OPENCHAIN JAPAN COMMUNITY DAY #32
メインセッション② - 業界最新動向の共有 -

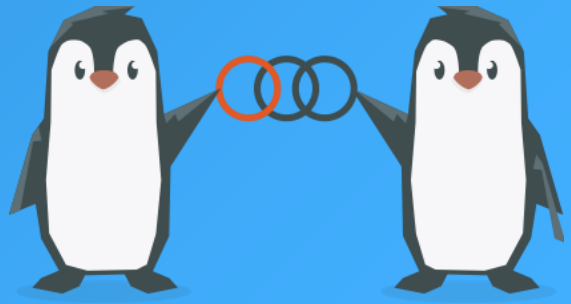
SBOM 動向

(16:10 pm - 16:30 pm)

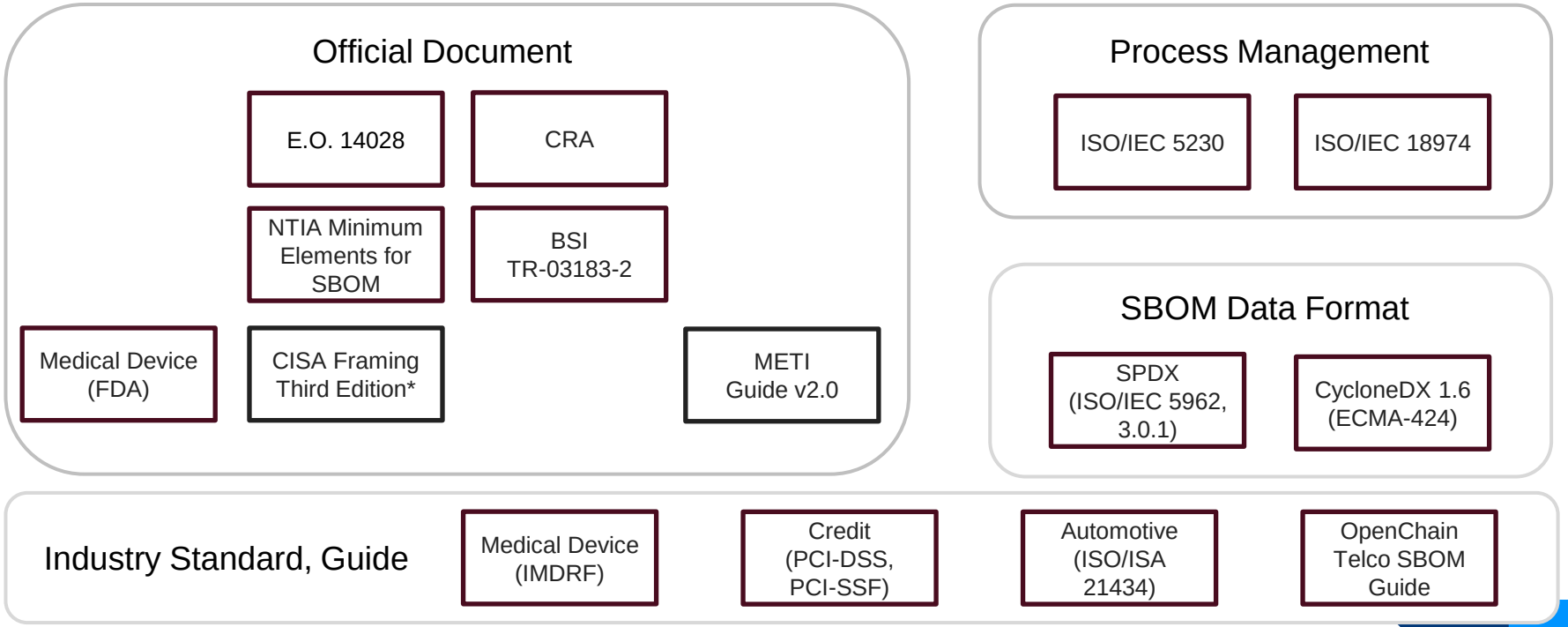
SBOM Sub Group,
OpenChain Japan Work Group

THE LINUX FOUNDATION

OPENCHAIN

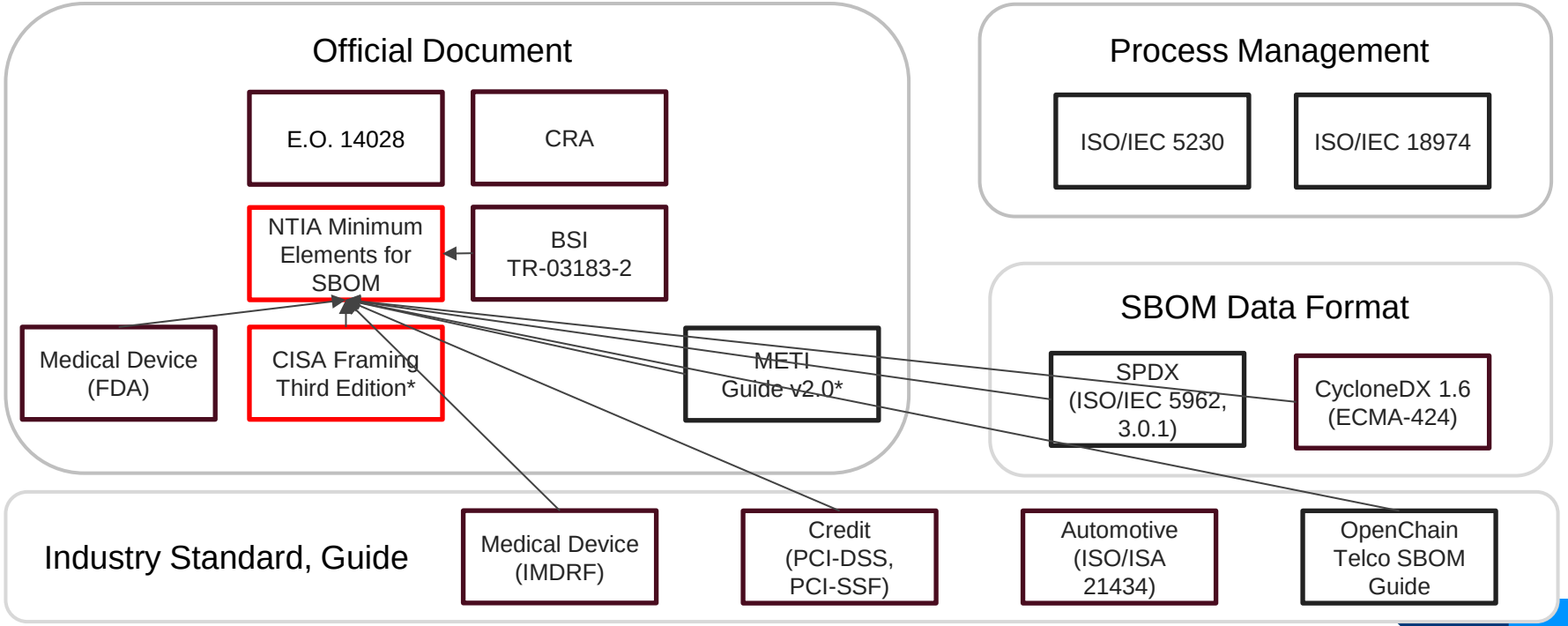


SBOM に関連する法令、規制、標準、ガイドラインの例



NTIA Minimum Elements を参照するものが多い

(※現在、米国では CISA にSBOMの管轄が移る)



CISA. SBOM-a-Rama Fall 2024

- SBOM-a-Rama
  - CISAが半期毎に主催するイベント
- 2024 Fall
  - September 11, 2024-September 12, 2024
  - Denver (In-person) + Virtual (Day2 は In-person のみ)
  - Virtual でもおよそ300名ほどが参加していた模様
- キーワードなど
  - AIBOM
  - bomctl
  - BOMOps
  - Healthcare ISAC, Medical Device
  - Korea’s effort to build an SBOM-based risk management framework
  - Mobile Industiry
  - OWASP TEA
  - Plugfest
  - SBOM Generation
  - Vulnerability

https://www.cisa.gov/news-events/events/sbom-rama-fall-2024

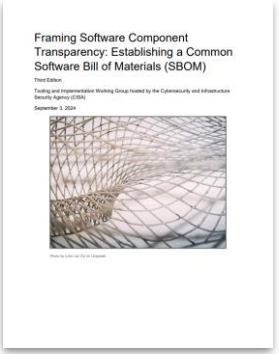
CISA. Framing Software Component Transparency: Establishing a Common Software Bill of Materials (SBOM) 3rd Ed.

- Maturity Levelslを定義
- Baseline Attributes の更新

★ Baseline Attributes が現時点でのライセンスコンプライアンス実務の観点が反映されている

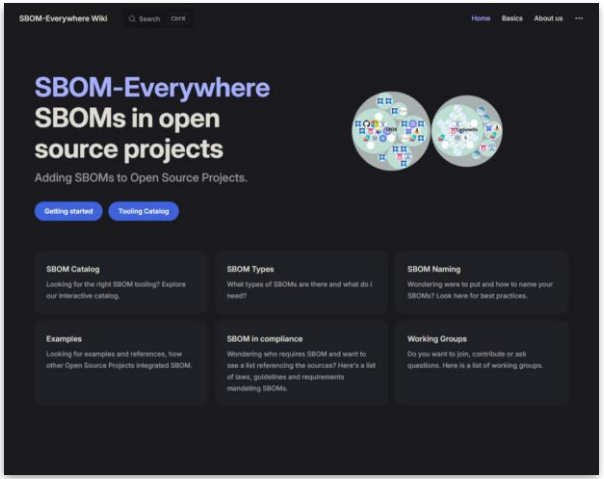
★ 本文書は NTIA の Minimum Elements を置き換えるものではないが、考慮すべき

CISA Baseline Attributes	NTIA Minimum Elements
SBOM Author Name	Author of SBOM Data
SBOM Timestamp	Timestamp
SBOM Type	
Component Supplier Name	Supplier Name
Component Name	Component Name
Component Version String	Version of the Component
Component Unique Identifier	Unique Identifiers
Component Cryptographic Hash	Component Hash
Component License	License Information
Component Copyright Holder	Copyright Information
SBOM Primary Component	Dependency Relationship
Component Relationships	Dependency Relationship
	External Data



https://www.cisa.gov/resources-tools/resources/framing-software-component-transparency-2024

SBOM-Everywhere

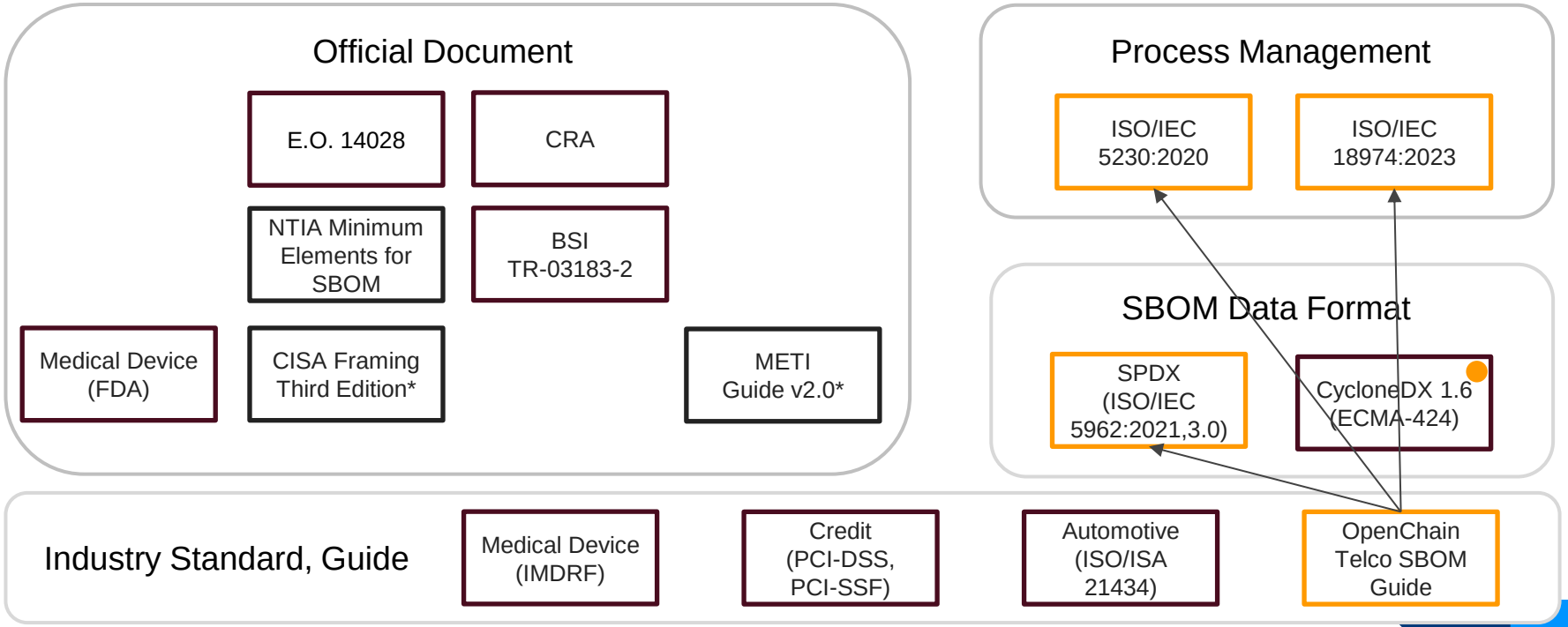


- OpenSSF の取組で、SBOM Landscape にて様々なSBOM関連ツールの情報を入手できる
- 生成、変換、検証、利用などのカテゴリから探すことができる
- SBOMファイルをフォーマット中立な protobom に変換して管理運用するアプローチが見られる: ex. bomctl

https://sbom-catalog.openssf.org/

THE LINUX FOUNDATION
OPENCHAIN

OpenChain Project はプロセスマネジメント要件でSBOMを重視



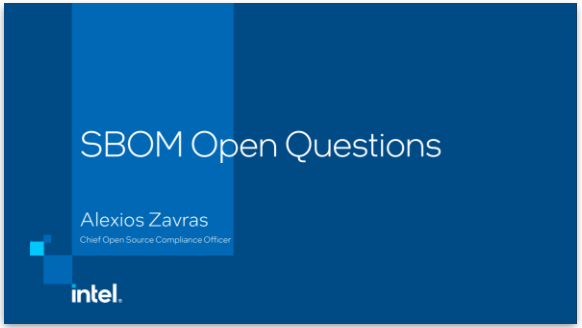
- OpenChain Prfoject ではSBOM標準仕様の一つとして議論の対象とする

Open Source Summit Europe 2024

- Open Source Summit
  - Linux Foundation が年に3回(米、欧、日)開催するイベント
- Europe 2024
  - Vienna, Austria, 16-18 September 2024
  - Videos
    - https://www.youtube.com/@LinuxfoundationOrg/playlists
  - Presentations
    - https://osseu2024.sched.com/

https://events.linuxfoundation.org/archive/2024/open-source-summit-europe/

Open Source Summit Europe 2024 (cont.)



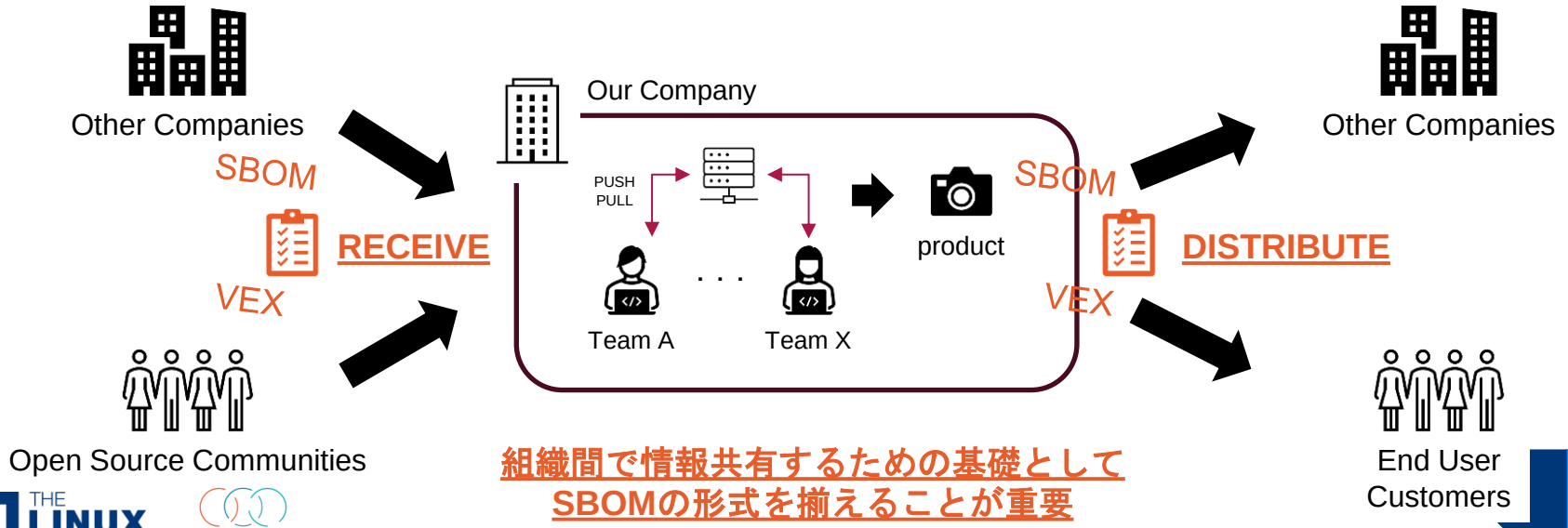
これら以外に Ad hocで “Unconference: Open Source Projects Road to Compliance SBoM, SPDX, etc.” が開催された

Open Source Summit Europe 2024: SPDX Ad hoc

- 9/17, SPDX 3.0 ad hoc
  - SPDX 3.0.1 をベースにISO化を図ることが改めて確認された
- 9/17, SPDX 3.0 translation ad hoc
  - SPDX Projectから、OSSJ2024の開催時期までに日本語版を用意したい旨の要望が示された

SPDX Lite : コンセプト

- SBOM仕様 SPDX のサブセット
- ライセンスコンプライアンスを目的として、SPDX仕様のうち最小限の情報要素に絞り込む



THE LINUX FOUNDATION
OPENCHAIN

SPDX Lite : ISO/IEC 5962:2021 (SPDX 2.2.1), SPDX 2.3

- **Software** Package Data Exchange (SPDX) の Annex H (v2.3ではG) として国際標準化
- パッケージを対象として、ライセンスコンプライアンス実務の観点で要素を厳選
- 国内企業でスプレッドシートでの利用事例アリ (SPDX Lite を独自拡張して利用するなど)


INTERNATIONAL STANDARD

ISO/IEC 5962

Information technology — SPDX® Specification V2.2.1

Annex H (informative)

SPDX file tags

H.1 Rationale

H.2 Format

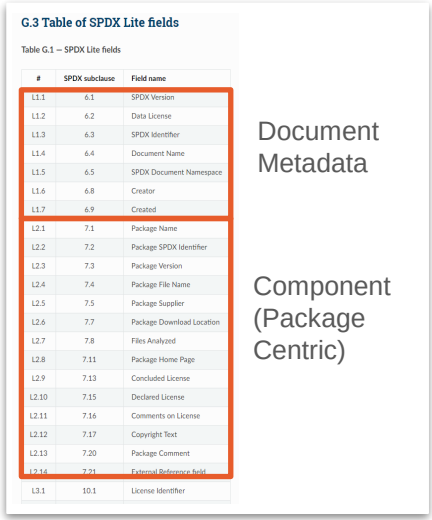
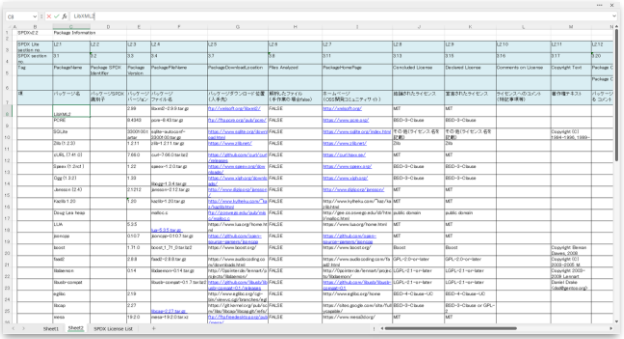

G.3 Table of SPDX Lite fields

Table G.1 — SPDX Lite fields

#	SPDX subclause	Field name
L1.1	6.1	SPDX Version
L1.2	6.2	Data License
L1.3	6.3	SPDX Identifier
L1.4	6.4	Document Name
L1.5	6.5	SPDX Document Namespace
L1.6	6.8	Creator
L1.7	6.9	Created
L2.1	7.1	Package Name
L2.2	7.2	Package SPDX Identifier
L2.3	7.3	Package Version
L2.4	7.4	Package File Name
L2.5	7.5	Package Supplier
L2.6	7.7	Package Download Location
L2.7	7.8	Files Analyzed
L2.8	7.11	Package Home Page
L2.9	7.13	Concluded License
L2.10	7.15	Declared License
L2.11	7.16	Comments on License
L2.12	7.17	Copyright Text
L2.13	7.20	Package Comment
L2.14	7.21	External Reference field
L3.1	10.1	License Identifier

Document Metadata

Component (Package Centric)



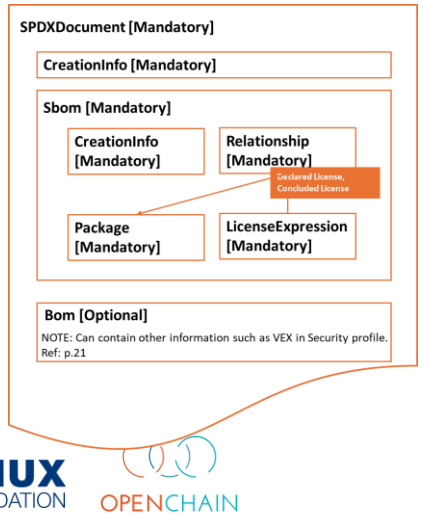
https://spdx.github.io/spdx-spec/v2.3/SPDX-Lite/

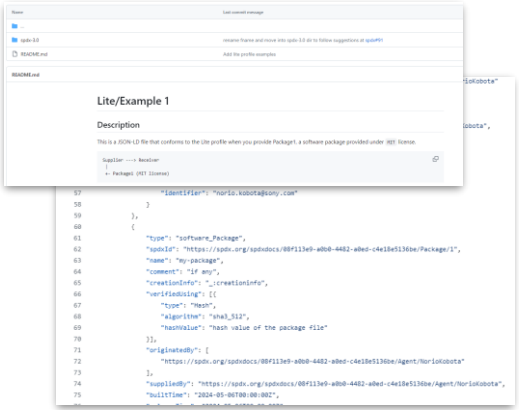
THE LINUX FOUNDATION

OPENCHAIN

SPDX Lite : SPDX 3.0 Lite Profile

- **System** Package Data Exchange (SPDX) 3.0 モデルへの変更に伴いデータ構造やエレメントを更新
- NTIA Minimum Elements を踏まえつつ実務観点でエレメントの追加やMadatory/Optional を定義 (現状、CISA Framing of Software Components Transparency 3rd ed. の Baseline Attributes もカバー)
- 作例を通じて Lite 及び Lite + Security Profile (VEX) の SBOMデータ表現の検証も進めている





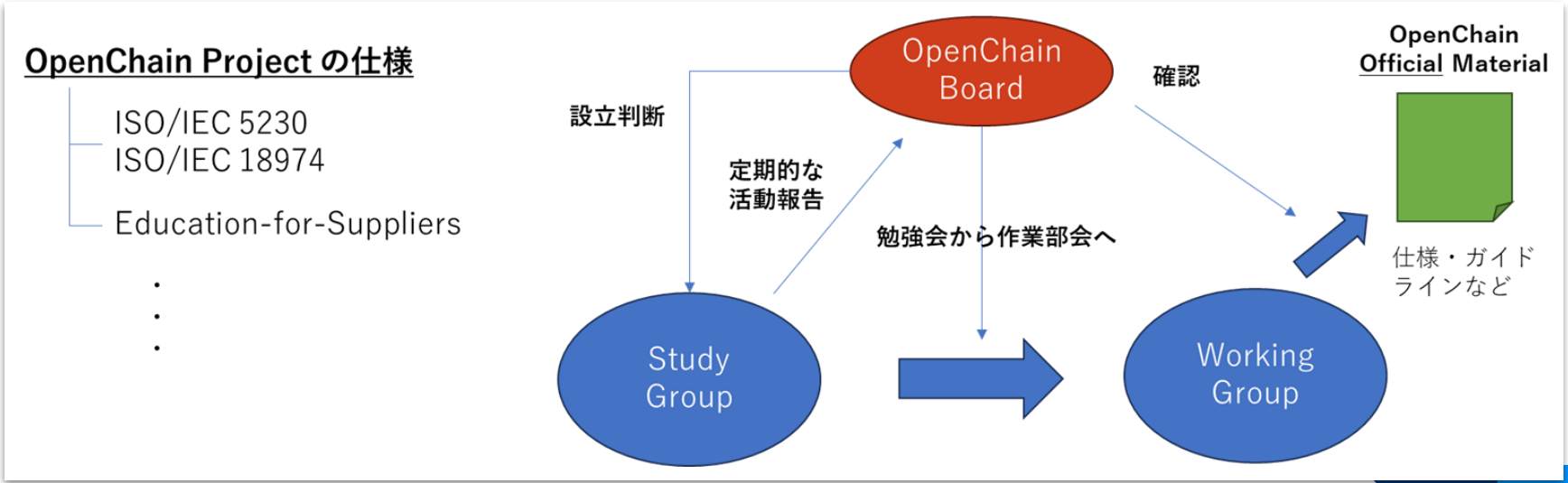
https://github.com/NorioKobota/spdx-examples/tree/lite-profile/lite/example1

https://github.com/no-ta/spdx-examples/tree/merge-lite-example-1/lite/example1-with-VEX/spdx-3.0

THE LINUX FOUNDATION
OPENCHAIN

OpenChain Project

- Working Group で仕様やガイドライン等を策定する
- 国際標準化する場合は JDF (Joint Development Foundation) と協力する



OpenChain SBOM Study Group

設立判断

OpenChain Board

定期的な
活動報告

SBOM Study Group

SBOMの利用にフォーカスを当て、製品やサービス開発など、他エンティティ間とのソフトウェアサプライチェーンにおける、実用的なSBOMのガイドラインやベストプラクティスを作成することを模索

日本で議論した結果を、グローバルコミュニティに持っていく

SBOM Study Group

毎月第4水曜
17:00 - 18:00 JST

OpenChain Japan SBOM SG

LFX

毎月第2火曜 16:00-18:00 JST
Automation & SBOM SG
情報共有・ドキュメントレビュー

LFX

毎週月曜 18:00-19:00 JST
Automation & SBOM SG
詳細議論・実作業

THE LINUX FOUNDATION
OPENCHAIN

Q&A

信頼に基づく秘密を共有してのディスカッションのために

これ以降のディスカッションは 「Chatham House Rule」 に基づいて執り行います
https://www.chathamhouse.org/about-us/chatham-house-rule

When a meeting, or part thereof, is held under the Chatham House Rule, participants are free to use the information received, but neither the identity nor the affiliation of the speaker(s), nor that of any other participant, may be revealed.

要点 1. この会議で知った情報を自由に使えます
2. 発言者や参加者を明かすことをしてはなりません

このルールに基づく参加に 同意頂けない方は退席 をお願いします

THE LINUX FOUNDATION
OPENCHAIN

SBOMやVEXに関して

- ビジョン
  - 企業を超えた協力関係により、SBOMやVEXを安心して使えるソフトウェアサプライチェーンを構築する
- ミッション
  - SBOMやVEXの標準化と普及を促進し、組織内および組織間のベストプラクティスを提供する
- ゴール
  - SBOMやVEXの標準化
  - ツールの開発、ノウハウの共有
  - コミュニティの構築
  - ベストプラクティスを収集、類型化し、ガイドに整理して提供する
  - (トレーニングの提供)

本件パネルディスカッションについて

- 想定されるゴール
  - SBOMやVEXに関する関心事項や課題(困りごと)の共有
  - Know-who や コネクション が拡がる
  - Japan-WG コミュニティへの参加者が増える

Under the Chatham House Rule

OPENCHAIN JAPAN COMMUNITY DAY #32
メインセッション② - 業界最新動向の共有 -

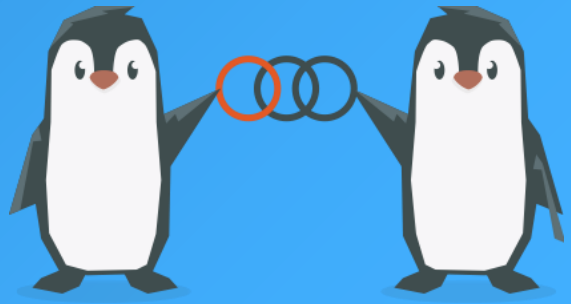
SBOM 動向：ディスカッション

(16:40 pm - 17:15 pm)

SBOM Sub Work Group,
OpenChain Japan Work Group

THE LINUX FOUNDATION

OPENCHAIN



SBOM提供の要求を受けるシーン

- 提供先毎に要求は様々
  - 提供先の社内システムに沿った個別対応を求められることも
- 最上流のサプライヤには、中間ステークホルダのさらにその先の提供先が示す要求フォーマットへの対応が求められる場合も
- SPDX Lite提案の背景には、フラグメントしないライセンスリストを、という課題の共有がある
- VEX対応はまだまだこれから
  - 技術/市場/業界の動向の把握には議論への参加も有効
  - コミュニティの主体的な活動が個別の商用ソリューションへの依存に対する解消への鍵か

AOSP(Android Open Source Project)

- AOSPはSBOM(SPDX2.3)の作成をサポート
  (https://source.android.com/docs/setup/create/create-sbom?hl=ja)

- SBOMの作成方法
  (https://source.android.com/docs/setup/create/create-sbom?hl=ja)

ソフトウェア部品構成表（SBOM）の作成

2022 年 2 月、米国国立標準技術研究所（NIST）が Secure Software Development Framework（SSDF）のバージョン 1.1 を公開しました。SSDF は、2021 年のサイバーセキュリティ大統領命令（EO）14028 に対応するために作成された、安全なソフトウェア開発手法に関する包括的なガイドラインです。

これらの要件の一部として、ソフトウェア リリースのコンポーネントを一覧にまとめた「ソフトウェア部品構成表（SBOM）」の提出を、米国政府から要請されることがあります。

注: このページの手順に沿って生成した SBOM には、米国商務省電気通信情報局（NTIA）によって公表されているソフトウェア部品構成表（SBOM）の最低限の要素が含まれています。また、Android ベースの SBOM ツールとプロダクト SBOM は Software Package Data Exchange（SPDX）2.3 形式に準拠しており、ソフトウェア パッケージに関連付けられているコンポーネントやメタデータの情報を SPDX 形式で伝達します。

SBOM は、自動的に Android 継続的インテグレーション（Android CI）ビルド用に生成されます。いずれかの CI ビルドを使用する場合は、以下の手順に沿ってビルドの SBOM を取得してください。それ以外の場合は、カスタム SBOM を生成するための手順を使用してください。

カスタム SBOM を生成する

バイナリやビルド、リリース ツール チェーンなど、プラットフォームに何かを追加する場合は、ソフトウェア部品構成表（SBOM）の最低限の要素を備えた SBOM 表現を提供する必要があります。カスタム SBOM を生成する手順は次のとおりです。

1. 次のコマンドを実行します。これにより、環境設定と SBOM のビルドが行われます。

```
$ $ source build/envsetup.sh
$ lunch TARGET
$ m sbom # Generates an SBOM
```

`TARGET` には、Android のビルドに使用したのと同じビルド ターゲット（たとえば `aosp_arm64-userdebug`）を指定します。

2. SBOM が正しくビルドされたことを確認するため、次のコマンドを実行します。

```
$ $ ls out/dist/sbom*
```

AOSP: 生成したSBOM


```
{
    "licenseId": "LicenseRef-frameworks-base-core-res-license",
    "name": "frameworks_base_core_res_license",
    "extractedText": "<text>\n   Copyright (c) 2005-2008, The Android Open Source Project\n\n   Licensed under the Apache License, Version 2.0 (the \"Licens
},
{
    "licenseId": "LicenseRef-frameworks-opt-net-ims-license",
    "name": "frameworks_opt_net_ims_license",
    "extractedText": "<text> Copyright (c) 2014, The Linux Foundation. All rights reserved.\n\n Redistribution and use in source and binary forms, with or w
},
{
    "licenseId": "LicenseRef-bionic-libc-license",
    "name": "bionic_libc_license",
    "extractedText": "<text> Copyright (c) 1993 John Brezak\n All rights reserved.\n\n Redistribution and use in source and binary forms, with or without\n
},
{
    "licenseId": "LicenseRef-bionic-libdl-license",
    "name": "bionic_libdl_license",
    "extractedText": "<text>Copyright (C) 2007 The Android Open Source Project\n\nLicensed under the Apache License, Version 2.0 (the \"License\");\nyou may
},
```


https://source.android.com/docs/setup/create/create-sbom?hl=ja

AOSP: Licenseの指定方法(Android.bpの記載例)

```
license {
    name: "frameworks_base_core_res_license",
    visibility: [":__subpackages__"],
    license_kinds: [
        "SPDX-license-identifier-Apache-2.0",
    ],
    license_text: [
        "NOTICE",
    ],
}
```

https://android.googlesource.com/platform/frameworks/base/+/master/core/res/Android.bp

AOSP: SBOMは自動生成される。だがしかし…

```
diff --git a/core/res/Android.bp b/core/res/Android.bp
index 6063062..c42517d 100644
--- a/core/res/Android.bp
+++ b/core/res/Android.bp

@@ -37,7 +37,6 @@
     visibility: [":__subpackages__"],
     license_kinds: [
         "SPDX-license-identifier-Apache-2.0",
-        "SPDX-license-identifier-GPL",
     ],
     license_text: [
         "NOTICE",
```

```
Fix incorrect licenses in frameworks/base

SPDX-license-identifier-GPL and SPDX-license-identifier-W3C are
false positives in the detector in binary files.

libs/usb/test/accessorytest/f_accessory.h is an original kernel
header file with GPL 2.0 license. Replace it with the file from
libs/usb/tests/AccessoryChat/accessorychat/linux/usb/f_accessory.h,
which is the same header after running it through bionic's script
to remove copyrightable information from the header.
```

https://android.googlesource.com/platform/frameworks/base/+/c1309ecfbab7d107f837e8f67d664a0d21b9174e%5E%21/#F0

SBOM出力機能に関する議論

- ツールチェイン、オープンソース、商用ソリューションのいずれであっても、SBOM出力結果には要注意
- 「標準」として提供される機能でも、その出力が要求に応える内容かどうかを検証すること
- 人手で修正するコストを見込むことや、そうしたスキルを持った人材の存在、なども重要

※会議後の追記事項：アップストリームへの貢献も重要

SPDXコミュニティー / こぼればなし

- SPDX v3.0.1 で Lite profile は仕様から外れかけた...

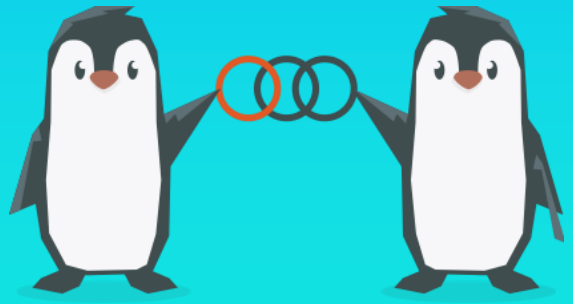
```
More specifically, the content of spdx/using will _not_
be part of the ISO standard – that's why we move them.

… (snip)

In general, think of the specification as the necessary
documents for _defining_ SPDX and the "using" repo as
helpful documentation on _how to use_ SPDX.

… (snip)

Looking at the current annex
https://github.com/spdx/spdx-
spec/blob/development/v3.0.1/docs/annexes/SPDX-Lite.md, I
```

アプローチ

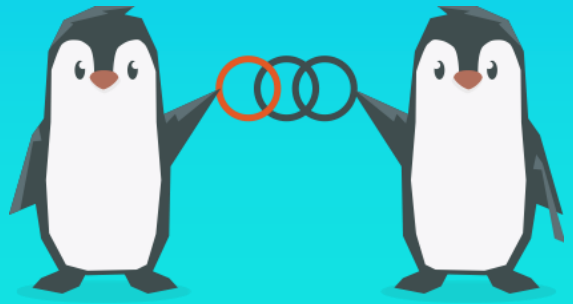
SPDX Lite の SPDX仕様化は市場要求に適うものであることへの理解を求める

1. 市場性を根拠とする理由付け
   a. ISO化の必要性: 取引要件として明確な共通認識のベース
   b. ライセンスコンプライアンスのためのSBOM要素を厳選　: 特定地域や特定産業に偏らず、産業中立的に幅広い分野で利用可
2. 仲間づくり
   a. OpenChain Telco-WGとの協業、その他の産業セクタをリードするLF傘下プロジェクトの主要メンバの意見を聞く
   b. OpenChain Project の General Manager からの理解も得る
3. コミュニケーション
   a. SPDX3.0.1仕様策定の主要メンバ (複数のTech Lead) らと、過去事例やこれまでの議論を丁寧に踏まえつつ、相互尊重と相互信頼に基づくやり取りを「重ねる」
   b. オンライン会議はタイムゾーンに配慮する

Thank you!



Appendix



Cybersecurity and Infrastructure Security Agency (CISA)

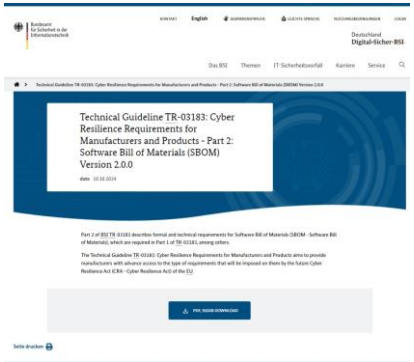
- 米国政府機関でサイバーセキュリティを扱う
  - SBOMを管轄する
    National Telecommunications and Information Administration (NTIA). Minimum Elements for Software Bill of Materials (SBOM) を更新する権限を有する
  - 上位機関は米 Department of Homeland Security (DHS)
- CISA SBOM Community
  - オープンに運営し、そこでのディスカッションの成果をまとめたホワイトペーパーなども発行する
- SBOM-a-Rama
  - 半期毎に開催し、CISAを含めて様々な組織におけるSBOM関連の動向を発信する

Software Bill of Materials (SBOM) | CISA

THE LINUX FOUNDATION
OPENCHAIN

BSI. TR-03183:
Cyber Resilience Requirements for Manufacturers and Products

- Bundesamt für Sicherheit in der Informationstechnik (BSI)
  - ドイツ連邦政府で情報セキュリティを担当する機関
- TR-03183 Part 2: Software Bill of Materials (SBOM)
  - CRAで要求されるSBOMに備えるためのガイドライン
  - 1.0 (2023-07-12), 1.1 (2023-11-28), 2.0 (2024-09-20)
- SBOM Content Requirements
  - SBOM Content の要件はライセンスコンプライアンスで活用されてきた経緯も踏まえている
- Level of Details
  - SBOMの詳細すなわち依存関係の深さや範囲をカテゴリ化して説明



★ Content や Level については、CISA や NTIA と必ずしも一致しない部分がある

経済産業省.
ソフトウェア管理に向けたSBOM（Software Bill of Materials）の導入に関する手引 ver2.0

- 対象
  - 中小企業も含むあらゆるソフトウェアサプライヤー
- 主な内容
  - SBOM導入に向けたプロセス
    - Phase 1. 環境構築、体制整備
    - Phase 2. 作成と共有
    - Phase 3. 運用と管理
  - 脆弱性管理プロセス (追加)
  - SBOM対応モデル (追加)
    - SBOMの作成や活用についてコストや効果を考慮した実施の選択肢とその内容を整理
  - SBOM取引モデル (追加)
    - 契約で規定することが期待される事項について、要求事項、責任、コスト負担、権利などの区分で整理

THE LINUX FOUNDATION
OPENCHAIN

OpenChain Telco SBOM Guide

- OpenChain Project Telco Work Group
  - テレコム産業の企業らが主体となっている活動
  - Nokiaがリード、Ericsson、KDDI、富士通、ソニー、東芝なども参加
- テレコム産業向けとしてSBOMのガイド
  - 要件
    - NTIA Minimum Elements
    - CISA SBOM types
  - SBOMデータフォーマット
    - ISO/IEC 5962:2021(SPDX 2.2.1), SPDX 2.3
    - JSON or Tag/Value
  - KDDIによる日本語版も公開
- Nokia が “OpenChain Telco SBOM validator” をリリース

★ 実質的に CISA Baseline Attributes をカバー
★ 産業中立的に活用できるものになっている

THE LINUX FOUNDATION
OPENCHAIN

ISO/IEC 5230 OpenChain Specification

- Open Source License Compliance
- SBOM Management: Identify, Review & Approval to Software Components

Requirements	
1. Program foundation	1.1 Policy
	1.2 Competence
	1.3 Awareness
	1.4 Program scope
	1.5 License obligations
2. Relevant tasks defined and supported	2.1 Access
	2.2 Effectively resourced
3. Open Source content review and approval	3.1 Bill of Materials
	3.2 License compliance
4. Compliance artifact creation and delivery	4.1 Compliance artifacts
5. Understanding open source community engagements	5.1 Contributions
6. Adherence to the specification requirements	6.1 Conformance
	6.2 Duration

THE LINUX FOUNDATION

OPENCHAIN

ISO/IEC 18974 OpenChain Security Assurance Specification

- Open Source Security Assurance (Vulnerability Management)
- Monitor and Manage Vulnerabilities across SDLC, with SBOM Management

Requirements	
1. Program foundation	**1.1 Policy**
	1.2 Competence
	1.3 Awareness
	1.4 Program Scope
	1.5 Standard Practice Implementation
2. Relevant tasks defined and supported	**2.1 Access**
	2.2 Effectively resourced
3. Open Source content review and approval	**3.1 Software Bill of Materials (SBOM)**
	3.2 Security Assurance
4. Adherence to the guideline requirements	**4.1 Completeness**
	4.2 Duration

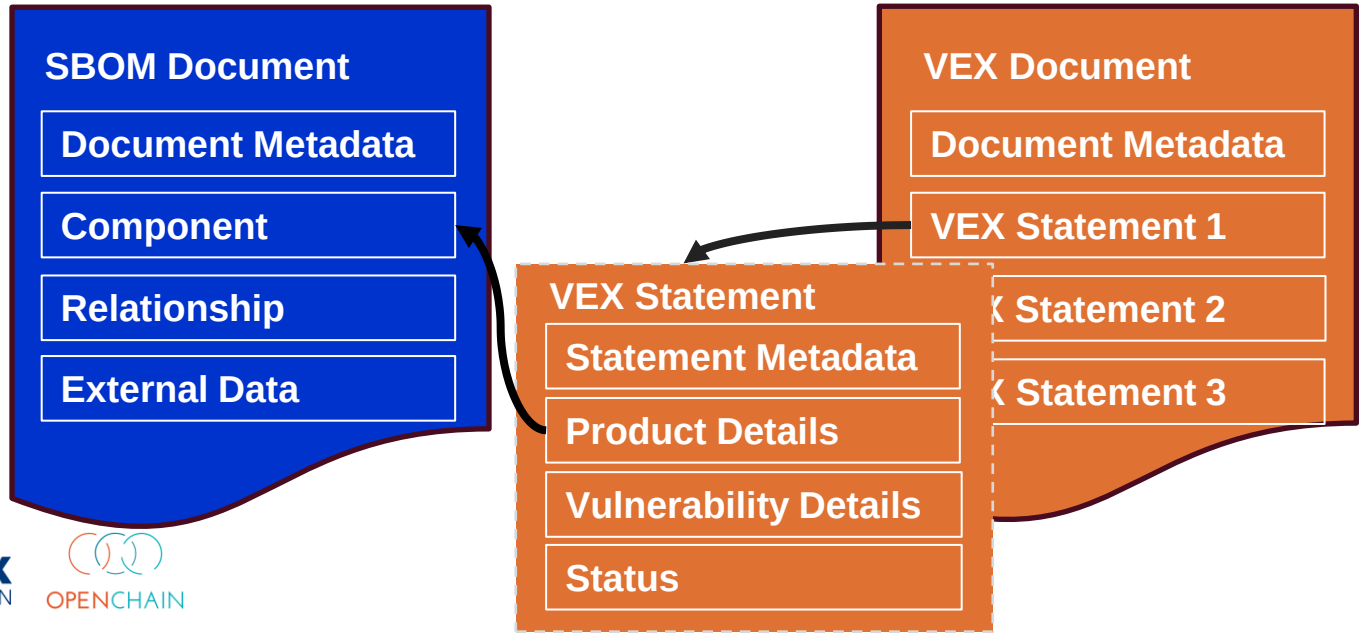
SBOM-VEX

SBOM

- Software Composition
- Provenance
- License Compliance

VEX

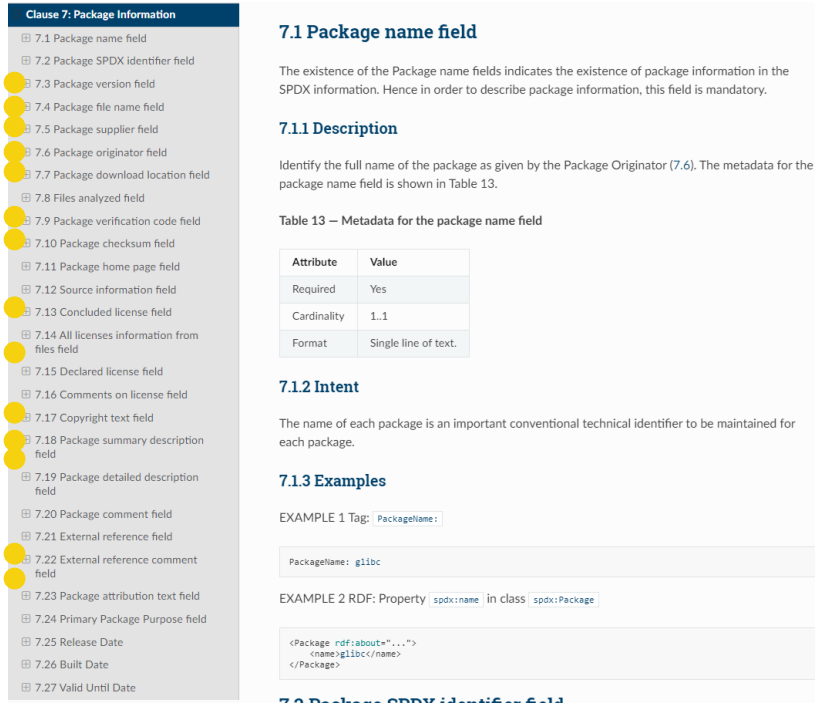
- Vulnerability Management
- Exploit
- Incident Response



THE LINUX FOUNDATION

OPENCHAIN

SPDX Lite：ライセンスコンプライアンス用のサブセット



Clause 7: Package Information

- 7.1 Package name field
- 7.2 Package SPDX identifier field
- 7.3 Package version field
- 7.4 Package file name field
- 7.5 Package supplier field
- 7.6 Package originator field
- 7.7 Package download location field
- 7.8 Files analyzed field
- 7.9 Package verification code field
- 7.10 Package checksum field
- 7.11 Package home page field
- 7.12 Source information field
- 7.13 Concluded license field
- 7.14 All licenses information from files field
- 7.15 Declared license field
- 7.16 Comments on license field
- 7.17 Copyright text field
- 7.18 Package summary description field
- 7.19 Package detailed description field
- 7.20 Package comment field
- 7.21 External reference field
- 7.22 External reference comment field
- 7.23 Package attribution text field
- 7.24 Primary Package Purpose field
- 7.25 Release Date
- 7.26 Built Date
- 7.27 Valid Until Date

7.1 Package name field

The existence of the Package name fields indicates the existence of package information in the SPDX information. Hence in order to describe package information, this field is mandatory.

7.1.1 Description

Identify the full name of the package as given by the Package Originator (7.6). The metadata for the package name field is shown in Table 13.

Table 13 — Metadata for the package name field

Attribute	Value
Required	Yes
Cardinality	1..1
Format	Single line of text.

7.1.2 Intent

The name of each package is an important conventional technical identifier to be maintained for each package.

7.1.3 Examples

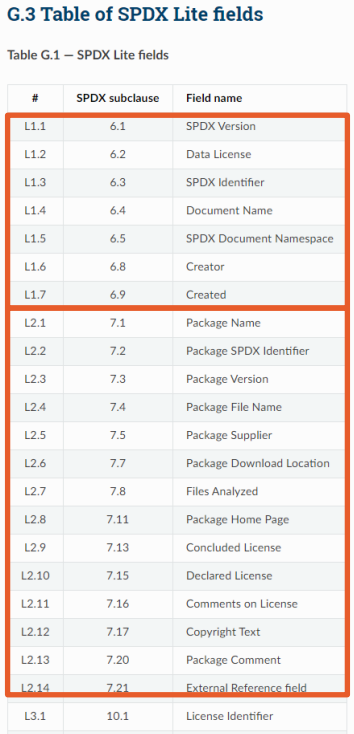
EXAMPLE 1 Tag: `PackageName:`

```
PackageName: glibc
```

EXAMPLE 2 RDF: Property `spdx:name` in class `spdx:Package`

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

https://spdx.github.io/spdx-spec/v2.3/package-information/



G.3 Table of SPDX Lite fields

Table G.1 — SPDX Lite fields

#	SPDX subclause	Field name
L1.1	6.1	SPDX Version
L1.2	6.2	Data License
L1.3	6.3	SPDX Identifier
L1.4	6.4	Document Name
L1.5	6.5	SPDX Document Namespace
L1.6	6.8	Creator
L1.7	6.9	Created
L2.1	7.1	Package Name
L2.2	7.2	Package SPDX Identifier
L2.3	7.3	Package Version
L2.4	7.4	Package File Name
L2.5	7.5	Package Supplier
L2.6	7.7	Package Download Location
L2.7	7.8	Files Analyzed
L2.8	7.11	Package Home Page
L2.9	7.13	Concluded License
L2.10	7.15	Declared License
L2.11	7.16	Comments on License
L2.12	7.17	Copyright Text
L2.13	7.20	Package Comment
L2.14	7.21	External Reference field
L3.1	10.1	License Identifier

Document Metadata

Component (Package Centric)

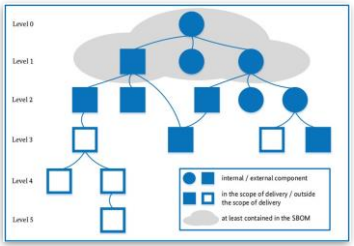
https://spdx.github.io/spdx-spec/v2.3/SPDX-Lite/

THE LINUX FOUNDATION

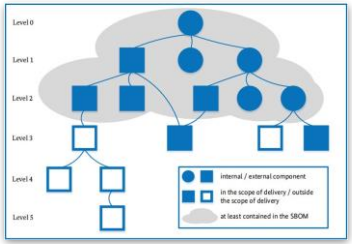
OPENCHAIN

CISA. “Maturity Levels” | BSI. “Level of Details”

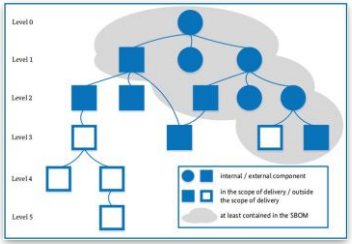
Maturity Levels	概要
Minimum Expected (CRAWL)	絶対的に最低限として想定されるもの。 直接依存するコンポーネントは扱うが、それよりも深い依存を扱わない。
Recommended Practice (WALK)	推奨されるアプローチ。 直接依存するコンポーネントと全て(full)の深さの依存までを扱う。
Aspiration Goal (RUN)	最も望まれる、いわば達成されるべき水準。 直接依存するコンポーネントと全て(full)の深さの依存に加え、リモートのコンポーネントも扱う。



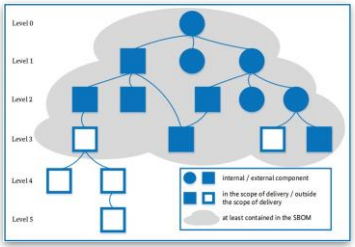
Top-level SBOM



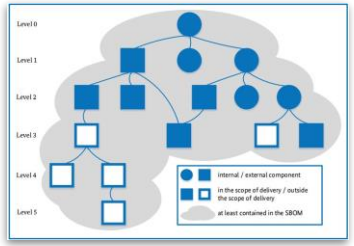
n-level SBOM



Transitive SBOM



Delivery item SBOM



Complete SBOM

SBOM Attributes Comparison (Overview)

NTIA Minimum Elements	CISA Baseline Attributes	EU TR-03183-2	IMDRF/FDA	PCI-DSS/PCI-SSF	OpenChain Telco SBOM guide	SPDX Lite (SPDX 3.0.1 Lite Profile)	CycloneDX 1.6 (ECMA-424)
					(6.1) SPDXVersion:	Core.CreationInfo.specVersion	specVersion
					(6.2) DataLicense:	Core.SpdxDocument.dataLicense	metadata.licenses
					(6.3) SPDXID:	Core.SpdxDocument.spdxId	serialNumber
		SBOM-URI			(6.4) DocumentName:	Core.SpdxDocument.name	
					(6.5) DocumentNamespace:	Core.SpdxDocument.namespaceMap, and other relevant properties	
Author of SBOM Data	SBOM Author Name	Creator of the SBOM	Author name	The name of the author who designed/developed the component or service.	(6.6) Creator:	Core.CreationInfo.createdBy	metadata.authors
Timestamp	SBOM Timestamp	Timestamp	Timestamp		(6.9) Created:	Core.CreationInfo.created	metadata.timestamp
Lifecycle: Phase	SBOM Type (Optional; Application)	(assumed Build SBOM)			(6.10) CreatorComment: (*1)	Software.Sbom.sbomType	metadata.lifecycles
Dependency Relationship	SBOM Primary Component	(primary component)	Relationship	A description of the relationship(s) between the component and service and other components/services embedded in or used by the software.	(11.1) Relationship: DESCRIBES (*1)	Software.Sbom.rootElement	metadata.component
Component Name	Component Name	Component name	Software component name	The name of the component or service as defined by the original supplier.	(7.1) PackageName:	Software.Package.name	components[].name
Version of the Component	Component Version String	Component version	Software component version	The version of the component or service as defined by the original supplier to differentiate it from previous or other updates.	(7.3) PackageVersion:	Software.Package.packageVersion	components[].version
Supplier Name	Component Supplier Name	Component creator	Software component vendor (supplier)	The original source/supplier of the component or service.	(7.5) PackageSupplier: (*2)	Software.Package.suppliedBy	metadata.supplier components[].supplier
Component Hash	Component Cryptographic Hash	Hash value of the executable component (Optional: Hash value of the source code of the component)	Component Hash		(7.10) PackageChecksum: (*1)	Software.Package.verifiedUsing	components[].hashes[]
		File name of the component			(7.4) PackageFileName:	Software.File.name	components[].evidence[].identity[].methods[].[{"technique":"filename","confidence":0.1,"value":"<FileName>"}] serialNumber+version
Other Unique Identifiers	Component Unique Identifier	Other unique identifiers	Unique Identifier	Any other identifiers provided by the original	(7.2) SPDXID: (6.5) DocumentNamespace: (7.2.1) ExternalRef:	Core.Artifact.spdxId Software.SoftwareArtifact.contentIdentifier Software.SoftwareArtifact.externalIdentifier (gitoid, cpe22, cpe23, cve, gitoid, packageUrl, swhid, swid, secondaryOther, other)	components[].cpe components[].purl components[].swid components[].omniborId components[].swhid components[].evidence.identity
Dependency Relationship	Component Relationships	Dependencies on other components	Relationship	A description of the relationship(s) between the component and service and other components/services embedded in or used by the software.	(11.1) Relationship: CONTAINS (*1)	Core.Relationship contains dependsOn hasStaticLink hasDynamicLink hasPrivateDependency hasOptionalDependency	dependencies[] components[].components
		-Executable property: "executable" and "non-executable" -Archive property: "archive" and "no archive" -Structured property: "structured" and "unstructured",			(5962: (7.20) PackageComment SPDX2.3: (7.24) Primary Package Purpose:	Software.Package.additionalPurpose	components[].type[]
		- Source code URI - URI of the deployable form of the component			(7.7) PackageDownloadLocation: (7.2.1) ExternalRef:	Software.Package.downloadLocation	components.externalReferences
		Associated licences			(7.14) PackageLicenseInfoFromFiles:	Lack of" PackageLicenseInfoFromFiles" equivalent property (See also: https://github.com/spdx/spdx-3-model/issues/303)	components[].evidence[].licenses[]
License Information	Component License (Application: Concluded)	Concluded Licences (Optional: Declared licences)	FDA: Licence		(7.13) PackageLicenseConcluded: (7.15) PackageLicenseDeclared:	Core.Relationship hasConcludedLicense hasDeclaredLicense	components[].licenses[] components[].licenses[].acknowledgement (concluded, concluded) components[].licenses[].license.id (proprietary) components[].evidence.licenses[]
License Information (Copyright)	Component Copyright Holder	(Not mentioned clearly, but assumed)			(7.17) PackageCopyrightText:	Software.SoftwareArtifact.copyrightText	components.copyright
Other Component Relationships	Component Relationships (Heritage or Pedigree Relationship)				(11.1) Relationship: GENERATED_FROM (11.1) Relationship: DESCENDANT_OF	Core.Relationship generatedFrom descendantOf	components[].pedigree
	Supplemental Information to Support Use Cases		FDA: Software lean forensport		(7.20) PackageComment or other comment data fields	Software.Package.supportLevel	
	Supplemental Information to Support Use Cases		- FDA: End-of-support date. - IMDRF: Life cycle of a device (e.g., a software component's end-of-support (EOS) date)		(5962: (7.20) PackageComment: SPDX2.3: (7.27) ValidUntilDate:	Software.Package.validUntilTime	

NOTE: *1: Not supported in SPDX Lite (ISO/IEC 5962:2021 (SPDX 2.2.1) and SPDX 2.3)
NOTE: *2: Not supported in SPDX Lite (ISO/IEC 5962:2021 (SPDX 2.2.1)

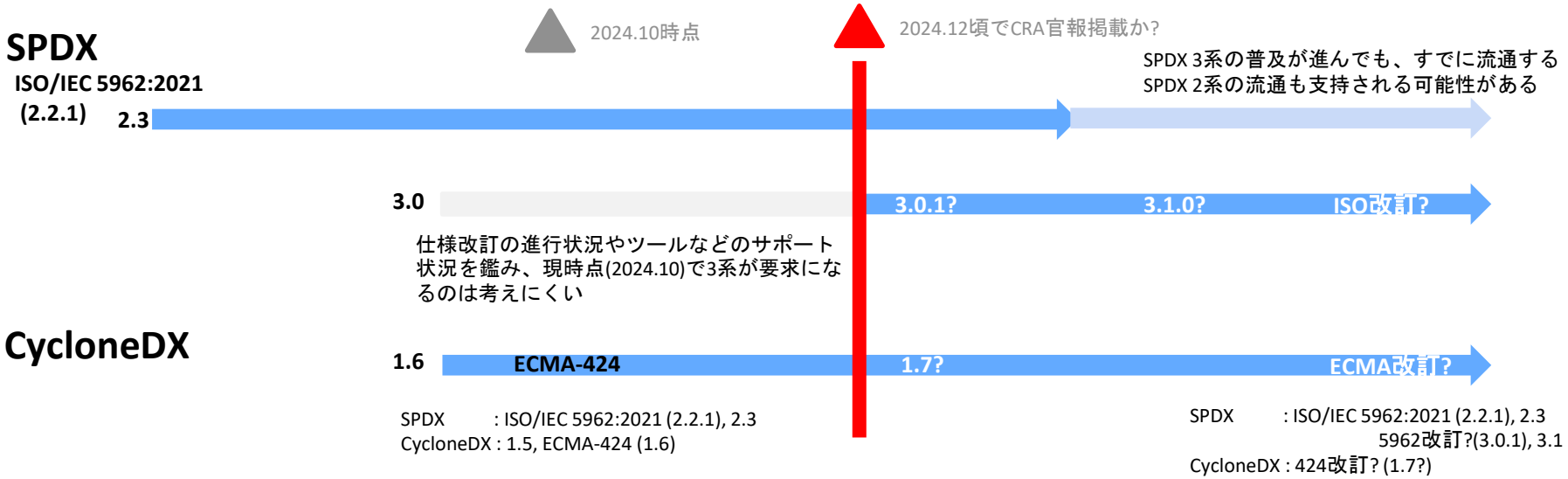
Need to discuss
Optional in specification
not specified

[WIP] SBOM element comparison
https://docs.google.com/spreadsheets/d/1SuGv1L3H_-Iq6dmH7DnjDgAa90LCRnoHB3DTfuWh0Jg/edit?gid=1936044844#gid=1936044844

THE LINUX FOUNDATION

SBOM フォーマット トレンド

SPDXは3系と2系(5962, v2.3.1)と併存する可能性がある

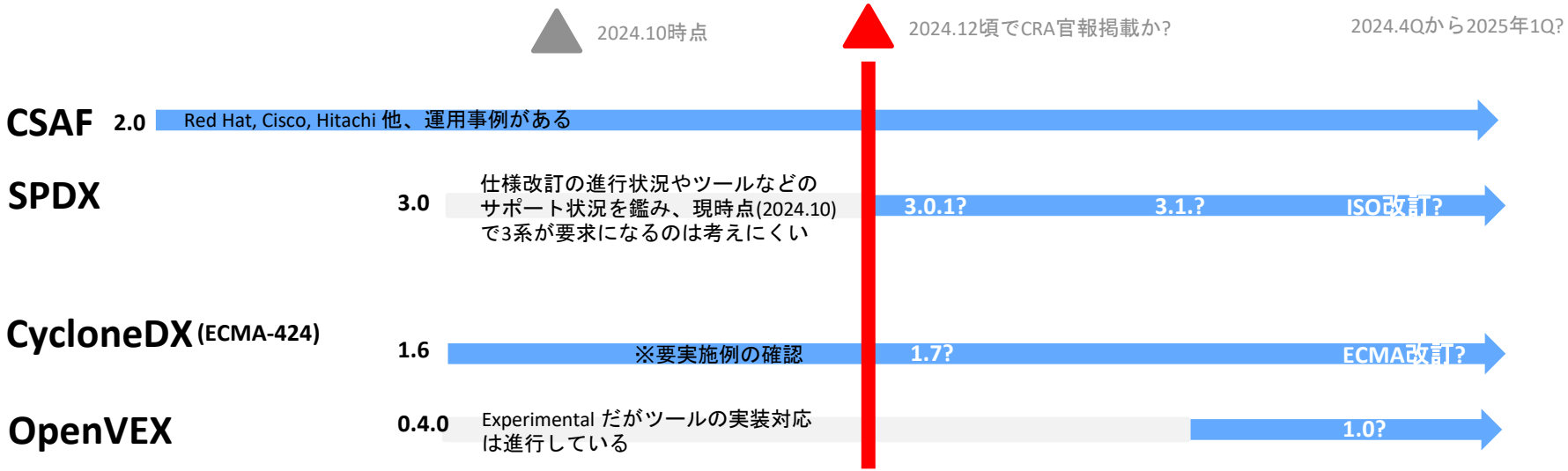


SPDX : 5962 (2.2.1), 2.3, 3.0.1?, 3.1?
CycloneDX : 1.5, 424 (1.6), 1.7?

THE LINUX FOUNDATION
OPENCHAIN

VEX フォーマット トレンド

4つのフォーマットが混在する可能性がある

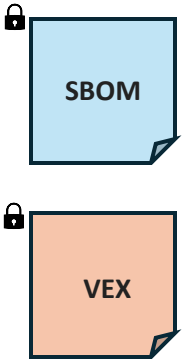


THE LINUX FOUNDATION
OPENCHAIN

SBOM と VEX の運用にみられる類型*

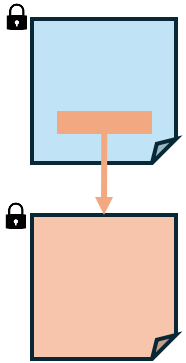
最新の状態の管理と共有が最も効果的に実施できる運用が求められる

ファイル: 分離 (s)
関連付: 無



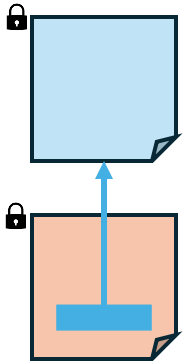
- 個別に更新が可能
- Consumerに関連付けに関する情報を別途提供する必要がある

ファイル: 分離 (s)
関連付: S→V



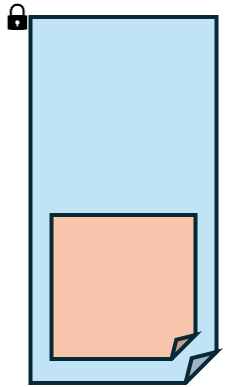
- VEXの更新の都度、SBOMも更新が必須

ファイル: 分離 (s)
関連付: V→S



- SBOMに更新なければVEXのみ更新が可能
- SBOMに更新あればVEXも更新が必要

ファイル: 埋込 (e)
関連付: 一体



- SBOMとVEXのどちらかに更新があれば、データファイルとして更新が必須

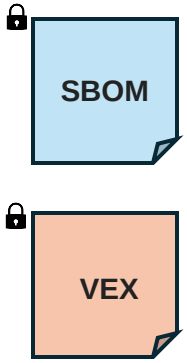
* 本検討のための独自分類

SBOM と VEX の運用で話題になる類型*

関連付けが必須な場合、s(V, S) または e での運用が想定されうる

ファイル: 分離 (s)

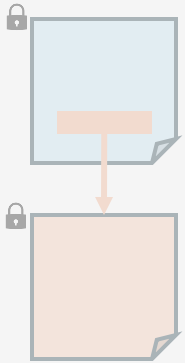
関連付: 無



- 現時点で導入が簡易
- [再掲] サプライチェーン間で最新版を把握できるようにする必要がある

ファイル: 分離 (s)

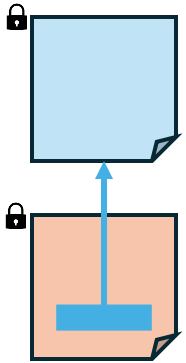
関連付: S→V



- 2021年頃は話題になることが多かったモデル
- VEXの性質を踏まえて、実運用を疑問視する意見がある

ファイル: 分離 (s)

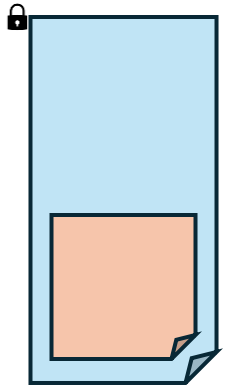
関連付: V→S



- 関連付ける場合はV→Sとする意見が増えつつある

ファイル: 埋込 (e)

関連付: 一体



- SBOMとVEXの対応付けとこれらの最新情報を誤りなく提供しやすい

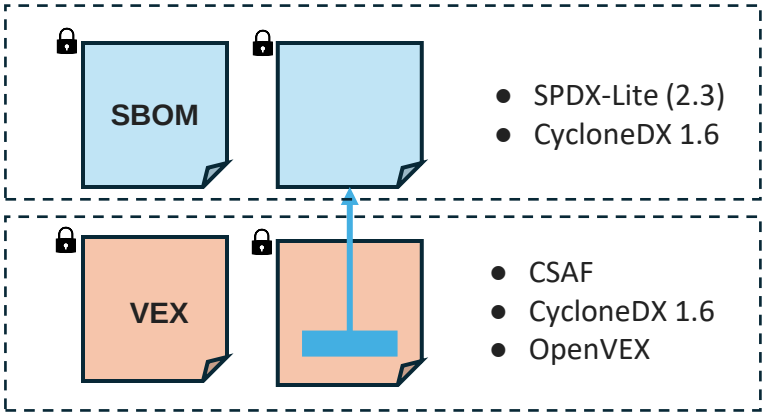
* 本検討のための独自分類

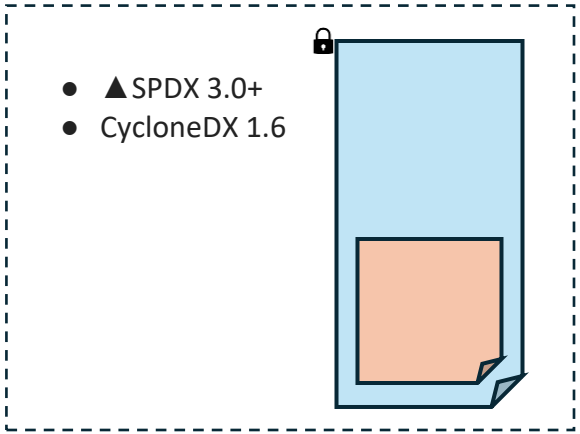
SBOM と VEX の組合せ実施例

SBOM　　：[実績を優先：SPDX-Lite (2.3)],　[記述事例の検証が必要：CycloneDX (1.6)]

VEX　　：[実績を優先：CSAF (記載量多)],　[分かりやすさを優先：OpenVEX (実績少)

SBOM+VEX　：[仕様の成熟度を優先：CycloneDX],　[今後の普及を待てる場合：SPDX 3.0+ (Lite + Security)]





※SBOMは、OpenChain Conformant な表現を前提にする

※SPDX 3.0+ は仕様及び実装例の検証がまだ必要な状況

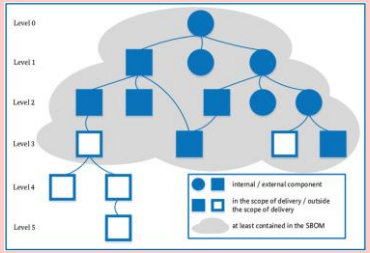
サプライチェーンにおける SBOM & VEX の実務の向上に向けて

実務的な課題の共有, ベストプラクティスの整理, ガイドの整備が重要

CISA. Maturity Levels

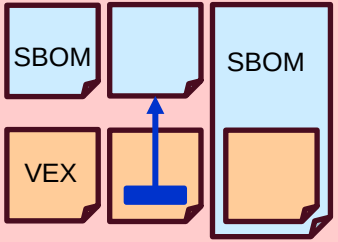
- Minimum Expected
- Recommended Practice
- Aspirational Goal

BSI. Level of Details



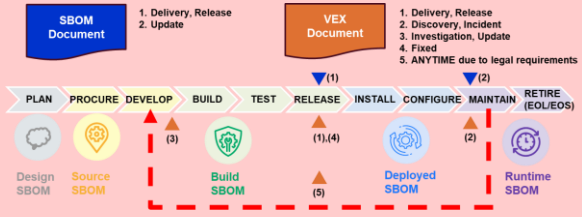
Attributes

SBOM & VEX



Operation

SDLC



"Delivery item SBOM" argued in: BSI. "Technical Guideline TR-03183: Cyber Resilience Requirements for Manufacturers and Products - Part 2: Software Bill of Materials (SBOM) Version 2.0"
CISA. "Framing Software Component Transparency: Establishing a Common Software Bill of Materials (SBOM)" 3rd Ed.
CISA. "When to Issue VEX Information". https://www.cisa.gov/resources-tools/resources/when-issue-vex-information
CISA. "Types of Software Bill of Materials (SBOM)". https://www.cisa.gov/resources-tools/resources/types-software-bill-materials-sbom

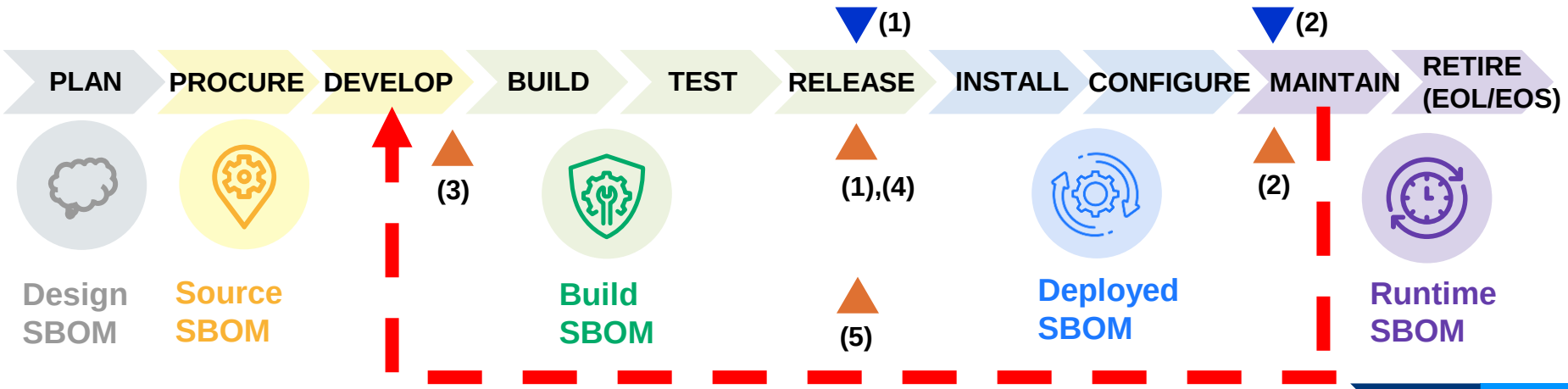
SBOM と VEX を SDLC にわたり管理しサプライチェーンで共有

SBOM Document

1. Delivery, Release
2. Update

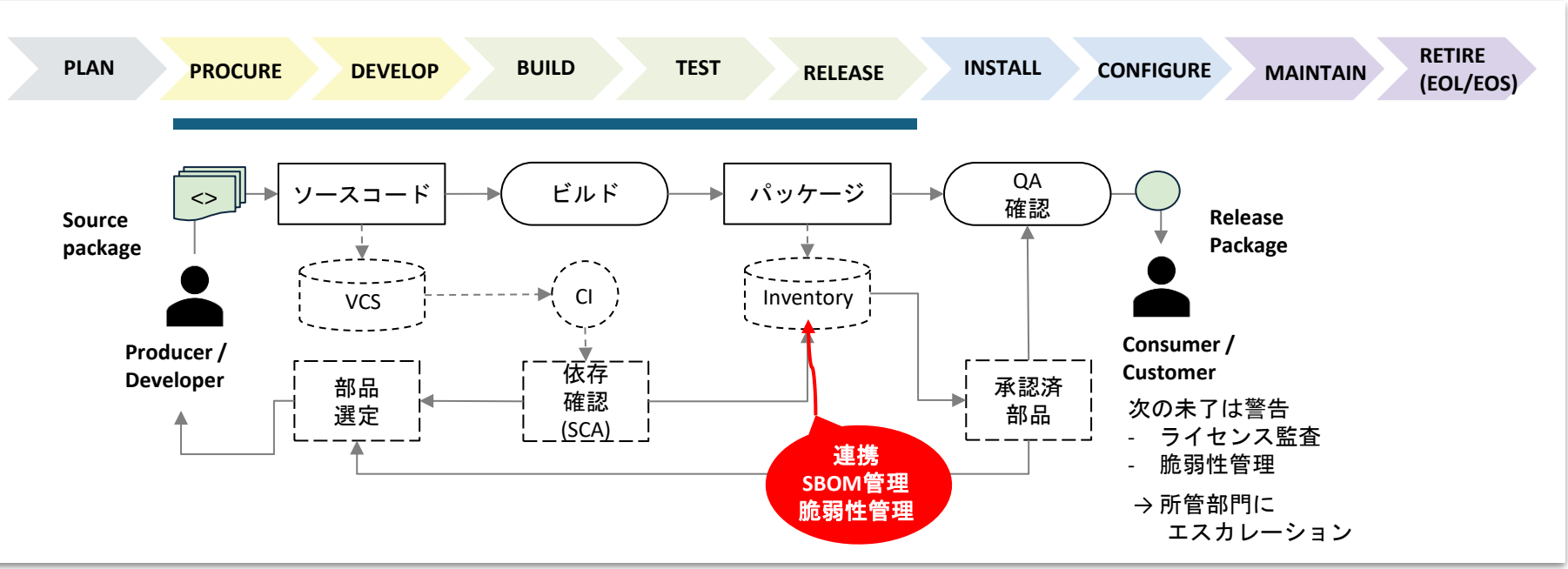
VEX Document

1. Delivery, Release
2. Discovery, Incident
3. Investigation, Update
4. Fixed
5. ANYTIME due to legal requirements



SDLCに渡る管理システム

構成管理システムと連携してSBOMやVEXを出力するように環境を整備する (下図はリリースまでの例)



VCS: Version Control System
CI: Continuous Integration
SCA: Software Composition Analysis
※ 一般的には、SCAに関連して、著作権、ライセンス、脆弱性その他の必要な情報の確認を実施することが知られている

End