives.gov/id/535413
Thank you!  Questions?

kstewart@linuxfoundation.org