Data Schema

# Description

This document describes the SPDX-based data schema used in FOSSology-Ninka output.

# Changelog

|  |  |  |
| --- | --- | --- |
| Date | Action | Who |
| 02/26/2014 | Document Creation | Doug Richardson |
| 02/26/2014 | Changed document format and uploaded to Github. | James Thompson |
| 02/26/2014 | Added document description; reorganized sections | Jon von Kampen |

# Data Schema

| Class | Member | Description |
| --- | --- | --- |
| Spdx\_Document |  |  |
|  | spdx\_version | The version of the SPDX document (ours will be 1.2) |
|  | data\_license | The license the SPDX document is under (I am thinking Creative Commons) |
|  | document\_comment | Comments on the document proper |
| Creator\_Information |  |  |
|  | creator | The creator of the document (ours will always be Tool: Fossology-Ninka) |
|  | created | When the document was created, will be in YYYY-MM-DDThh:mm:ssZ format with leading zeroes when appropriate. |
|  | creator\_comment | Comments upon the creation of the document |
|  | license\_list\_version | The version of the license list used (currently 1.19) |
| Package |  |  |
|  | package\_name | The name of the package used |
|  | package\_version | The version of the package used |
|  | package\_file\_name | The full file name of the package used (for example, ninka-1.1.tar.bz2) |
|  | package\_supplier | The supplier of the package (i.e. sourceforge) |
|  | package\_originator | The originator of the package used (may be different then the supplier) |
|  | package\_download\_location | Where the package was downloaded from (beyond the scope of our scanner) |
|  | package\_verification\_code | A combined SHA-1 checksum of EACH FILE IN THE PACKAGE (excluding the SPDX-file) |
|  | package\_checksum | A checksum for the package as a holistic entity (SHA-1 format) |
|  | package\_home\_page | The home page of the package (beyond the scope of our scanner) |
|  | source\_information | Any other relevant background about the package (beyond the scope of our scanner) |
|  | package\_license\_concluded | HUMAN CONFIRMED licenses within the package (obviously beyond the scope of our scanner) |
|  | All\_license\_info\_from\_files | All the license info from files (doesn’t need to indicate the file itself, just the license with it, goes line by line for the licenses) |
|  | package\_license\_declared | The licenses the package claims to have (this is where our scanner’s findings will generally be) |
|  | package\_license\_comments | Comments on the declared license (if there are conflicts, we will indicate them in here) |
|  | package\_copyright\_text | This is the copyright portion declared within the package (this is beyond the scope of our scanner, but it can be found with FOSSOLOGY) |
|  | package\_summary | A short description about the package (beyond the scope of our scanner) |
|  | package \_description | A more detailed description of the package (beyond the scope of our scanner) |
| Other License Information |  |  |
|  | license\_ID | The reference id of the license involved (i.e. LicenseRef-1) |
|  | extracted\_text | The text portion of the license, used for future reference |
|  | license\_name | The name of the extracted license (for example, Aphero GPL version 3) |
|  | license\_cross\_reference | Where one can look up this license |
|  | license\_comment | Comments on this extracted license |
| File\_Information |  |  |
|  | file\_name | The FULL PATH of the file name scanned (i.e. /splitter/splitter.pl) |
|  | file\_type | Valid options are SOURCE, BINARY, ARCHIVE, or OTHER |
|  | file\_checksum | A SHA-1 checksum of the file proper |
|  | concluded\_license | A HUMAN-VERIFIED license or licenses of the file (will always be blank) |
|  | license\_info\_on\_file | The “license declared” part of the file (again, this is where our scanner puts its output) |
|  | file\_license\_comments | Comments of the license of the file (if conflicts occur, this is where the output is) |
|  | file\_copyright\_text | The copyright information on the individual file (can be found with FOSSOLOGY, but beyond the scope of our project) |
|  | artifact\_of\_project\_name | If this file is an artifact of a specific file (will always be blank for our scanner) |
|  | artifact\_of\_project\_homepage | The homepage of the project this file is a part of (beyond the scope of our scanner, will always be blank) |
|  | artifact\_of\_project\_URI | The URI of the project the file is a part of (beyond the scope of our scanner, will always be blank) |
|  | file\_comment | Comments on the specific file (may be there if the scanner turns up an error, otherwise beyond the scope of our scanner) |
|  | file\_notice | A legal notice found in the file (beyond the scope of our scanner) |
|  | file\_contributor | A notice as to who contributed the file for scanning (beyond the scope of our scanner). |
|  | file\_dependencies | Dependencies the file has (beyond the scope of our scanner) |

## Handling Out-of-scope Fields

If something is indicated to be beyond the scope of the scanner, the field will always be left blank. It will be there for the sake of a complete document spec and data, but the scanner itself can’t do anything in that regard. It will be included in the schema in case an update is made that will allow these fields to be manipulated through this tool.