



OPENCHAIN

05-07-2018

OpenChain Workgroup - The Linux Foundation

All materials are made available under the Creative Commons CC0 1.0 Universal license.

Antitrust Policy Notice

- › Linux Foundation meetings involve participation by industry competitors, and it is the intention of the Linux Foundation to conduct all of its activities in accordance with applicable antitrust and competition laws. It is therefore extremely important that attendees adhere to meeting agendas, and be aware of, and not participate in, any activities that are prohibited under applicable US state, federal or foreign antitrust and competition laws.
- › Examples of types of actions that are prohibited at Linux Foundation meetings and in connection with Linux Foundation activities are described in the Linux Foundation Antitrust Policy available at <http://www.linuxfoundation.org/antitrust-policy>. If you have questions about these matters, please contact your company counsel, or if you are a member of the Linux Foundation, feel free to contact Andrew Updegrave of the firm of Gesmer Updegrave LLP, which provides legal counsel to the Linux Foundation.

Agenda

- Roll call
- Project Update
- Specification Work Team
- Conformance Work Team

Project Update

- OpenChain Conformance - New Entities
- OpenChain @ Legal and Licensing Workshop
- OpenChain Japan Work Group
- OpenChain Case Studies

Specification Working Group

- Now that version 1.2 has been released ...
- Kick off the drafting of the next version
 - Revisit guiding principles
 - Discuss objectives
 - Version number: 1.3? 1.5? 2.0?
 - Highlight key issues to tackle
 - <https://github.com/OpenChain-Project/Specification/issues>
 - Discuss roadmap

Specification Working Group Guiding Principles

1. Build trust around the use of open source in constructing Software Solutions that are shared with others (**with focus on license compliance**)
2. Less is More
 - Avoid boiling the ocean - Focus specifically on providing the necessary and sufficient requirements of a “quality” compliance program
 - Focus on meaningful pain points based on actual practice use cases
3. Focus of the what and why (avoid the how and when)
 - Embrace the implementation of different practices to solve a given requirement
 - Avoid providing specific legal advice or specific best practices
4. Function as an open development initiative - open to all to contribute - inclusion via discussion and consensus that adhere to these guiding principles

Specification Working Group Target Audience (Value Recipient)

- The Benefactors:
 1. Recipients of Software (e.g, supply chain, organization's customers)
 2. Legal Group - Assurance that engineering is doing the right thing

- Focus on Open Source License Compliance

Conformance Working Group

Conformance Questionnaire Format

- Localization <https://certification.openchainproject.org/> requires a questionnaire format that supports multiple languages
- The current format is a CSV file maintained at <https://github.com/OpenChain-Project/conformance-questionnaire>
- Anyone on the conformance team should feel comfortable suggesting updates to the questionnaire in the format maintained in github
- We need to agree on the new format before we can proceed with localizing the website

Conformance Working Group

Proposal

- Multiple files – one per language
- Positive: Multiple files support a more natural localization workflow where there is a primary file edited by the conformance team and localization files maintained by local language experts
- Negative: Need to upload multiple files for any update. Can be solved by enhancing the website admin functions to pull changes from Github as a set of files.
- JSON file format
- Positives: Structure format, removes the need for section titles file, easier to validate, better for Github tracking
- Negative: More complex to read and update

Conformance Working Group

JSON Example

```
{
  "language": "eng",
  "sections": [
    {
      "sectionNumber": 1,
      "sectionName": "G1",
      "sectionTitle": "G1: Know Your FOSS Responsibilities",
      "questions": [
        {
          "questionNumber": "1.a",
          "specReferenceNumbers": [ "1.1", "1.1.1" ],
          "questionText": "Do you have a documented policy that governs FOSS license compliance of the Supplied Software distribution (e.g., via training, internal wiki, or other practical communication method)?",
          "answerType": "YES_NO",
          "correctAnswer": "Yes",
          "evidencePrompt": "",
          "evidenceValidation": "",
          "subQuestionOfNumber": ""
        },
        {
          "questionNumber": "1.b",
          "specReferenceNumbers": [ "1.1" ],
          "questionText": "Is the policy internally communicated?",
          "answerType": "YES_NO",
          "correctAnswer": "Yes",
          "evidencePrompt": "",
          "evidenceValidation": "",
          "subQuestionOfNumber": ""
        }
      ]
    },
    {
      "sectionNumber": 2,
      "sectionName": "G2",
      "sectionTitle": "G2: Assign Responsibility for Achieving Compliance",
      "questions": [
        {
```



Conformance Working Group

Format Criteria

- Easy to maintain (by us)
- Not requiring expensive proprietary tools to maintain
- Allow for collaboration and change tracking using github (e.g. binary files or tools that do not maintain the same order would be a problem for this criteria)
- Easy to upload into the conformance website (e.g. single file would be best)
- Support easy localization
- Effort needed to update the site software

Conformance Working Group

Format Options

- Multiple CSV files – one per language supported – with the same column definitions as the current CSV file
- Single CSV file with an additional column containing the language of the text. We would basically duplicate the rows and replace the question and prompt text with the language local versions.
- Single XML file – Create an XML format text with elements and attributes to describe questions. This is similar to the approach taken by the SPDX legal team for describing licenses.
- Multiple XML files – One XML file per language
- Microsoft Excel or OpenOffice calc file with one sheet per language
- JSON

Conformance Working Group

Format	1 – Easy to maintain	2 – no prop. Tools	3 – <u>github</u>	4 – easy upload	5 – localization	6 – Update Effort
A – Multiple CSV	OK – Same as today except for localization	Yes – text editor works fine	OK as long as Excel is not used (messes up the encoding and changes every row)	Not very convenient – need to track and upload multiple files	Text encoding is not specified, can be a problem	Low effort – leverages current format
B – Single CSV	OK – A bit more work to observe the language column	Yes – text editor works fine	OK as long as Excel is not used (messes up the encoding and changes every row)	Good – Single file	Text encoding is not specified, can be a problem	Low effort – leverages current format
C – Single XML	More difficult – need to edit XML files where is a bit more challenging than a CSV file	Maybe – Can be edited with a text editor but there are XML editors which are easier to use	OK as long as an XML editor doesn't re-arrange everything	Good – single file	Preferred format for localization	Medium – Requires XML parsing, new format support
D – Multiple XML files	More difficult – need to edit XML files where is a bit more challenging than a CSV file	Maybe – Can be edited with a text editor but there are XML editors which are easier to use	OK as long as an XML editor doesn't re-arrange everything	Not very convenient – need to track and upload multiple files	Preferred format for localization	Medium – Requires XML parsing, new format support
E – Excel or OpenOffice	Easy to use tools	Most users have access to Excel or OpenOffice Calc	Bad - Although the internal format is XML, the tools typically re-arrange the text	Good – single file	Text encoding is not typically specified	Difficult – more proprietary format with limited open source library support
F – Single JSON file	Easier than XML but more challenging than CSV	Yes – text editor works fine	Best for <u>github</u> – text format, easy to see differences	Best – single file, removes the need for sections CSV file	Support from several tools, UTF-8 encoding standard	Medium – requires some parsing update but JSON is already implemented. Easier than XML.
G – Multiple JSON files	Easier than XML but more challenging than CSV	Yes – text editor works fine	Best for <u>github</u> – text format, easy to see differences	Not very convenient – need to track and upload multiple files	Support from several tools, UTF-8 encoding standard	Medium – requires some parsing update but JSON is already implemented. Easier than XML.



Thoughts?
Questions?