An Unexpected Journey
Implementing License Matching using the SPDX License List
Kris Reeves
• kris@pressbuttonllc.com
• https://github.com/myndzi
• https://www.npmjs.com/profile/myndzi
• myndzi @ freenode

Gary O’Neall
• gary@sourceauditor.com
• https://github.com/goneall
• http://sourceauditor.com
Agenda

• Where we are today
  • SPDX License List master files maintained in a spreadsheet and text files
  • Web accessible license list (generated from master files)
  • Human readable matching guidelines with minimal markup
  • Java libraries which partially implements matching guidelines

• The Journey ...
How the SPDX License List is Maintained

- git.spdx.org license-list & github.com/spdx/license-list
- Spreadsheet
- SPDX Tools License RDFa Generator
- licenses.json
- indexes.html (RDFa)
- [spdxid].html (RDFa)
- [spdxid].json
- Other Formats (text, raw HTML)
- License Template Files (text)
SPDX License List Repository

• A Git repository of the SPDX License List is maintained at git.spdx.org.

• Web access to the repository: [http://git.spdx.org/?p=license-list.git;a=summary](http://git.spdx.org/?p=license-list.git;a=summary) — or - [https://github.com/spdx/license-list](https://github.com/spdx/license-list)

• Tags are created for each release of the license list in the format “v” followed by the version number.
SPDX License List

The SPDX License List is a list of commonly found open source software licenses for the purposes of being able to easily and efficiently identify such licenses in an SPDX document. The SPDX License List includes a standardized short identifier, full name for each license, vetted license text, other basic information, and a canonical permanent URL. By providing a short identifier, users can efficiently refer to a license without having to redundantly reproduce the full license.

Other information relevant to the SPDX License List:

- **Overview**: For general information about the SPDX License List, including principles for inclusion of a license and an explanation of the field contained on the list.
- **Request New License**: For instructions on how to propose additional licenses be added to the SPDX License List.
- **Matching Guidelines**: Guidelines for what constitutes a license match to the SPDX License List when generating an SPDX file.
- **Master Files**: The HTML pages you see here are generated from the master files for the SPDX License List. The master files include both a spreadsheet listing all the licenses and fields; and the text for each license in a .txt file. These files are available in a Git repository.

SPDX License List version 1.19.

<table>
<thead>
<tr>
<th>License Type</th>
<th>Full name</th>
<th>Identifier</th>
<th>OSI Approved</th>
<th>License Text</th>
</tr>
</thead>
<tbody>
<tr>
<td>Academic Free License v1.1</td>
<td>AFL-1.1</td>
<td>Y</td>
<td>License Text</td>
<td></td>
</tr>
<tr>
<td>Academic Free License v1.2</td>
<td>AFL-1.2</td>
<td>Y</td>
<td>License Text</td>
<td></td>
</tr>
<tr>
<td>Academic Free License v2.0</td>
<td>AFL-2.0</td>
<td>Y</td>
<td>License Text</td>
<td></td>
</tr>
<tr>
<td>Academic Free License v2.1</td>
<td>AFL-2.1</td>
<td>Y</td>
<td>License Text</td>
<td></td>
</tr>
<tr>
<td>Academic Free License v3.0</td>
<td>AFL-3.0</td>
<td>Y</td>
<td>License Text</td>
<td></td>
</tr>
</tbody>
</table>
SPDX License Pages

• Location: http://spdx.org/licenses/[spdx-short-id] where spdx-short-id is the SPDX short identifier for the license

BSD 2-clause "Simplified" License

Full name
BSD 2-clause "Simplified" License

Short identifier
BSD-2-Clause

Other web pages for this license
• http://www.opensource.org/licenses/BSD-2-Clause

Notes
None

Text
Copyright (c) <YEAR>, <OWNER>
All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1) Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2) Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Standard License Header
Just Release - JSON Access

• What is JSON?
  • JSON (JavaScript Object Notation) is a lightweight data-interchange format.
  • Extremely popular with parsers available in several languages

• Location
  • The index json file (http://spdx.org/licenses/license.json) contains the license list version, the license ID’s, if the license is deprecated, link to the detailed license JSON file
  • Each license JSON file contains terms fully describing the license
What is RDFa?

RDFa (Resource Descriptor Framework in Attributes) extends HTML (and XHTML) with attributes allowing RDF data to be easily extracted from the same web pages which can also be rendered using web browsers.

Several parsers and tools are available – see [http://rdfa.info](http://rdfa.info).

Location

- The index html file ([http://spdx.org/licenses/index.html](http://spdx.org/licenses/index.html)) contains same information as the JSON files.
- Each license HTML file contains terms fully describing the license.
JSON Names / RDFa Terms

- License ID - http://spdx.org/rdf/terms#licenseId
- License Name - http://www.w3.org/2000/01/rdf-schema#label or http://spdx.org/rdf/terms#name
- License Header - http://spdx.org/rdf/terms#standardLicenseHeader
- License Template - http://spdx.org/rdf/terms#licenseTemplate
- OSI Approved - http://spdx.org/rdf/terms#isOsiApproved
- Cross Reference URL - http://www.w3.org/2002/07/owl#sameAs or http://www.w3.org/2000/01/rdf-schema#seeAlso
- isDeprecatedLicense – True if the license ID has been deprecated
Matching Guidelines

• Guidelines to be used for the purposes of matching licenses and license exceptions against those included on the SPDX License List
• Available at https://spdx.org/spdx-license-list/matching-guidelines
• Some are easy to implement in tools (synonyms, white space), others are more difficult (e.g. what is extraneous text at the end of a license?)
Replaceable and Optional Text

• Included in the license templates (see SPDX specification, Appendix II)
• Optional Format: <<beginOptional; name=Name>> optional text<<endOptional>>
• Variable Format: <<var; name=Name; original=original text; match=Regex>>
• Template text can be accessed through the JSON and/or RDFa files on spdx.org/licenses
SPDX Tools Library

• Open source (Apache 2.0 licensed) Java code implementing tools for the SPDX standard

• Binary .jar files available:
  • Maven central repository the group ID org.spdx and the artifact ID spdx-tools

• Source code available:
  • git.spdx.org – spdx-tools project & [https://github.com/spdx/tools](https://github.com/spdx/tools)

• Bugs / enhancement ideas can be reported:
  • [bugs.linuxfoundation.org](https://bugs.linuxfoundation.org) – project SPDX/tools and Github issues

• Contributions welcome!
The Journey...
It Begins

• “Can you get a list of licenses in use?”
• Find Node-license-finder, send off a list
  • What is the need that drives the question?
• Approach is not really ideal
  • Node.js projects can wind up with lots of dependencies
  • New dependencies can be brought in easily over time
  • How about some way to validate automatically that unapproved licenses don’t slip in?
A Node Project’s Dependencies
Node License Validator

• Uses Node License Finder under the hood
• List available licenses
• Assert acceptable licenses / exceptions
• …what’s this?

```
npm i node-license-validator
how2 learn more
mynd2i published 3 months ago
1.2.2 is the latest of 5 releases

npm i node-license-validator@1.2.2

```

```
ISC
```

```
json-pare-helpfulerror@1.0.3: MIT
- lcid@1.0.0: MIT
- minimatch@2.0.10: ISC
- ntl@1.4.0: MIT

node-license-validator@1.2.2: ISC, MIT, WTFPL
- npm-package-data@2.3.3: BSD, BSD-Z-Clause
- npm-package-arg@4.1.0: ISC
- number-is-nan@1.0.0: MIT
- once@1.2.3: ISC
- os-locale@1.4.0: MIT
- path-is-absolute@1.0.0: MIT
- read-installed@4.0.3: ISC
- read-package-json@2.0.3: ISC
- readdir-scoped-modules@0.1.2: ISC
- semver@4.3.6: ISC
- slide@1.1.6: ISC
- spdx-compare@0.1.2: MIT
- spdx-correction@1.0.2: Apache, Apache-2.0
- spdx-exceptions@1.0.4: Expires-BY-3.0

krisre:0/bash-1:tmux~ "kris.re dev" 14:20 13-Mar-19
```
License Matching is Hard

• …so some people take the easy route
• Node license finder:

```javascript
var patterns = [ {
    'name': 'BSD',
    'regex': [ /(?:^|\s)BSD\s/ ]
}, {
    'name': 'GPL',
    'regex': [
        /(?:^|\s)GPL\s/,
        /(?:^|\s)GPLv\d\s/
    ]
}, {
    'name': 'LGPL',
    'regex': [ /(?:^|\s)LGPL\s/ ]
}, ... ];
```
Available Packages

• Node License Finder (iandotkelly)
  • [https://www.npmjs.com/package/nlf](https://www.npmjs.com/package/nlf)
  • Prints licenses used by dependencies recursively
  • Provides a summary of used licenses

• License Checker (davglass)
  • [https://www.npmjs.com/package/license-checker](https://www.npmjs.com/package/license-checker)
  • Nicer formatting
  • No summary

• Licensecheck (marcello3d)
  • Actually uses license text to match, SPDX templates
  • Doesn’t implement matching guidelines
  • Not maintained
License Matching is Hard (2)

- License-checker:

```javascript
var MIT_LICENSE = /permission is hereby granted, free of charge, to any/;
var BSD_LICENSE = /edistribution and use in source and binary forms, with or without/;
var ISC_LICENSE = /The ISC License/;
var MIT = /MIT\b/;
```

- Licensecheck:

```javascript
function normalizeText(text) {
  return text.replace(/[^a-z0-9\s]/ig, '')
    .toLowerCase()
    .trim()
    .split(/\[\s\n]+/)
    .join(' ')
}
```
Rolling My Own

• Why Node?
  • Matches tool chain, no need to bring in JRE
  • Existing packages I can take advantage of

• Why SPDX?
  • Large list of real world licenses and templates
  • package.json standardized on using SPDX identifiers
  • Well-thought-out and –defined matching rules

• This will be easy, just follow the guideline!
About Those Matching Guidelines...

• Easy: Section 4 – capitalization
  • Capitalization, alternate spellings, etc. are computer-easy

• Troublesome: Section 7 – bullets and numbering
  • Easy gotchas: templates have a wide variety of formatting
    • Symbols such as * or -, even bullet character: • (ASCII or UTF-8?)
    • Letters, numbers, with various formatting, e.g. (A) or 12.
    • Roman numerals: ii. is valid, but to. is not

• …what?: Section 12 – Extraneous text
  • How is a computer to determine what text is extraneous?

• Some of this is tricky, but for matching purposes we don’t have to get it all 100% perfect
And Then, the Templates…

• Templates include markup for sections that guidelines say to ignore, e.g. copyright declarations
  • The markup assigns these chunks to variables, indicating they’re important to hang on to

• Extraneous text is included, but usually not marked optional

• Some templates (ISC, at the time) include organization-specific text that’s not marked up and not covered by matching guidelines

• Some licenses in the wild just vary widely
  • BSD-3-Clause:
    ( neither the names?( of)? .+ nor the names of (the|its|his|her|their|project) contributors may be | the names?( of)? .+ may not be )

• Pretty much have to patch/override/extend some of the templates
How’s It Perform?

- Top ~ 250 packages on NPM, and all their dependencies
  - 1185 package.json files
  - 1167 potential ‘license’ sections in README markdown
  - 881 plain ‘LICENSE’ files
  - 45 ‘LICENSE’ files in markdown
  - 8 potential ‘license’ sections in plain text README

- Licensecheck v0.3.0 stock
  - 24 unknown licenses, 1396 unmatched licenses

- Licensecheck fork
  - 21 unknown licenses, 1393 unmatched licenses

- Doesn’t seem like we gained much...
  - package.json is reasonably reliable
  - NPM packages strongly skewed towards only a few licenses
What Did We Gain?

- Correctness
  - Implemented the matching guidelines fully
- Take advantage of markup
  - We can handle more variance in formatting
    - In the sample data used it doesn’t appear to be much of a problem
  - Variant licenses can be handled by the template data rather than manual supplements in the matcher code
  - Can utilize variable assignment from matching markup in dependent tools
Where Are We Going From Here?

• More markup
  • Make the SPDX repository responsible for specifying human-easy but computer-hard matching rules

• Better markup
  • Move to XML for widespread familiarity and tool support

• Easier contribution
  • Move to Github for easier contribution, review by general public
More Markup

• Many templates don’t have markup
  • Make all templates a consistent format for consumption
  • Still won’t address places that need markup that don’t have it yet

• Some information is kept in a side channel (spreadsheet)
  • Resolved by “better markup” being able to include this information in the license template itself

• Expand templates that don’t have markup
  • Aided by making contribution easier
Better Markup

• Use XML
  • Available everywhere, well-understood
• Make the templates reasonably easy for humans to work with
• Single source of truth in a diff-friendly format
• Proposal (work in progress): http://wiki.spdx.org/view/Legal_Team/Template_proposal
Easier Contribution

• Use Github
  • Bugzilla not as friendly to find or interact with
  • Built in tools that make proposed changes and discussions about them easy
  • Extant spirit of collaboration and contribution

• Distribute the work
  • Getting away from the spreadsheet makes it easier for multiple contributors
From Here to There – Data

• Normalizing code written for matching not nearly sufficient for automatic translation to new XML proposal
• Full license list pass by hand with the aid of a tool to section off chunks of license templates
  • Aware of existing SPDX markup and can convert it
  • Can identify bullets and tag them
  • Can bring in related XML fields
• Systematic review by legal team contributors to validate generated XML
• https://github.com/myndzi/license-list/tree/xml-test/src (WIP)
From Here to There - Process

• Work with each license as a single source of truth
• Make it easy for people to interact and contribute when they have a need
• Make it easy for SPDX team to review and implement change
Once We’re There

• Users of SPDX-consuming tools can easily communicate needs with the SPDX team
• Authors of SPDX-consuming tools can perform matching easier and with less code, and the result will be more consistent across tools
• Contributors to the SPDX project can more easily control and manage changes and additions
Questions or comments?

Join us Thursday afternoon
Backup Slides: Additional Information on SPDX Tools
The primary interfaces to the SPDX standard licenses are the following methods in the `org.spdx.rdfparser.SPDXLicenseInfoFactory` class:

- `String getLicenseListVersion()`
- `SPDXStandardLicense getStandardLicenseById(String licenseId)`
- `String[] getStandardLicenseIds()`
- `isStandardLicenseId(String licenseId)`
- `SPDXLicenseInfo parseSPDXLicenseString(String licenseString)`
licenseTemplate Package

• The package org.spdx.licenseTemplate contains classes to manage the license templates used in SPDX standard licenses.
• See the JavaDocs for this package (located in the doc directory of the source code or in the javadoc jar file in Maven) for a description of the interface to this package.
• The primary interfaces are in the org.spdx.licenseTemplate. SpdxLicenseTemplateHelper class.
LicenseCompareHelper

• The class `org.spdx.compare.LicenseCompareHelper` contains methods to compare license text to determine if they are equivalent according to the license matching guidelines maintained by the SPDX workgroup
  • `boolean LicenseCompareHelper.isTextStandardLicense(SPDXStandardLicense license, String licenseText)`