NLUUG 2024

# Software Bill of Materials

Alexios Zavras

# Alexios Zavras – about me

- Greek, living in Munich
- Intel's Chief Open Source Compliance Officer, working at Open Source Program Office
- "Open Source" since 1983

- First NLUUG event in 1988
- Involved in SBOM/SPDX since 2011

# Software is complex

- Nowadays almost always a combination of components
- 80 – 20 rule

# Software Bill of Materials (SBOM)

An SBOM is a formal record containing details and supply chain relationships of components used in building software.

- Components include libraries and modules
- Components can be open source or proprietary
- Components can be freely available or paid
- Data can be widely available or access-restricted

# Who should use an SBOM?

- Any organization concerned about better supporting their software products internally and better supporting their customers

- Different views
  - Produce / Consume (Use / Integrate)

- Commonly required as part of any product's BOM, so necessary information is available:
  - Contractual – negotiated terms, implementation strategies
  - Legal – compliance with licensing and regulatory obligations
  - Technical – identification of software or component dependencies and supply chain risk, vulnerability and asset management

# Why have an SBOM?

- Legal compliance
  - License obligations, Open Source or not
  - Comply with *all* obligations of *all* licenses of *all* components
    - Straightforward
    - But not trivial or easy
- Export
- Security

# Why have an SBOM?

- Legal compliance
- Export
- **Security**
  - NTIA, FDA, NERC, ENISA

# Do YOU know...

... whether you are affected by $VULNERABILITY?




# Do YOU know...

## ... how to detect and remediate complex attacks?

SUPPLY CHAIN ATTACK
Attackers insert malicious code into a DLL component of legitimate software. The compromised DLL is distributed to organizations that use the related software.

EXECUTION, PERSISTENCE
When the software starts, the compromised DLL loads, and the inserted malicious code calls the function that contains the backdoor capabilities.

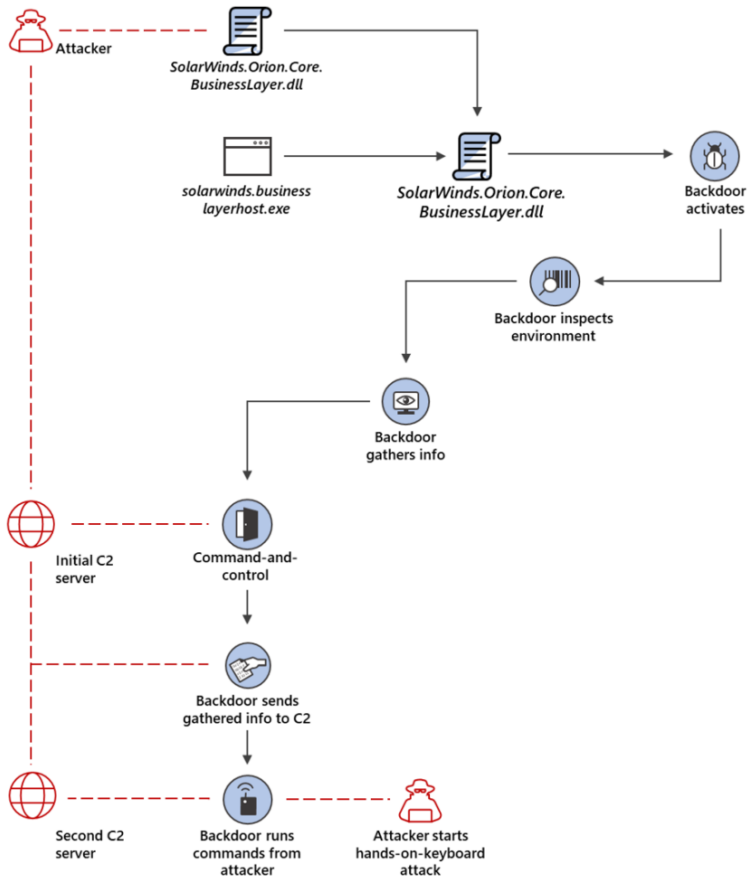
DEFENSE EVASION
The backdoor has a lengthy list of checks to make sure it's running in an actual compromised network.

RECON
The backdoor gathers system info

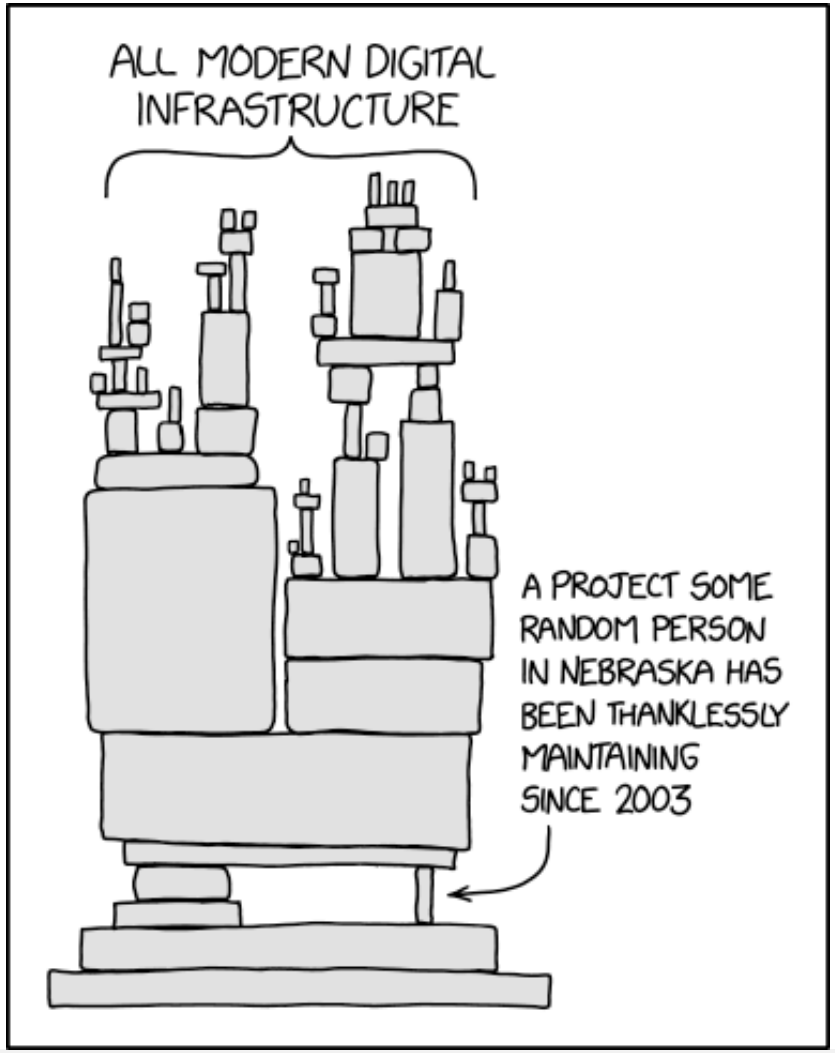
INITIAL C2
The backdoor connects to a command-and-control server. The domain it connects to is partly based on info gathered from system, making each subdomain unique. The backdoor may receive an additional C2 address to connect to.

EXFILTRATION
The backdoor sends gathered information to the attacker.

HANDS-ON-KEYBOARD ATTACK
The backdoor runs commands it receives from attackers. The wide range of backdoor capabilities allow attackers to perform additional activities, such as credential theft, progressive privilege escalation, and lateral movement.



# Most do not know what software is running




Dependency, by xkcd, CC-BY-NC-2.5

# Regulation

- Regulation is ~~coming~~ here!

- US
  - EO 14028 on Improving the Nation's Cybersecurity; May 2021
  - National Cybersecurity Strategy Implementation Plan; July 2023
- EU
  - Cyber Resilience Act (CRA); December 2023
- Germany
- Japan
- …

# Need for a Bill of Materials

A comprehensive list of software components,
with information on:

| | | |
|---|---|---|
| ▪ Name | zlib | gcc |
| ▪ License | Zlib license | GPLv3 |
| ▪ Version | 1.3 | 13.2 |
| ▪ Origin | https://zlib.net | https://gcc.gnu.org |
| ▪ … | | “not modified” |

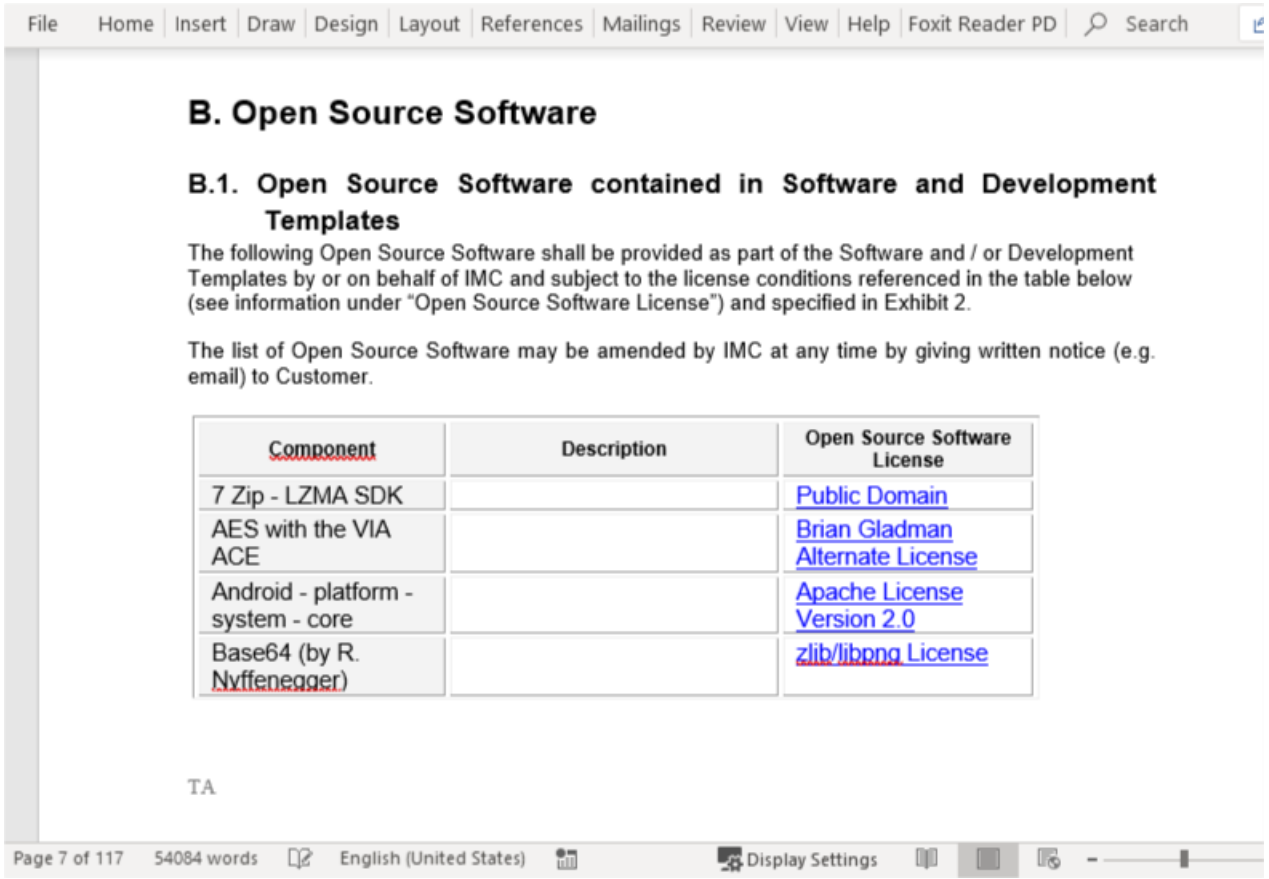
# Contents of a minimum viable SBOM

| Data Field | Description |
|---|---|
| **Supplier Name** | The name of an entity that creates, defines, and identifies components. |
| **Component Name** | Designation assigned to a unit of software defined by the original supplier. |
| **Version of the Component** | Identifier used by the supplier to specify a change in software from a previously identified version. |
| **Other Unique Identifiers** | Other identifiers that are used to identify a component, or serve as a look-up key for relevant databases. |
| **Dependency Relationship** | Characterizing the relationship that an upstream component X is included in software Y. |
| **Author of SBOM Data** | The name of the entity that creates the SBOM data for this component. |
| **Timestamp** | Record of the date and time of the SBOM data assembly. |

The Minimum Elements for an SBOM, by US Department of Commerce

# How to deliver this information?

# How to deliver this information?



File Home Insert Draw Design Layout References Mailings Review View Help Foxit Reader PD Search

## B. Open Source Software

### B.1. Open Source Software contained in Software and Development Templates

The following Open Source Software shall be provided as part of the Software and / or Development Templates by or on behalf of IMC and subject to the license conditions referenced in the table below (see information under "Open Source Software License") and specified in Exhibit 2.

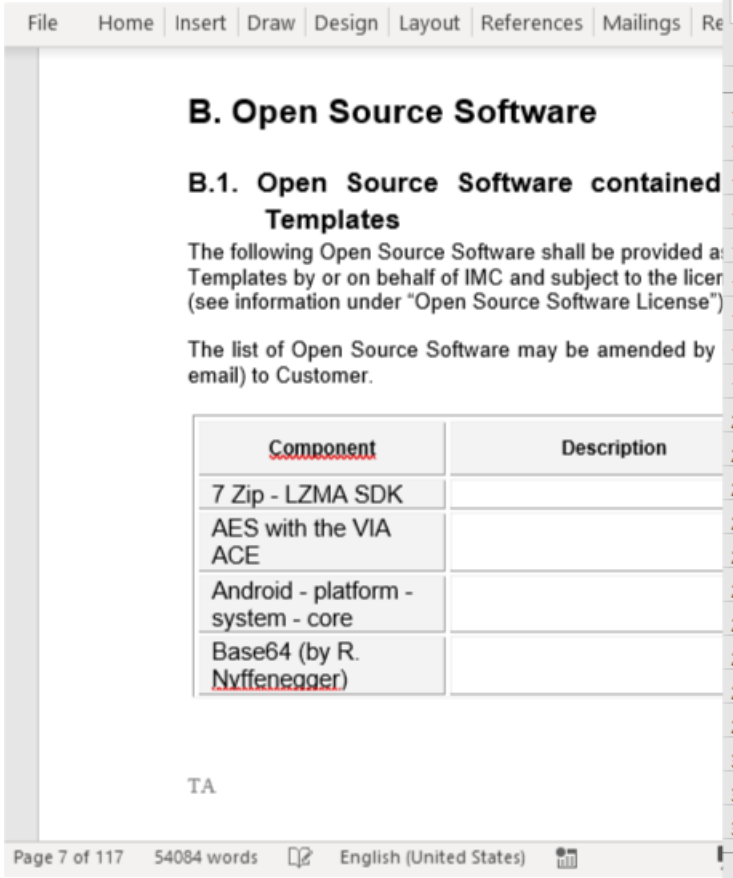
The list of Open Source Software may be amended by IMC at any time by giving written notice (e.g. email) to Customer.

| Component | Description | Open Source Software License |
|---|---|---|
| 7 Zip - LZMA SDK | | Public Domain |
| AES with the VIA ACE | | Brian Gladman Alternate License |
| Android - platform - system - core | | Apache License Version 2.0 |
| Base64 (by R. Nyffenegger) | | zlib/libpng License |

TA

Page 7 of 117 54084 words English (United States) Display Settings

# How to deliver this information?



File Home Insert Draw Design Layout References Mailings Re

## B. Open Source Software

### B.1. Open Source Software contained Templates

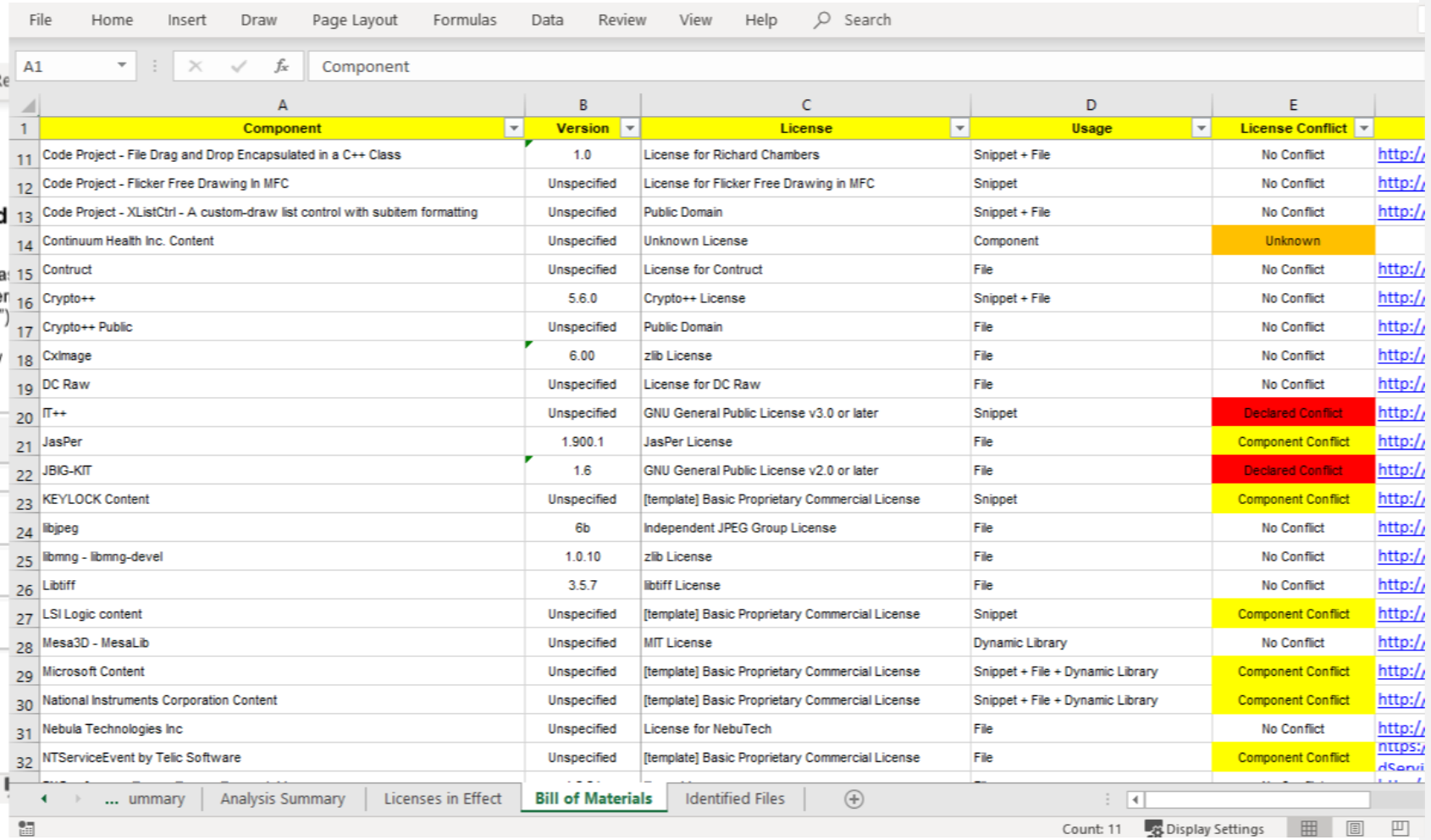
The following Open Source Software shall be provided a
Templates by or on behalf of IMC and subject to the licer
(see information under "Open Source Software License")

The list of Open Source Software may be amended by
email) to Customer.

| Component | Description |
|---|---|
| 7 Zip - LZMA SDK | |
| AES with the VIA ACE | |
| Android - platform - system - core | |
| Base64 (by R. Nyffenegger) | |

TA

Page 7 of 117 54084 words English (United States)



File Home Insert Draw Page Layout Formulas Data Review View Help Search

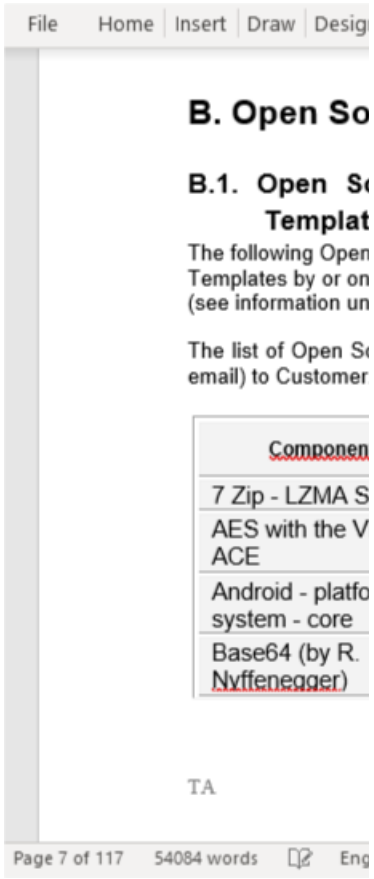
A1 Component

| | A | B | C | D | E | |
|---|---|---|---|---|---|---|
| 1 | Component | Version | License | Usage | License Conflict | |
| 11 | Code Project - File Drag and Drop Encapsulated in a C++ Class | 1.0 | License for Richard Chambers | Snippet + File | No Conflict | http:// |
| 12 | Code Project - Flicker Free Drawing In MFC | Unspecified | License for Flicker Free Drawing in MFC | Snippet | No Conflict | http:// |
| 13 | Code Project - XListCtrl - A custom-draw list control with subitem formatting | Unspecified | Public Domain | Snippet + File | No Conflict | http:// |
| 14 | Continuum Health Inc. Content | Unspecified | Unknown License | Component | Unknown | |
| 15 | Contruct | Unspecified | License for Contruct | File | No Conflict | http:// |
| 16 | Crypto++ | 5.6.0 | Crypto++ License | Snippet + File | No Conflict | http:// |
| 17 | Crypto++ Public | Unspecified | Public Domain | File | No Conflict | http:// |
| 18 | CxImage | 6.00 | zlib License | File | No Conflict | http:// |
| 19 | DC Raw | Unspecified | License for DC Raw | File | No Conflict | http:// |
| 20 | IT++ | Unspecified | GNU General Public License v3.0 or later | Snippet | Declared Conflict | http:// |
| 21 | JasPer | 1.900.1 | JasPer License | File | Component Conflict | http:// |
| 22 | JBIG-KIT | 1.6 | GNU General Public License v2.0 or later | File | Declared Conflict | http:// |
| 23 | KEYLOCK Content | Unspecified | [template] Basic Proprietary Commercial License | Snippet | Component Conflict | http:// |
| 24 | libjpeg | 6b | Independent JPEG Group License | File | No Conflict | http:// |
| 25 | libmng - libmng-devel | 1.0.10 | zlib License | File | No Conflict | http:// |
| 26 | Libtiff | 3.5.7 | libtiff License | File | No Conflict | http:// |
| 27 | LSI Logic content | Unspecified | [template] Basic Proprietary Commercial License | Snippet | Component Conflict | http:// |
| 28 | Mesa3D - MesaLib | Unspecified | MIT License | Dynamic Library | No Conflict | http:// |
| 29 | Microsoft Content | Unspecified | [template] Basic Proprietary Commercial License | Snippet + File + Dynamic Library | Component Conflict | http:// |
| 30 | National Instruments Corporation Content | Unspecified | [template] Basic Proprietary Commercial License | Snippet + File + Dynamic Library | Component Conflict | http:// |
| 31 | Nebula Technologies Inc | Unspecified | License for NebuTech | File | No Conflict | http:// |
| 32 | NTServiceEvent by Telic Software | Unspecified | [template] Basic Proprietary Commercial License | File | Component Conflict | https:// |

... ummary | Analysis Summary | Licenses in Effect | Bill of Materials | Identified Files

Count: 11 Display Settings

# How to deliver this information?




```
439.    vue-property-decorator (8.3.0, MIT, https://github.com/kaorun343/vue-property-decorator)
440.    vue-router (3.1.5, MIT, https://github.com/vuejs/vue-router)
441.    watchpack (1.6.0, MIT, https://github.com/webpack/watchpack)
442.    webpack (4.41.5, MIT, https://github.com/webpack/webpack)
443.    webpack-sources (1.4.3, MIT, https://github.com/webpack/webpack-sources)
444.    websocket-client (0.56.0, BSD, https://github.com/websocket-client/websocket-client.git)
445.    which (1.3.1, ISC, https://github.com/isaacs/node-which)
446.    word-wrap (1.2.3, MIT, https://github.com/jonschlinkert/word-wrap)
447.    worker-farm (1.7.0, MIT, https://github.com/rvagg/node-worker-farm)
448.    wrappy (1.0.2, ISC, https://github.com/npm/wrappy)
449.    write (1.0.3, MIT, https://github.com/jonschlinkert/write)
450.    xmltodict (0.12.0, MIT, https://github.com/martinblech/xmltodict)
451.    xtend (4.0.2, MIT, https://github.com/Raynos/xtend)
452.    y18n (4.0.0, ISC, https://github.com/yargs/y18n)
453.    yallist (3.1.1, ISC, https://github.com/isaacs/yallist)



# 0.    @babel/code-frame License Info Follows

        MIT License

        Copyright (c) 2014-present Sebastian McKenzie and other contributors

        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
```

# SPDX

## ~~Software~~ System Package Data Exchange

# System Package Data Exchange – ISO/IEC 5962:2021

Standards for communicating the component and metadata information associated with software

- Specification
- License List
- Tools


Working groups:

- Technical
- Legal
- Outreach

# SPDX License List

List of (common) Open Source licenses

- Currently more than 650 licenses and 65 exceptions
- For each one, several data:
  - name, short identifier, canonical license text, reference URL, is OSI approved, is FSF libre, standard header text

Matching guidelines to determine if text matches license text

- Canonical license text is templatized

# SPDX License List short identifiers

Authoritative list of names and short identifiers

- MIT, BSD-3-Clause, GPL-2.0-or-later, ...

- Expressions

```
GPL-2.0-only OR BSD-3-Clause

EPL-2.0 OR MPL-2.0
```

# Use of SPDX identifiers in source files

```
SPDX-License-Identifier: Apache-2.0
```

- Easy to use, machine-readable
  - Just adds one comment line
  - Makes it easy to know the license for a file
  - Satisfies the DCO requirement for a license reference per file

- Concise standard format

# SPDX Documents

Collecting all information about a software delivery

- Descriptive
  - Detailed Bill of Materials (aka manifest) of the software contents
- Flexible
  - Formats for automatic processing (XML, JSON, YAML), for manual editing (tag:value), and for non-technical (spreadsheet)
- Accurate
  - Focus on capturing facts; allow interpretations

# Example SPDXv2 Document

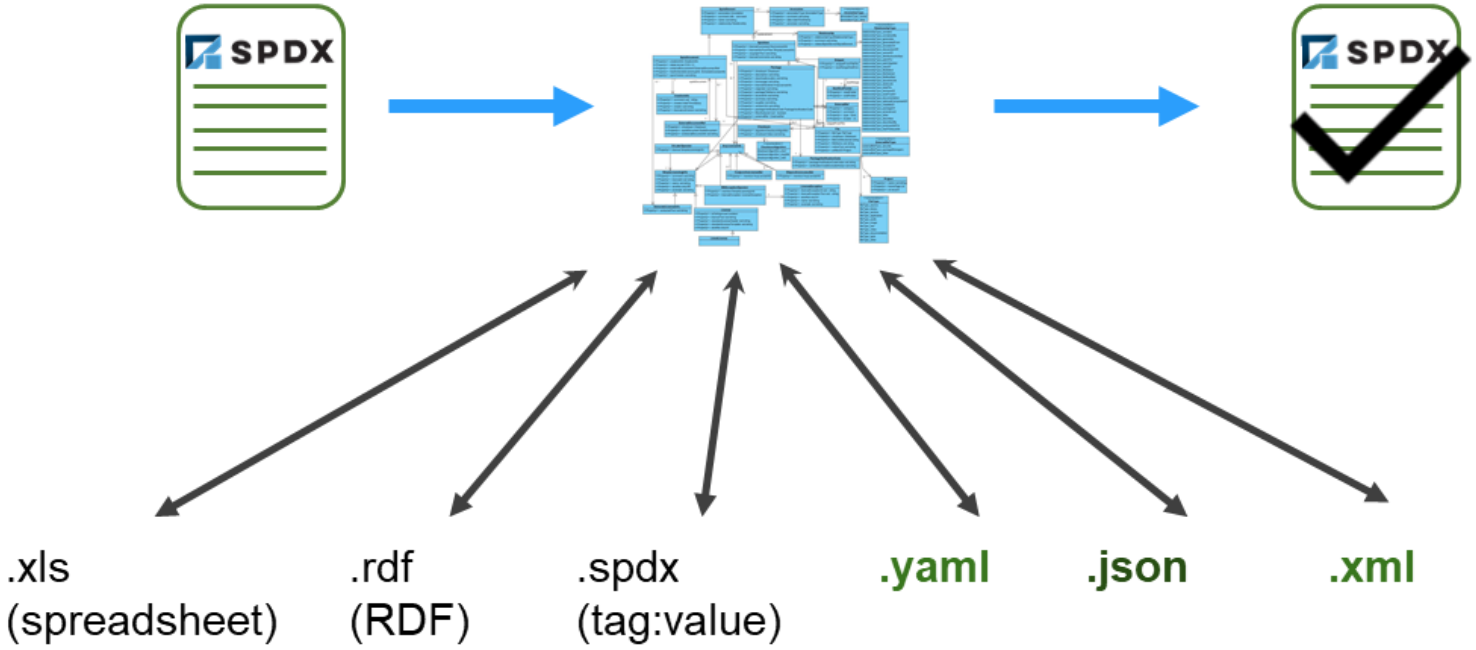
```
DocumentName: SPDX-DemoSoftware-v2.0
SPDXID: SPDXRef-DOCUMENT
DocumentComment: <text>This document was
created using SPDX 2.0</text>

## Creation Information
Creator: Person: Jane Doe
(jane.doe@corp.com)
Creator: Organization: Big Company
Creator: Tool: LicenseFind-1.0
Created: 2022-11-01T18:30:22Z
LicenseListVersion: 3.18
```

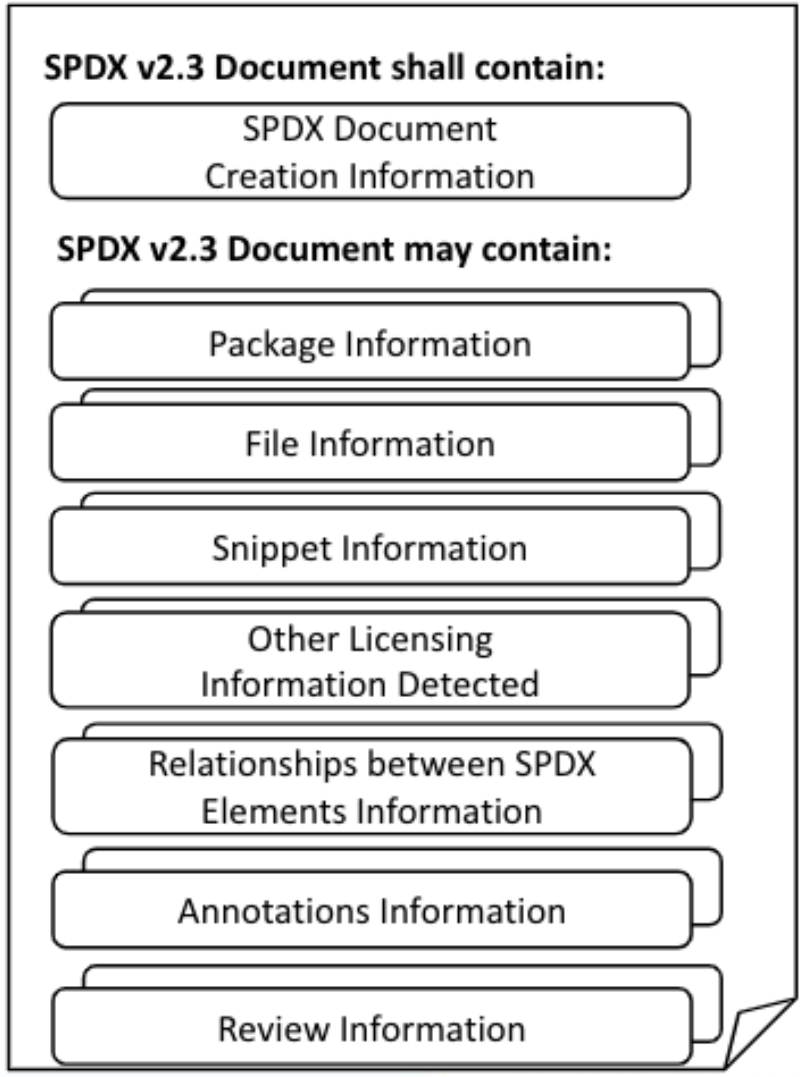
```
FileName: ./package/foo.c
FileType: TEXT
FileChecksum: SHA1:
d6a770ba38583ed4bb4525bd96e50461655d2758
LicenseInfoInFile: Apache-2.0
LicenseInfoInFile: LicenseRef-1
LicenseConcluded: Apache-2.0
FileCopyrightText: <text>Copyright 2008-
2015 John Smith</text>
FileComment: <text>The concluded license
was taken from the package; the info was
found in the COPYING.txt file in the top-
level directory.</text>

LicenseID: LicenseRef-1
ExtractedText: <text>This software…
```

# Model supports diverse file formats



# Structure of an SPDXv2 Document

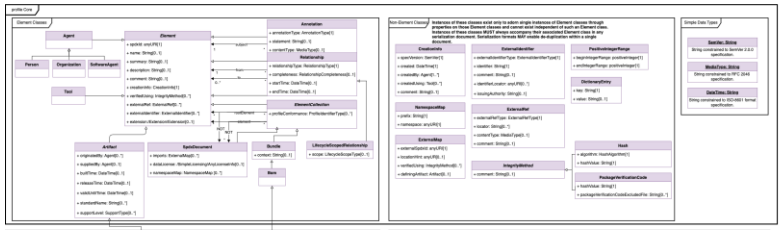


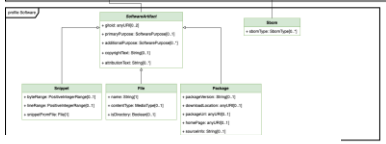
# Released: SPDX 3.0

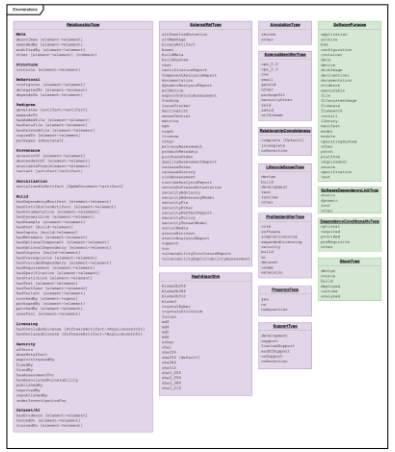
- Major undertaking
- Abstracted information to be more widely useful
- Refactored to CORE and PROFILEs
  - CORE is minimum needed to describe artifacts and relationships
  - PROFILEs for each Area of Interest:
    Licensing, Vulnerabilities, Provenance, …

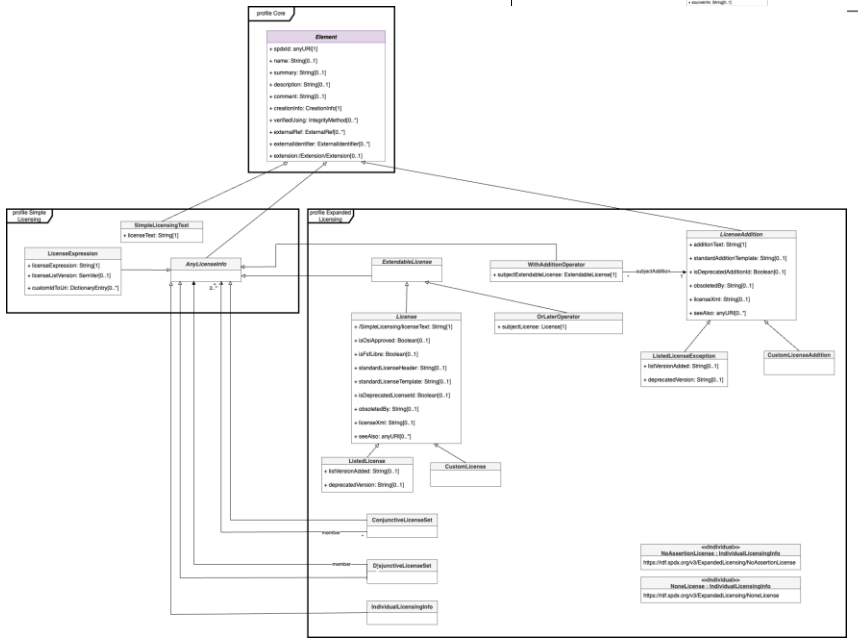
- (Finally) Released in April!

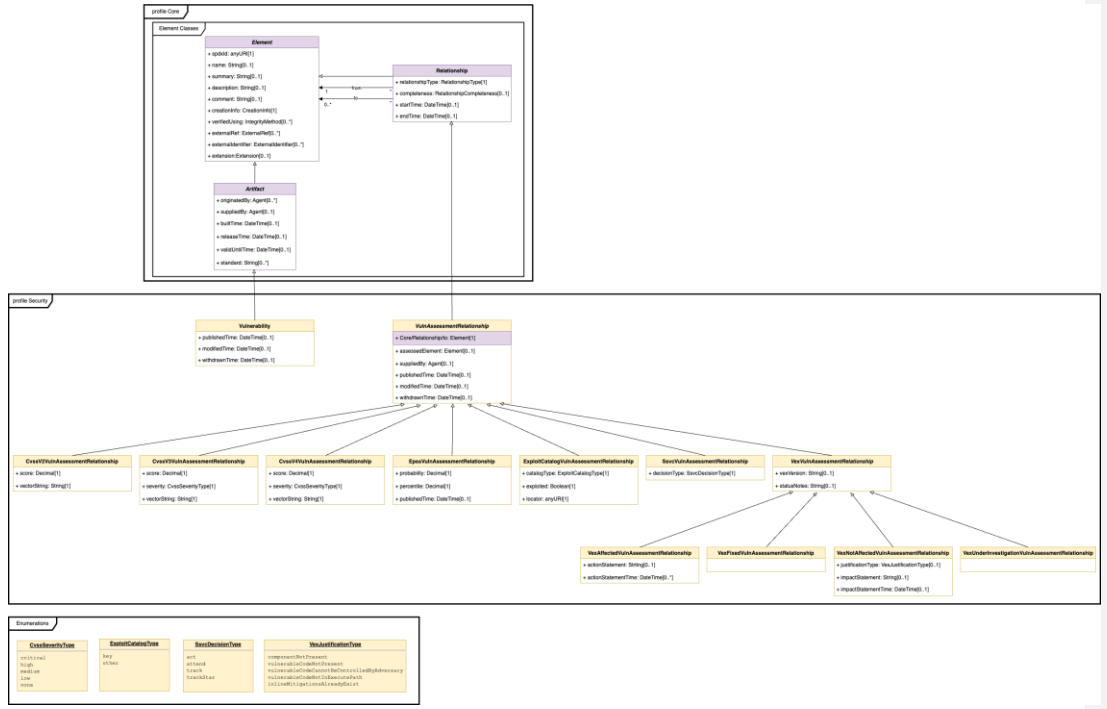
# SPDXv3 is graph-based data











# SPDXv3 Profiles

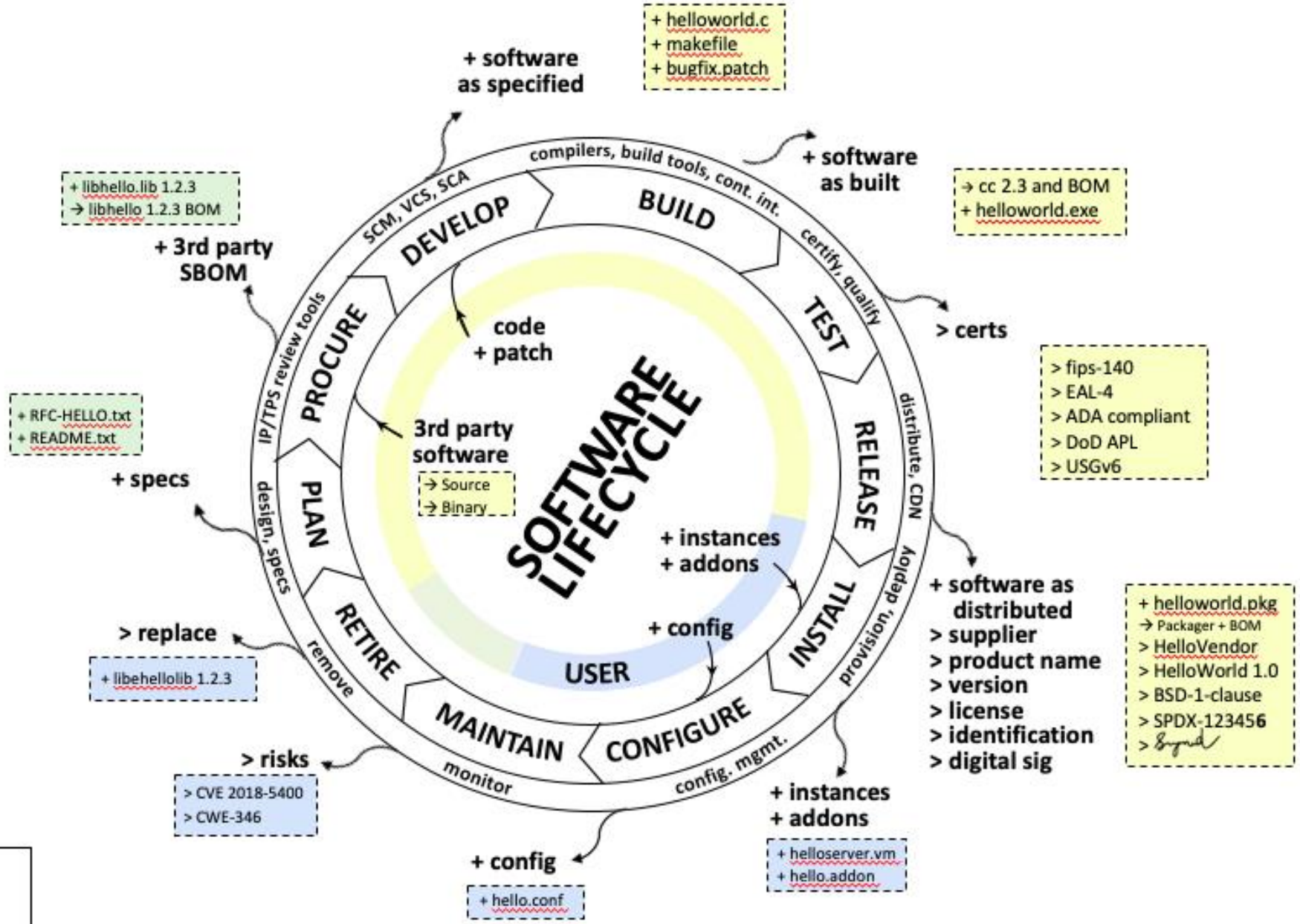
- Core, Software
- Licensing
- Security
- Build
- AI / Dataset

- In progress: FuSa, Operations, SaaS, Hardware, ...

# However...

## Real life is not that simple

# Software Lifecycle & Bill of Materials Generation



# Types of SBOMs

| | |
|---|---|
| Design | SBOM of intended, planned software project or product with included components (some of which may not yet exist) for a new software artifact. |
| Source | SBOM created directly from the development environment, source files, and included dependencies used to build a product artifact. |
| Build | SBOM generated as part of the process of building the software to create a releasable artifact (e.g., executable or package) from data such as source files, dependencies, built components, build process ephemeral data, and other SBOMs. |
| Analyzed | SBOM generated through analysis of artifacts (e.g., executables, packages, containers, and virtual machine images) after its build. Such analysis generally requires a variety of heuristics. In some contexts, this may also be referred to as a "third-party" SBOM. |
| Deployed | SBOM provides an inventory of software that is present on a system. This may be an assembly of other SBOMs that combines analysis of configuration options, and examination of execution behavior in a (potentially simulated) deployment environment. |
| Runtime | SBOM generated through instrumenting the system running the software, to capture only components present in the system, as well as external call-outs or dynamically loaded components. In some contexts, this may also be referred to as an "Instrumented" or "Dynamic" SBOM. |

# Tools

# Tool functional classification taxonomy

| Category | Type | Description |
|---|---|---|
| Produce | Build | SBOM is automatically created as part of building a software artifact and contains information about the build |
| | Analyze | Analysis of source or binary files will generate the SBOM by inspection of the artifacts and any associated sources |
| | Edit | A tool to assist a person manually entering or editing SBOM data |
| Consume | View | Be able to understand the contents in human readable form (e.g., picture, figures, tables, text, etc.). Use to support decision making & business processes |
| | Diff | Be able to compare multiple SBOMs and clearly see the differences (e.g., comparing two versions of a piece of software) |
| | Import | Be able to discover, retrieve, and import an SBOM into your system for further processing and analysis |
| Transform | Translate | Change from one file type to another file type while preserving the same information |
| | Merge | Multiple sources of SBOM and other data can be combined together for analysis and audit purposes |
| | Tool support | Support use in other tools by APIs, object models, libraries, transport, or other reference sources |

# Tools classifications

- Licensed under:
  - Open Source
  - Proprietary
- SBOM Type
- List keeps expanding…
- Level:
  - Libraries
  - Purpose-specific
  - Complete applications
  - Integrated environments
- Ecosystem

# SPDX: Open for participation!

## To everyone

# Participate!

Teams

- Technical
- Legal
- Outreach

- Mailing lists
- Meetings
- GitHub

Groups

- AI
- Build
- Data
- Defects
- Functional Safety
- Hardware
- …

All information on https://spdx.dev and https://github.com/spdx


intel