

Software Package Data Exchange (SPDX®) Specification

Draft: 2.0rc3-20150303

Copyright © 2010-2015 Linux Foundation and its Contributors. This work is licensed under the Creative Commons Attribution License 3.0 Unported (CC-BY-3.0) reproduced in its entirety in Appendix V herein. All other rights are expressly reserved.

With thanks to Adam Cohn, Andrew Back, Ann Thornton, Bill Schineller, Bruno Cornec, Ciaran Farrell, Daniel German, Debra McGlade, Dennis Clark, Ed Warnicke, Eran Strod, Eric Thomas, Esteban Rockett, Gary O'Neill, Guillaume Rousseau, Hassib Khanafer, Jack Manbeck, Jaime Garcia, Jeff Luszcz, Jilayne Lovejoy, John Ellis, Karen Copenhaver, Kate Stewart, Kim Weins, Kirsten Newcomer, Liang Cao, Marc-Etienne Vargenau, Mark Gisi, Marshall Clow, Martin Michlmayr, Martin von Willebrand, Matt Germonprez, Michael J. Herzog, Michel Ruffin, Nuno Brito, Paul Madick, Peter Williams, Phil Robb, Philip Odenice, Philip Koltun, Pierre Lapointe, Rana Rahal, Sameer Ahmed, Scott K Peterson, Scott Lamons, Scott Sterling, Shane Coughlan, Steve Cropper, Stuart Hughes, Tom Callaway, Tom Vidal, Thomas F. Incorvia, Venkata Krishna and Zachary McFarland for their contributions and assistance.

TABLE OF CONTENTS

- [1 Rationale](#)
 - [1.1 Charter](#)
 - [1.2 Definition](#)
 - [1.3 Why is a common format for data exchange needed?](#)
 - [1.4 What does this specification cover?](#)
 - [1.5 What is not covered in the specification?](#)
 - [1.6 Format Requirements:](#)
 - [1.7 Conformance](#)
 - [1.8 Differences from \[SPDX Specification 1.2\]\(#\)](#)
- [2 Document Creation Information](#)
 - [2.1 \[SPDX Version\]\(#\)](#)
 - [2.2 \[Data License\]\(#\)](#)
 - [2.3 \[SPDX Identifier\]\(#\)](#)
 - [2.4 \[Document Name\]\(#\)](#)
 - [2.5 \[SPDX Document Namespace\]\(#\)](#)
 - [2.6 \[External Document References\]\(#\)](#)
 - [2.7 \[License List Version\]\(#\)](#)
 - [2.8 \[Creator\]\(#\)](#)
 - [2.9 \[Created\]\(#\)](#)
 - [2.10 \[Creator Comment\]\(#\)](#)
 - [2.11 \[Document Comment\]\(#\)](#)
- [3 Package Information](#)
 - [3.1 \[Package Name\]\(#\)](#)
 - [3.2 \[Package SPDX Identifier\]\(#\)](#)
 - [3.3 \[Package Version\]\(#\)](#)
 - [3.4 \[Package File Name\]\(#\)](#)
 - [3.5 \[Package Supplier\]\(#\)](#)
 - [3.6 \[Package Originator\]\(#\)](#)
 - [3.7 \[Package Download Location\]\(#\)](#)
 - [3.8 \[Package Verification Code\]\(#\)](#)
 - [3.9 \[Package Checksum\]\(#\)](#)
 - [3.10 \[Package Home Page\]\(#\)](#)
 - [3.11 \[Source Information\]\(#\)](#)
 - [3.12 \[Concluded License\]\(#\)](#)
 - [3.13 \[All Licenses Information from Files\]\(#\)](#)
 - [3.14 \[Declared License\]\(#\)](#)
 - [3.15 \[Comments on License\]\(#\)](#)
 - [3.16 \[Copyright Text\]\(#\)](#)
 - [3.17 \[Package Summary Description\]\(#\)](#)
 - [3.18 \[Package Detailed Description\]\(#\)](#)
 - [3.19 \[Package Comment\]\(#\)](#)

- 4 [File Information](#)
 - 4.1 [File Name](#)
 - 4.2 [File SPDX Identifier](#)
 - 4.3 [File Type](#)
 - 4.4 [File Checksum](#)
 - 4.5 [Concluded License](#)
 - 4.6 [License Information in File](#)
 - 4.7 [Comments on License](#)
 - 4.8 [Copyright Text](#)
 - 4.9 [Artifact of Project Name](#)
 - 4.10 [Artifact of Project Homepage](#)
 - 4.11 [Artifact of Project Uniform Resource Identifier](#)
 - 4.12 [File Comment](#)
 - 4.13 [File Notice](#)
 - 4.14 [File Contributor](#)
 - 4.15 [File Dependencies \(deprecated\)](#)
- 5 [Other Licensing Information Detected](#)
 - 5.1 [License Identifier](#)
 - 5.2 [Extracted Text](#)
 - 5.3 [License Name](#)
 - 5.4 [License Cross Reference](#)
 - 5.5 [License Comment](#)
- 6 [Relationships between SPDX Elements](#)
 - 6.1 [Relationship](#)
 - 6.2 [Relationship Comment](#)
- 7 [Annotations](#)
 - 7.1 [Annotator](#)
 - 7.2 [Annotation Date](#)
 - 7.3 [Annotation Type](#)
 - 7.4 [SPDX Identifier Reference](#)
 - 7.5 [Annotation Comment](#)
- 8 [Review Information \(deprecated\)](#)
 - 8.1 [Reviewer \(deprecated\)](#)
 - 8.2 [Review Date \(deprecated\)](#)
 - 8.3 [Review Comment \(deprecated\)](#)

[Appendix I: SPDX License List](#)

[Licenses with Short Form Identifiers](#)

[Exceptions List](#)

[Deprecated Licenses](#)

[Appendix II: License Matching Guidelines and Templates](#)

[Appendix III: RDF Data Model Implementation and Identifier Syntax](#)

[Agent and Tool Identifiers](#)

[Appendix IV: License Expression Syntax \(beta draft 0.98\)](#)

[Overview](#)

[Simple License Expressions](#)

Composite License Expressions

- 1) [Disjunctive OR Operator](#)
- 2) [Conjunctive AND Operator](#)
- 3) [Exception WITH Operator](#)
- 4) [Order of Precedence and Parentheses](#)

[Appendix V: Creative Commons Attribution License 3.0 Unported](#)

1 Rationale

1.1 Charter

To create a set of data exchange standards that enable companies and organizations to share license and component information (metadata) for software packages and related content with the aim of facilitating license and other policy compliance.

1.2 Definition

The Software Package Data Exchange (SPDX®) specification is a standard format for communicating the components, licenses, and copyrights associated with a software package. An SPDX file is associated with a particular software package and contains information about that package in the SPDX format.

1.3 Why is a common format for data exchange needed?

Companies and organizations (collectively “Organizations”) are widely using and reusing open source and other software packages. Compliance with the associated licenses requires a set of analysis activities and due diligence that each Organization performs independently, which may include a manual and/or automated scan of software and identification of associated licenses followed by manual verification. Software development teams across the globe use the same open source packages, but little infrastructure exists to facilitate collaboration on the analysis or share the results of these analysis activities. As a result, many groups are performing the same work leading to duplicated efforts and redundant information. The SPDX working group seeks to create a data exchange format so that information about software packages and related content may be collected and shared in a common format with the goal of saving time and improving data accuracy.

1.4 What does this specification cover?

1.4.1 SPDX Document Creation Information: Meta data to associate analysis results with a specific version of the SPDX file and license for use, and provide information on how, when, and by whom the SPDX file was created.

1.4.2 Package Information: Facts that are common properties of the entire package.

1.4.3 File Information: Facts (e.g. copyrights, licenses) that are specific to each file included in the package.

1.4.4 Other Licensing Information: A list of common licenses likely to be encountered and a standardized naming convention for referring to these licenses and other licenses also found within an SPDX document. This naming convention will also be the basis for extending this set of common licenses over time.

1.4.5 Relationships: Information on how Documents, Packages & Files relate to each other.

1.4.6 Annotations: Information about when and by whom the SPDX file was reviewed.

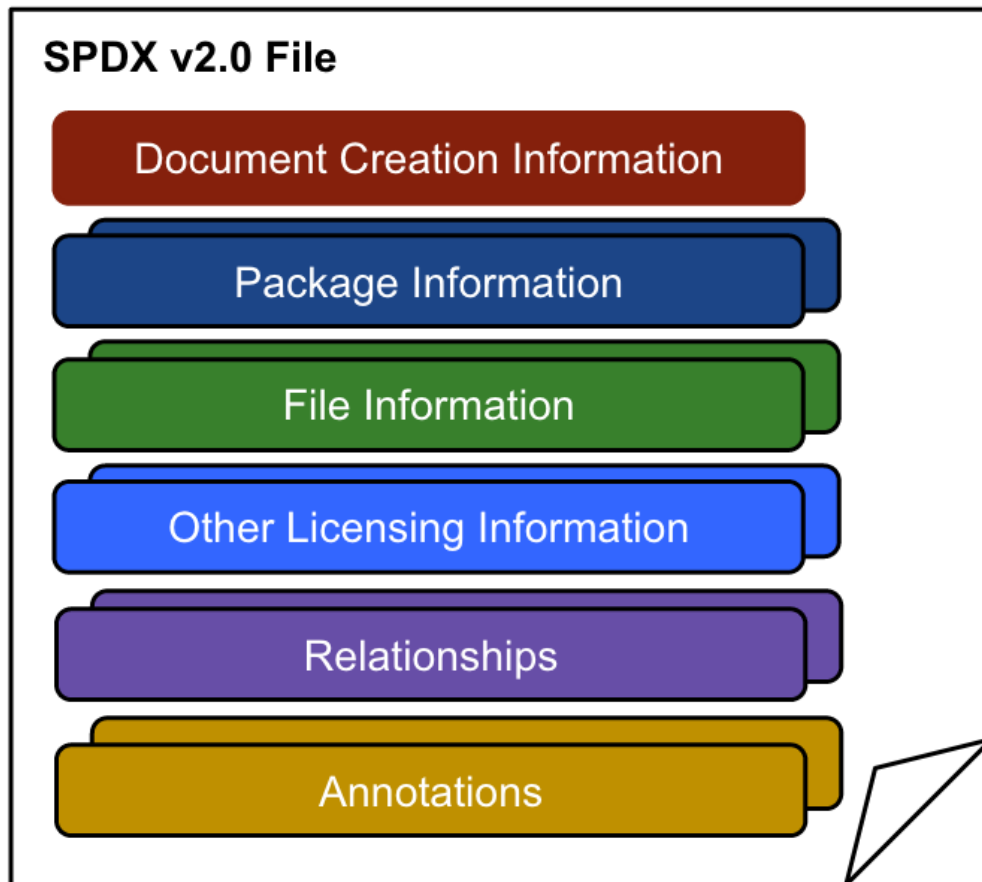


Figure 1 Overview of SPDX 2.0 document contents.

1.5 What is not covered in the specification?

- 1.5.1 Information that cannot be derived from an inspection (whether manual or using automated tools) of the package to be analyzed.
- 1.5.2 How the data stored in an SPDX file is used by the recipient.
- 1.5.3 Any identification of any patent(s) which may or may not relate to the package.
- 1.5.4 Legal interpretation of the licenses or any compliance actions that have been or may need to be taken.

1.6 Format Requirements:

- 1.6.1 Must be in a human readable form.

- 1.6.2** Must be in a syntax that a software tool can read and write.
- 1.6.3** Must be suitable to be checked for syntactic correctness independent of how it was generated (human or tool).
- 1.6.4** The SPDX file character set must support UTF-8 encoding.
- 1.6.5** Must permit automated specification syntax validation.
- 1.6.6** Resource Description Framework (RDF) can be used to represent this information, as can an annotate tag value flat text file.
- 1.6.7** Interoperability with an annotate tag format and the RDF format will be preserved.
- 1.6.8** Tags and RDF properties are case sensitive.

1.7 Conformance

1.7.1 A file can be designated an SPDX file, if it is compliant with the requirements of the SPDX Trademark License (See the [SPDX Trademark Page](#)).

1.7.2 The official copyright notice to be used with any verbatim reproduction and/or distribution of this SPDX Specification 2.0 is:

"Official SPDX[®] Specification 2.0. Copyright © 2010-2015 Linux Foundation and its Contributors. Licensed under the Creative Commons Attribution License 3.0 Unported. All other rights are expressly reserved."

1.7.3 The official copyright notice to be used with any non-verbatim reproduction and/or distribution of this SPDX Specification, including without limitation any partial use or combining this SPDX Specification with another work, is:

"This is not an official SPDX Specification. Portions herein have been reproduced from SPDX[®] Specification 2.0 found at spdx.org. These portions are Copyright © 2010-2015 Linux Foundation and its Contributors, and are licensed under the Creative Commons Attribution License 3.0 Unported by the Linux Foundation and its Contributors. All other rights are expressly reserved by Linux Foundation and its Contributors."

1.8 Differences from SPDX Specification 1.2

1.8.1 Abstraction has been applied to the underlying model with the inclusion of SPDX elements. With SPDX 2.0, the concept of an SPDX element is introduced (see Appendix III). This includes SPDX Documents, SPDX Files, and SPDX Packages, each of which gets associated with an SPDX Identifier which is denoted by "SPDXRef-".

1.8.2 SPDX relationships have been added to allow any SPDX element to have a relationship to other SPDX elements. This can be used for a variety of purposes including documenting a hierarchy of sub-packages, documenting the origin of an SPDX element and documenting modifications or corrections (annotations) to an SPDX element.

- 1.8.3** The ability to reference SPDX elements outside the current SPDX document itself (external references).
- 1.8.4** Additional file types are now supported.
- 1.8.5** Additional checksum algorithms are now supported.
- 1.8.6** Review Information Section is deprecated. It is recommended to provide document reviews with Annotations (Section 7).
- 1.8.7** A richer License Expression syntax has been introduced and documented in Appendix IV.

2 Document Creation Information

One instance is required for each SPDX file produced. It provides the necessary information for forward and backward compatibility for processing tools.

Fields:

2.1 SPDX Version

2.1.1 Purpose: Provide a reference number that can be used to understand how to parse and interpret the rest of the file. It will enable both future changes to the specification and to support backward compatibility. The version number consists of a major and minor version indicator. The major field will be incremented when incompatible changes between versions are made (one or more sections are created, modified or deleted). The minor field will be incremented when backwards compatible changes are made.

2.1.2 Intent: Here, parties exchanging information in accordance with SPDX specification need to provide 100% transparency as to which SPDX specification such Identification Information is conforming to.

2.1.3 Cardinality: Mandatory, one.

2.1.4 Data Format: “SPDX-M.N”

where:

M is major version number

N is minor version number.

2.1.5 Tag: “SPDXVersion:”

Example:

SPDXVersion: SPDX-2.0

2.1.6 RDF: spdx:specVersion

Example:

```
<SpdxDocument rdf:about="">
  <specVersion>SPDX-2.0</specVersion>
</SpdxDocument>
```

2.2 Data License

2.2.1 Purpose: Compliance with the SPDX specification includes populating the SPDX fields therein with data related to such fields ("SPDX-Metadata"). The SPDX specification contains numerous fields where an SPDX author may provide relevant explanatory text in SPDX-Metadata. Without opining on the lawfulness of "database rights" (in jurisdictions where applicable), such explanatory text is copyrightable subject matter in most Berne Convention countries. By using the SPDX specification, or any portion hereof, you hereby agree that any copyright rights (as

determined by your jurisdiction) in any SPDX-Metadata, including without limitation explanatory text, shall be subject to the terms of the Creative Commons CC0 1.0 Universal license. For SPDX-Metadata not containing any copyright rights, you hereby agree and acknowledge that the SPDX-Metadata is provided to you "as-is" and without any representations or warranties of any kind concerning the SPDX-Metadata, express, implied, statutory or otherwise, including without limitation warranties of title, merchantability, fitness for a particular purpose, non infringement, or the absence of latent or other defects, accuracy, or the presence or absence of errors, whether or not discoverable, all to the greatest extent permissible under applicable law.

2.2.2 Intent: This is to alleviate any concern that content (the data or database) in an SPDX file is subject to any form of intellectual property right that could restrict the re-use of the information or the creation of another SPDX file for the same project(s). This approach avoids intellectual property and related restrictions over the SPDX file, however individuals can still contract with each other to restrict release of specific collections of SPDX files (which map to software bill of materials) and the identification of the supplier of SPDX files.

2.2.3 Cardinality: Mandatory, one.

2.2.4 Data Format: "CC0-1.0"

2.2.5 Tag: "DataLicense:"

Example:

DataLicense: CC0-1.0

2.2.6 RDF: spdx:dataLicense

Example:

```
<SpdxDocument rdf:about="">
  <dataLicense rdf:resource="http://spdx.org/licenses/CC0-1.0" />
</SpdxDocument>
```

2.3 SPDX Identifier

2.3.1 Purpose: Identify the current SPDX document which may be referenced in relationships by other files, packages internally and documents externally. To reference another SPDX document in total, this identifier should be used with the external document identifier preceding it. See the "Relationships between SPDX Elements" section for examples .

2.3.2 Intent: Provide a way for the document to refer to itself in relationship to other elements.

2.3.3 Cardinality: Mandatory, one.

2.3.4 DataFormat: "SPDXRef-DOCUMENT"

2.3.5 Tag: "SPDXID:"

Example:

SPDXID: SPDXRef-DOCUMENT

- 2.3.6 RDF:** The URI for the document is the document namespace appended by “#SPDXRef-DOCUMENT”

```
<spdx:SpdxDocument
rdf:about="http://spdx.org/spdxdocs/spdx-example-444504E0-4F89-41D3-9A0C-0305E82C33
01#SPDXRef-DOCUMENT">
...
</spdx:SpdxDocument>
```

2.4 Document Name

- 2.4.1 Purpose:** Identify name of this document as designated by creator.
- 2.4.2 Intent:** Here, the name of each document is an important convention and easier to refer to than the URI.
- 2.4.3 Cardinality:** Mandatory, one.
- 2.4.4 DataFormat:** single line of text.
- 2.4.5 Tag:** “DocumentName:”
- 2.4.6 RDF:** property spdx:name in class Document

Example:

```
DocumentName: glibc-v2.3
DocumentName: ubuntu-14.04
```

Example:

```
<SpdxDocument rdf:about="">
  <name>glibc-v2.3</name>
</SpdxDocument>

<SpdxDocument rdf:about="">
  <name>ubuntu-14.04</name>
</SpdxDocument>
```

2.5 SPDX Document Namespace

- 2.5.1 Purpose:** This field provided an SPDX document specific namespace as a unique absolute [Uniform Resource Identifier](#) (URI) as specified in [RFC 2396](#), with the exception of the ‘#’ delimiter. The URI must be unique for the SPDX document including the specific version of the SPDX document. If the SPDX document is updated, thereby creating a new version, a new URI for the updated document must be used. There can only be one URI for an SPDX document and only one SPDX document for a given URI. The SPDX Document URI cannot contain a URI “part” (e.g. the “#” character), since the “#” is used to uniquely identify SPDX element identifiers.

2.5.2 Intent: The URI provides an unambiguous mechanism for other SPDX documents to reference SPDX elements within this SPDX document. See section 3.5 for a description on how external documents are referenced. Although it is not required, the URI can be constructed in a way which provides information on how the SPDX document can be found. For example, the URI can be a URL referencing the SPDX document itself if it is available on the internet. A best practice for creating the URI for SPDX documents available on the public internet is `http://[CreatorWebsite]/[pathToSpdx]/[DocumentName]-[UUID]` where:

- CreatorWebsite is a website hosted by the creator of the document. (e.g. an SPDX document provided by SPDX would be `spdx.org`)
- PathToSpdx is a path to where SPDX documents are stored on the website (e.g. `/spdx/spdxdocs`)
- DocumentName is a name given to the SPDX Document itself, typically the (set of) package name(s) followed by the version. (see section 2.4).
- UUID is a [universally unique identifier](#). The UUID could be a version 4 random UUID which can be generated from the [Online UUID Generator](#) or a version 5 UUID generated from a sha1 checksum known to be unique for this specific SPDX document version.
- If the creator does not own their own website, a default SPDX CreatorWebsite and PathToSpdx can be used 'spdx.org/spdxdocs'. Note that the SPDX documents are not currently stored or accessible on this website. The URI is only used to create a unique ID following the above conventions.

Note that the URI does not have to be accessible. It is only intended to provide a unique ID. In many cases, the URI will point to a web accessible document, but this should not be assumed to be the case.

2.5.3 Cardinality: Mandatory, one.

2.5.4 Data Format: unique absolute [Uniform Resource Identifier](#) (URI) as specified in [RFC 2396](#), with the exception of the '#' character.

The URI must be unique for the SPDX document including the specific version of the SPDX document. If the SPDX document is updated, thereby creating a new version, a new URI for the updated document must be used. There can only be one URI for an SPDX document and only one SPDX document for a given URI.

The SPDX Document URI cannot contain a URI "part" (e.g. the "#" delimiter), since the "#" is used to uniquely identify SPDX element identifiers.

2.5.5 Tag: "DocumentNamespace:"

Example:

DocumentNamespace:<http://spdx.org/spdxdocs/spdx-tools-v1.2-3F2504E0-4F89-41D3-9A0C-0305E82C3301>

2.5.6 RDF: The unique ID is the URI for the SPDX document

Example:

```
<SpdxDocument
rdf:about="http://spdx.org/spdxdocs/spdx-tools-v1.2-3F2504E0-4F89-41D3-9A0C-0305E82C3301">
  <rdfs:comment>
    This document was created using SPDX 2.0 using licenses from the web site.
  </rdfs:comment>
</SpdxDocument>
```

2.6 External Document References

2.6.1 Purpose: Identify any external SPDX documents referenced within this SPDX document.

2.6.2 Intent: SPDX elements may be related to other SPDX elements. These elements may be in external SPDX documents. An SPDX element could be a file, package, or SPDX document.

2.6.3 Cardinality: Optional, one or many.

2.6.4 Data Format: DocumentRef-[idString] [SPDX Document URI] [Checksum]

where

[idString] is a unique string containing letters, numbers, ".", "-", or "+".

[SPDX Document Namespace] is the unique ID for the external document as defined in section 2.5 of that referenced document,

[Checksum] is a checksum of the external document following the checksum format defined in section 3.9.

2.6.5 Tag: "ExternalDocumentRef:"

Example:

ExternalDocumentRef: DocumentRef-spdx-tool-1.2

<http://spdx.org/spdxdocs/spdx-tools-v1.2-3F2504E0-4F89-41D3-9A0C-0305E82C3301> SHA1:
d6a770ba38583ed4bb4525bd96e50461655d2759

2.6.6 RDF: property `spdx:externalDocumentRef` in class `spdx:Document` range `ExternalDocumentRef`. The `ExternalDocumentRef` contains two properties:

- `spdxDocument` - the `SpdxDocument` being referenced
- `checksum` - the checksum of the referenced SPDX document

Example:

```
<externalDocumentRef>
  <ExternalDocumentRef>
```

```
<spdx:externalDocumentId>DocumentRef-spdx-tool-1.2</spdx:externalDocumentId>
```

```
<spdxDocument
```

```
  rdf:about="http://spdx.org/spdxdocs/spdx-tools-v1.2-3F2504E0-4F89-41D3-9A0C-0305E82C3301"
/>
```

```
  <checksum>
```

```
    <Checksum>
```

```
      <algorithm rdf:resource="checksumAlgorithm_sha1"/>
```

```
      <checksumValue>d6a770ba38583ed4bb4525bd96e50461655d2758
```

```
    </checksumValue>
```

```
  </Checksum>
```

```

    </checksum>
  </ExternalDocumentRef>
</externalDocumentRef>

```

Notes: in RDF, a namespace can be created for the external document reference if a short form name for the external reference is desired.

2.7 License List Version

2.7.1 Purpose: An optional field for creators of the SPDX file to provide the version of the SPDX License List used when the SPDX file was created.

2.7.2 Intent: Recognizing that licenses are added to the SPDX License List over time, and with each subsequent version, the intent is to provide recipients of the SPDX file with the version of the License List used. This anticipates that in the future, an SPDX file may have used a version of the License List that is older than the then current one.

2.7.3 Cardinality: Optional, one.

2.7.4 Data Format: “M.N”
 where:
 M is major version number
 N is minor version number.

2.7.5 Tag: “LicenseListVersion:”

Example:
 LicenseListVersion: 1.19

2.7.6 RDF: property licenseListVersion in class spdx:CreationInfo

Example:
 <CreationInfo>
 <licenseListVersion>2.0</licenseListVersion>
 </CreationInfo>

2.8 Creator

2.8.1 Purpose: Identify who (or what, in the case of a tool) created the SPDX file. If the SPDX file was created by an individual, indicate the person's name. If the SPDX file was created on behalf of a company or organization, indicate the entity name. If the SPDX file was created using a software tool, indicate the name and version for that tool. If multiple participants or tools were involved, use multiple instances of this field. Person name or organization name may be designated as “anonymous” if appropriate.

2.8.2 Intent: Here, the generation method will assist the recipient of the SPDX file in assessing the general reliability/accuracy of the analysis information.

2.8.3 Cardinality: Mandatory, one or many.

2.8.4 Data Format: single line of text with the following keywords:

"Person: person name" and optional "(email)"
 "Organization: organization" and optional "(email)"
 "Tool: toolidentifier-version"

2.8.5 Tag: "Creator:"

Example:

Creator: Person: Jane Doe ()

Creator: Organization: ExampleCodeInspect ()

Creator: Tool: LicenseFind-1.0

2.8.6 RDF: property spdx:creator in class spdx:CreationInfo

Example:

<CreationInfo>

<creator> Person: Jane Doe () </creator>

<creator> Organization: ExampleCodeInspect () </creator>

<creator> Tool: LicenseFind-1.0 </creator>

</CreationInfo>

2.9 Created

2.9.1 Purpose: Identify when the SPDX file was originally created. The date is to be specified according to combined date and time in UTC format as specified in ISO 8601 standard. This field is distinct from the fields in section 7, which involves the addition of information during a subsequent review.

2.9.2 Intent: Here, the time stamp can serve as a verification as to whether the analysis needs to be updated.

2.9.3 Cardinality: Mandatory, one.

2.9.4 Data Format: YYYY-MM-DDThh:mm:ssZ

where:

YYYY is year

MM is month with leading zero

DD is day with leading zero

T is delimiter for time

hh is hours with leading zero in 24 hour time

mm is minutes with leading zero

ss is seconds with leading zero

Z is universal time indicator

2.9.5 Tag: "Created:"

Example:

Created: 2010-01-29T18:30:22Z

2.9.6 RDF: property spdx:created in class spdx:CreationInfo

Example:
 <CreationInfo>
 <created> 2010-01-29T18:30:22Z </created>
 </CreationInfo>

2.10 Creator Comment

2.10.1 Purpose: An optional field for creators of the SPDX file to provide general comments about the creation of the SPDX file or any other relevant comment not included in the other fields.

2.10.2 Intent: Here, the intent is to provide recipients of the SPDX file with comments by the creator of the SPDX file.

2.10.3 Cardinality: Optional, one.

2.10.4 Data Format: free form text that can span multiple lines. In tag format this is delimited by <text> .. </text>, in RDF, it is delimited by <rdfs:comment>.

2.10.5 Tag: "CreatorComment:"

Example:
 CreatorComment: <text>
 This package has been shipped in source and binary form.
 The binaries were created with gcc 4.5.1 and expect to link to
 compatible system run time libraries.
 </text>

2.10.6 RDF: property rdfs:comment in class spdx:CreationInfo

Example:
 <CreationInfo>
 <rdfs:comment> This package has been shipped in source and binary form.
 The binaries were created with gcc 4.5.1 and expect to link to
 compatible system run time libraries. </rdfs:comment>
 </CreationInfo>

2.11 Document Comment

2.11.1 Purpose: An optional field for creators of the SPDX file content to provide comments to the consumers of the SPDX document.

2.11.2 Intent: Here, the intent is to provide readers/reviewers with comments by the creator of the SPDX file about the SPDX document.

2.11.3 Cardinality: Optional, one.

2.11.4 Data Format: free form text that can span multiple lines. In tag format this is delimited by

<text> .. </text>, in RDF, it is delimited by <rdfs:comment>.

2.11.5 Tag: “DocumentComment:”

Example:

DocumentComment: <text>

This document was created using SPDX 2.0 using licenses from the web site.

</text>

2.11.6 RDF: property rdfs:comment in class SpdxDocument

Example:

<SpdxDocument rdf:about="">

 <rdfs:comment>

 This document was created using SPDX 2.0 using licenses from the web site.

 </rdfs:comment>

</SpdxDocument>

3 Package Information

One instance of the Package Information is required per package being analyzed. In SPDX 2.0, it is not necessary to have a package wrapping a set of files. A package can contain sub-packages, but the information in this section is a reference to the entire contents of the package listed.

In tag:value format, the order in which package and files occur is syntactically significant.

- A new package information section is denoted by the Package Name field.
- All Package Information fields must be grouped together before the start of a Files section.
- All files contained in a package must immediately follow the package.
- A new package information section (via Package Name) denotes the start of an other package.
- Sub-packages should not be nested inside a package section, but should be separate and should use a Relationship to clarify.
- Annotations on the package and Relationships from the package may appear after the package information before any file information.

Fields:

3.1 Package Name

3.1.1 Purpose: Identify the full name of the package as given by Package Originator.

3.1.2 Intent: Here, the name of each package is an important conventional technical identifier to be maintained for each package.

3.1.3 Cardinality: Mandatory, one.

3.1.4 DataFormat: single line of text.

3.1.5 Tag: "PackageName:"

Example:

PackageName: glibc

3.1.6 RDF: property spdx:name in class spdx:Package

Example:

```
<Package rdf:about="">
  <name>glibc</name>
</Package>
```

3.2 Package SPDX Identifier

3.2.1 Purpose: Uniquely identify any element in an SPDX document which may be referenced by other elements. These may be referenced internally and externally with the addition of the SPDX Document Identifier.

3.2.2 Intent: There may be several versions of the same package within an SPDX document.

Each element needs to be able to be referred to uniquely so that relationships between elements can be clearly articulated.

3.2.3 Cardinality: Mandatory, one.

3.2.4 DataFormat: “SPDXRef-[idString]”

where **[idString]** is a unique string containing letters, numbers, “.”, “-”.

3.2.5 Tag: “SPDXID:”

Example:

SPDXID: SPDXRef-1

3.2.6 RDF: The URI for the element will follow the form:

[SpxDocumentURI]#SPDXRef-[idString] where [SpxDocumentURI] is the URI for the SPDX Document containing the element.

Example using xml:base:

<rdf:RDF xml:base=

["http://acme.com/spdxdocs/acmeproj/v1.2/1BE2A4FF-5F1A-48D3-8483-28A9B0349A1B"](http://acme.com/spdxdocs/acmeproj/v1.2/1BE2A4FF-5F1A-48D3-8483-28A9B0349A1B)>

...

<Package rdf:ID="SPDXRef-1">

...

</Package>

Example using document URI:

<Package rdf:about=

["http://acme.com/spdxdocs/acmeproj/v1.2/1BE2A4FF-5F1A-48D3-8483-28A9B0349A1B#SPDXRef-1"](http://acme.com/spdxdocs/acmeproj/v1.2/1BE2A4FF-5F1A-48D3-8483-28A9B0349A1B#SPDXRef-1)>

...

</Package>

3.3 Package Version

3.3.1 Purpose: Identify the version of the package.

3.3.2 Intent: The versioning of a package is a useful for identification purposes and for indicating later changes for the package version.

3.3.3 Cardinality: Optional, one.

3.3.4 DataFormat: single line of text.

3.3.5 Tag: “PackageVersion:”

Example:

```
PackageVersion: 2.11.1
```

3.3.6 RDF: property spdx:versionInfo **in class** spdx:Package**Example:**

```
<Package rdf:about="">
  <versionInfo>2.11.1</versionInfo>
</Package>
```

3.4 Package File Name

3.4.1 Purpose: Provide the actual file name of the package. This may include the packaging and compression methods used as part of the file name.

3.4.2 Intent: Here, the actual file name of the compressed file containing the package is a significant technical element that needs to be included with each package identification information.

3.4.3 Cardinality: Optional, one.

3.4.4 Data Format: single line of text.

3.4.5 Tag: "PackageFileName:"

Example:

```
PackageFileName: glibc-2.11.1.tar.gz
```

3.4.6 RDF: property spdx:packageFileName **in class** spdx:Package**Example:**

```
<Package rdf:about="">
  <packageFileName>glibc 2.11.1.tar.gz</packageFileName>
</Package>
```

3.5 Package Supplier

3.5.1 Purpose: Identify the actual distribution source for the package identified in the SPDX file. This may or may not be different from the originating distribution source for the package. The name of the Package Supplier must be an organization or recognized author and not a web site. For example, Sourceforge is a host website, not a supplier, the supplier for <http://sourceforge.net/projects/bridge/> is "The Linux Foundation." NOASSERTION should be used if:

- (i) the SPDX file creator has attempted to but cannot reach a reasonable objective determination of who the supplier is;
- (ii) the project is orphaned and was obtained from a public website; or
- (iii) the SPDX file creator has intentionally provided no information (no meaning should be implied by doing so).

3.5.2 Intent: This field assists with understanding the point of distribution for the code in the package. This field is vital for ensuring that downstream package recipients can address any

ambiguity or concerns that might arise with the information in the SPDX file or the contents of the package it documents.

3.5.3 Cardinality: Optional, one.

3.5.4 Data Format: single line of text with the following keywords | “NOASSERTION”

"Person:" person name and optional "("email")"

"Organization:" organization name and optional "("email")"

3.5.5 Tag: “PackageSupplier:”

Example:

PackageSupplier: Person: Jane Doe (jane.doe@example.com)

3.5.6 RDF: property spdx:supplier in class spdx:Package

Example:

<Package rdf:about="">

<supplier>Person: Jane Doe (jane.doe@example.com) </supplier>

</Package>

3.6 Package Originator

3.6.1 Purpose: If the package identified in the SPDX file originated from a different person or organization than identified as Package Supplier (see section 4.4 above), this field identifies from where or whom the package originally came. In some cases a package may be created and originally distributed by a different third party than the Package Supplier of the package. For example, the SPDX file identifies the package glibc and Red Hat as the Package Supplier, but Free Software Foundation is the Package Originator. NOASSERTION should be used if:

- (i) the SPDX file creator has attempted to but cannot reach a reasonable objective determination of who the originator is;
- (ii) the project is orphaned and was obtained from a public website; or
- (iii) the SPDX file creator has intentionally provided no information (no meaning should be implied by doing so).

3.6.2 Intent: This field assists with understanding the point of origin of the code in the package. This field is vital for understanding who originally distributed a package and should help in addressing any ambiguity or concerns that might arise with the information in the SPDX file or the contents of the Package it documents.

3.6.3 Cardinality: Optional, one.

3.6.4 Data Format: single line of text with the following keywords | “NOASSERTION”

"Person:" person name and optional "("email")"

"Organization:" organization name and optional "("email")"

3.6.5 Tag: “PackageOriginator:”

Example:

PackageOriginator: Organization: ExampleCodeInspect (contact@example.com)

3.6.6 RDF: property spdx:originator in class spdx:Package

Example:

```
<Package rdf:about="">
  <originator>Organization: ExampleCodeInspect (contact@example.com)
</originator>
</Package>
```

3.7 Package Download Location

3.7.1 Purpose: This section identifies the download Universal Resource Locator (URL), or a specific location within a version control system (VCS) for the package at the time that the SPDX file was created. If there is no public (or internal) URL, then it is explicitly marked as NONE. If there is insufficient knowledge about whether a public or internal download mechanism exists or not, then NOASSERTION should be used.

3.7.2 Intent: Here, where and how to download the exact package being referenced is critical verification and tracking data.

3.7.3 Cardinality: Mandatory, one.

3.7.4 Data Format: uniform resource locator | VCS location | “NONE” | “NOASSERTION”

For version-controlled files, the VCS location syntax is similar to a URL and has the:

```
<vcs_tool>+<transport>://<host_name>[/<path_to_repository>][@<revision_tag_or_branch>][#<sub_path>]
```

This VCS location compact notation (inspired and mostly adopted from https://pip.pypa.io/en/latest/reference/pip_install.html#vcs-support as of 20150220) supports referencing locations in version control systems such as Git, Mercurial, Subversion and Bazaar, and specifies the type of VCS tool using url prefixes: “git+”, “hg+”, “bzt+”, “svn+” and specific transport schemes such as SSH or HTTPS.

Specifying sub-paths, branch names, a commit hash, a revision or a tag name is recommended, and supported using the “@” delimiter for commits and the “#” delimiter for sub-paths.

Using user names and password in the host_name is not supported and should be considered as an error. User access control to URLs or VCS repositories must be handled outside of an SPDX document.

In VCS location compact notations, the trailing slashes in <host_name>, <path_to_repository> are not significant. Leading and trailing slashes in <sub_path> are not significant.

3.7.5 Tag: “PackageDownloadLocation:”

Example for a plain URL:

PackageDownloadLocation: <http://ftp.gnu.org/gnu/glibc/glibc-ports-2.15.tar.gz>

Example for Git:

SPDX supports git, git+git, git+https git+http and git+ssh transports. git and git+git are equivalent.

Here are the supported forms:

```
PackageDownloadLocation: git://git.myproject.org/MyProject
PackageDownloadLocation: git+https://git.myproject.org/MyProject.git
PackageDownloadLocation: git+http://git.myproject.org/MyProject
PackageDownloadLocation: git+ssh://git.myproject.org/MyProject.git
PackageDownloadLocation: git+git://git.myproject.org/MyProject
PackageDownloadLocation: git+git@git.myproject.org:MyProject
```

To specify a sub-path to a file or directory inside a repository use the "#" delimiter:

```
PackageDownloadLocation: git://git.myproject.org/MyProject#src/somefile.c
PackageDownloadLocation: git+https://git.myproject.org/MyProject#src/Class.java
```

To specify branch names, a commit hash or a tag name, use the "@" delimiter:

```
PackageDownloadLocation: git://git.myproject.org/MyProject.git@master
PackageDownloadLocation: git+https://git.myproject.org/MyProject.git@v1.0
PackageDownloadLocation:
git://git.myproject.org/MyProject.git@da39a3ee5e6b4b0d3255bfef95601890afd80709
```

Sub-paths and branch names or commit hash can be combined too:

```
PackageDownloadLocation: git+https://git.myproject.org/MyProject.git@master#src/MyClass.cpp
PackageDownloadLocation:
git+https://git.myproject.org/MyProject@da39a3ee5e6b4b0d3255bfef95601890afd80709#lib/variable.rb
```

Example for Mercurial:

SPDX supported schemes are: hg+http, hg+https, hg+static-http and hg+ssh.

The supported forms are:

```
PackageDownloadLocation: hg+http://hg.myproject.org/MyProject
PackageDownloadLocation: hg+https://hg.myproject.org/MyProject
PackageDownloadLocation: hg+ssh://hg.myproject.org/MyProject
```

To specify a sub-path to a file or directory inside a repository use the "#" delimiter:

```
PackageDownloadLocation: hg+https://hg.myproject.org/MyProject#src/somefile.c
PackageDownloadLocation: hg+https://hg.myproject.org/MyProject#src/Class.java
```

To pass branch names, a commit hash, a tag name or a local branch name use the "@" delimiter:

```
PackageDownloadLocation: hg+https://hg.myproject.org/MyProject@da39a3ee5e6b
PackageDownloadLocation: hg+https://hg.myproject.org/MyProject@2019
PackageDownloadLocation: hg+https://hg.myproject.org/MyProject@v1.0
PackageDownloadLocation: hg+https://hg.myproject.org/MyProject@special_feature
```

Sub-paths and branch names or commit hash can be combined too:

```
PackageDownloadLocation: hg+https://hg.myproject.org/MyProject@master#src/MyClass.cpp
PackageDownloadLocation: hg+https://hg.myproject.org/MyProject@da39a3ee5e6b#lib/variable.rb
```

Example for Subversion:

SPDX supports the URL schemes svn, svn+svn, svn+http, svn+https, svn+ssh. svn and svn+svn are equivalent.

The supported forms are:

```
PackageDownloadLocation: svn://svn.myproject.org/svn/MyProject
PackageDownloadLocation: svn+svn://svn.myproject.org/svn/MyProject
```


PackageDownloadLocation: svn+http://svn.myproject.org/svn/MyProject/trunk
 PackageDownloadLocation: svn+https://svn.myproject.org/svn/MyProject/trunk

To specify a sub-path to a file or directory inside a repository use the "#" delimiter:

PackageDownloadLocation: svn+https://svn.myproject.org/MyProject#src/somefile.c
 PackageDownloadLocation: svn+https://svn.myproject.org/MyProject#src/Class.java

This support is less important for SVN since the URL path can also contain sub-paths; this two forms are equivalent:

PackageDownloadLocation: svn+https://svn.myproject.org/MyProject/trunk#src/somefile.c
 PackageDownloadLocation: svn+https://svn.myproject.org/MyProject/trunk/src/somefile.c

You can specify a revision using the "@" delimiter:

PackageDownloadLocation: svn+https://svn.myproject.org/svn/MyProject/trunk@2019

Sub-paths and revisions can be combined too:

PackageDownloadLocation: svn+https://svn.myproject.org/MyProject@123#src/MyClass.cpp
 PackageDownloadLocation: svn+https://svn.myproject.org/MyProject/trunk@1234#lib/variable/variable.rb

Example for Bazaar:

SPDX supports Bazaar using the bzd+http, bzd+https, bzd+ssh, bzd+sftp, bzd+ftp and bzd+lp schemes.

The supported forms are:

PackageDownloadLocation: bzd+https://bzd.myproject.org/MyProject/trunk
 PackageDownloadLocation: bzd+http://bzd.myproject.org/MyProject/trunk
 PackageDownloadLocation: bzd+sftp://myproject.org/MyProject/trunk
 PackageDownloadLocation: bzd+ssh://myproject.org/MyProject/trunk
 PackageDownloadLocation: bzd+ftp://myproject.org/MyProject/trunk
 PackageDownloadLocation: bzd+lp:MyProject

To specify a sub-path to a file or directory inside a repository use the "#" delimiter:

PackageDownloadLocation: bzd+https://bzd.myproject.org/MyProject/trunk#src/somefile.c
 PackageDownloadLocation: bzd+https://bzd.myproject.org/MyProject/trunk#src/Class.java

You can specify a revision or tag using the "@" delimiter:

PackageDownloadLocation: bzd+https://bzd.myproject.org/MyProject/trunk@2019
 PackageDownloadLocation: bzd+http://bzd.myproject.org/MyProject/trunk@v1.0

Sub-paths and revisions can be combined too:

PackageDownloadLocation: bzd+https://bzd.myproject.org/MyProject/trunk@2019#src/somefile.c

3.7.6 RDF: property spdx:downloadLocation in class spdx:Package

Example:

```
<Package rdf:about="">
  <downloadLocation>
    http://ftp.gnu.org/gnu/glibc/glibc-ports-2.15.tar.gz
  </downloadLocation>
</Package>
```

```
<Package rdf:about="">
  <downloadLocation>
```

```
git+git://git.myproject.org/MyProject.git@da39a3ee5e6b4b0d3255bfef95601890afd80709#egg=MyProject
```

```

    </downloadLocation>
  </Package>

```

3.8 Package Verification Code

3.8.1 Purpose: This field provides an independently reproducible mechanism identifying specific contents of a package based on the actual files (except the SPDX file itself, if it is included in the package) that make up each package and that correlates to the data in this SPDX file. This identifier enables a recipient to determine if any file in the original package (that the analysis was done on) has been changed and permits inclusion of an SPDX file as part of a package.

3.8.2 Intent: Providing a unique identifier based on the files inside each package, eliminates confusion over which version or modification of a specific package the SPDX file refers to. It also permits one to embed the SPDX file within the package without altering the identifier.

3.8.3 Cardinality: Mandatory, one.

3.8.4 Algorithm:

```

verificationcode = 0
filelist = templist = ""
for all files in the package {
  if file is an "excludes" file, skip it /* exclude SPDX analysis file(s) */
  append templist with "SHA1(file)/n"
}
sort templist in ascending order by SHA1 value
filelist = templist with "/n"s removed. /* ordered sequence of SHA1 values with no separators */
verificationcode = SHA1(filelist)

```

Where SHA1(file) applies a SHA1 algorithm on the contents of file and returns the result in lowercase hexadecimal digits.

Required sort order: '0','1','2','3','4','5','6','7','8','9','a','b','c','d','e','f' (ASCII order)

3.8.5 Data Format: single line of text with 160 bit binary represented as 40 lowercase hexadecimal digits

3.8.6 Tag: "PackageVerificationCode:" (and optionally "(excludes: FileName)") where FileName is as specified in 6.1.

Example:

```
PackageVerificationCode: d6a770ba38583ed4bb4525bd96e50461655d2758 (excludes: ./package.spdx)
```

4.8.7 RDF: spdx:packageVerificationCodeValue, spdx:packageVerificationCodeExcludedFile in class spdx:PackageVerificationCode

Example:

```

<Package rdf:about="">
  <packageVerificationCode>
    <PackageVerificationCode>
      <packageVerificationCodeValue>d6a770ba38583ed4bb4525bd96e50461655d2758
    </packageVerificationCodeValue>
    <packageVerificationCodeExcludedFile> ./package.spdx
  </packageVerificationCodeExcludedFile>
</PackageVerificationCode>

```

```

</packageVerificationCode>
</Package>

```

3.9 Package Checksum

3.9.1 Purpose: This field provides an independently reproducible mechanism that permits unique identification of a specific package that correlates to the data in this SPDX file. This identifier enables a recipient to determine if any file in the original package has been changed. If the SPDX file is to be included in a package, this value should not be calculated. The SHA-1 algorithm will be used to provide the checksum by default.

3.9.2 Intent: Here, by providing a unique identifier of the package, confusion over which version or modification of a specific package the SPDX file references should be eliminated.

3.9.3 Cardinality: Optional, one or many.

3.9.4 Algorithms that can be used: SHA1, SHA256, MD5

3.9.5 Data Format: There are three components, an algorithm identifier (“SHA1”), a colon separator “:” and a bit value represented as lowercase hexadecimal digits (appropriate as output to the algorithm).

3.9.6 Tag: “PackageChecksum:”

Example:

```

PackageChecksum: SHA1: 85ed0817af83a24ad8da68c2b5094de69833983c
PackageChecksum: SHA256:
11b6d3ee554eedf79299905a98f9b9a04e498210b59f15094c916c91d150efcd
PackageChecksum: MD5: 624c1abb3664f4b35547e7c73864ad24

```

3.9.7 RDF: properties spdx:algorithm, spdx:checksumValue in **class** spdx:checksum

Example:

```

<Package rdf:about="">
  <checksum>
    <Checksum>
      <algorithm rdf:resource="http://spdx.org/rdf/terms#checksumAlgorithm_sha1"/>
      <checksumValue> 85ed0817af83a24ad8da68c2b5094de69833983c
    </checksumValue>
    </Checksum>
    <Checksum>
      <algorithm rdf:resource="http://spdx.org/rdf/terms#checksumAlgorithm_sha256"/>
      <checksumValue>
        11b6d3ee554eedf79299905a98f9b9a04e498210b59f15094c916c91d150efcd
      </checksumValue>
    </Checksum>
    <Checksum>
      <algorithm rdf:resource="http://spdx.org/rdf/terms#checksumAlgorithm_md5"/>
      <checksumValue> 624c1abb3664f4b35547e7c73864ad24
    </Checksum>
  </checksum>
</Package>

```

```

        </checksumValue>
    </Checksum>

    </checksum>
</Package>

```

3.10 Package Home Page

3.10.1 Purpose: This field provides a place for the SPDX file creator to record a web site that serves as the package's home page. This link can also be used to reference further information about the package referenced by the SPDX file creator.

3.10.2 Intent: Here, by providing a link to the package's home page, the SPDX file creator can provide additional information useful for analysis. This saves the recipient of the SPDX file who is looking for more info from having to search for and verify a match between the package and the associated project homepage.

3.10.3 Cardinality: Optional, one.

3.10.4 Data Format: uniform resource locator | "NONE" | "NOASSERTION"

3.10.5 Tag: "PackageHomePage:"

Example:

PackageHomePage: <http://ftp.gnu.org/gnu/glibc>

3.10.6 RDF: property doap:homepage in **class** spdx:Package

Example:

```

<Package rdf:about="">
    <doap:homepage >http://ftp.gnu.org/gnu/glibc/""</doap:homepage>
</Package>

```

3.11 Source Information

3.11.1 Purpose: This field provides a place for the SPDX file creator to record any relevant background information or additional comments about the origin of the package. For example, this field might include comments indicating whether the package was pulled from a source code management system or has been repackaged.

3.11.2 Intent: Here, by providing a comment field, the SPDX file creator can provide additional information to describe any anomalies or discoveries in the determination of the origin of the package.

3.11.3 Cardinality: Optional, one.

3.11.4 Data Format: free form text that can span multiple lines.
In tag format this is delimited by <text> .. </text>.

3.11.5 Tag: "PackageSourceInfo:"**Example:**

```
PackageSourceInfo: <text>
uses glibc-2_11-branch from git://sourceware.org/git/glibc.git. </text>
```

3.11.6 RDF: spdx:sourceInfo**Example:**

```
<Package rdf:about="">
  <sourceInfo>uses glibc-2_11-branch from git://sourceware.org/git/glibc.git.
</sourceInfo>
</Package>
```

3.12 Concluded License

3.12.1 Purpose: This field contains the license the SPDX file creator has concluded as governing the package or alternative values, if the governing license cannot be determined. The options to populate this field are limited to:

- (a) the SPDX License List short form identifier, if the concluded license is on the SPDX License List;
- (b) a reference to the license text denoted by the LicenseRef-[idString], if the concluded license is not on the SPDX License List;
- (c) NOASSERTION should be used if:
 - (i) the SPDX file creator has attempted to but cannot reach a reasonable objective determination of the Concluded License;
 - (ii) the SPDX file creator is uncomfortable concluding a license, despite some license information being available;
 - (iii) the SPDX file creator has made no attempt to determine a Concluded License;
 - (iv) the SPDX file creator has intentionally provided no information (no meaning should be implied by doing so); or
 - (v) there is no licensing information from which to conclude a license for the package.

With respect to (a) and (b) above, if there is more than one concluded license, all should be included. If the package recipient has a choice of multiple licenses, then each of the choices should be recited as a "disjunctive" license. If the Concluded License is not the same as the Declared License, a written explanation should be provided in the Comments on License field (section 3.15). With respect to (c), a written explanation in the Comments on License field (section 3.15) is preferred.

3.12.2 Intent: Here, the intent is for the SPDX file creator to analyze the license information in package, and other objective information, e.g., COPYING file, together with the results from any scanning tools, to arrive at a reasonably objective conclusion as to what license governs the package.

3.12.3 Cardinality: Mandatory, one.

3.12.4 Data Format: <short form identifier in Appendix I> | <license set>
 [ExternalDocumentRef:]"LicenseRef"-[idString] |
 | "NONE" | "NOASSERTION"

where:

[ExternalDocumentRef:] is an optional reference to an external SPDX document as described in section 2.7

[idString] is a unique string containing letters, numbers, “.”, “-” or “+”
 <license set> to be used when there is a choice between licenses
 (“disjunctive license”), they must be separated with “OR” and enclosed
 in parentheses. When multiple licenses apply (“conjunctive license”), they
 must be separated with an “AND” and enclosed in parentheses.
 See Appendix IV for License Expression syntax.

3.12.5 Tag: “PackageLicenseConcluded:”

Example:

PackageLicenseConcluded: LGPL-2.0

Example:

PackageLicenseConcluded: (LGPL-2.0 OR LicenseRef-3)

3.12.6 RDF: property spdx:licenseConcluded in class spdx:Package

Example:

```
<Package rdf:about="">
  <licenseConcluded rdf:resource="http://spdx.org/licenses/LGPL-2.0" />
</Package>
```

Example:

```
<Package rdf:about="">
  <licenseConcluded>
    <DisjunctiveLicenseSet>
      <member rdf:resource="http://spdx.org/licenses/LGPL-2.0" />
      <member rdf:resource="LicenseRef-3" />
    </DisjunctiveLicenseSet>
  </licenseConcluded>
</Package>
```

3.13 All Licenses Information from Files

3.13.1 Purpose: This field is to contain a list of all licenses found in the package. The relationship between licenses (i.e., conjunctive, disjunctive) is not specified in this field – it is simply a listing of all licenses found. The options to populate this list are limited to:

- (a) the SPDX License List short form identifier, if a detected license is on the SPDX License List;
- (b) a reference to the license, denoted by LicenseRef-[idString], if the detected license is not on the SPDX License List;
- (c) NONE, if no license information is detected in any of the files; or
- (d) NOASSERTION, if the SPDX file creator has not examined the contents of the actual files or if the SPDX file creator has intentionally provided no information (no meaning should be implied by doing so).

3.13.2 Intent: Here, the intention is to capture all license information detected in the actual files.

3.13.3 Cardinality: Mandatory, one or many.

3.13.4 Data Format: <short form identifier in Appendix I> |

```
[ExternalDocumentRef:]"LicenseRef"-[idString] |
| "NONE" | "NOASSERTION"
```

where:

[ExternalDocumentRef:] is an optional reference to an external SPDX document as described in section 2.7

[idString] is a unique string containing letters, numbers, ".", "-", or "+"

3.13.5 Tag: "PackageLicenseInfoFromFiles:"

Example:

PackageLicenseInfoFromFiles: GPL-2.0

PackageLicenseInfoFromFiles: LicenseRef-1

PackageLicenseInfoFromFiles: LicenseRef-2

3.13.6 RDF: property spdx:licenseInfoFromFiles in class spdx:Package

Example:

```
<Package rdf:about="">
```

```
<licenseInfoFromFiles rdf:resource=" http://spdx.org/licenses/GPL-2.0" />
```

```
<licenseInfoFromFiles rdf:resource="LicenseRef-1" />
```

```
<licenseInfoFromFiles rdf:resource="LicenseRef-2" />
```

```
</Package>
```

3.14 Declared License

3.14.1 Purpose: This field lists the licenses that have been declared by the authors of the package. Any license information that does not originate from the package authors, e.g. license information from a third party repository, should not be included in this field. The options to populate this field are limited to:

(a) the SPDX License List short form identifier, if the license is on the SPDX License List;

(b) a reference to the license, denoted by LicenseRef-[idString], if the declared license is not on the SPDX License List;

(c) NONE, if no license information is detected in any of the files; or

(d) NOASSERTION, if the SPDX file creator has not examined the contents of the package or if the SPDX file creator has intentionally provided no information (no meaning should be implied by doing so).

With respect to "a" and "b" above, if license information for more than one license is contained in the file, all should be reflected in this field. If the license information offers the package recipient a choice of licenses, then each of the choices should be recited as "disjunctive" licenses.

3.14.2 Intent: This is simply the license identified in text in one or more files (for example COPYING file) in the source code package. This field is not intended to capture license information obtained from an external source, such as the package website. Such information can be included in Concluded License (section 3.12). This field may have multiple declared licenses, if multiple licenses are declared at the package level.

3.14.3 Cardinality: Mandatory, one.

3.14.4 Data Format: <short form identifier in Appendix I> | <license set>
 [ExternalDocumentRef:]"LicenseRef"-[idString] |
 | "NONE" | "NOASSERTION"

where:

[ExternalDocumentRef:] is an optional reference to an external SPDX document as described in section 2.7

[idString] is a unique string containing letters, numbers, ".", "-" or "+"

<license set> to be used when there is a choice between licenses ("disjunctive license"), they must be separated with "OR" and enclosed in parentheses. When multiple licenses apply ("conjunctive license"), they must be separated with an "AND" and enclosed in parentheses. See Appendix IV for License Expression syntax.

3.14.5 Tag: "PackageLicenseDeclared:"

Example:

PackageLicenseDeclared: LGPL-2.0

Example:

PackageLicenseDeclared: (LGPL-2.0 AND LicenseRef-3)

3.14.6 RDF: property spdx:licenseDeclared in **class** spdx:Package

Example:

```
<Package rdf:about="">
  <licenseDeclared rdf:resource="http://spdx.org/licenses/LGPL-2.0" />
</Package>
```

Example:

```
<Package rdf:about="">
  <licenseDeclared>
    <ConjunctiveLicenseSet>
      <member rdf:resource="http://spdx.org/licenses/LGPL-2.0" />
      <member rdf:resource="LicenseRef-3" />
    </ConjunctiveLicenseSet>
  </licenseDeclared>
</Package>
```

3.15 Comments on License

3.15.1 Purpose: This field provides a place for the SPDX file creator to record any relevant background information or analysis that went in to arriving at the Concluded License for a package. If the Concluded License does not match the Declared License or License Information from Files, this should be explained by the SPDX file creator. Its is also preferable to include an explanation here when the Concluded License is NOASSERTION.

3.15.2 Intent: Here, the intent is to provide the recipient of the SPDX file with a detailed explanation of how the Concluded License was determined if it does not match the License Information from the files or the source code package, is marked NOASSERTION, or other helpful information relevant to determining the license of the package.

3.15.3 Cardinality: Optional, one.

3.15.4 Data Format: free form text that can span multiple lines.
In tag format this is delimited by `<text> .. </text>`,
in RDF, it is delimited by `<licenseComments>`.

3.15.5 Tag: "PackageLicenseComments:"

Example:

PackageLicenseComments: `<text>`

The license for this project changed with the release of version x.y. The version of the project included here post-dates the license change.

`</text>`

3.15.6 RDF: property `spdx:licenseComments` in **class** `spdx:Package`

Example:

`<Package rdf:about="">`

`<licenseComments>`

This package has been shipped in source and binary form.

The binaries were created with gcc 4.5.1 and expect to link to compatible system run time libraries.

`</licenseComments>`

`</Package>`

3.16 Copyright Text

3.16.1 Purpose: Identify the copyright holders of the package, as well as any dates present. This will be a free form text field extracted from the package information files. The options to populate this field are limited to:

- (a) any text related to a copyright notice, even if not complete;
- (b) NONE if the package contains no copyright information whatsoever; or
- (c) NOASSERTION, if the SPDX file creator has not examined the contents of the package or if the SPDX file creator has intentionally provided no information (no meaning should be implied by doing so).

3.16.2 Intent: Record any copyright notices for the package.

3.16.3 Cardinality: Mandatory, one.

3.16.4 Data Format: free form text that can span multiple lines | "NOASSERTION" | "NONE"

3.16.5 Tag: "PackageCopyrightText:"

In tag format multiple lines are delimited by `<text> .. </text>`.

Example:

PackageCopyrightText: `<text>`

Copyright 2008-2010 John Smith

```
</text>
```

3.16.6 **RDF: property** spdx:copyrightText in **class** spdx:Package

Example:

```
<Package rdf:about="">
  <copyrightText>
    Copyright 2008-2010 John Smith
  </copyrightText>
</Package>
```

3.17 Package Summary Description

3.17.1 Purpose: This field is a short description of the package.

3.17.2 Intent: Here, the intent is to allow the SPDX file creator to provide concise information about the function or use of the package without having to parse the source code of the actual package.

3.17.3 Cardinality: Optional, one.

3.17.4 Data Format: free form text that can span multiple lines.

3.17.5 Tag: “PackageSummary:”
In tag format multiple lines are delimited by <text> .. </text>.

Example:

```
PackageSummary: <text> GNU C library. </text>
```

3.17.6 **RDF: property** spdx:summary in **class** spdx:Package

Example:

```
<Package rdf:about="">
  <summary> GNU C library.</summary>
</Package>
```

3.18 Package Detailed Description

3.18.1 Purpose: This field is a more detailed description of the package. It may also be extracted from the packages itself.

3.18.2 Intent: Here, the intent is to provide recipients of the SPDX file with a detailed technical explanation of the functionality, anticipated use, and anticipated implementation of the package. This field may also include a description of improvements over prior versions of the package.

3.18.3 Cardinality: Optional, one.

3.18.4 Data Format: free form text than can span multiple lines.

- 3.18.5 Tag: “PackageDescription:”**
In tag format multiple lines are delimited by <text> .. </text>.

Example:

PackageDescription: <text>

The GNU C Library defines functions that are specified by the ISO C standard, as well as additional features specific to POSIX and other derivatives of the Unix operating system, and extensions specific to GNU systems.

</text>

- 3.18.6 RDF: property spdx:description in class spdx:Package**

Example:

<Package rdf:about="">

<description>

The GNU C Library defines functions that are specified by the ISO C standard, as well as additional features specific to POSIX and other derivatives of the Unix operating system, and extensions specific to GNU systems.

</description>

</Package>

3.19 Package Comment

- 3.19.1 Purpose:** This field provides a place for the SPDX file creator to record any general comments about the package being described.

- 3.19.2 Intent:** Here, the intent is to provide the recipient of the SPDX document with more information determined after careful analysis of a package.

- 3.19.3 Cardinality:** Optional, one.

- 3.19.4 Data Format:** free form text that can span multiple lines.

- 3.19.5 Tag: “PackageComment:”**
In tag format multiple lines are delimited by <text> .. </text>.

Example:

PackageComment: <text>

The concluded license was found in the COPYING.txt file in the xyz directory.

</text>

- 3.19.6 RDF: property rdfs:comment in class spdx:Package**

Example:

<Package rdf:about="">

<rdfs:comment>

The concluded license was found in the COPYING.txt file in the xyz directory.

</rdfs:comment>

</Package>

4 File Information

One instance of the File Information is required for each file in the software package. It provides important meta information about a given file including licenses and copyright. Each instance should include the following fields. In SPDX 2.0, it is not necessary to have a package wrapping a set of files.

When implementing tag:value format, the positioning of File elements is syntactically significant.

- The first field to start off the description of a File must be the File Name in tag:value format.
- Other file information is associated with the File Name that precede it.
- Files are assumed to be associated with the Package that immediately precedes it, if a package exists. Presence of a new package signals the end of the set of files associated with the original package, unless an explicit Relationship is used.
- If a Package contains files, the File Information section must follow its Package Information section.
- If a File is not part of any package, it must precede any Package section reference in the SPDX document.
- Annotations on the file and Relationships from the file may appear after the file information, before the next file or Package section.

4.1 File Name

4.1.1 Purpose: Identify the full path and filename that corresponds to the file information in this section.

4.1.2 Intent: To aid finding the correct file which corresponds to the file information.

4.1.3 Cardinality: Mandatory, one.

4.1.4 Data Format: A relative filename with the root of the package archive or directory. See <http://www.ietf.org/rfc/rfc3986.txt> for syntax.

4.1.5 Tag: "FileName:"

Example:

FileName: ./package/foo.c

4.1.6 RDF: property spdx:fileName in **class** spdx:File

Example:

```
<File rdf:about="e">
  <fileName>./package/foo.c</fileName>
</File>
```

4.2 File SPDX Identifier

4.2.1 Purpose: Uniquely identify any element in an SPDX document which may be referenced by other elements. These may be referenced internally and externally with the addition of the SPDX Document Identifier.

4.2.2 Intent: There may be several versions of the same file within an SPDX document. Each element needs to be able to be referred to uniquely so that relationships between elements can be clearly articulated.

4.2.3 Cardinality: Mandatory, one.

4.2.4 DataFormat: “SPDXRef-[idString]”

where **[idString]** is a unique string containing letters, numbers, “.”, “-”.

4.2.5 Tag: “SPDXID:”

Example:

SPDXID: SPDXRef-1

4.2.6 RDF: The URI for the element will follow the form: [SpxDocumentURI]#SPDXRef-[idString] where [SpxDocumentURI] is the URI for the SPDX Document containing the element.

Example using xml:base:

```
<rdf:RDF xml:base=
"http://acme.com/spdxdocs/acmeproj/v1.2/1BE2A4FF-5F1A-48D3-8483-28A9B0349A1B"
...
<File rdf:ID="SPDXRef-1">
...
</File>
```

Example using document URI:

```
<File rdf:about=
"http://acme.com/spdxdocs/acmeproj/v1.2/1BE2A4FF-5F1A-48D3-8483-28A9B0349A1B#SPDXRef-1">
...
</File>
```

4.3 File Type

4.3.1 Purpose: This field provides information about the type of file identified. File Type is intrinsic to the file, independent of how the file is being used. A file may have more than one file type assigned to it, however the options to populate this field are limited to:

- a) SOURCE if the file is human readable source code (.c, .html, etc.);
- b) BINARY if the file is a compiled object, target image or binary executable (.o, .a, etc.);
- c) ARCHIVE if the file represents an archive (.tar, .jar, etc.);
- d) APPLICATION if the file is associated with a specific application type (MIME type of application/*);
- e) AUDIO if the file is associated with an audio file (MIME type of audio/*, ie. .mp3);
- f) IMAGE if the file is associated with an picture image file (MIME type of image/*, ie. .jpg, .gif);
- g) TEXT if the file is human readable text file (MIME type of text/*);
- h) VIDEO if the file is associated with a video file type (MIME type of video/*);
- i) DOCUMENTATION if the file serves as documentation;
- j) SPDX if the file is an SPDX document;

- k) OTHER if the file doesn't fit into the above categories (generated artifacts, data files, etc.)

4.3.2 Intent: Here, this field is a reasonable estimation of the file type, from a developer perspective.

4.3.3 Cardinality: Optional, multiple.

4.3.4 Data Format: “SOURCE” | “BINARY” | “ARCHIVE” | “APPLICATION” | “AUDIO” | “IMAGE” | “TEXT” | “VIDEO” | “DOCUMENTATION” | “SPDX” | “OTHER”

4.3.5 Tag: "FileType:"

Example:

FileType: BINARY

Example: (for a README.TXT)

FileType: TEXT

FileType: DOCUMENTATION

Example (foo.exe)

FileType: BINARY

FileType: APPLICATION

4.3.6 RDF: property spdx:fileType in **class** spdx:File

Example:

```
<File rdf:about="file1">
  <fileType rdf:resource="fileType_binary" />
</File>
```

Example: (where file2 is a README.TXT)

```
<File rdf:about="file2">
  <fileType rdf:resource="fhttp://spdx.org/rdf/terms#fileType_text" />
  <fileType rdf:resource="fhttp://spdx.org/rdf/terms#fileType_documentation" />
</File>
```

4.4 File Checksum

4.4.1 Purpose: Provide a unique identifier to match analysis information on each specific file in a package.

4.4.2 Intent: Here, by providing a unique identifier of each file, confusion over which version/modification of a specific file should be eliminated.

4.4.3 Cardinality: Mandatory, one SHA1, others may be optionally provided.

4.4.4 Algorithm: SHA1() is to be used on the file. Other algorithms that can be provided optionally include SHA256(), MD5().

4.4.5 Data Format: There are three components, an algorithm identifier (SHA1), a separator (":") and a 160 bit value represented as 40 lowercase hexadecimal digits. For other algorithms, an appropriate number of hexadecimal digits is expected.

4.4.6 Tag: "FileChecksum:"

Example:

FileChecksum: SHA1: d6a770ba38583ed4bb4525bd96e50461655d2758

FileChecksum: MD5: 624c1abb3664f4b35547e7c73864ad24

4.4.7 RDF: property spdx:Checksum in **class** spdx:File

Example:

```
<File rdf:about="">
  <checksum>
    <Checksum>
      <algorithm rdf:resource="http://spdx.org/rdf/terms#checksumAlgorithm_sha1"/>
      <checksumValue>d6a770ba38583ed4bb4525bd96e50461655d2758
    </checksumValue>
    </Checksum>
    <Checksum>
      <algorithm rdf:resource="http://spdx.org/rdf/terms#checksumAlgorithm_md5"/>
      <checksumValue> 624c1abb3664f4b35547e7c73864ad24
    </checksumValue>
    </Checksum>
  </checksum>
</File>
```

4.5 Concluded License

4.5.1 Purpose: This field contains the license the SPDX file creator has concluded as governing the file or alternative values if the governing license cannot be determined. The options to populate this field are limited to:

- (a) the SPDX License List short form identifier, if the concluded license is on the SPDX License List;
- (b) a reference to the licenses, denoted by LicenseRef-[idString], if the concluded license is not on the SPDX License List;
- (c) NOASSERTION should be used if:
 - (i) the SPDX file creator has attempted to but cannot reach a reasonable objective determination of the concluded license;
 - (ii) the SPDX file creator is uncomfortable concluding a license, despite some license information being available;
 - (iii) the SPDX file creator has made no attempt to arrive at a concluded license;
 - (iv) the SPDX file creator has intentionally provided no information (no meaning should be implied by doing so); or
 - (v) there is no license information from which to conclude a license for the file.

With respect to “a” and “b” above, if there is more than one concluded license, all should be included. If the package recipient has a choice of multiple licenses, then each of the choices should be recited as a “disjunctive” or “conjunctive” license, as appropriate. If the Concluded License is not the same as the License Information in File, a written explanation should be provided in the Comments on License field (section 4.7). With respect to (c), a written explanation in the Comments on License field is preferred.

4.5.2 Intent: Here, the intent is for the SPDX file creator to analyze the License Information in file (section 4.6) and other objective information, e.g., “COPYING FILE,” along with the results from any scanning tools, to arrive at a reasonably objective conclusion as to what license governs the file.

4.5.3 Cardinality: Mandatory, one.

4.5.4 Data Format: <short form identifier in Appendix I> | <license set>
 [ExternalDocumentRef:]“LicenseRef”-[idString] |
 | “NONE” | “NOASSERTION”

where:

[ExternalDocumentRef:] is an optional reference to an external SPDX document as described in section 2.7

[idString] is a unique string containing letters, numbers, “.”, “-” or “+”

<license set> to be used when there is a choice between licenses (“disjunctive license”), they must be separated with “OR” and enclosed in parentheses. When multiple licenses apply (“conjunctive license”), they must be separated with an “AND” and enclosed in parentheses. See Appendix IV for License Expression syntax.

4.5.5 Tag: “LicenseConcluded:”

Example:

LicenseConcluded: LGPL-2.0

Example:

LicenseConcluded: (LGPL-2.0 OR LicenseRef-2)

4.5.6 RDF: property spdx:licenseConcluded in **class** spdx:File

Example:

```
<File rdf:about="file">
  <licenseConcluded>LGPL-2.0</licenseConcluded>
</File>
```

Example:

```
<File rdf:about="">
  <licenseConcluded>
    <DisjunctiveLicenseSet>
      <member rdf:resource="http://spdx.org/licenses/LGPL-2.0"/>
      <member rdf:resource="LicenseRef-2"/>
    </DisjunctiveLicenseSet>
  </licenseConcluded>
</File>
```

4.6 License Information in File

4.6.1 Purpose: This field contains the license information actually found in the file, if any. Any license information not actually in the file, e.g., “COPYING.txt” file in a top level directory, should not be reflected in this field. This information is most commonly found in the header of the file, although it may be in other areas of the actual file. The options to populate this field are limited to:

- (a) the SPDX License List short form identifier, if the license is on the SPDX License List;
- (b) a reference to the license, denoted by LicenseRef-[idString], if the license is not on the SPDX License List;
- (c) NONE, if the actual file contains no license information whatsoever; or
- (d) NOASSERTION, if the SPDX file creator has not examined the contents of the actual file or the SPDX file creator has intentionally provided no information (no meaning should be implied by doing so).

With respect to “a” and “b” above, if license information for more than one license is contained in the file or if the license information offers the package recipient a choice of licenses, then each of the choices should be listed as a separate entry.

4.6.2 Intent: Here, the intent is to provide the license information actually in the file, as compared to the Concluded License field.

4.6.3 Cardinality: Mandatory, one or many.

4.6.4 Data Format: <short form identifier in Appendix I> |
 [ExternalDocumentRef:]“LicenseRef”-[idString] |
 | “NONE” | “NOASSERTION”

where:

[ExternalDocumentRef:] is an optional reference to an external SPDX document as described in section 2.7

[idString] is a unique string containing letters, numbers, “.”, “-” or “+”

4.6.5 Tag: “LicenseInfoInFile:”

Example:

LicenseInfoInFile: GPL-2.0

LicenseInfoInFile: LicenseRef-2

4.6.6 RDF: property spdx:licenseInfoInFile in **class** spdx:File

Example:

```
<File rdf:about="file1">
  <licenseInfoInFile rdf:resource="http://spdx.org/licenses/GPL-2.0" />
  <licenseInfoInFile rdf:resource="LicenseRef-2" />
</File>
```

4.7 Comments on License

4.7.1 Purpose: This field provides a place for the SPDX file creator to record any relevant background references or analysis that went in to arriving at the Concluded License for a file. If the Concluded License does not match the License Information in File, this should be explained by the SPDX file creator. It is also preferable to include an explanation here when the Concluded License is NOASSERTION.

4.7.2 Intent: Here, the intent is to provide the recipient of the SPDX file with a detailed explanation of how the Concluded License was determined if it does not match the License Information in File, is marked NOASSERTION, or other helpful information relevant to determining the license of the file.

4.7.3 Cardinality: Optional, one.

4.7.4 Data Format: free form text that can span multiple lines

4.7.5 Tag: "LicenseComments:"
In tag format multiple lines are delimited by <text> .. </text>.

Example:

LicenseComments: <text>

The concluded license was taken from the package level that the file was included in. This information was found in the COPYING.txt file in the xyz directory.

</text>

4.7.6 RDF: property spdx:licenseComments in **class** spdx:File

Example:

<File rdf:about="">

<licenseComments>

The concluded license was taken from the package level that the file was included in. This information was found in the COPYING.txt file in the xyz directory. This package has been shipped in source and binary form.

</licenseComments>

</File>

4.8 Copyright Text

4.8.1 Purpose: Identify the copyright holder of the file, as well as any dates present. This will be a freeform text field extracted from the actual file. The options to populate this field are limited to:

- (a) any text relating to a copyright notice, even if not complete;
- (b) NONE, if the file contains no copyright information whatsoever; or
- (c) NOASSERTION, if the SPDX creator has not examined the contents of the actual file or if the SPDX creator has intentionally provided no information (no meaning should be implied from the absence of an assertion).

4.8.2 Intent: Record any copyright notice for the package.

4.8.3 Cardinality: Mandatory, one.

4.8.4 Data Format: free form text that can span multiple lines | "NONE" | "NOASSERTION"

4.8.5 Tag: "FileCopyrightText:"
In tag format multiple lines are delimited by <text> .. </text>.

Example:

FileCopyrightText: <text> Copyright 2008-2010 John Smith </text>

4.8.6 RDF: property spdx:copyrightText in class spdx:File

Example:

```
<File rdf:about="">
  <copyrightText>
    Copyright 2008-2010 John Smith
  </copyrightText>
</File>
```

4.9 Artifact of Project Name

4.9.1 Purpose: To indicate that a file has been derived from a specific project.

4.9.2 Intent: To make it easier for recipients of the SPDX file to determine the original source of the identified file. If the project is not described in an SPDX Document, then ArtifactOf can be used. If the project is described in another SPDX Document, then Relationship should be used.

4.9.3 Cardinality: Optional, one or many.

4.9.4 Data Format: single line of text. In Tag/value format the ArtifactOfProjectName must precede any optional ArtifactOf optional properties (e.g. ArtifactOfHomePage and ArtifactOfURI).

4.9.5 Tag: "ArtifactOfProjectName:"

Example:

ArtifactOfProjectName: Jena

4.9.6 RDF: property spdx:artifactOf/doap:Project/doap:name

Example:

```
<File>
  <artifactOf>
    <doap:Project>
      <doap:name>Jena</doap:name>
    </doap:Project>
  </artifactOf>
</File>
```

4.10 Artifact of Project Homepage

- 4.10.1 Purpose:** To indicate the location of the project from which the file has been derived.
- 4.10.2 Intent:** To make it easier for recipients of the SPDX file to determine the original source of the identified file.
- 4.10.3 Cardinality:** Optional, one or many.
- 4.10.4 Data Format:** uniform resource locator | “UNKNOWN”. In Tag/value format all optional ArtifactOf fields must follow immediately below the ArtifactOfProjectName.

4.10.5 Tag: "ArtifactOfProjectHomePage:"

Example:

ArtifactOfProjectHomePage: <http://www.openjena.org/>

4.10.6 RDF: spdx:artifactOf/doap:Project/doap:homepage

Example:

```
<File>
  <artifactOf>
    <doap:Project>
      <doap:homepage >http://www.openjena.org/</doap:homepage>
    </doap:Project>
  </artifactOf>
</File>
```

4.11 Artifact of Project Uniform Resource Identifier

- 4.11.1 Purpose:** To provide a linkage to the project resource in the DOAP document and permit interoperability between the different formats supported.
- 4.11.2 Intent:** To make it easier for recipients of the SPDX file to determine the original source of the identified file.
- 4.11.3 Cardinality:** Optional, one or many.
- 4.11.4 Data Format:** uniform resource identifier. In Tag/value format all optional ArtifactOf fields must follow immediately below the ArtifactOfProjectName.

4.11.5 Tag: "ArtifactOfProjectURI:"

Example:

ArtifactOfProjectURI: <http://subversion.apache.org/doap.rdf>

4.11.6 RDF: spdx:artifactOf/doap

Example:

```
<File>
  <artifactOf rdf:resource="http://subversion.apache.org/" />
```

```

</File>
  <!-- Note: within the DOAP file at http://subversion.apache.org/doap.rdf
        the value "http://subversion.apache.org/" is the URI of the describes
        resource of type doap:Project -->

```

4.12 File Comment

4.12.1 Purpose: This field provides a place for the SPDX file creator to record any general comments about the file.

4.12.2 Intent: Here, the intent is to provide the recipient of the SPDX file with more information determined after careful analysis of a file.

4.12.3 Cardinality: Optional, one.

4.12.4 Data Format: free form text that can span multiple lines

4.12.5 Tag: "FileComment:"
In tag format multiple lines are delimited by <text> .. </text>.

Example:

```
FileComment: <text>
```

```

The concluded license was taken from the package level that the file was included in.
This information was found in the COPYING.txt file in the xyz directory.
</text>

```

4.12.6 RDF: property rdfs:comments in **class** spdx:File

Example:

```

<File rdf:about="">
  <rdfs:comment>
    The concluded license was taken from the package level that the file
    was included in. This information was found in the COPYING.txt file
    in the xyz directory. This package has been shipped in source and binary form.
  </rdfs:comment>
</File>

```

4.13 File Notice

4.13.1 Purpose: This field provides a place for the SPDX file creator to record legal notices potentially found in the file. This may or may not include copyright statements.

4.13.2 Intent: Here, the intent is to provide the recipient of the SPDX file with notices that may require additional review or publication as a requirement of distribution.

4.13.3 Cardinality: Optional, one.

4.13.4 Data Format: free form text that can span multiple lines

4.13.5 Tag: "FileNotice:"
In tag format multiple lines are delimited by <text> .. </text>.

Example:

FileNotice: <text>

Copyright (c) 2001 Aaron Lehmann aaroni@vitelus.com

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

</text>

4.13.6 RDF: property noticeText in class spdx:File**Example:**

<File rdf:about="">

<noticeText>

Copyright (c) 2001 Aaron Lehmann aaroni@vitelus.com

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

</noticeText>

</File>

4.14 File Contributor

4.14.1 Purpose: This field provides a place for the SPDX file creator to record file contributors. Contributors could include names of copyright holders and/or authors who may not be copyright

holders, yet contributed to the file content.

4.14.2 Intent: Here, the intent is to provide the recipient of the SPDX file with a list of one or more contributors (credits). This is one way of providing acknowledgement to the contributors of a file. This would be useful, for example, if a recipient company wanted to contact copyright holders to inquire about alternate licensing.

4.14.3 Cardinality: Optional, one or many.

4.14.4 Data Format: free form text on a single line.

4.14.5 Tag: "FileContributor:"
In tag format single line per contributor.

Example:

FileContributor: Modified by Paul Mundt lethal@linux-sh.org

FileContributor: The Regents of the University of California

FileContributor: IBM Corporation

4.14.6 RDF: property fileContributor in **class** spdx:File

Example:

```
<File rdf:about="">
  <fileContributor> Modified by Paul Mundt lethal@linux-sh.org </fileContributor>
  <fileContributor> The Regents of the University of California </fileContributor>
  <fileContributor> IBM Corporation </fileContributor>
</File>
```

4.15 File Dependencies (deprecated)

This field is deprecated in SPDX 2.0 in favor of using Section 8 which provides more granularity about relationships.

4.15.1 Purpose: The field provides a place for the SPDX file creator to record a list of other files (referenceable within this SPDX file) which the file is a derivative of and/or depends on for the build (e.g., source file or build script for a binary program or library). The list of files may not necessarily represent the list of all file dependencies, but possibly the ones that impact the licensing and/or may be needed as part of the file distribution obligation.

4.15.2 Intent: Here, the intent is to provide the recipient of the SPDX file with file dependency information based on the build system that created the file. These other files may impact the licensing of the file and/or may be required to satisfy the distribution obligation of the file (e.g., source files subject to a copyleft license).

4.15.3 Cardinality: Optional, one or many.

4.15.4 Data Format: Reference to the file within the SPDX document. For the Tag format, this will be the filename. For the RDF format, it will be a reference to the actual file node.

4.15.5 Tag: "FileDependency:"

Example:

FileDependency:./busybox-1.20.2/shell/match.h

FileDependency:./busybox-1.20.2/shell/match.c

FileDependency:./busybox-1.20.2/shell/ash.c

4.15.6 RDF: property `spdx:fileDependency` in class `spdx:File`**Example:**

```

<File rdf:nodeID="A0">
  <fileName>./package/source1.java</fileName>
</File>

<File rdf:nodeID="A1">
  <fileName>./package/source2.java</fileName>
</File>

<File rdf:nodeID="A3">
  <fileName>./package/source3.java</fileName>
</File>

<File rdf:about="">
  <fileName>./package/mylibrary.jar</fileName>
  <fileDependency rdf:nodeID="A0"/>
  <fileDependency rdf:nodeID="A1"/>
  <fileDependency rdf:nodeID="A2"/>
</File>

```

5 Other Licensing Information Detected

This section is used for any detected, declared or concluded licenses that are NOT on the SPDX License List. For the most up-to-date version of the list see: <http://spdx.org/licenses/>. The SPDX License List can also be found here in Appendix I.

One instance should be created for every unique license or licensing information reference detected in package that does not match one of the licenses on the SPDX License List. Each license instance should have the following fields.

Fields:

5.1 License Identifier

5.1.1 Purpose: Provide a locally unique identifier to refer to licenses that are not found on the SPDX License List. This unique identifier can then be used in the packages and files sections of the SPDX file (sections 4 and 6, respectively).

5.1.2 Intent: Create a human readable short form license identifier for a license not on the SPDX License List. This identifier should be unique within the SPDX file. In previous versions of SPDX, the references were required to be sequential numbers, but with version 1.2, creators may specify references that are easier for humans to remember and mentally map.

5.1.3 Cardinality: Conditional (mandatory, one) if license is not on SPDX License List.

5.1.4 Data Format: "LicenseRef-"[idString]
where
[idString] is a unique string containing letters, numbers, ".", "-", or "+".

5.1.5 Tag: "LicenseID:"

Examples:

LicenseID: LicenseRef-1

LicenseID: LicenseRef-Bereware-4.2

5.1.6 RDF: property spdx:licenseID in class spdx:ExtractedLicensingInfo

Examples:

```
<ExtractedLicensingInfo rdf:about="licenseRef-1">
```

```
  <licenseID>LicenseRef-1</licenseID>
```

```
</ExtractedLicensingInfo>
```

```
<ExtractedLicensingInfo rdf:about="licenseRef-Bereware-4.2">
```

```
  <licenseID>LicenseRef-Bereware-4.2</licenseID>
```

```
</ExtractedLicensingInfo>
```

5.2 Extracted Text

5.2.1 Purpose: Provide a copy of the actual text of the license reference extracted from the package or file that is associated with the License Identifier Assigned to aid in future analysis.

5.2.2 Intent: Provide the actual text as found in the package or file for a license that is not on the SPDX License List.

5.2.3 Cardinality: Conditional (Mandatory, one) if there is an Identifier Assigned.

5.2.4 Data Format: free form text field that may span multiple lines.

5.2.5 Tag: "ExtractedText:"
In tag format multiple lines are delimited by <text> .. </text>.

Example:

ExtractedText: <text>"THE BEER-WARE LICENSE" (Revision 42):
phk@FreeBSD.ORG wrote this file. As long as you retain this notice you
can do whatever you want with this stuff. If we meet some day, and you think this stuff is worth it, you can buy me a
beer in return Poul-Henning Kamp </text>

5.2.6 RDF: property spdx:extractedText in class spdx:ExtractedLicensingInfo

Example:

```
<ExtractedLicensingInfo rdf:about="licenseRef-Beerware-42">
<licenseId>LicenseRef-Beerware-4.2</licenseId>
<extractedText> "THE BEER-WARE LICENSE" (Revision 42):
phk@FreeBSD.ORG wrote this file. As long as you retain this notice you
can do whatever you want with this stuff. If we meet some day, and you think this stuff is worth it, you can buy me a
beer in return Poul-Henning Kamp
</extractedText>
</ExtractedLicensingInfo>
```

5.3 License Name

5.3.1 Purpose: Common name of the license not on the SPDX list. If there is no common name or it is not known, please use NOASSERTION.

5.3.2 Intent: Provides a human readable name suitable for use as a title or label of the license when showing compact lists of licenses from the SPDX data to humans.

5.3.3 Cardinality: Conditional (mandatory, one) if license is not on SPDX License List.

5.3.4 Data Format: single line of text | "NOASSERTION".

5.3.5 Tag: "LicenseName:"

Example:

LicenseName: Beer-Ware License (Version 42)

5.3.6 **RDF:** property `spdx:licenseName` in class `spdx:ExtractedLicensingInfo`

Example:

```
<ExtractedLicensingInfo rdf:about="licenseRef-Beerware-42">
  <name>Beer-Ware License (Version 42)</name>
</ExtractedLicensingInfo>
```

5.4 License Cross Reference

5.4.1 **Purpose:** Provide a pointer to the official source of a license that is not included in the SPDX License List, that is referenced by the License Identifier Assigned.

5.4.2 **Intent:** Canonical source for a license currently not on the SPDX License List.

5.4.3 **Cardinality:** Conditional (optional, one or more) if license is not on SPDX License List.

5.4.4 **Data Format:** uniform resource locator

5.4.5 **Tag:** "LicenseCrossReference:"

Example:

LicenseCrossReference: <http://people.freebsd.org/~phk/>

5.4.6 **RDF:** property `rdfs:seeAlso` in class `spdx:ExtractedLicensingInfo`

Example:

```
<ExtractedLicensingInfo rdf:about="licenseRef-1">
  <rdfs:seeAlso>http://people.freebsd.org/~phk/</rdfs:seeAlso>
</ExtractedLicensingInfo>
```

5.5 License Comment

5.5.1 **Purpose:** This field provides a place for the SPDX file creator to record any general comments about the license.

5.5.2 **Intent:** Here, the intent is to provide the recipient of the SPDX file with more information determined after careful analysis of a license, or addition cross references.

5.5.3 **Cardinality:** Optional, one.

5.5.4 **Data Format:** free form text that can span multiple lines

5.5.5 **Tag:** "LicenseComment:"
In tag format multiple lines are delimited by `<text> .. </text>`.

Example:

LicenseComment: `<text>`
The beerware license has a couple of other standard variants.
`</text>`

5.5.6 **RDF:** property `rdfs:comment` in class `spdx:ExtractedLicensingInfo`**Example:**

```
<ExtractedLicensingInfo rdf:about="licenseRef-1">
  <rdfs:comment> The beerware license has a couple of other standard variants.
</rdfs:comment>
</ExtractedLicensingInfo>
```

6 Relationships between SPDX Elements

6.1 Relationship

6.1.1 Purpose: This field provides information about the relationship between two SPDX elements. For example, you can represent a relationship between two different Files, between a Package and a File, between two Packages, or between one SPDXDocument and another SPDXDocument. The relationships between two elements that are supported are:

Relationship	Description	Example
DESCRIBES	Is to be used when SPDXRef-DOCUMENT describes SPDXRef-A	An SPDX document 'glibc.spdx' describes package 'glibc'. Note this is a logical relationship to help organize related items within an SPDX document that is mandatory if more than one package or set of files not in a package is present.
DESCRIBED_BY	Is to be used when SPDXRef-A is described by SPDXREF-Document	TO DO: fill in example pointing to an external document.
CONTAINS	Is to be used when SPDXRef-A contains SPDXRef-B.	An ARCHIVE file 'bar.tgz' contains a SOURCE file 'foo.c'.
CONTAINED_BY	Is to be used when SPDXRef-A is contained by SPDXRef-B.	A SOURCE file foo.c is contained by ARCHIVE file 'bar.tgz'
GENERATES	Is to be used when SPDXRef-A generates the SPDXRef-B.	A SOURCE file 'makefile.mk' generates a BINARY file 'a.out'
GENERATED_FROM	Is to be used when SPDXRef-A was generated from SPDXRef-B.	A BINARY file 'a.out' has been generated from a SOURCE file 'makefile.mk'. A BINARY file 'foolib.a' is generated from a SOURCE file 'bar.c'.
ANCESTOR_OF	Is to be used when SPDXRef-A is an ancestor (same lineage	A SOURCE file 'makefile.mk' is a version of the original ancestor SOURCE file 'makefile2.mk'

	but pre-dates) SPDXRef-B	
DESCENDANT_OF	Is to be used when SPDXRef-A is a descendant of (same lineage but postdates) SPDXRef-B.	A SOURCE file 'makefile2.mk' is a descendant of the original SOURCE file 'makefile.mk'
VARIANT_OF	Is to be used when SPDXRef-A is a variant of (same lineage but not clear which came first) SPDXRef-B.	A SOURCE file 'makefile2.mk' is a variant of SOURCE file 'makefile.mk' if they differ by some edit, but there is no way to tell which came first (no reliable date information).
DISTRIBUTION_ARTIFACT	Is to be used when distributing SPDXRef-A requires that SPDXRef-B also be distributed.	A BINARY file 'foo.o' requires that the ARCHIVE file 'bar-sources.tgz' be made available on distribution.
PATCH_FOR	Is to be used when SPDXRef-A is a patch file for (to be applied to) SPDXRef-B.	A SOURCE file 'foo.diff' is a patch file for SOURCE file 'foo.c'.
PATCH_APPLIED	Is to be used when SPDXRef-A is a patch file that has been applied to SPDXRef-B.	A SOURCE file 'foo.diff' is a patch file that has been applied to SOURCE file 'foo-patched.c'.
COPY_OF	Is to be used when SPDXRef-A is an exact copy of SPDXRef-B.	A BINARY file 'alib.a' is an exact copy of BINARY file 'a2lib.a'.
FILE_ADDED	Is to be used when SPDXRef-A is a file added to SPDXRef-B.	A SOURCE file 'foo.c' has been added to package ARCHIVE 'bar.tgz'.
FILE_DELETED	Is to be used when SPDXRef-A is a file was deleted from to SPDXRef-B.	A SOURCE file 'foo.diff' has been deleted from package ARCHIVE 'bar.tgz'.
FILE_MODIFIED	Is to be used when SPDXRef-A is a file that was modified from SPDXRef-B.	A SOURCE file 'foo.c' has been modified from SOURCE file 'foo.orig.c'.

EXPANDED_FROM_ARCHIVE	Is to be used when SPDXRef-A is expanded from the archive SPDXRef-B.	A SOURCE file 'foo.c', has been expanded from the archive ARCHIVE file 'xyz.tgz'.
DYNAMIC_LINK	Is to be used when SPDXRef-A dynamically links to SPDXRef-B.	An APPLICATION file 'myapp' dynamically links to BINARY file 'zlib.so'.
STATIC_LINK	Is to be used when SPDXRef-A statically links to SPDXRef-B.	An APPLICATION file 'myapp' statically links to BINARY 'zlib.a'.
DATA_FILE_OF	Is to be used when SPDXRef-A is a data file used in SPDXRef-B.	An IMAGE file 'kitty.jpg' is a data file of an APPLICATION 'hellokitty'.
TEST_CASE_OF	Is to be used when SPDXRef-A is a test case used in testing SPDXRef-B.	A SOURCE file testMyCode.java is a unit test file used to test an APPLICATION MyPackage.
BUILD_TOOL_OF	Is to be used when SPDXRef-A is used to to build SPDXRef-B.	A SOURCE file 'makefile.mk' is used to build an APPLICATION 'zlib'.
DOCUMENTATION_OF	Is to be used when SPDXRef-A provides documentation of SPDXRef-B.	A DOCUMENTATION file 'readme.txt' documents the APPLICATION 'zlib'.
OPTIONAL_COMPONENT_OF	Is to be used when SPDXRef-A is an optional component of SPDXRef-B.	A SOURCE file 'fool.c' (which is in the contributors directory) may or may not be included in the build of APPLICATION 'atthebar'.
METAFILE_OF	Is to be used when SPDXRef-A is a metafile of SPDXRef-B.	A SOURCE file 'pom.xml' is a metafile of the APPLICATION 'Apache Xerces'.
PACKAGE_OF	Is to be used when SPDXRef-A is used as a package as part of SPDXRef-B.	A Linux distribution contains an APPLICATION package gawk as part of the distribution MyLinuxDistro.
AMENDS	Is to be used when (current) SPDXRef-DOCUMENT amends the SPDX information in SPDXRef-B.	(Current) SPDX document A version 2 contains a correction to a previous version of the SPDX document A version 1. Note the reserved identifier SPDXRef-DOCUMENT for the current document is required.
PREREQUISITE_FOR	Is to be used when	A library bar.dll is a prerequisite

	SPDXRef-A is a prerequisite for SPDXRef-B	for Application foo.exe
HAS_PREREQUISITE	Is to be used when SPDXRef-A has as a prerequisite SPDXRef-B	An application foo.exe has prerequisite bar.dll
OTHER	Is to be used for a relationship which has not been defined in the formal SPDX specification. A description of the relationship should be included in the Relationship comments field.	

6.1.2 Intent: Here, this field is a reasonable estimation of the relation between two identified elements (i.e. files or packages, or documents), from a developer perspective.

6.1.3 Cardinality: Optional, multiple.

6.1.4 Data Format:

[ExternalDocumentRef:]SPDXID <relationship> [ExternalDocumentRef]:SPDXID
 where [ExternalDocumentRef:] is an optional reference to an external SPDX document as described in section 2.7

where SPDXID is a string containing letters, numbers, ".", "-", as described in sections (2.3, 3.2, 4.2).

where <relationship> is one of the documented relationship types in table 6.1.1.

6.1.5 Tag: "Relationship:"

Examples:

Relationship: SPDXRef-grep CONTAINS SPDXRef-make

RelationshipComment: Package grep contains file make

Relationship: SPDXRef-DOCUMENT AMENDS DocumentRef-SPDXA:SPDXRef-DOCUMENT

RelationshipComment: This current document is an amendment of the SPDXA document.

6.1.6 RDF: property relationship in any SpdxElement

Examples:

```
<SpdxElement rdf:about="#SPDXRef-45">
  <relationship>
    <Relationship>
      <spdx:relatedSpdxElement>
        <spdx:SpdxElement
```

```

rdf:about="http://spdx.org/spdxdocs/spdx-tools-v1.2-3F2504E0-4F89-41D3-9A0C-0305E82C3301#
SPDXRef-ToolsElement"/>
  </spdx:relatedSpdxElement>

```

```

<relationshipType>http://spdx.org/rdf/terms#relationshipType_contains</relationshipType>
  </Relationship>
</relationship>
...
</SpdxElement>

```

6.2 Relationship Comment

6.2.1 Purpose: This field provides a place for the SPDX file creator to record any general comments about the relationship.

6.2.2 Intent: Here, the intent is to provide the recipient of the SPDX file with more information determined after careful analysis of the relationship between two elements in an SPDX file.

6.2.3 Cardinality: Optional, one.

6.2.4 Data Format: free form text that can span multiple lines, refers only to the immediately preceding relationship.

6.2.5 Tag: “RelationshipComment:”
 In tag format multiple lines are delimited by <text> .. </text>.
 A RelationshipComment: must be the line immediately after a “Relationship:”

Example:

```

RelationshipComment: <text>
The package foo.tgz is a pre-requisite for building executable bar.
</text>

```

6.2.6 RDF: property rdfs:comments in class spdx:Relationship

Example:

```

<Relationship rdf:about="">
  <rdfs:comment>
    The package foo.tgz is a pre-requisite for building executable bar.
  </rdfs:comment>
  ...
</Relationship>

```

7 Annotations

7.1 Annotator

7.1.1 Purpose: This field identifies the person, organization or tool that has commented on a file, package, or the entire document.

7.1.2 Intent: It may also be important for participants in the software supply chain to validate and add information on ambiguous files, and packages.

7.1.3 Cardinality: Conditional (Mandatory, one), if there is an Annotation.

7.1.4 Data Format: single line of text with the following keywords.
 "Person: person name" and optional "(email)"
 "Organization: organization" and optional "(email)"
 "Tool: tool identifier - version"

7.1.5 Tag: "Annotator:"

Example:

Annotator: Person: Jane Doe ()

7.1.6 RDF: property spdx:annotator in **class** spdx:Annotation

Example:

```
<Annotation>
  <annotator> Person: Jane Doe () </annotator>
</Annotations>
```

7.2 Annotation Date

7.2.1 Purpose: Identify when the comment was made. This is to be specified according to the combined date and time in the UTC format, as specified in the ISO 8601 standard.

7.2.2 Intent: Here, the Annotation Date can serve as a verification as to when the actual review was done.

7.2.3 Cardinality: Conditional (Mandatory, one), if there is an Annotation.

7.2.4 Data Format: YYYY-MM-DDThh:mm:ssZ

where:

YYYY is year

MM is month with leading zero

DD is day with leading zero

T is delimiter for time

hh is hours with leading zero in 24 hour time

mm is minutes with leading zero

ss is seconds with leading zero
Z is universal time indicator

7.2.5 Tag: "AnnotationDate:"

Example:

AnnotationDate: 2010-01-29T18:30:22Z

7.2.6 RDF: property spdx:annotationDate in **class** spdx:Annotation

Example:

```
</Annotation>
  <annotationDate> 2010-01-29T18:30:22Z </annotation Date>
</Annotation>
```

7.3 Annotation Type

7.3.1 Purpose: This field describes the type of annotation.

7.3.2 Intent: This allows the type of annotation to be recorded.

7.3.3 Cardinality: Conditional (Mandatory, one), if there is an Annotation.

7.3.4 Data Format: "REVIEW" | "OTHER"

7.3.5 Tag: "AnnotationType:"

Example:

AnnotationType: REVIEW

7.3.6 RDF: property rdfs:comment in **class** spdx:Annotation

Example:

```
<Annotation>
  <spdx:annotationType rdf:resource="http://spdx.org/rdf/terms#annotationType_other"/>
</Annotation >
```

7.4 SPDX Identifier Reference

7.4.1 Purpose: Uniquely identify the element in an SPDX document which is being referenced. These may be referenced internally and externally with the addition of the SPDX Document Identifier.

7.4.2 Intent: There may be several versions of the same package or file within an SPDX document. Each element needs to be able to be referred to uniquely so that relationships between elements can be clearly articulated.

7.4.3 Cardinality: Conditional (Mandatory, one), if there is an Annotation.

7.4.4 DataFormat: [ExternalDocumentRef:]SPDXID

where:

- [ExternalDocumentRef:] is an optional reference to an external SPDX document as described in section 2.6
- SPDXID is a unique string containing letters, numbers, ".", "-". as described in Sections 2.3, 3.2 and 4.2.

7.4.5 Tag: "SPDXREF:"**Example:**

SPDXREF: SPDXRef-45

7.4.6 RDF:

For RDF, the annotations are a property of the SPDX element it is annotating.

```
<SpdxElement rdf:about="#SPDXRef-45">
  <annotation>
    <Annotation>
      ...
    </Annotation>
  </annotation>
</SpdxElement rdf:about="#SPDXRef-45">
```

7.5 Annotation Comment

7.5.1 Purpose: This optional free form text field permits the annotator to provide commentary on the analysis.

7.5.2 Intent: This allows the annotator to provide independent assessment and note any points where there is disagreement with the analysis.

7.5.3 Cardinality: Conditional (Mandatory, one), if there is an Annotation.

7.5.4 Data Format: free form text that can span multiple lines.

7.5.5 Tag: "AnnotationComment:"

In tag format multiple lines are delimited by <text> .. </text>.

Example:

AnnotationComment: <text>

All of the licenses seen in the file, are matching what was seen during manual inspection. There are some terms that can influence the concluded license, and some alternatives may be possible, but the concluded license is one of the options.
</text>

7.5.6 RDF: property rdfs:comment in class spdx:Annotation

Example:

```
<Annotation>
  <rdfs:comment>
```

All of the licenses seen in the file, are matching what was seen during manual inspection. There are some terms that can influence the concluded license, and some alternatives may

be possible, but the concluded license is one of the options.
</rdfs:comment>
</Annotation >

8 Review Information (deprecated)

The review information section is included for compatibility with SPDX 1.2, and is deprecated in SPDX 2.0. Any review information should use an Annotation (as described in Section 7) with an annotation type of `annotationType_review`.

Review information can be added after the initial SPDX file has been created. The set of fields are optional and multiple instances can be added. Once a Reviewer entry is added, the Review Date associated with the review is mandatory. The Created date should not be modified as a result of the addition of information regarding the conduct of a review. A Review Comments is optional.

Fields:

8.1 Reviewer (deprecated)

This field is deprecated in SPDX 2.0.

8.1.1 Purpose: This field identifies the person, organization or tool that has reviewed the SPDX file. This field is optional and thus there is no requirement for any reviewer to add a set of review information to the file. This can be considered as an equivalent to “signed off” or “reviewed by.” Additional reviewers can be added after the original version of the SPDX file is created and be appended to the original file.

8.1.2 Intent: Here, as time progresses certain reviewers will begin to gain credibility as reliable. This field intends to make such information transparent. It may also be important for participants in the software supply chain to validate whether upstream providers have reviewed the SPDX file.

8.1.3 Cardinality: Optional, one.

8.1.4 Data Format: single line of text with the following keywords.
 "Person: person name" and optional "(email)"
 "Organization: organization" and optional "(email)"
 "Tool: tool identifier - version"

8.1.5 Tag: "Reviewer:"

Example:

Reviewer: Person: Jane Doe ()

8.1.6 RDF: property `spdx:reviewer` in **class** `spdx:Review`

Example:

```
<Review>
  <reviewer> Person: Jane Doe () </reviewer>
</Review>
```

8.2 Review Date (deprecated)

This field is deprecated in SPDX 2.0.

8.2.1 Purpose: Identify when the review was done. This is to be specified according to the combined date and time in the UTC format, as specified in the ISO 8601 standard.

8.2.2 Intent: Here, the ReviewDate can serve as a verification as to when the actual review was done.

8.2.3 Cardinality: Conditional (Mandatory, one), if there is a Reviewer.

8.2.4 Data Format: YYYY-MM-DDThh:mm:ssZ
where:

YYYY is year
MM is month with leading zero
DD is day with leading zero
T is delimiter for time
hh is hours with leading zero in 24 hour time
mm is minutes with leading zero
ss is seconds with leading zero
Z is universal time indicator

8.2.5 Tag: "ReviewDate:"

Example:

ReviewDate: 2010-01-29T18:30:22Z

8.2.6 RDF: property spdx:reviewDate in **class** spdx:Review

Example:

```
<Review>
  <reviewDate> 2010-01-29T18:30:22Z </reviewDate>
</Review>
```

8.3 Review Comment (deprecated)

This field is deprecated in SPDX 2.0.

8.3.1 Purpose: This optional free form text field permits the reviewer to provide commentary on the analysis.

8.3.2 Intent: This allows the reviewer to provide independent assessment and note any points where there is disagreement with the analysis.

8.3.3 Cardinality: Optional, one.

8.3.4 Data Format: free form text that can span multiple lines.

8.3.5 Tag: "ReviewComment:"
In tag format multiple lines are delimited by <text> .. </text>.

Example:

ReviewComment: <text>

All of the licenses seen in the file, are matching what was seen during manual inspection. There are some terms that can influence the concluded license, and some alternatives may be possible, but the concluded license is one of the options.

</text>

8.3.6 RDF: property rdfs:comment in class spdx:Review

Example:

<Review>

 <rdfs:comment>

All of the licenses seen in the file, are matching what was seen during manual inspection. There are some terms that can influence the concluded license, and some alternatives may be possible, but the concluded license is one of the options.

 </rdfs:comment>

</Review>

Appendix I: 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.

The following table contains the full names and short identifiers for the SPDX License List, v2.0 which was released February 2015. For the full and most up-to-date version of the SPDX License List, please see <http://spdx.org/licenses/>

You may propose additional licenses be added to the SPDX License List by following the process at <http://spdx.org/spdx-license-list/request-new-license>

Guidelines for what constitutes a license match to the SPDX License List when generating an SPDX file can be found here: <http://spdx.org/spdx-license-list/matching-guidelines>

I.1 Licenses with Short Form Identifiers

Full Name of License	SPDX License Identifier
3dfx Glide License	Glide
Abstyles License	Abstyles
Academic Free License v1.1	AFL-1.1
Academic Free License v1.2	AFL-1.2
Academic Free License v2.0	AFL-2.0
Academic Free License v2.1	AFL-2.1
Academic Free License v3.0	AFL-3.0
Academy of Motion Picture Arts and Sciences BSD	AMPAS
Adaptive Public License 1.0	APL-1.0
Adobe Glyph List License	Adobe-Glyph
Adobe Postscript AFM License	APAFML
Adobe Systems Incorporated Source Code License Agreement	Adobe-2006

Affero General Public License v1.0	AGPL-1.0
Afmparse License	Afmparse
Aladdin Free Public License	Aladdin
Amazon Digital Services License	ADSL
AMD's plpa_map.c License	AMDPLPA
ANTLR Software Rights Notice	ANTLR-PD
Apache License 1.0	Apache-1.0
Apache License 1.1	Apache-1.1
Apache License 2.0	Apache-2.0
Apple MIT License	AML
Apple Public Source License 1.0	APSL-1.0
Apple Public Source License 1.1	APSL-1.1
Apple Public Source License 1.2	APSL-1.2
Apple Public Source License 2.0	APSL-2.0
Artistic License 1.0	Artistic-1.0
Artistic License 1.0 (Perl)	Artistic-1.0-Perl
Artistic License 1.0 w/clause 8	Artistic-1.0-cl8
Artistic License 2.0	Artistic-2.0
Attribution Assurance License	AAL
Bahyph License	Bahyph
Barr License	Barr
Beerware License	Beerware
BitTorrent Open Source License v1.0	BitTorrent-1.0
BitTorrent Open Source License v1.1	BitTorrent-1.1
Boost Software License 1.0	BSL-1.0

Borceux license	Borceux
BSD 2-clause "Simplified" License	BSD-2-Clause
BSD 2-clause FreeBSD License	BSD-2-Clause-FreeBSD
BSD 2-clause NetBSD License	BSD-2-Clause-NetBSD
BSD 3-clause "New" or "Revised" License	BSD-3-Clause
BSD 3-clause Clear License	BSD-3-Clause-Clear
BSD 4-clause "Original" or "Old" License	BSD-4-Clause
BSD Protection License	BSD-Protection
BSD with attribution	BSD-3-Clause-Attribution
BSD-4-Clause (University of California-Specific)	BSD-4-Clause-UC
bzip2 and libbzip2 License v1.0.5	bzip2-1.0.5
bzip2 and libbzip2 License v1.0.6	bzip2-1.0.6
Caldera License	Caldera
CeCILL Free Software License Agreement v1.0	CECILL-1.0
CeCILL Free Software License Agreement v1.1	CECILL-1.1
CeCILL Free Software License Agreement v2.0	CECILL-2.0
CeCILL-B Free Software License Agreement	CECILL-B
CeCILL-C Free Software License Agreement	CECILL-C
Clarified Artistic License	CIArtistic
CMU License	MIT-CMU
CNRI Python License	CNRI-Python
CNRI Python Open Source GPL Compatible License Agreement	CNRI-Python-GPL-Compati ble
Code Project Open License 1.02	CPOL-1.02
Common Development and Distribution License 1.0	CDDL-1.0
Common Development and Distribution License 1.1	CDDL-1.1

Common Public Attribution License 1.0	CPAL-1.0
Common Public License 1.0	CPL-1.0
Computer Associates Trusted Open Source License 1.1	CATOSL-1.1
Condor Public License v1.1	Condor-1.1
Creative Commons Attribution 1.0	CC-BY-1.0
Creative Commons Attribution 2.0	CC-BY-2.0
Creative Commons Attribution 2.5	CC-BY-2.5
Creative Commons Attribution 3.0	CC-BY-3.0
Creative Commons Attribution 4.0	CC-BY-4.0
Creative Commons Attribution No Derivatives 1.0	CC-BY-ND-1.0
Creative Commons Attribution No Derivatives 2.0	CC-BY-ND-2.0
Creative Commons Attribution No Derivatives 2.5	CC-BY-ND-2.5
Creative Commons Attribution No Derivatives 3.0	CC-BY-ND-3.0
Creative Commons Attribution No Derivatives 4.0	CC-BY-ND-4.0
Creative Commons Attribution Non Commercial 1.0	CC-BY-NC-1.0
Creative Commons Attribution Non Commercial 2.0	CC-BY-NC-2.0
Creative Commons Attribution Non Commercial 2.5	CC-BY-NC-2.5
Creative Commons Attribution Non Commercial 3.0	CC-BY-NC-3.0
Creative Commons Attribution Non Commercial 4.0	CC-BY-NC-4.0
Creative Commons Attribution Non Commercial No Derivatives 1.0	CC-BY-NC-ND-1.0
Creative Commons Attribution Non Commercial No Derivatives 2.0	CC-BY-NC-ND-2.0
Creative Commons Attribution Non Commercial No Derivatives 2.5	CC-BY-NC-ND-2.5
Creative Commons Attribution Non Commercial No Derivatives 3.0	CC-BY-NC-ND-3.0
Creative Commons Attribution Non Commercial No Derivatives 4.0	CC-BY-NC-ND-4.0
Creative Commons Attribution Non Commercial Share Alike 1.0	CC-BY-NC-SA-1.0
Creative Commons Attribution Non Commercial Share Alike 2.0	CC-BY-NC-SA-2.0

Creative Commons Attribution Non Commercial Share Alike 2.5	CC-BY-NC-SA-2.5
Creative Commons Attribution Non Commercial Share Alike 3.0	CC-BY-NC-SA-3.0
Creative Commons Attribution Non Commercial Share Alike 4.0	CC-BY-NC-SA-4.0
Creative Commons Attribution Share Alike 1.0	CC-BY-SA-1.0
Creative Commons Attribution Share Alike 2.0	CC-BY-SA-2.0
Creative Commons Attribution Share Alike 2.5	CC-BY-SA-2.5
Creative Commons Attribution Share Alike 3.0	CC-BY-SA-3.0
Creative Commons Attribution Share Alike 4.0	CC-BY-SA-4.0
Creative Commons Zero v1.0 Universal	CC0-1.0
Crossword License	Crossword
CUA Office Public License v1.0	CUA-OPL-1.0
Cube License	Cube
Deutsche Freie Software Lizenz	D-FSL-1.0
diffmark license	diffmark
Do What The F*ck You Want To Public License	WTFPL
DOC License	DOC
Dotseqn License	Dotseqn
DSDP License	DSDP
dvipdfm License	dvipdfm
Eclipse Public License 1.0	EPL-1.0
Educational Community License v1.0	ECL-1.0
Educational Community License v2.0	ECL-2.0
eGenix.com Public License 1.1.0	eGenix
Eiffel Forum License v1.0	EFL-1.0
Eiffel Forum License v2.0	EFL-2.0
Enlightenment License (e16)	MIT-advertising

enna License	MIT-enna
Entessa Public License v1.0	Entessa
Erlang Public License v1.1	ErlPL-1.1
EU DataGrid Software License	EUDatagrid
European Union Public License 1.0	EUPL-1.0
European Union Public License 1.1	EUPL-1.1
Eurosym License	Eurosym
Fair License	Fair
feh License	MIT-feh
Frameworkx Open License 1.0	Frameworkx-1.0
FreelImage Public License v1.0	FreelImage
Freetype Project License	FTL
FSF Unlimited License	FSFUL
FSF Unlimited License (with License Retention)	FSFULLR
Giftware License	Giftware
GL2PS License	GL2PS
Glulxe License	Glulxe
GNU Affero General Public License v3.0	AGPL-3.0
GNU Free Documentation License v1.1	GFDL-1.1
GNU Free Documentation License v1.2	GFDL-1.2
GNU Free Documentation License v1.3	GFDL-1.3
GNU General Public License v1.0 only	GPL-1.0
GNU General Public License v2.0 only	GPL-2.0
GNU General Public License v3.0 only	GPL-3.0
GNU Lesser General Public License v2.1 only	LGPL-2.1
GNU Lesser General Public License v3.0 only	LGPL-3.0

GNU Library General Public License v2 only	LGPL-2.0
gnuplot License	gnuplot
gSOAP Public License v1.3b	gSOAP-1.3b
Haskell Language Report License	HaskellReport
Historic Permission Notice and Disclaimer	HPND
IBM PowerPC Initialization and Boot Software	IBM-pibs
IBM Public License v1.0	IPL-1.0
ICU License	ICU
ImageMagick License	ImageMagick
iMatix Standard Function Library Agreement	iMatix
Imlib2 License	Imlib2
Independent JPEG Group License	IJG
Intel ACPI Software License Agreement	Intel-ACPI
Intel Open Source License	Intel
IPA Font License	IPA
ISC License	ISC
JasPer License	JasPer-2.0
JSON License	JSON
LaTeX Project Public License 1.3a	LPPL-1.3a
LaTeX Project Public License v1.0	LPPL-1.0
LaTeX Project Public License v1.1	LPPL-1.1
LaTeX Project Public License v1.2	LPPL-1.2
LaTeX Project Public License v1.3c	LPPL-1.3c
Latex2e License	Latex2e
Lawrence Berkeley National Labs BSD variant license	BSD-3-Clause-LBNL
Leptonica License	Leptonica

libpng License	Libpng
libtiff License	libtiff
Lucent Public License v1.02	LPL-1.02
Lucent Public License Version 1.0	LPL-1.0
MakeIndex License	MakeIndex
Matrix Template Library License	MTLL
Microsoft Public License	MS-PL
Microsoft Reciprocal License	MS-RL
MirOS Licence	MirOS
MIT +no-false-attribs license	MITNFA
MIT License	MIT
Motosoto License	Motosoto
Mozilla Public License 1.0	MPL-1.0
Mozilla Public License 1.1	MPL-1.1
Mozilla Public License 2.0	MPL-2.0
Mozilla Public License 2.0 (no copyleft exception)	MPL-2.0-no-copyleft-exception
mpich2 License	mpich2
Multics License	Multics
Mup License	Mup
NASA Open Source Agreement 1.3	NASA-1.3
Naumen Public License	Naumen
Net Boolean Public License v1	NBPL-1.0
NetCDF license	NetCDF
Nethack General Public License	NGPL
Netizen Open Source License	NOSL

Netscape Public License v1.0	NPL-1.0
Netscape Public License v1.1	NPL-1.1
Newsletr License	Newsletr
No Limit Public License	NLPL
Nokia Open Source License	Nokia
Non-Profit Open Software License 3.0	NPOSL-3.0
Noweb License	Noweb
NRL License	NRL
NTP License	NTP
Nunit License	Nunit
OCLC Research Public License 2.0	OCLC-2.0
ODC Open Database License v1.0	ODbL-1.0
ODC Public Domain Dedication & License 1.0	PDDL-1.0
Open Group Test Suite License	OGTSL
Open LDAP Public License 2.2.2	OLDAP-2.2.2
Open LDAP Public License v1.1	OLDAP-1.1
Open LDAP Public License v1.2	OLDAP-1.2
Open LDAP Public License v1.3	OLDAP-1.3
Open LDAP Public License v1.4	OLDAP-1.4
Open LDAP Public License v2.0 (or possibly 2.0A and 2.0B)	OLDAP-2.0
Open LDAP Public License v2.0.1	OLDAP-2.0.1
Open LDAP Public License v2.1	OLDAP-2.1
Open LDAP Public License v2.2	OLDAP-2.2
Open LDAP Public License v2.2.1	OLDAP-2.2.1
Open LDAP Public License v2.3	OLDAP-2.3
Open LDAP Public License v2.4	OLDAP-2.4

Open LDAP Public License v2.5	OLDAP-2.5
Open LDAP Public License v2.6	OLDAP-2.6
Open LDAP Public License v2.7	OLDAP-2.7
Open LDAP Public License v2.8	OLDAP-2.8
Open Market License	OML
Open Public License v1.0	OPL-1.0
Open Software License 1.0	OSL-1.0
Open Software License 1.1	OSL-1.1
Open Software License 2.0	OSL-2.0
Open Software License 2.1	OSL-2.1
Open Software License 3.0	OSL-3.0
OpenSSL License	OpenSSL
PHP License v3.0	PHP-3.0
PHP License v3.01	PHP-3.01
Plexus Classworlds License	Plexus
PostgreSQL License	PostgreSQL
psfrag License	psfrag
psutils License	psutils
Python License 2.0	Python-2.0
Q Public License 1.0	QPL-1.0
Qhull License	Qhull
Rdisc License	Rdisc
RealNetworks Public Source License v1.0	RPSL-1.0
Reciprocal Public License 1.1	RPL-1.1
Reciprocal Public License 1.5	RPL-1.5
Red Hat eCos Public License v1.1	RHeCos-1.1

Ricoh Source Code Public License	RSCPL
Ruby License	Ruby
Sax Public Domain Notice	SAX-PD
Saxpath License	Saxpath
SCEA Shared Source License	SCEA
Scheme Widget Library (SWL) Software License Agreement	SWL
SGI Free Software License B v1.0	SGI-B-1.0
SGI Free Software License B v1.1	SGI-B-1.1
SGI Free Software License B v2.0	SGI-B-2.0
SIL Open Font License 1.0	OFL-1.0
SIL Open Font License 1.1	OFL-1.1
Simple Public License 2.0	SimPL-2.0
Sleepycat License	Sleepycat
SNIA Public License 1.1	SNIA
Standard ML of New Jersey License	SMLNJ
SugarCRM Public License v1.1.3	SugarCRM-1.1.3
Sun Industry Standards Source License v1.1	SISSL
Sun Industry Standards Source License v1.2	SISSL-1.2
Sun Public License v1.0	SPL-1.0
Sybase Open Watcom Public License 1.0	Watcom-1.0
TCL/TK License	TCL
The Unlicense	Unlicense
TMate Open Source License	TMate
TORQUE v2.5+ Software License v1.1	TORQUE-1.1
Trusster Open Source License	TOSL
Unicode Terms of Use	Unicode-TOU

University of Illinois/NCSA Open Source License	NCSA
Vim License	Vim
VOSTROM Public License for Open Source	VOSTROM
Vovida Software License v1.0	VSL-1.0
W3C Software Notice and License (1998-07-20)	W3C-19980720
W3C Software Notice and License (2002-12-31)	W3C
Wsuipa License	Wsuipa
X.Net License	Xnet
X11 License	X11
Xerox License	Xerox
XFree86 License 1.1	XFree86-1.1
xinetd License	xinetd
XPP License	xpp
XSkat License	XSkat
Yahoo! Public License v1.0	YPL-1.0
Yahoo! Public License v1.1	YPL-1.1
Zed License	Zed
Zend License v2.0	Zend-2.0
Zimbra Public License v1.3	Zimbra-1.3
Zimbra Public License v1.4	Zimbra-1.4
zlib License	Zlib
zlib/libpng License with Acknowledgement	zlib-acknowledgement
Zope Public License 1.1	ZPL-1.1
Zope Public License 2.0	ZPL-2.0
Zope Public License 2.1	ZPL-2.1

I.2 Exceptions List

Full Name of Exception	SPDX License Exception
Autoconf exception 2.0	Autoconf-exception-2.0
Autoconf exception 3.0	Autoconf-exception-3.0
Bison exception 2.2	Bison-exception-2.2
Classpath exception 2.0	Classpath-exception-2.0
eCos exception 2.0	eCos-exception-2.0
Font exception 2.0	Font-exception-2.0
GCC Runtime Library exception 2.0	GCC-exception-2.0
GCC Runtime Library exception 3.1	GCC-exception-3.1
WxWindows Library Exception 3.1	WxWindows-exception-3.1

I.3 Deprecated Licenses

Full Name of License	Deprecated SPDX License Identifier
eCos license version 2.0	eCos-2.0
GNU General Public License v1.0 or later	GPL-1.0+
GNU General Public License v2.0 or later	GPL-2.0+
GNU General Public License v2.0 w/Autoconf exception	GPL-2.0-with-autoconf-exception
GNU General Public License v2.0 w/Bison exception	GPL-2.0-with-bison-exception
GNU General Public License v2.0 w/Classpath exception	GPL-2.0-with-classpath-exception
GNU General Public License v2.0 w/Font exception	GPL-2.0-with-font-exception
GNU General Public License v2.0 w/GCC Runtime Library exception	GPL-2.0-with-GCC-exception
GNU General Public License v3.0 or later	GPL-3.0+
GNU General Public License v3.0 w/Autoconf exception	GPL-3.0-with-autoconf-exception

GNU General Public License v3.0 w/GCC Runtime Library exception	GPL-3.0-with-GCC-exception
GNU Lesser General Public License v2.1 or later	LGPL-2.1+
GNU Lesser General Public License v3.0 or later	LGPL-3.0+
GNU Library General Public License v2 or later	LGPL-2.0+
Standard ML of New Jersey License	StandardML-NJ
wxWindows Library License	WXwindows

Appendix II: License Matching Guidelines and Templates

The SPDX License List Matching Guidelines provide guidelines for matching licenses found while creating a SPDX file to the licenses found on the SPDX License List. There is no intent to make a judgment or interpretation, but merely to ensure that when one SPDX creator identifies a license as “BSD 3-clause,” for example, it is indeed the same license as what someone else identifies as “BSD 3-clause” and the same license as what is listed on the SPDX License List. The SPDX License Matching Guidelines can be found here: <https://spdx.org/spdx-license-list/matching-guidelines>.

Some of the matching guidelines require additional information for implementation and to encourage consistent matching results. For example, when text can be replaceable (e.g. the third clause and disclaimer in the BSD licenses, or the third, fourth, and fifth clauses of Apache 1.1) or omitted (e.g. extra text after the noted “end of license terms”) while still affording a match. Such guidelines are indicated via markup in the license template. Not all licenses on the SPDX License List will have template markup.

SPDX License List Template Access

The SPDX License List download includes a .txt file with the license text for each license. Template markup for licenses that require it, are included within the .txt file as per the description below. The download location for the current version of the matching guidelines can be found in the link at the top of the page <http://spdx.org/licenses>.

RDFa Access: The template text for the license can be accessed using the RDF tag `licenseTemplate` on the web page containing the license.

Template Format

A template is composed of text with zero or more rules embedded in it.

A rule is a variable section of a license wrapped between double angle brackets “<<>>” and is composed of 5 fields. Each field is separated with a semi-colon “;”. Rules cannot be embedded within other rules. Rule fields begin with a case sensitive tag followed by an equal sign “=”.

Rule fields:

`original`: the original text of the rule. It is recommended but not required that this is the first field.. This field is required.

`match`: a POSIX ERE regular expression (see below). This field is required.

`name`: name of the field in the template. This field is required.

`type`: one of “required” or “optional” (case sensitive). This field is required.

`example`: another example of the use of the text. This field is optional.

“required” indicates that text matching the regular expression must be supplied to be a matched license.

“optional” indicates that no text is required, but can be optionally supplied.

The POSIX ERE regular expression has the following restrictions and extensions:

Semicolons are escaped with `\;`

POSIX Bracket Extensions are not allowed

Example:

```
<<original=THE AUTHOR OR
```

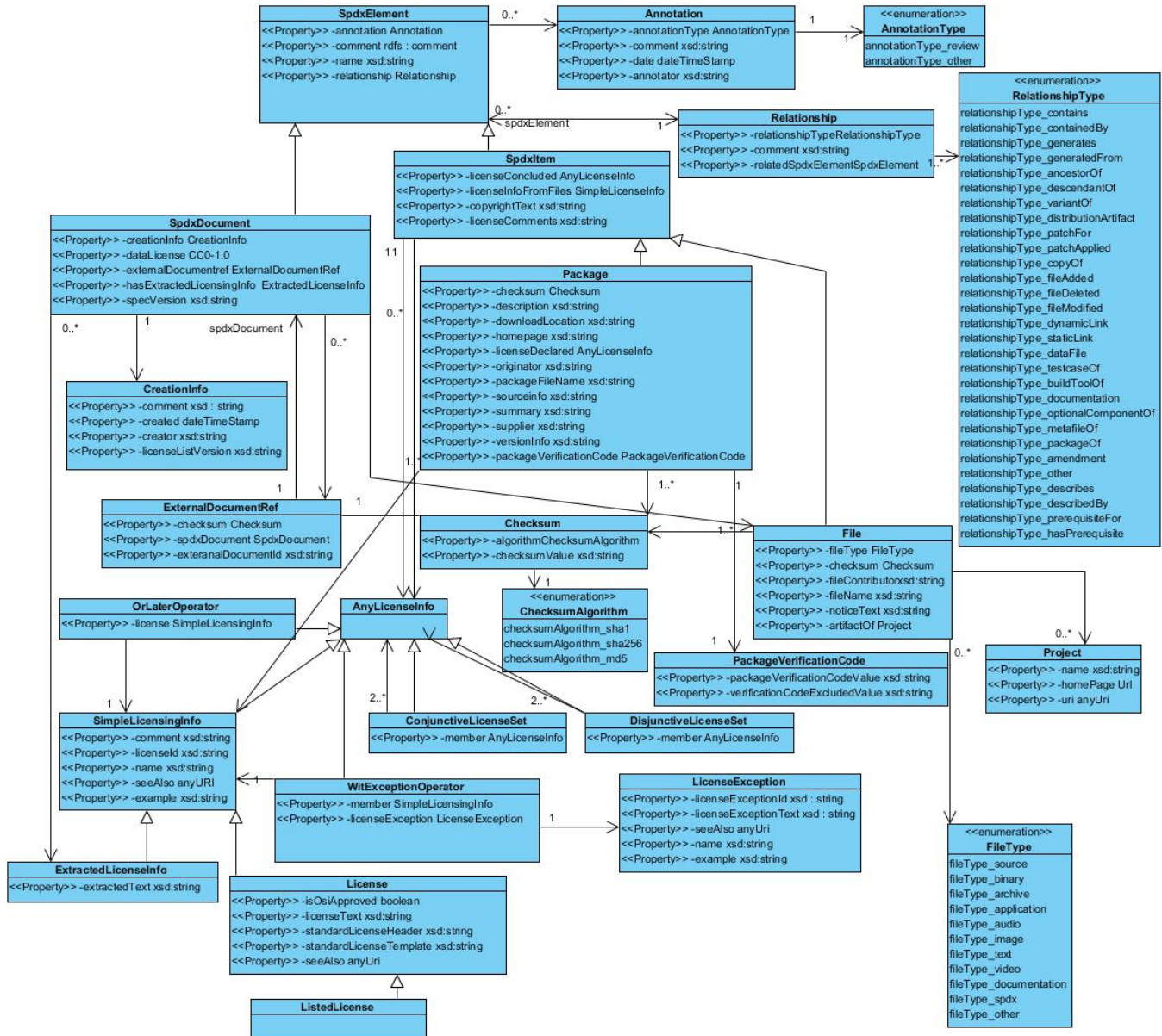

CONTRIBUTORS;match=+;name=copyrightHolderLiability;type=required;example=dmg inc.>>

Appendix III: RDF Data Model Implementation and Identifier Syntax

SPDX® Vocabulary Specification

See: <http://spdx.org/rdf/ontology/spdx-2-0>

Version: 2.0r2050223



Copyright © 2010-2015 Linux Foundation and its Contributors. All other rights are expressly reserved.

Licensed under the [Creative Commons Attribution License 3.0 Unported](https://creativecommons.org/licenses/by/3.0/).

Agent and Tool Identifiers

Fields that identify entities that have acted in relation to the SPDX file are single line of text which name the agent or tool and, optionally, provide contact information. For example, "Person: Jane Doe (jane.doe@example.com)", "Organization: ExampleCodeInspect (contact@example.com)" and "Tool: LicenseFind - 1.0". The exact syntax of agent and tool identifications is described below in [ABNF](#).

```

agent           = person / organization
tool            = "Tool: " name 0*1( " " DASH " " version)

person          = "Person: " name 0*1contact-info
organization    = "Organization: " name 0*1contact-info

name            = 1*( UNRESERVED ) / U+0022 1*( VCHAR-SANS-QUOTE ) U+0022
contact-info    = " (" email-addr ")"
email-addr      = local-name-atom *( "." local-name-atom ) "@" domain-name-atom 1*(
"." domain-name-atom )

idstring        = 1*(ALPHA / DIGIT / "-" / "+" / "." )

version         = 1*VCHAR-SANS-QUOTE

local-name-atom = 1*( ALPHA / DIGIT /      ; Printable US-ASCII
"! " / "# " /      ; characters not including
"$ " / "% " /      ; specials.
"& " / "! " /
"* " / "+" /
"_ " / "/" /
"= " / "? " /
"^ " / " " /
"` " / "{ " /
"| " / "}" /
"~ " )

domain-name-atom = 1*( ALPHA / DIGIT / "-" )

DASH            = U+2010 / U+2212 /      ; hyphen, minus, em dash and
U+2013 / U+2014      ; en dash

UNRESERVED      = U+0020-U+0027 /      ; visible unicode characters
U+0029-U+0080 /      ; except '(' and dashes
U+00A0-U+200F /
U+2011-U+2027 /
U+202A-U+2211 /
U+2213-U+E01EF

VCHAR-SANS-QUOTE = U+0020-U+0021 /      ; visible unicode characters
U+0023-U+0080 /      ; except quotation mark
U+00a0-U+E01EF

```

Appendix IV: License Expression Syntax (beta draft 0.98)

Overview

Often a single license can be used to represent the licensing terms of a source code or binary file, but there are situations where a single license identifier is not sufficient. A common example is when software is offered under a choice of one or more licenses (e.g., GPL-2.0 OR BSD-3-Clause). Another example is when a set of licenses is needed to represent a binary program constructed by compiling and linking two (or more) different source files each governed by different licenses (e.g., LGPL-2.1 and BSD-3-Clause respectively).

The exact syntax of license expressions is described below in [ABNF](#).

```
license-expression = simple-license-identifier /
                    simple-license-identifier 1*( "WITH" ) license-exception-id /
                    "(" simple-license-identifier 1*( "AND" / "OR" ) license-expression ")"

simple-license-identifier = 1*(license-id / license-id"+" / license-ref)

idstring = 1*(ALPHA / DIGIT / "-" / "+" / "." )
license-ref = "LicenseRef-"[idString]
license-id = <short form license identifier in Appendix I.1>
license-exception-id = <short form license exception identifier in Appendix I.2>
```

In the following sections we describe in more detail **<SPDX-License-Expression>** construct, a licensing expression string that enables a more accurate representation of the licensing terms of modern day software. A valid **<SPDX-License-Expression>** string consists of either:

- (i) a simple licensed expression, such as a single license identifier; or
- (ii) a more complex expression constructed by combining smaller valid expressions using Boolean license operators.

Simple License Expressions

A simple **<SPDX-License-Expression>** is composed one of the following:

- An **SPDX License List Short Form Identifier**. For example:

<SPDX-License-Expression-1> = GPL-2.0

- An **SPDX License List Short Form Identifier** with a unary "+" operator suffix to represent the current version of the license or any later version:

<SPDX-License-Expression-1> = GPL-2.0+

- A SPDX user defined license reference, denoted by **LicenseRef-[idString]**. For example:

<SPDX-License-Expression-1> = LicenseRef-23

<SPDX-License-Expression-2> = LicenseRef-MIT-Style-1

Composite License Expressions

More expressive composite license expressions can be constructed using "OR," "AND," and "WITH" operators similar to constructing mathematical expressions using arithmetic operators. For the tag/value format, any license expression that consists of more than one license identifier and/or LicenseRef, should be encapsulated by parentheses: (). This has been specified to facilitate expression parsing. Nested parentheses can also be used to specify an order of precedence which is discussed in more detail in subsection (4).

1) Disjunctive OR Operator

If presented with a choice between two or more licenses, use the disjunctive binary "OR" operator to construct a new **<SPDX-License-Expression>**, where both the left and right operands are valid **<SPDX-License-Expression>** values. The syntax for the "OR" operator is:

<SPDX-License-Expression-3> = (<SPDX-License-Expression-1> OR <SPDX-License-Expression-2>)

For example, when given a choice between the LGPL-2.1 or MIT licenses, a valid expression would be: **(LGPL-2.1 OR MIT)**, where:

<SPDX-License-Expression-1> = LGPL-2.1
<SPDX-License-Expression-2> = MIT
<SPDX-License-Expression-3> = (LGPL-2.1 OR MIT)

An example representing a choice between three different licenses would be: **(LGPL-2.1 OR MIT OR BSD-3-Clause)**, where:

<SPDX-License-Expression-1> = LGPL-2.1
<SPDX-License-Expression-2> = MIT
<SPDX-License-Expression-3> = BSD-3-Clause
<SPDX-License-Expression-4> = (LGPL-2.1 OR MIT OR BSD-3-Clause)

2) Conjunctive AND Operator

If required to simultaneously comply with two or more licenses, use the conjunctive binary "AND" operator to construct a new **<SPDX-License-Expression>**, where both the left and right operands are a valid **<SPDX-License-Expression>** values. The syntax for the "AND" operator is:

<SPDX-License-Expression-3> = (<SPDX-License-Expression-1> AND <SPDX-License-Expression-2>)

For example, when one is required to comply with both the LGPL-2.1 AND MIT licenses, a valid conjunctive expression would be: **LGPL-2.1 AND MIT**, where:

<SPDX-License-Expression-1> = LGPL-2.1
<SPDX-License-Expression-2> = MIT
<SPDX-License-Expression-3> = LGPL-2.1 AND MIT

An example where three licenses apply would be: **LGPL-2.1 AND MIT AND BSD-2-Clause**, where:

<SPDX-License-Expression-1> = LGPL-2.1
<SPDX-License-Expression-2> = MIT
<SPDX-License-Expression-3> = BSD-2-Clause
<SPDX-License-Expression-4> = (LGPL-2.1 AND MIT AND BSD-2-Clause)

3) Exception WITH Operator

Sometimes a set of license terms apply except under special circumstances. In this case, use the binary “**WITH**” operator to construct a new **<SPDX-License-Expression>** to represent the special exception situation. A valid license expression is where the left operand is a valid **<SPDX-License-Expression>** value and the right operand is an **SPDX License List Exception Identifier (SPDX LL Exception Identifier)** that represents the special exception terms. SPDX License LL Identifiers can be found on the SPDX License List. The syntax for the “**WITH**” operator is:

<SPDX-License-Expression-2> = (<SPDX-License-Expression-1> WITH <SPDX-LL-Exception-Identifier>)

For example, when the Bison exception is to be applied to GPL-2.0+, the expression would be: **(GPL-2.0+ WITH Bison-exception-2.2)**, where:

<SPDX-License-Expression-1> = GPL-2.0+
<SPDX-LL-Exception-Identifier> = Bison-exception-2.2
<SPDX-License-Expression-2> = (GPL-2.0+ WITH Bison-exception-2.2)

If the applicable exception is not found on the SPDX License List, then use **LicenseRef-[idString]** to express the license and exception.

4) Order of Precedence and Parentheses

The order of application of the operators in an expression matters (similar to mathematical operators). The default operator order of precedence of an **<SPDX-License-Expression>** is:

1. +
2. WITH
3. AND
4. OR

where a lower order operator is applied before a higher order operator. For example, the following expression:

(LGPL-2.1 OR BSD-3-Clause AND MIT)

represents a license choice between either **LGPL-2.1** and the expression **BSD-3-Clause AND MIT** because the **AND** operator takes precedence over (is applied before) the **OR** operator.

When required to express an order of precedence that is different from the default order an **<SPDX-License-Expression>** can be encapsulated in one or more pairs of parentheses: **()**, to indicate that the operators found inside the parentheses takes precedence over operators outside. This is also similar to the use of parentheses in an algebraic expression e.g., **(5+7)/2**. For instance, the following expression:

((LGPL-2.1+ OR BSD-3-Clause) AND MIT)

states the **OR** operator should be applied before the **AND** operator. That is, one should first select between the LGPL-2.1+ and the BSD-3-Clause license before applying the MIT license.

Appendix V: Creative Commons Attribution License 3.0 Unported

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

- a. **"Adaptation"** means a work based upon the Work, or upon the Work and other pre-existing works, such as a translation, adaptation, derivative work, arrangement of music or other alterations of a literary or artistic work, or phonogram or performance and includes cinematographic adaptations or any other form in which the Work may be recast, transformed, or adapted including in any form recognizably derived from the original, except that a work that constitutes a Collection will not be considered an Adaptation for the purpose of this License. For the avoidance of doubt, where the Work is a musical work, performance or phonogram, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered an Adaptation for the purpose of this License.
- b. **"Collection"** means a collection of literary or artistic works, such as encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(f) below, which, by reason of the selection and arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined above) for the purposes of this License.
- c. **"Distribute"** means to make available to the public the original and copies of the Work or Adaptation, as appropriate, through sale or other transfer of ownership.
- d. **"Licensor"** means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.
- e. **"Original Author"** means, in the case of a literary or artistic work, the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers, musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organization that transmits the broadcast.
- f. **"Work"** means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works

expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.

- g. **"You"** means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.
- h. **"Publicly Perform"** means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.
- i. **"Reproduce"** means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.

2. Fair Dealing Rights. Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

- a. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections;
- b. to create and Reproduce Adaptations provided that any such Adaptation, including any translation in any medium, takes reasonable steps to clearly label, demarcate or otherwise identify that changes were made to the original Work. For example, a translation could be marked "The original work was translated from English to Spanish," or a modification could indicate "The original work has been modified.";
- c. to Distribute and Publicly Perform the Work including as incorporated in Collections; and,
- d. to Distribute and Publicly Perform Adaptations.
- e. For the avoidance of doubt:
 - i. **Non-waivable Compulsory License Schemes.** In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;
 - ii. **Waivable Compulsory License Schemes.** In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor waives the exclusive right to collect such royalties for any exercise by You of the rights granted under this License; and,
 - iii. **Voluntary License Schemes.** The Licensor waives the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License.

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. Subject to Section 8(f), all rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

- a. You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. This Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this License. If You create a Collection, upon notice from any Licensor You must, to the extent practicable, remove from the Collection any credit as required by Section 4(b), as requested. If You create an Adaptation, upon notice from any Licensor You must, to the extent practicable, remove from the Adaptation any credit as required by Section 4(b), as requested.
- b. If You Distribute, or Publicly Perform the Work or any Adaptations or Collections, You must, unless a request has been made pursuant to Section 4(a), keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or if the Original Author and/or Licensor designate another party or parties (e.g., a sponsor institute, publishing entity, journal) for attribution ("Attribution Parties") in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; (ii) the title of the Work if supplied; (iii) to the extent reasonably practicable, the URI, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and (iv) , consistent with Section 3(b), in the case of an Adaptation, a credit identifying the use of the Work in the Adaptation (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). The credit required by this Section 4 (b) may be implemented in any reasonable manner; provided, however, that in the case of a Adaptation or Collection, at a minimum such credit will appear, if a credit for all contributing authors of the Adaptation or Collection appears, then as part of these credits and in a manner at least as prominent as the credits for the other contributing authors. For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this License, You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties.
- c. Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law, if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any Adaptations or Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author's honor or reputation. Licensor agrees that in those jurisdictions (e.g. Japan), in which any exercise of the right granted in Section 3(b) of this License (the right to make Adaptations) would be deemed to be a distortion, mutilation, modification or other derogatory action prejudicial to the Original Author's honor and reputation, the Licensor will waive or not assert, as appropriate, this Section, to the fullest extent permitted by the applicable national law, to enable You to reasonably exercise Your right under Section 3(b) of this License (right to make Adaptations) but not otherwise.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

- a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Adaptations or Collections from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.
- b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

- a. Each time You Distribute or Publicly Perform the Work or a Collection, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.
- b. Each time You Distribute or Publicly Perform an Adaptation, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.
- c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.
- d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.
- e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.
- f. The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention for the Protection of Literary and Artistic Works (as amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be

included in the License; this License is not intended to restrict the license of any rights under applicable law.