Open Security Controls Assessment Language
The Anatomy of OSCAL Models
- Implementation Layer -

OSCAL 101 Series - Lecture #3

THE LECTURE WILL START SOON!

National Institute of Standards and Technology
U.S. Department of Commerce

Presenter: **Dr. Michaela Iorga**
OSCAL Strategic Outreach Director

THE PRESENTATION IS BEEING RECORDED!

- NIST is hosting a series of monthly educational workshops, on the third Tuesday of each month, 11:00-12:00 EST.
- **Purpose**: improve OSCAL adoption by expanding the OSCAL community of interest (COI) through the onboarding of members who have no previous knowledge of OSCAL.
- Schedule and info: https://csrc.nist.gov/Projects/open-security-controls-assessment-language/oscal-education-workshops

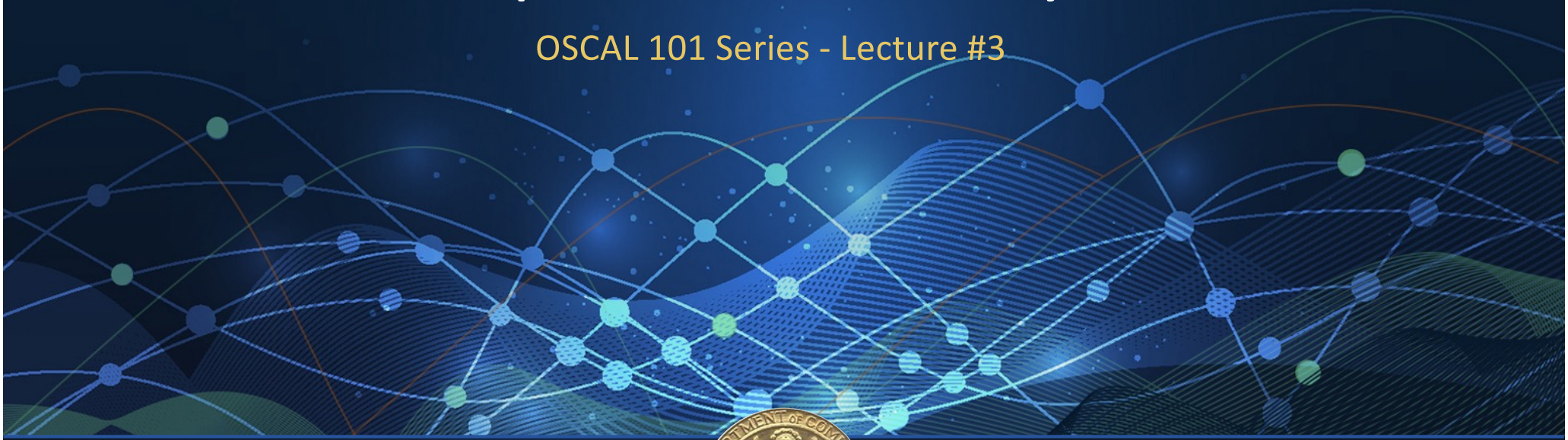
Welcome to the Lecture #3

Agenda

- Brief Recap of Lecture #1 & #2
- Continue with the Anatomy of OSCAL models
  - The Component Definition Model
  - The System Security Plan Model

Open Security Controls Assessment Language
The Anatomy of OSCAL Models
- Implementation Layer -

OSCAL 101 Series - Lecture #3

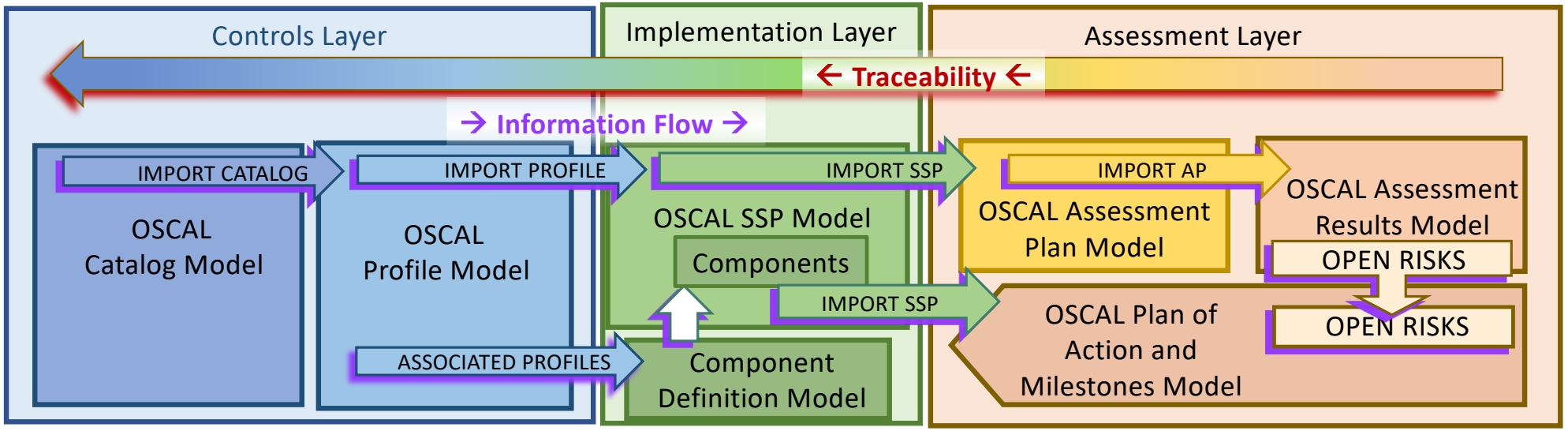


National Institute of Standards and Technology
U.S. Department of Commerce

Presenter: **Dr. Michaela Iorga**
OSCAL Strategic Outreach Director

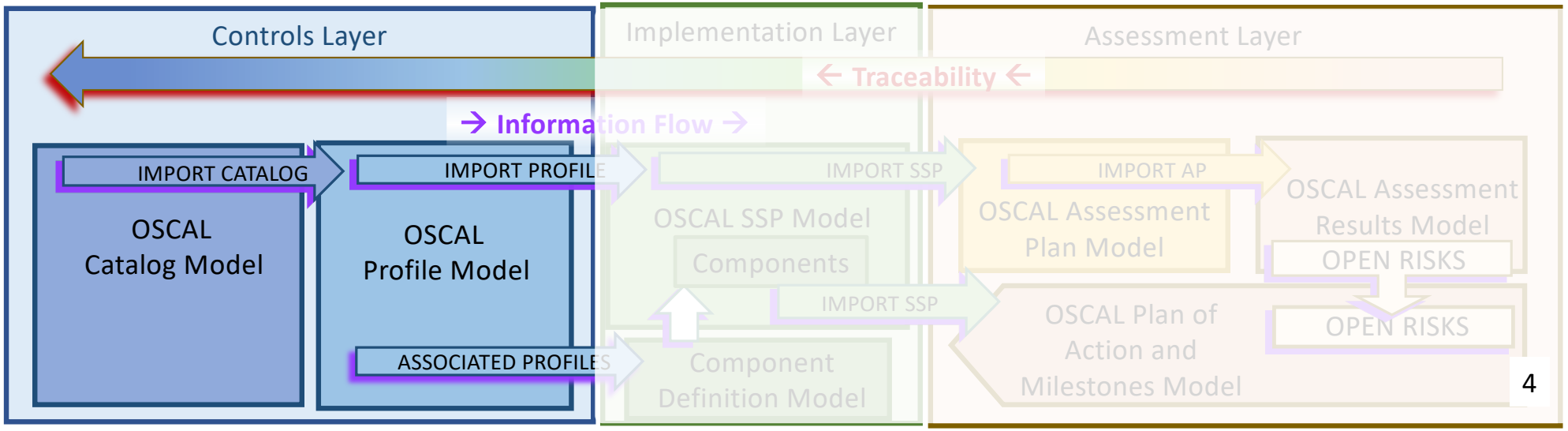
Recap: Lectures #1 & #2

- OSCAL is a standardized, flexible, open-source language designed to express security controls and their associated implementations and assessment methods in machine-readable formats (XML, JSON, and YAML). OSCAL content can be easily transformed into human-friendly formats.
- OSCAL:
  - Enables automated traceability
  - Provides a standards-based foundation for the next generation GRCs
  - Helps improve the risk management posture, consistency, and interoperability.



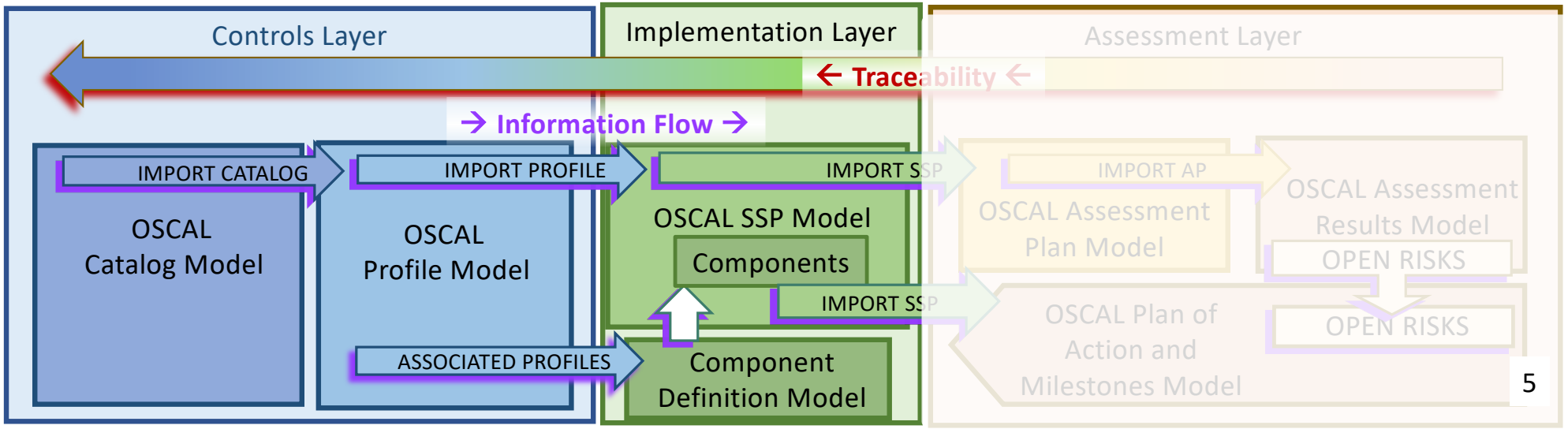
Recap: Lectures #1 & #2

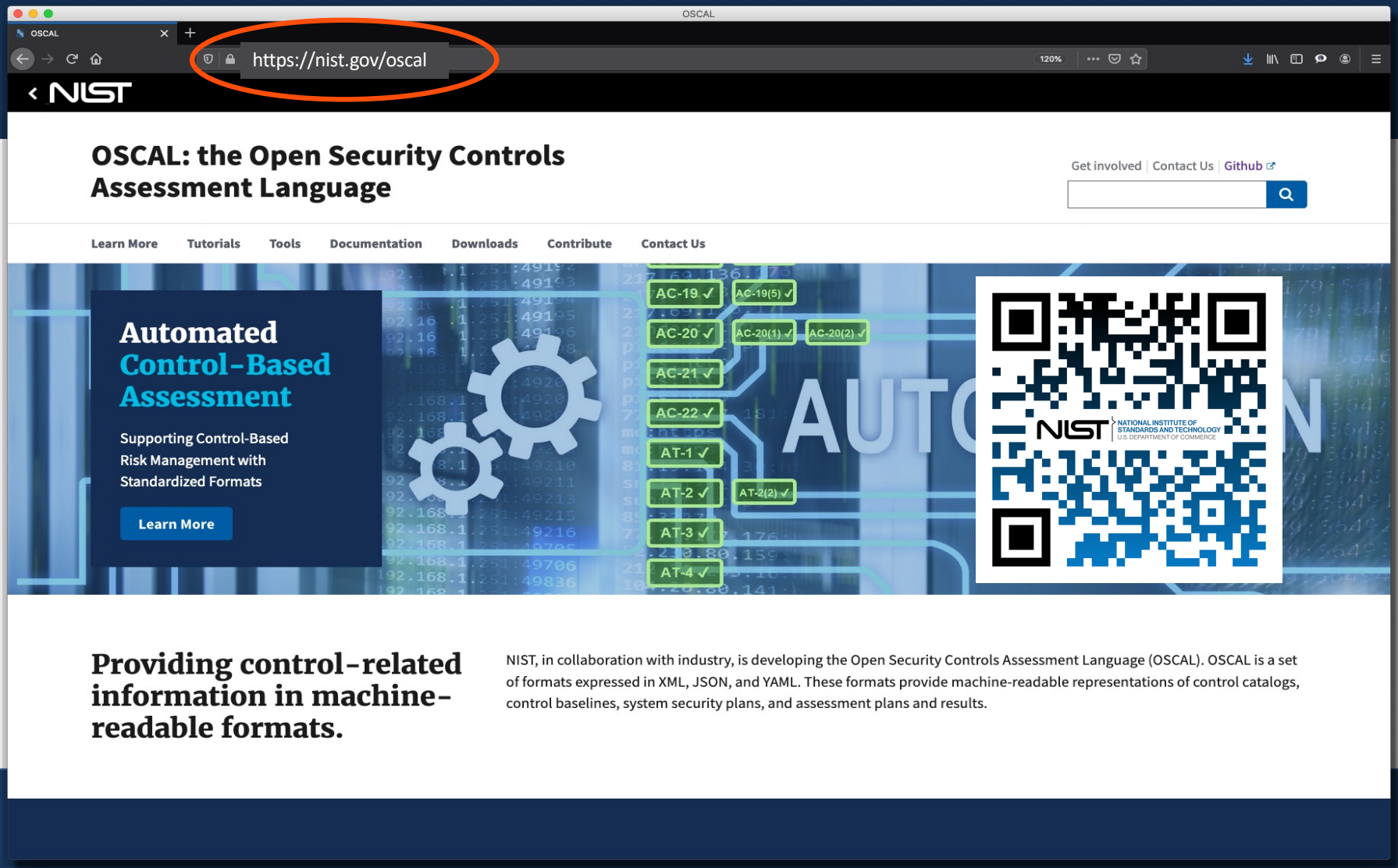
- OSCAL is a standardized, flexible, open-source language designed to express security controls and their associated implementations and assessment methods in machine-readable formats (XML, JSON, and YAML). OSCAL content can be easily transformed into human-friendly formats.
- OSCAL:
  - Enables automated traceability
  - Provides a standards-based foundation for the next generation GRCs
  - Helps improve the risk management posture, consistency, and interoperability.



Recap: Lectures #1 & #2

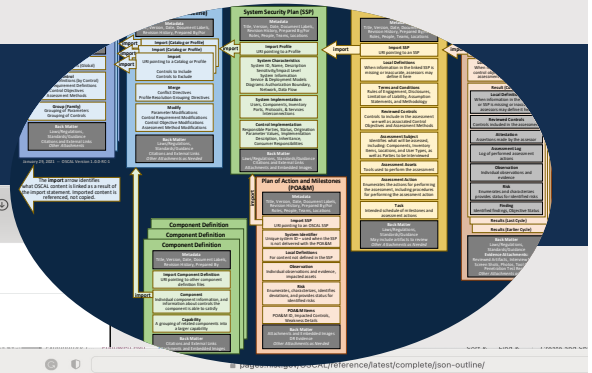
- OSCAL is a standardized, flexible, open-source language designed to express security controls and their associated implementations and assessment methods in machine-readable formats (XML, JSON, and YAML). OSCAL content can be easily transformed into human-friendly formats.
- OSCAL:
  - Enables automated traceability
  - Provides a standards-based foundation for the next generation GRCs
  - Helps improve the risk management posture, consistency, and interoperability.




OSCAL
https://nist.gov/oscal
NIST
OSCAL: the Open Security Controls Assessment Language
Get involved | Contact Us | Github
Learn More Tutorials Tools Documentation Downloads Contribute Contact Us
Automated Control-Based Assessment
Supporting Control-Based Risk Management with Standardized Formats
Learn More
AC-19 ✓
AC-19(5) ✓
AC-20 ✓
AC-20(1) ✓
AC-20(2) ✓
AC-21 ✓
AC-22 ✓
AT-1 ✓
AT-2 ✓
AT-2(2) ✓
AT-3 ✓
AT-4 ✓
NIST NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY U.S. DEPARTMENT OF COMMERCE
Providing control-related information in machine-readable formats.
NIST, in collaboration with industry, is developing the Open Security Controls Assessment Language (OSCAL). OSCAL is a set of formats expressed in XML, JSON, and YAML. These formats provide machine-readable representations of control catalogs, control baselines, system security plans, and assessment plans and results.

Recap: OSCAL Models' Outline

https://pages.nist.gov/OSCAL/reference/latest/complete/xml-outline/



OSCAL

About Learn Concepts Reference Downloads Tools Contribute Contact Us

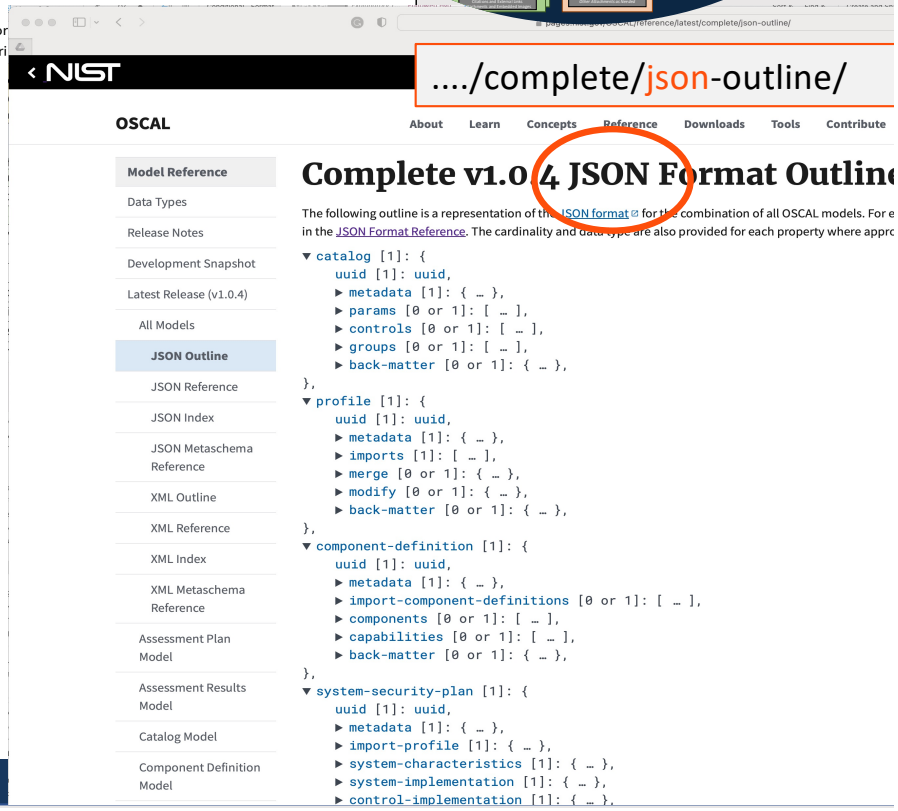
- Model Reference
- Data Types
- Release Notes
- Development Snapshot
- Latest Release (v1.0.4)
- All Models
  - JSON Outline
  - JSON Reference
  - JSON Index
  - JSON Metaschema Reference
  - XML Outline
  - XML Reference
  - XML Index
  - XML Metaschema Reference
  - Assessment Plan Model
  - Assessment Results Model
  - Catalog Model
  - Component Definition Model
  - Plan of Action and Milestones Model

Complete v1.0.4 XML Format Outline

The following outline is a representation of the XML format for the combination of all OSCAL models. For each element or attribute, see the corresponding entry in the XML Format Reference. The cardinality and data type are also provided for each element or attribute.

```
▼ <catalog uuid="uuid"> [1]
    ▶ <metadata> … </metadata> [1]
    ▶ <param id="token" class="token" depends-on="token"> … </param> [0 to ∞]
    ▶ <control id="token" class="token"> … </control> [0 to ∞]
    ▶ <group id="token" class="token"> … </group> [0 to ∞]
    ▶ <back-matter> … </back-matter> [0 or 1]
</catalog>
▼ <profile uuid="uuid"> [1]
    ▶ <metadata> … </metadata> [1]
    ▶ <import href="uri-reference"> … </import> [1 to ∞]
    ▶ <merge> … </merge> [0 or 1]
    ▶ <modify> … </modify> [0 or 1]
    ▶ <back-matter> … </back-matter> [0 or 1]
</profile>
▼ <component-definition uuid="uuid"> [1]
    ▶ <metadata> … </metadata> [1]
    ▶ <import-component-definition href="uri-reference"/> [0 to ∞]
    ▶ <component uuid="uuid" type="string"> … </component> [0 to ∞]
    ▶ <capability uuid="uuid" name="string"> … </capability> [0 to ∞]
    ▶ <back-matter> … </back-matter> [0 or 1]
</component-definition>
▼ <system-security-plan uuid="uuid"> [1]
    ▶ <metadata> … </metadata> [1]
    ▶ <import-profile href="uri-reference"> … </import-profile> [1]
    ▶ <system-characteristics> … </system-characteristics> [1]
    ▶ <system-implementation> … </system-implementation> [1]
    ▶ <control-implementation> … </control-implementation> [1]
    ▶ <back-matter> … </back-matter> [0 or 1]
</system-security-plan>
▶ <assessment-plan uuid="uuid"> … </assessment-plan> [1]
▶ <assessment-results uuid="uuid"> … </assessment-results> [1]
▶ <plan-of-action-and-milestones uuid="uuid"> … </plan-of-action-and-milestones> [1]
```

7



NIST

…./complete/json-outline/

OSCAL

About Learn Concepts Reference Downloads Tools Contribute

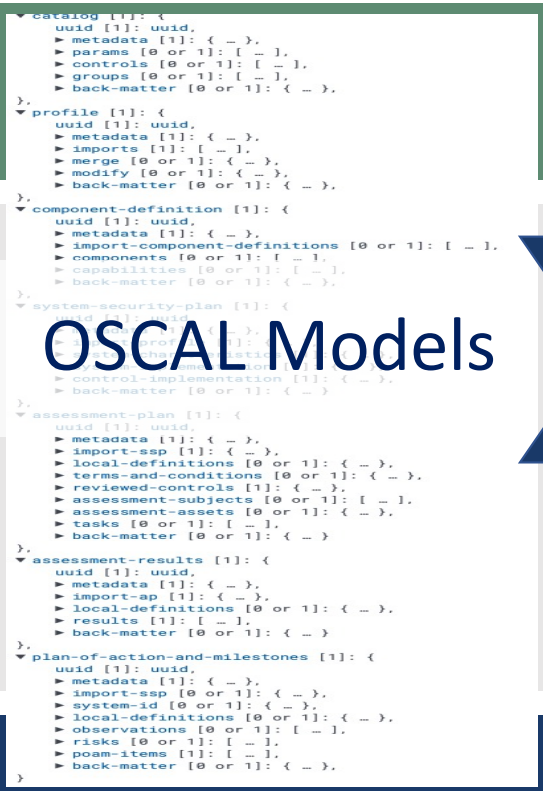
- Model Reference
- Data Types
- Release Notes
- Development Snapshot
- Latest Release (v1.0.4)
- All Models
  - JSON Outline
  - JSON Reference
  - JSON Index
  - JSON Metaschema Reference
  - XML Outline
  - XML Reference
  - XML Index
  - XML Metaschema Reference
  - Assessment Plan Model
  - Assessment Results Model
  - Catalog Model
  - Component Definition Model

Complete v1.0.4 JSON Format Outline

The following outline is a representation of the JSON format for the combination of all OSCAL models. For each element or attribute, see the corresponding entry in the JSON Format Reference. The cardinality and data type are also provided for each property where appropriate.

```
▼ catalog [1]: {
    uuid [1]: uuid,
    ▶ metadata [1]: { … },
    ▶ params [0 or 1]: [ … ],
    ▶ controls [0 or 1]: [ … ],
    ▶ groups [0 or 1]: [ … ],
    ▶ back-matter [0 or 1]: { … },
},
▼ profile [1]: {
    uuid [1]: uuid,
    ▶ metadata [1]: { … },
    ▶ imports [1]: [ … ],
    ▶ merge [0 or 1]: { … },
    ▶ modify [0 or 1]: { … },
    ▶ back-matter [0 or 1]: { … },
},
▼ component-definition [1]: {
    uuid [1]: uuid,
    ▶ metadata [1]: { … },
    ▶ import-component-definitions [0 or 1]: [ … ],
    ▶ components [0 or 1]: [ … ],
    ▶ capabilities [0 or 1]: [ … ],
    ▶ back-matter [0 or 1]: { … },
},
▼ system-security-plan [1]: {
    uuid [1]: uuid,
    ▶ metadata [1]: { … },
    ▶ import-profile [1]: { … },
    ▶ system-characteristics [1]: { … },
    ▶ system-implementation [1]: { … },
    ▶ control-implementation [1]: { … },
```

Recap: OSCAL Models >>> OSCAL Content >>> OSCAL Tools



OSCAL Models

https://github.com/usnistgov/OSCAL

OSCAL Content Generation

OSCAL Content in Action

https://github.com/usnistgov/oscal-content

Name	Provider/Developer	Description	Type
Compliance trestle	IBM	A python SDK and command line tool which manipulates OSCAL structures and supports transformation of data into OSCAL.	open source
OSCAL Java Library	NIST OSCAL Project	A Java-based programming API for reading and writing content conformant to the OSCAL XML, JSON, and YAML based models.	open source
OSCAL React Component Library	Easy Dynamics	A library of reusable React components and an example user interface application that provides a direct UI into OSCAL	open source
XSLT Tooling	NIST OSCAL Project		open source
XML Jelly Sandwich	Wendell Piez (NIST)	Interactive XSLT in the browser includes OSCAL demonstrations.	open source
Xacta 360			license
Atlasity: Continuous Compliance Automation	C2 Labs	development of OSCAL v1.0 content for Catalogs, Profiles, System Security Plans and Components. Additional detail can be found in this blog post: Atlasity Delivers Free Tools to Create OSCAL Content.	community edition

OSCAL Editorial Tools

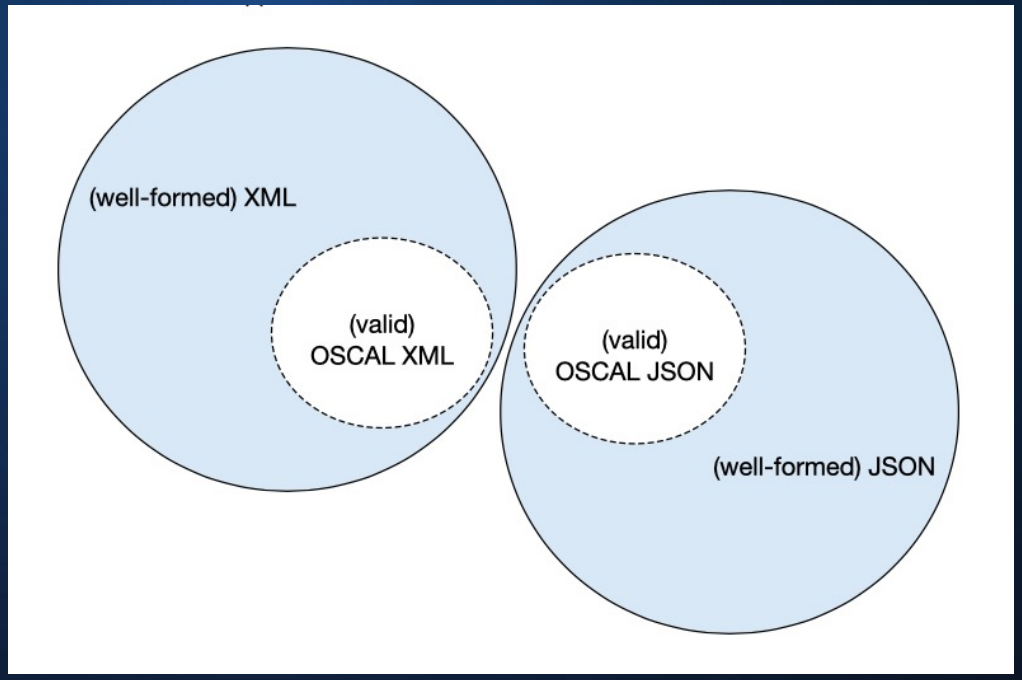
OSCAL GRC Tools

https://github.com/usnistgov/oscal-tools

Recap - OSCAL Content Validation

https://pages.nist.gov/OSCAL/concepts/validation/

"well-formed" vs "valid" OSCAL content



XML Schema Validators:
https://www.w3.org/XML/Schema#Tools

JSON Schema Validators:
https://json-schema.org/implementations.html#validators

Recap - Common OSCAL Structure

Complete v1.0.4 JSON Format Outline

The following outline is a representation of the JSON format for the combination of all OSCAL models. For each p
in the JSON Format Reference. The cardinality and data type are also provided for each property where appropriat

```
▼ catalog [1]: {
    uuid [1]: uuid,
    ▶ metadata [1]: { … },
    ▶ params [0 or 1]: [ … ],
    ▶ controls [0 or 1]: [ … ],
    ▶ groups [0 or 1]: [ … ],
    ▶ back-matter [0 or 1]: { … },
},
▼ profile [1]: {
    uuid [1]: uuid,
    ▶ metadata [1]: { … },
    ▶ imports [1]: [ … ],
    ▶ merge [0 or 1]: { … },
    ▶ modify [0 or 1]: { … },
    ▶ back-matter [0 or 1]: { … },
},
▼ component-definition [1]: {
    uuid [1]: uuid,
    ▶ metadata [1]: { … },
    ▶ import-component-definitions [0 or 1]: [ … ],
    ▶ components [0 or 1]: [ … ],
    ▶ capabilities [0 or 1]: [ … ],
    ▶ back-matter [0 or 1]: { … },
}
```

Root Element & Root UUID

Body (Model Specific)

Root Element & Root UUID

Body (Model Specific)

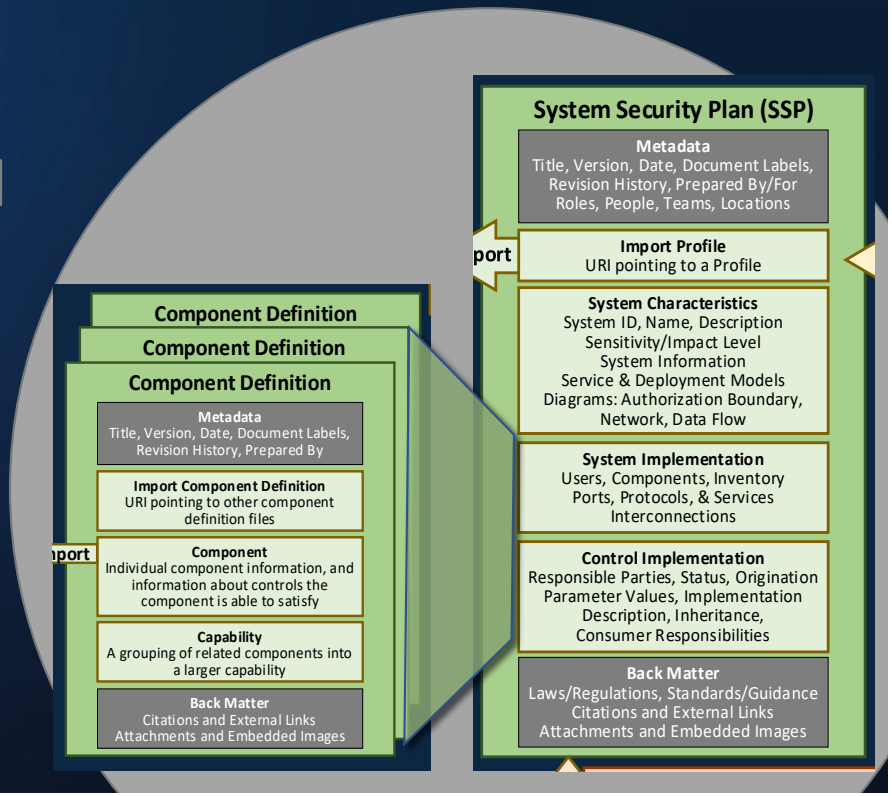
Root Element & Root UUID

Body (Model Specific)

OSCAL Implementation Layer

Component Definition Model

System Security Plan (SSP) Model



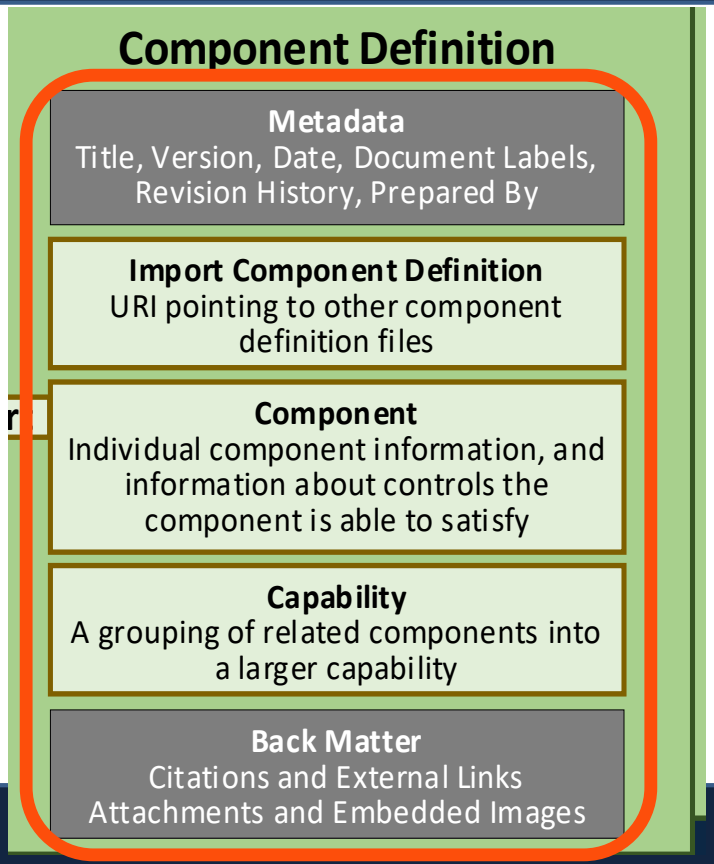
The Anatomy of a Component Definition

https://pages.nist.gov/OSCAL/reference/latest/component-definition/

Component Definition Model v1.0.4 JSON Format Outline

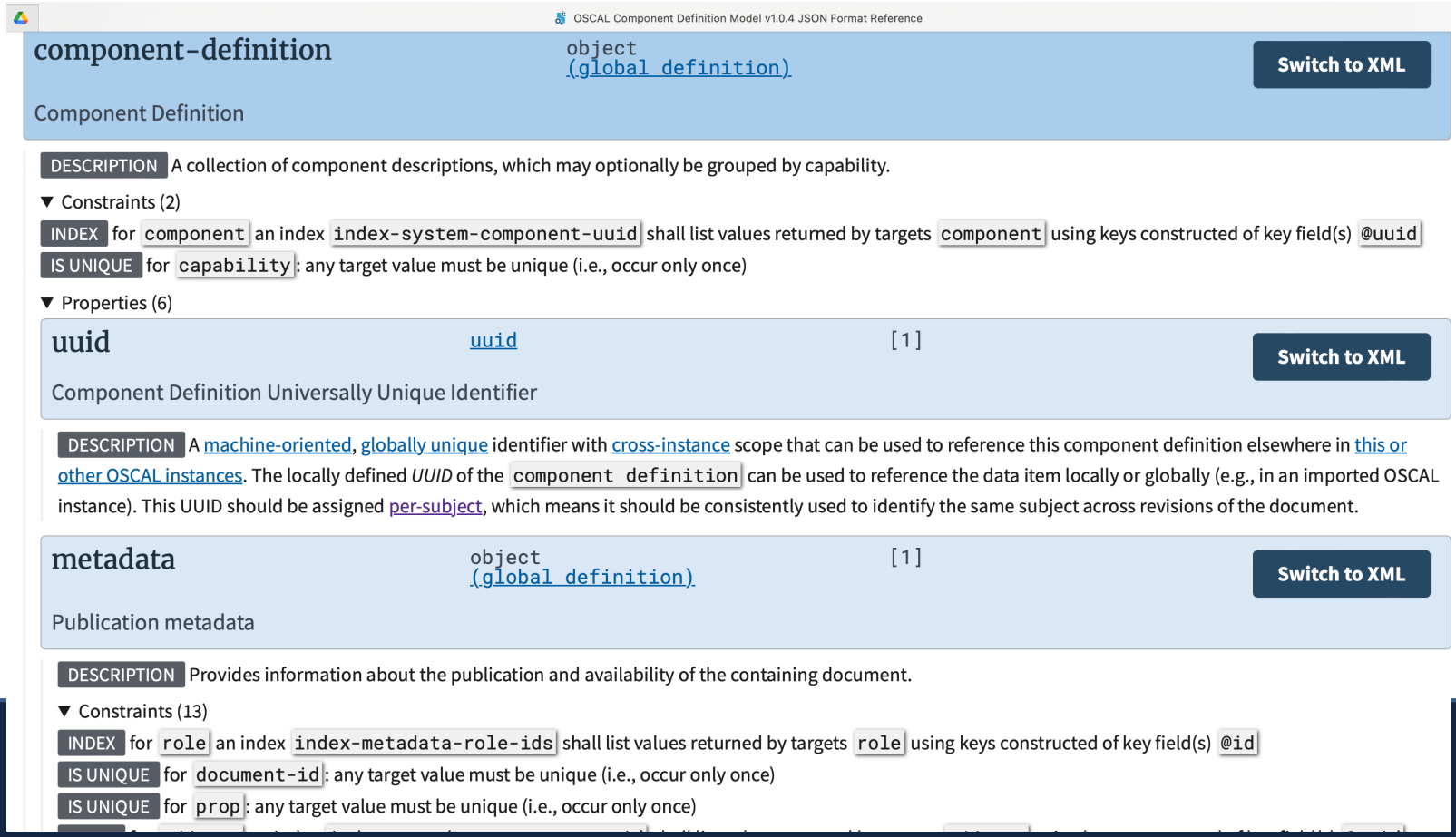
The following outline is a representation of the JSON format for this model. For each property, the name links to the corresponding entry in the JSON Format Reference. The cardinality and data type are also provided for each property where appropriate.

```
▼ component-definition [1]: {
    uuid [1]: uuid,
  ▶ metadata [1]: { … },
  ▶ import-component-definitions [0 or 1]: [ … ],
  ▶ components [0 or 1]: [ … ],
  ▶ capabilities [0 or 1]: [ … ],
  ▶ back-matter [0 or 1]: { … },
}
```



The Anatomy of a Component Definition - Body

https://pages.nist.gov/OSCAL/reference/latest/component-definition/json-reference/#/component-definition

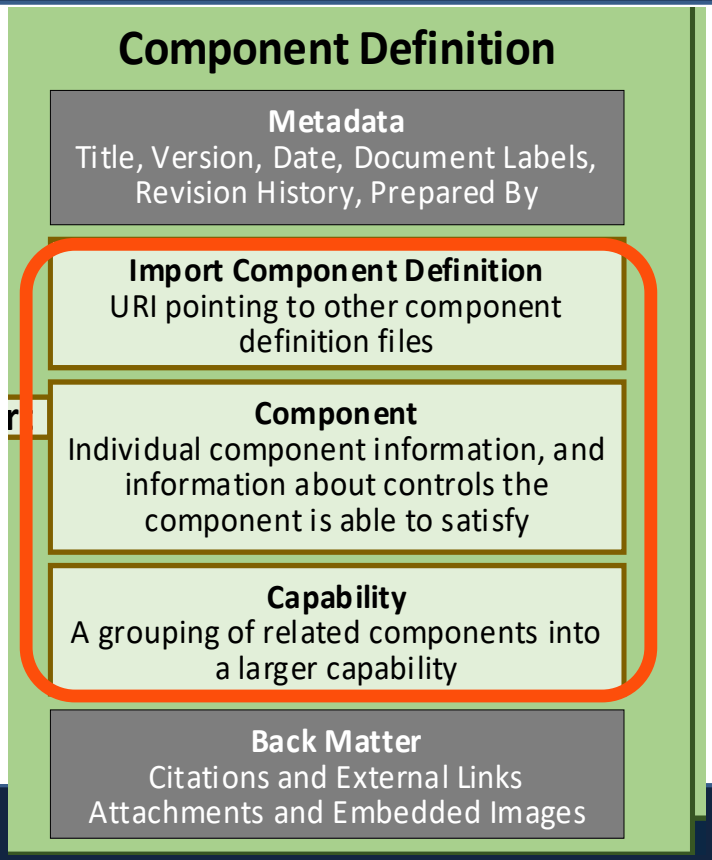


The Anatomy of a Component Definition - Body

https://pages.nist.gov/OSCAL/reference/latest/component-definition/

```
▼ component-definition [1]: {
      uuid [1]: uuid,
    ▶ metadata [1]: { … },
    ▶ import-component-definitions [0 or 1]
    ▶ components [0 or 1]: [ … ],
    ▶ capabilities [0 or 1]: [ … ],
    ▶ back-matter [0 or 1]: { … },
  }
```

Root Element & Root UUID



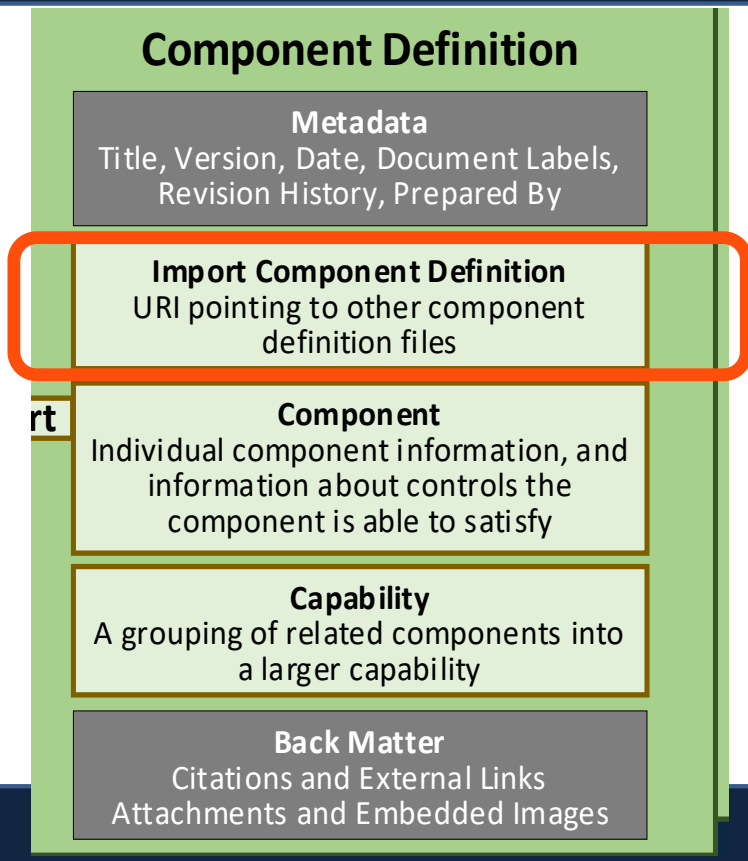
The Anatomy of a Component Definition - Body

https://pages.nist.gov/OSCAL/reference/latest/component-definition/

Loads a component definition from another resource.

```
▼ import-component-definitions [0 or 1]: [
    An array of import-component-definition objects
    [1 to ∞] {
        href [1]: uri-reference
    }

mponents [0 or 1]: [
    An array of component objects [1 to ∞] {
        uuid [1]: uuid,
        type [1]: string,
        title [1]: markup-line,
        description [1]: markup-multiline,
        purpose [0 or 1]: markup-line,
      ▶ props [0 or 1]: [ … ],
      ▶ links [0 or 1]: [ … ],
      ▶ responsible-roles [0 or 1]: [ … ],
      ▶ protocols [0 or 1]: [ … ],
      ▶ control-implementations [0 or 1]: [ … ],
        remarks [0 or 1]: markup-multiline,
    }
```

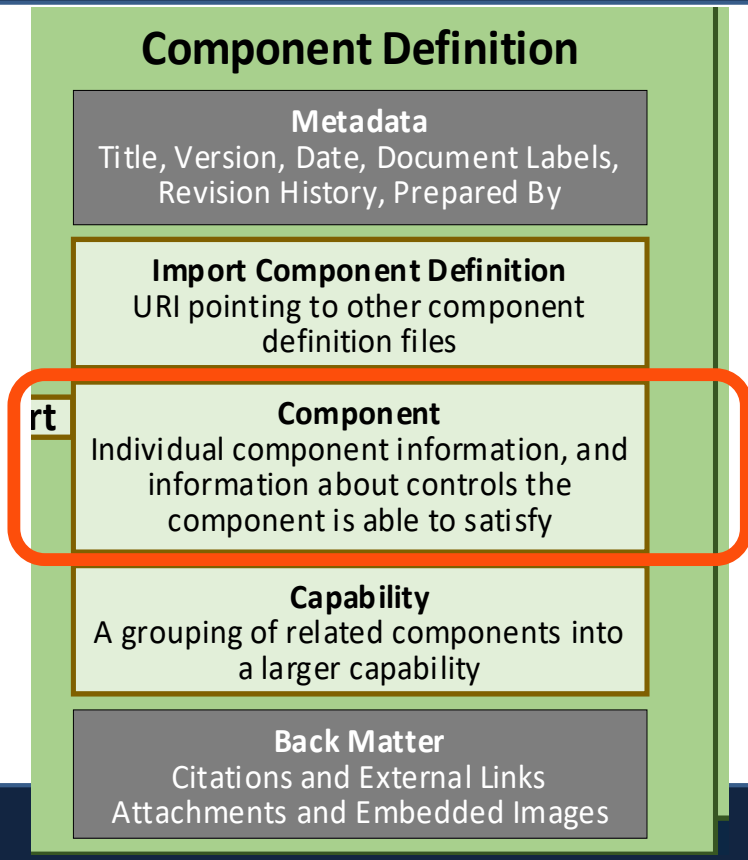


The Anatomy of a Component Definition - Components

https://pages.nist.gov/OSCAL/reference/latest/component-definition/

```
▼ import-component-definitions [0 or 1]: [
      An array of import-component-definition objects
      [1 to ∞] {
          href [1]: uri-reference
      }
  ]
▼ components [0 or 1]: [
      An array of component objects [1 to ∞] {
          uuid [1]: uuid,
          type [1]: string,
          title [1]: markup-line,
          description [1]: markup-multiline,
          purpose [0 or 1]: markup-line,
        ▶ props [0 or 1]: [ … ],
        ▶ links [0 or 1]: [ … ],
        ▶ responsible-roles [0 or 1]: [ … ],
        ▶ protocols [0 or 1]: [ … ],
        ▶ control-implementations [0 or 1]: [ … ],
          remarks [0 or 1]: markup-multiline,
      }
```

A defined component that can be part of an implemented system.

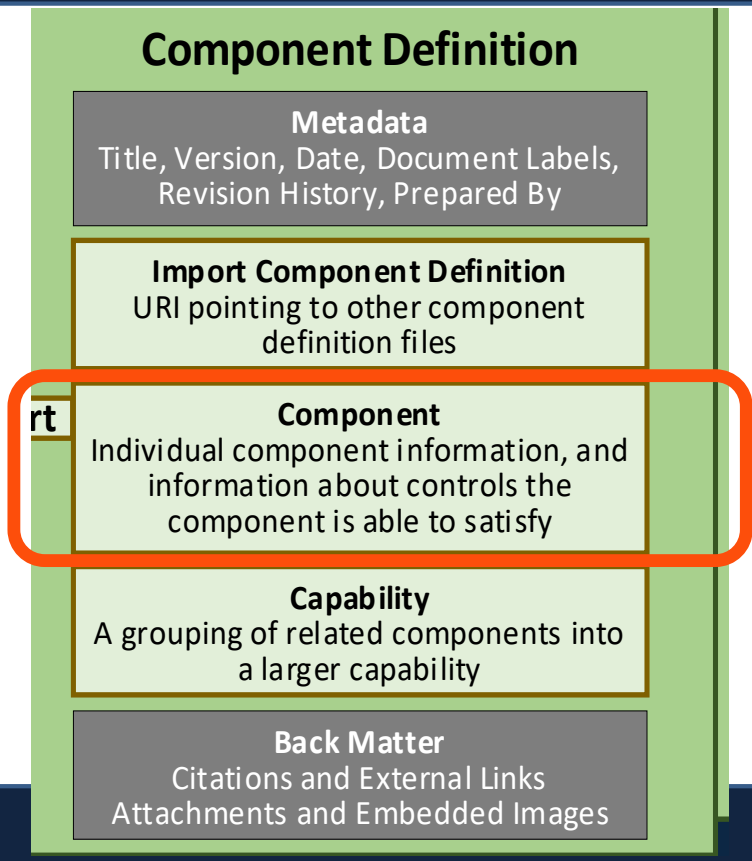


The Anatomy of a Component Definition - Components

https://pages.nist.gov/OSCAL/reference/latest/component-definition/

Defines how the component or capability supports a set of controls.

```
▼ control-implementations [0 or 1]: [
    An array of control-implementation objects [1 to ∞] {
      uuid [1]: uuid,
      source [1]: uri-reference,
      description [1]: markup-multiline,
      ▶ props [0 or 1]: [ … ],
      ▶ links [0 or 1]: [ … ],
      ▶ set-parameters [0 or 1]: [ … ],
      ▼ implemented-requirements [1]: [
          An array of implemented-requirement objects [1 to ∞] {
            uuid [1]: uuid,
            control-id [1]: token,
            description [1]: markup-multiline,
            ▶ props [0 or 1]: [ … ],
            ▶ links [0 or 1]: [ … ],
            ▶ set-parameters [0 or 1]: [ … ],
            ▶ responsible-roles [0 or 1]: [ … ],
            ▶ statements [0 or 1]: [ … ],
            remarks [0 or 1]: markup-multiline,
          }
      ],
    }
],
```

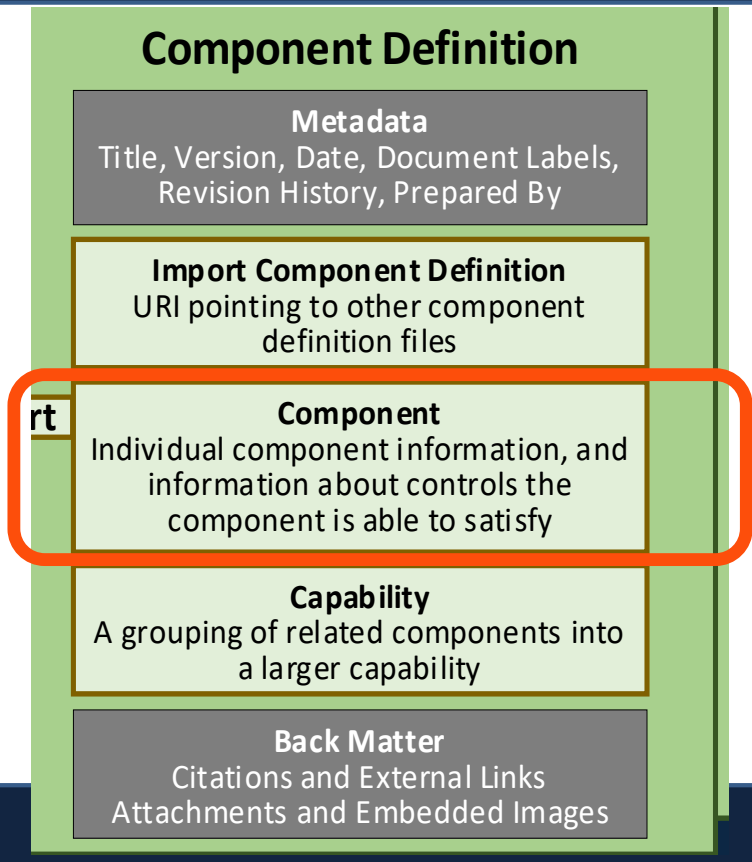


The Anatomy of a Component Definition - Components

https://pages.nist.gov/OSCAL/reference/latest/component-definition/

Describes how the component implements an individual control

```
▼ control-implementations [0 or 1]: [
    An array of control-implementation objects [1 to ∞] {
        uuid [1]: uuid,
        source [1]: uri-reference,
        description [1]: markup-multiline,
        ▶ props [0 or 1]: [ … ],
        ▶ links [0 or 1]: [ … ],
        ▶ set-parameters [0 or 1]: [ … ],
        ▼ implemented-requirements [1]: [
            An array of implemented-requirement objects [1 to ∞] {
                uuid [1]: uuid,
                control-id [1]: token,
                description [1]: markup-multiline,
                ▶ props [0 or 1]: [ … ],
                ▶ links [0 or 1]: [ … ],
                ▶ set-parameters [0 or 1]: [ … ],
                ▶ responsible-roles [0 or 1]: [ … ],
                ▶ statements [0 or 1]: [ … ],
                remarks [0 or 1]: markup-multiline,
            }
        ],
    }
],
```

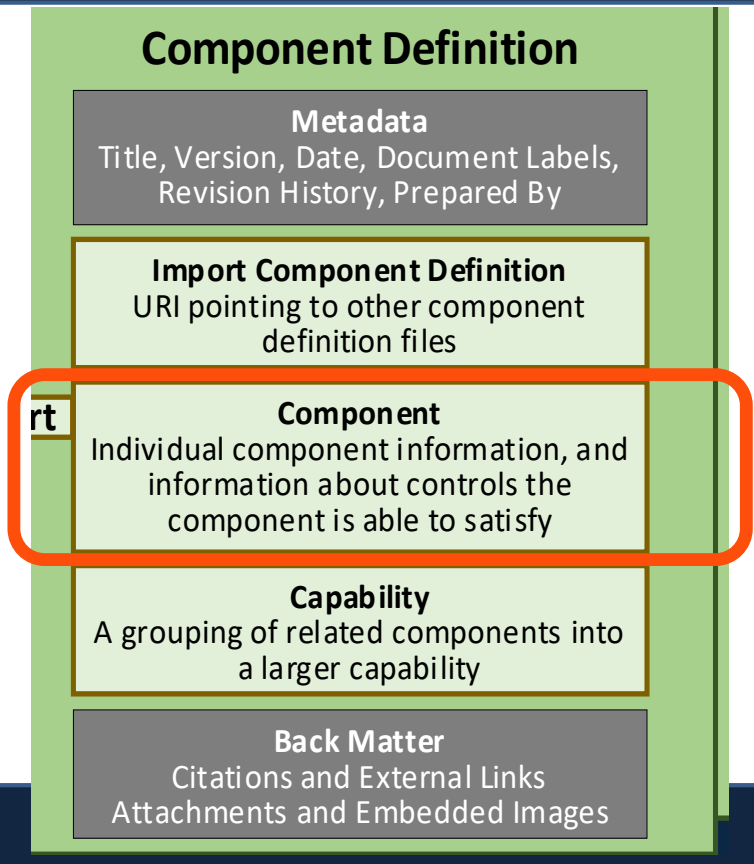


The Anatomy of a Component Definition - Components

https://pages.nist.gov/OSCAL/reference/latest/component-definition/

```
▼ control-implementations [0 or 1]: [
    An array of control-implementation objects [1 to ∞] {
        uuid [1]: uuid,
        source [1]: uri-reference,
        description [1]: markup-multiline,
        ▶ props [0 or 1]: [ … ],
        ▶ links [0 or 1]: [ … ],
        ▶ set-parameters [0 or 1]: [ … ],
        ▼ implemented-requirements [1]: [
            An array of implemented-requirement objects [1 to ∞] {
                uuid [1]: uuid,
                control-id [1]: token,
                description [1]: markup-multiline,
                ▶ props [0 or 1]: [ … ],
                ▶ links [0 or 1]: [ … ],
                ▶ set-parameters [0 or 1]: [ … ],
                ▶ responsible-roles [0 or 1]: [ … ],
                ▶ statements [0 or 1]: [ … ],
                remarks [0 or 1]: markup-multiline,
            }
        ],
    }
],
```

Identifies which statements within a control are addressed.



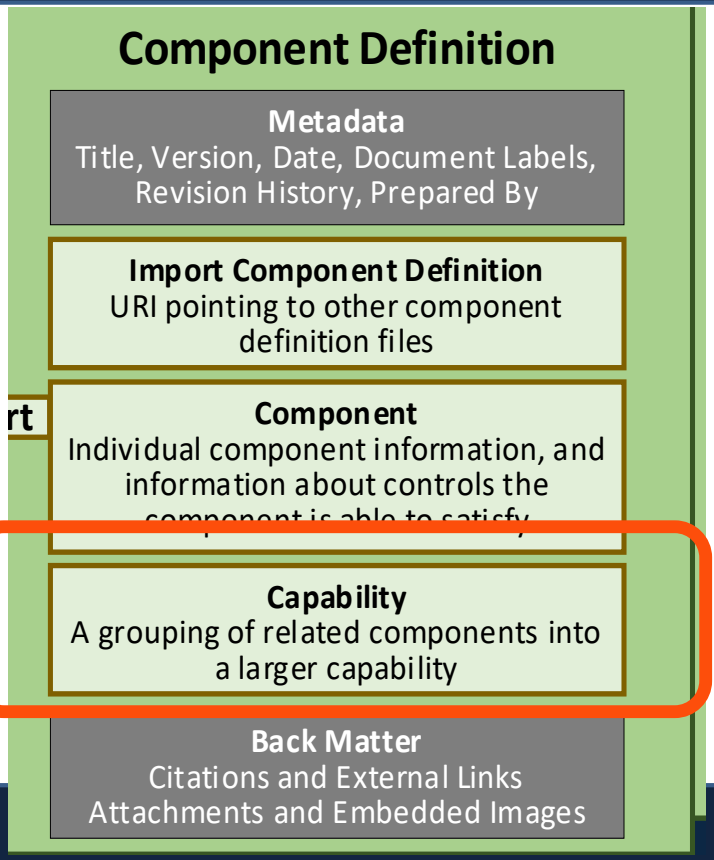
The Anatomy of a Component Definition - Capabilities

https://pages.nist.gov/OSCAL/reference/latest/component-definition/

A grouping of other components and/or capabilities.

```
▼ capabilities [0 or 1]:
    An array of capability objects [1 to ∞] {
        uuid [1]: uuid,
        name [1]: string,
        description [1]: markup-multiline,
      ▶ props [0 or 1]: [ … ],
      ▶ links [0 or 1]: [ … ],
      ▶ incorporates-components [0 or 1]: [ … ],
      ▶ control-implementations [0 or 1]: [ … ],
        remarks [0 or 1]: markup-multiline,
    }
  ],
▼ back-matter [0 or 1]: {
    ▶ resources [0 or 1]: [ … ],
  },
}
```

Model Reference



An Example of a Component Definition Instance

https://github.com/usnistgov/oscal-content/blob/main/examples/component-definition/json/example-component-definition.json

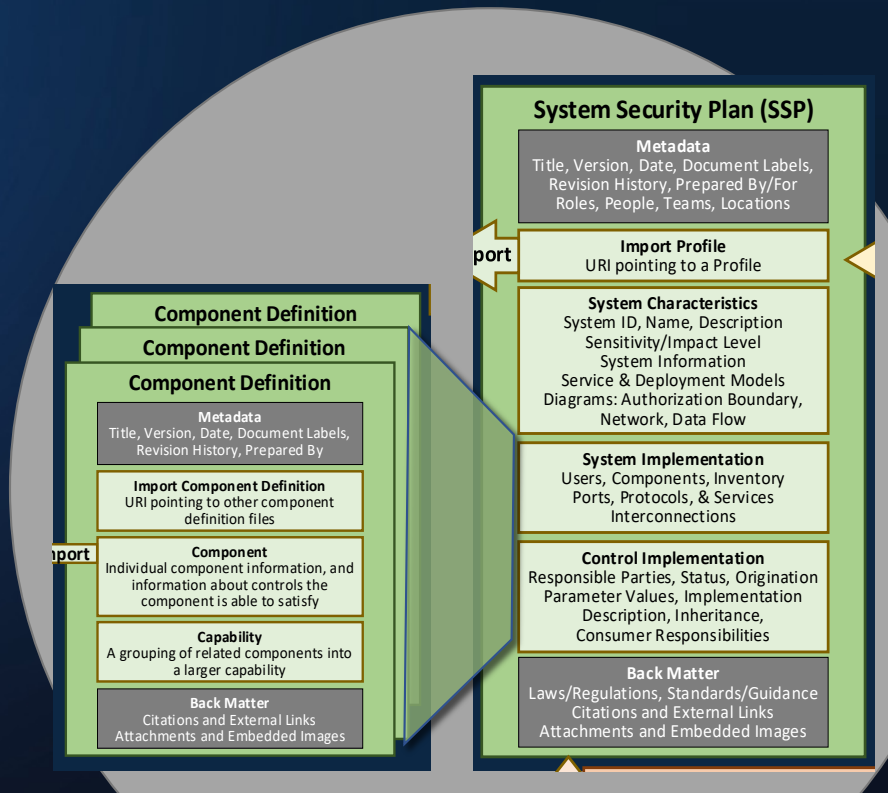

```
{
  "component-definition": {
    "uuid": "a7ba800c-a432-44cd-9075-0862cd66da6b",
    "metadata": {…
    },
    "components": [
      {
        "uuid": "91f646c5-b1b6-4786-9ec3-2305a044e217",
        "type": "software",
        "title": "MongoDB",
        "description": "MongoDB is a source-available, cross-platform document-oriented database program.
        "purpose": "Provides a NoSQL database service",
        "responsible-roles": […
        ],
        "protocols": […
        ],
        "control-implementations": [
          {
            "uuid": "49f0b690-ed9f-4f32-aae0-625b77aa6d27",
            "source": "https://github.com/usnistgov/oscal-content/blob/master/nist.gov/SP800-53/rev5/xml/
            "description": "MongoDB control implementations for NIST SP 800-53 revision 5.",
            "implemented-requirements": [
              {
                "uuid": "cf8338c5-fb6e-4593-a4a8-b3c4946ee2a0",
                "control-id": "sc-8.1",
                "description": "MongoDB supports TLS 1.x to encrypt data in transit, preventing unauthori
              },
              {
                "uuid": "cf8338c5-fb6e-4593-a4a8-b3c4946ee2a0",
                "control-id": "sa-4.9",
                "description": "Must ensure that MongoDB only listens for network connections on authoriz
              }
```


NOTE: some properties (e.g. `responsible-roles`, `protocols`, etc.) are collapsed

OSCAL Implementation Layer

Component Definition Model

System Security Plan (SSP) Model



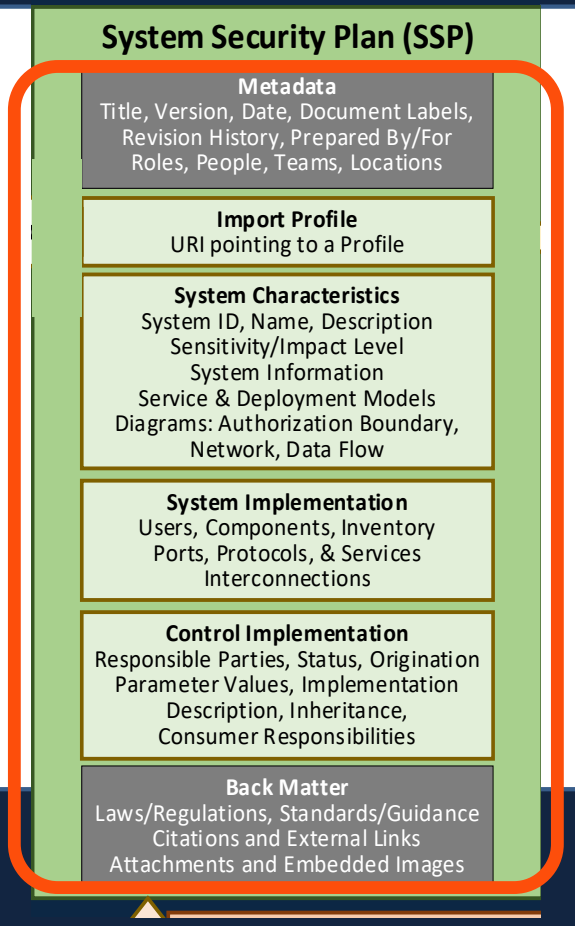
The Anatomy of the SSP Model – Body

https://pages.nist.gov/OSCAL/reference/latest/system-security-plan/json-outline/

System Security Plan Model v1.0.4 JSON Format Outline

The following outline is a representation of the JSON format for this model. For each property, the name links to the corresponding entry in the JSON Format Reference. The cardinality and data type are also provided for each property where appropriate.

```
▼ system-security-plan [1]: {
    uuid [1]: uuid,
  ▶ metadata [1]: { … },
  ▶ import-profile [1]: { … },
  ▶ system-characteristics [1]: { … },
  ▶ system-implementation [1]: { … },
  ▶ control-implementation [1]: { … },
  ▶ back-matter [0 or 1]: { … }
}
```

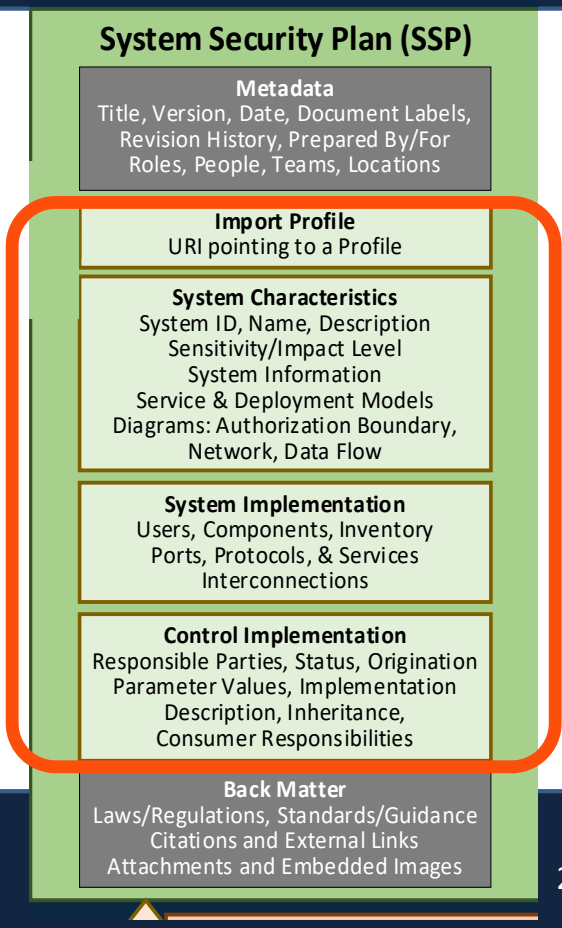


The Anatomy of the SSP Model – Body

https://pages.nist.gov/OSCAL/reference/latest/system-security-plan/json-outline/

```
▼ system-security-plan [1]: {
    uuid [1]: uuid,
  ▶ metadata [1]: { … },
  ▶ import-profile [1]: { … },
  ▶ system-characteristics [1]: { … },
  ▶ system-implementation [1]: { … },
  ▶ control-implementation [1]: { … },
  ▶ back-matter [0 or 1]: { … }
}
```

Root Element & Root UUID



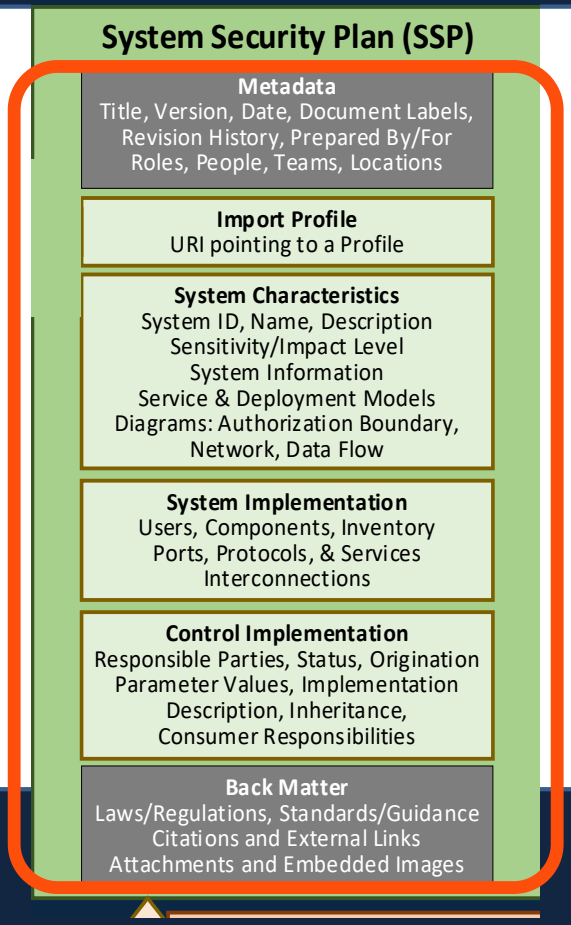
The Anatomy of the SSP Model – Body

https://pages.nist.gov/OSCAL/reference/latest/system-security-plan/json-outline/

System Security Plan Model v1.0.4 JSON Format Outline

The following outline is a representation of the JSON format for this model. For each property, the name links to the corresponding entry in the JSON Format Reference. The cardinality and data type are also provided for each property where appropriate.

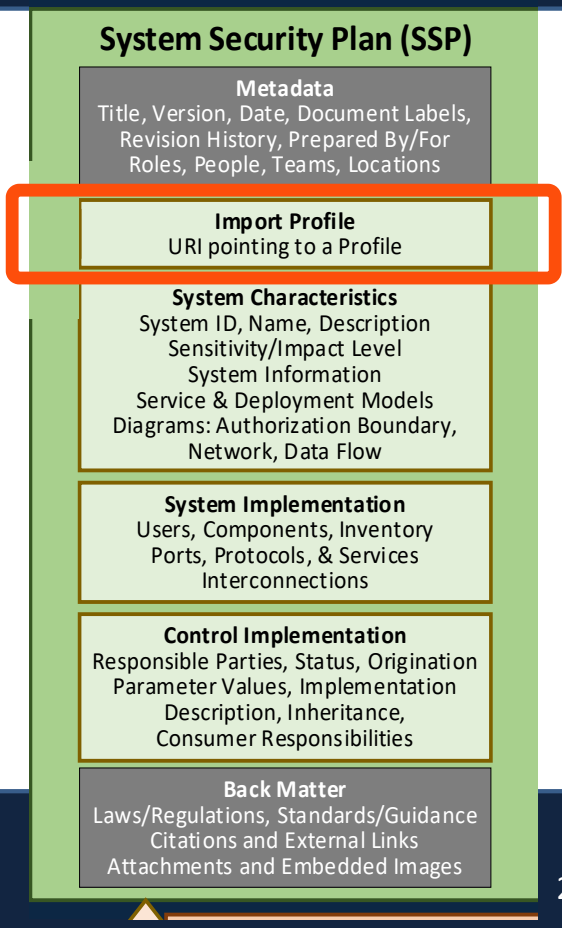
```
▼ system-security-plan [1]: {
    uuid [1]: uuid,
    ▶ metadata [1]: { … },
    ▶ import-profile [1]: { … },
    ▶ system-characteristics [1]: { … },
    ▶ system-implementation [1]: { … },
    ▶ control-implementation [1]: { … },
    ▶ back-matter [0 or 1]: { … }
}
```



The Anatomy of the SSP Model – Body

https://pages.nist.gov/OSCAL/reference/latest/system-security-plan/json-outline/

```
▼ system-security-plan [1]: {
    uuid [1]: uuid,
    ▶ metadata [1]: { … },
    ▼ import-profile [1]: {
        href [1]: uri-reference,
        remarks [0 or 1]: markup-multiline
    },
    ▶ system-characteristics [1]: { … },
    ▶ system-implementation [1]: { … },
    ▶ control-implementation [1]: { … },
    ▶ back-matter [0 or 1]: { … }
}
```



The Anatomy of the SSP Model – Body

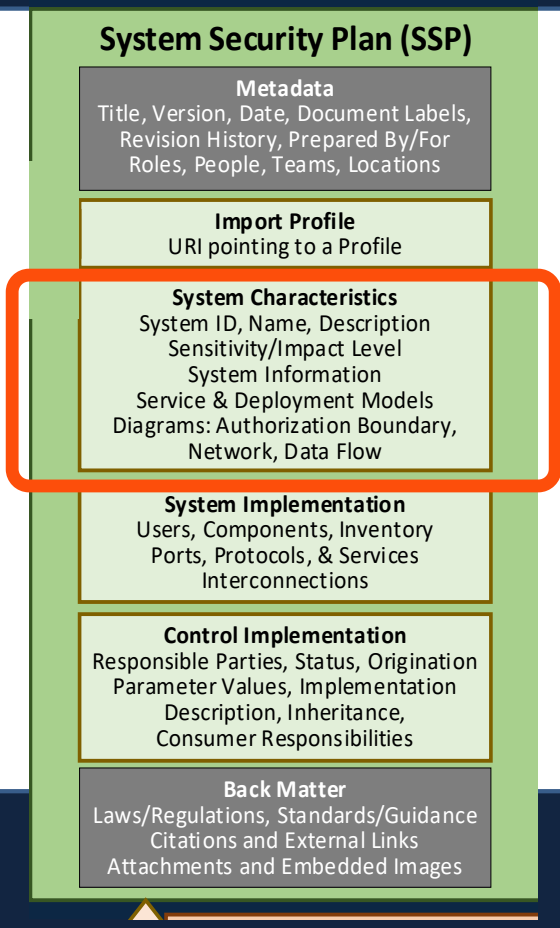
https://pages.nist.gov/OSCAL/reference/latest/system-security-plan/json-outline/

The overall information system sensitivity categorization, such as defined by FIPS-199.

Contains details about all information types that are stored, processed, or transmitted by the system, such as privacy information, and those defined in NIST SP 800-60

The overall level of expected impact caused by unauthorized disclosure, modification, or loss of access to the information.

```
▼ system-security-plan [1]: {
    uuid [1]: uuid,
    ▶ metadata [1]: { … },
    ▶ import-profile [1]: { … },
    ▼ system-characteristics [1]: {
        ▶ system-ids [1]: [ … ],
        system-name [1]: string,
        system-name-short [0 or 1]: string,
        description [1]: markup-multiline,
        ▶ props [0 or 1]: [ … ],
        ▶ links [0 or 1]: [ … ],
        date-authorized [0 or 1]: date,
        security-sensitivity-level [1]: string,
        ▶ system-information [1]: { … },
        ▶ security-impact-level [1]: { … },
        ▶ status [1]: { … },
        ▶ authorization-boundary [1]: { … },
        ▶ network-architecture [0 or 1]: { … },
        ▶ data-flow [0 or 1]: { … },
        ▶ responsible-parties [0 or 1]: [ … ],
        remarks [0 or 1]: markup-multiline,
    },
    ▶ system-implementation [1]: { … },
```



The Anatomy of the SSP Model – Body

https://pages.nist.gov/OSCAL/reference/latest/system-security-plan/json-outline/

A description of another authorized system from which this system inherits capabilities that satisfy security requirements. Another term for this concept is a *common control provider*.

```
▼ component-definition [1]: {
    uuid [1]: uuid,
    ▶ metadata [1]: { … },
    ▶ import-component-definitions [0 or 1]: [ … ]
    ▼ components [0 or 1]: [
        An array of component objects [1 to ∞] {
            uuid [1]: uuid,
            type [1]: string,
            title [1]: markup-line,
            description [1]: markup-multiline,
            purpose [0 or 1]: markup-line,
            ▶ props [0 or 1]: [ … ],
            ▶ links [0 or 1]: [ … ],
            ▶ responsible-roles [0 or 1]: [ … ],
            ▶ protocols [0 or 1]: [ … ],
            ▶ control-implementations [0 or 1]: [
            remarks [0 or 1]: markup-multiline,
        }
    ],
    ▶ capabilities [0 or 1]: [ … ],
    ▶ back-matter [0 or 1]: { … },
}
```

```
▼ system-security-plan [1]: {
    uuid [1]: uuid,
    ▶ metadata [1]: { … },
    ▶ import-profile [1]: { … },
    ▼ system-implementation [1]: {
        ▶ props [0 or 1]: [ … ],
        ▶ links [0 or 1]: [ … ],
        ▶ leveraged-authorizations [0 or 1]: [ … ],
        ▶ users [1]: [ … ],
        ▼ components [1]: [
            An array of component objects [1 to ∞] {
                uuid [1]: uuid,
                type [1]: string,
                title [1]: markup-line,
                description [1]: markup-multiline,
                purpose [0 or 1]: markup-line,
                ▶ props [0 or 1]: [ … ],
                ▶ links [0 or 1]: [ … ],
                ▶ status [1]: { … },
                ▶ responsible-roles [0 or 1]: [ … ],
                ▶ protocols [0 or 1]: [ … ],
                remarks [0 or 1]: markup-multiline,
            }
        ],
        ▶ inventory-items [0 or 1]: [ … ],
        remarks [0 or 1]: markup-multiline,
    },
    ▶ control-implementation [1]: { … },
```

System Security Plan (SSP)

Metadata
Title, Version, Date, Document Labels, Revision History, Prepared By/For Roles, People, Teams, Locations

Import Profile
URI pointing to a Profile

System Characteristics
System ID, Name, Description
Sensitivity/Impact Level
System Information
Service & Deployment Models
Diagrams: Authorization Boundary, Network, Data Flow

System Implementation
Users, Components, Inventory
Ports, Protocols, & Services
Interconnections

Control Implementation
Responsible Parties, Status, Origination
Parameter Values, Implementation Description, Inheritance, Consumer Responsibilities

Back Matter
Laws/Regulations, Standards/Guidance
Citations and External Links
Attachments and Embedded Images

The Anatomy of the SSP Model – Body

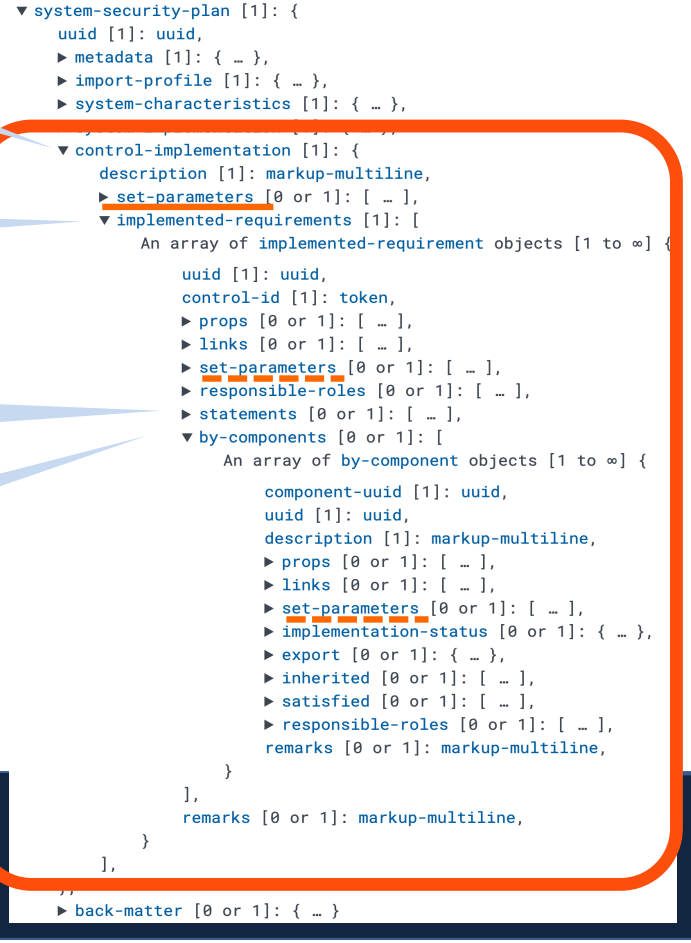
https://pages.nist.gov/OSCAL/reference/latest/system-security-plan/json-outline/

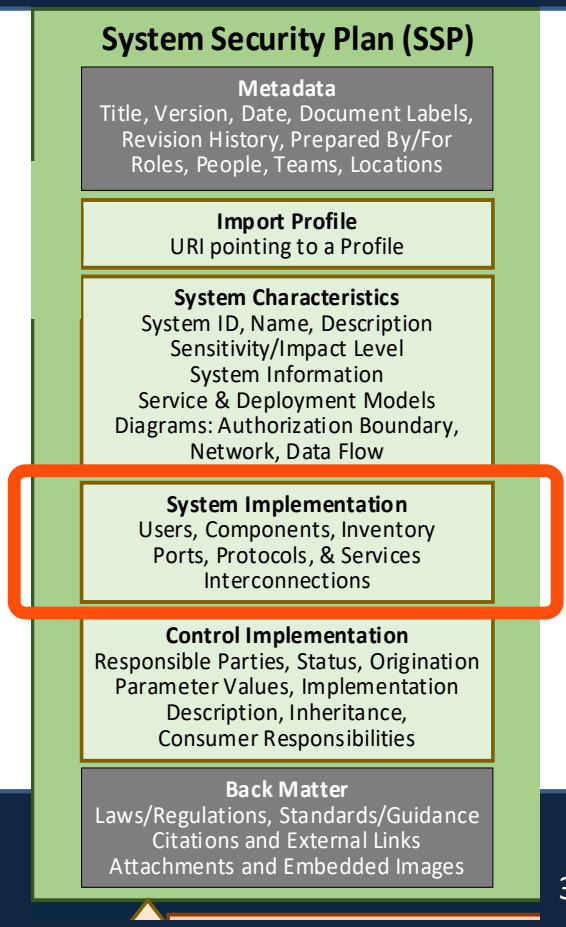
Describes how the system satisfies a set of controls.

Describes how the system satisfies the requirements of an individual control.

Identifies which statements within a control are addressed. The information can be provided for each component of the system

Defines how the referenced component implements a set of controls.

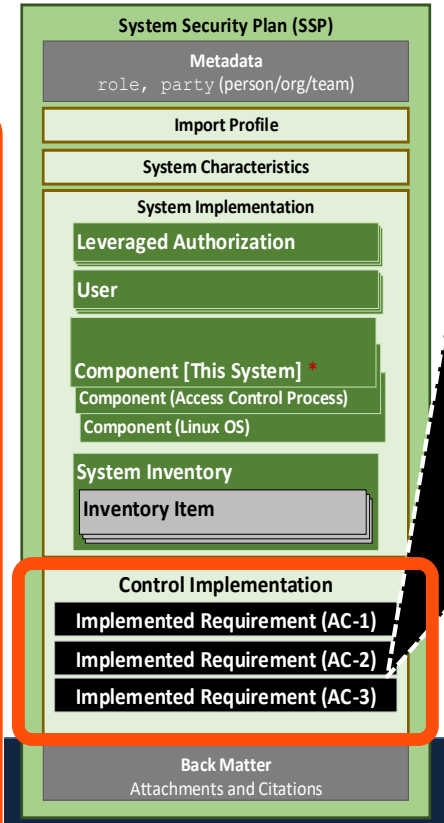


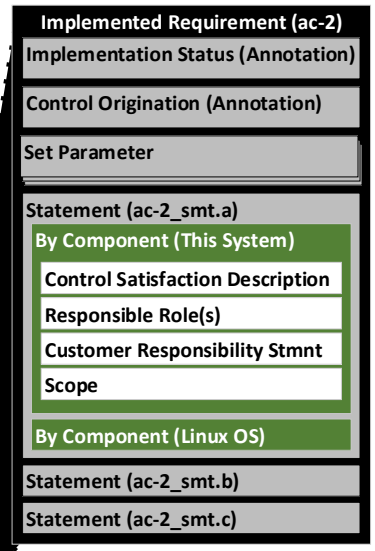


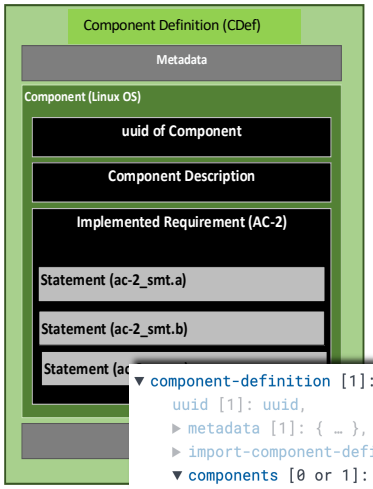
The Anatomy of the SSP Model – Body

https://pages.nist.gov/OSCAL/reference/latest/system-security-plan/json-outline/

```
▼ system-security-plan [1]: {
    uuid [1]: uuid,
    ▶ metadata [1]: { … },
    ▶ import-profile [1]: { … },
    ▶ system-characteristics [1]: { … },
    ▶ system-implementation [1]: { … },
    ▼ control-implementation [1]: {
        description [1]: markup-multiline,
        ▶ set-parameters [0 or 1]: [ … ],
        ▼ implemented-requirements [1]: [
            An array of implemented-requirement objects [1 to ∞]
                uuid [1]: uuid,
                control-id [1]: token,
                ▶ props [0 or 1]: [ … ],
                ▶ links [0 or 1]: [ … ],
                ▶ set-parameters [0 or 1]: [ … ],
                ▶ responsible-roles [0 or 1]: [ … ],
                ▶ statements [0 or 1]: [ … ],
                ▼ by-components [0 or 1]: [
                    An array of by-component objects [1 to ∞] {
                        component-uuid [1]: uuid,
                        uuid [1]: uuid,
                        description [1]: markup-multiline,
                        ▶ props [0 or 1]: [ … ],
                        ▶ links [0 or 1]: [ … ],
                        ▶ set-parameters [0 or 1]: [ … ],
                        ▶ implementation-status [0 or 1]: { … },
                        ▶ export [0 or 1]: { … },
                        ▶ inherited [0 or 1]: [ … ],
                        ▶ satisfied [0 or 1]: [ … ],
                        ▶ responsible-roles [0 or 1]: [ … ],
                        remarks [0 or 1]: markup-multiline,
                    }
                ],
                remarks [0 or 1]: markup-multiline,
            }
        ],
    }
    ▶ back-matter [0 or 1]: { … }
```








```
▼ component-definition [1]: {
    uuid [1]: uuid,
    ▶ metadata [1]: { … },
    ▶ import-component-definitions [0 or 1]: [ … ],
    ▼ components [0 or 1]: [
        An array of component objects [1 to ∞] {
            uuid [1]: uuid,
            type [1]: string,
            title [1]: markup-line,
            description [1]: markup-multiline,
            purpose [0 or 1]: markup-line,
            ▶ props [0 or 1]: [ … ],
            ▶ links [0 or 1]: [ … ],
            ▶ responsible-roles [0 or 1]: [ … ],
            ▶ protocols [0 or 1]: [ … ],
            ▶ control-implementations [0 or 1]: [
            remarks [0 or 1]: markup-multiline,
        }
    ],
    ▶ capabilities [0 or 1]: [ … ],
    ▶ back-matter [0 or 1]: { … },
}
```

Simple OSCAL SSP Examples

- Please visit https://github.com/usnistgov/oscal-content/tree/main/examples/ssp for some simple SSP examples in:
  - XML
  - JSON
  - YAML

Thank you!

OSCAL is a community-driven program!

Please join us!

OSCAL Catalog Tutorial:
https://pages.nist.gov/OSCAL/learn/tutorials/control/basic-catalog/

https://www.nist.gov/OSCAL

Contact us at: oscal@nist.gov

Subscribe to our mailing lists: oscal-dev@list.nist.gov or oscal-updates@list.nist.gov

Chat with us on Gitter: https://gitter.im/usnistgov-OSCAL/Lobby

Collaborate with us on GitHub: https://github.com/usnistgov/OSCAL

Join our COI meetings: https://pages.nist.gov/OSCAL/contribute/#community-meetings

OSCAL News / Events

Open Floor Discussion

NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
U.S. DEPARTMENT OF COMMERCE

Ground Rules of Engagement

- Keep the discussion respectful by:
  - using welcoming and inclusive language
  - being respectful of differing viewpoints and experiences
  - gracefully accepting constructive criticism
  - wait for one speaker to finish before speaking
- Speak from your own experience instead of generalizing.
- Do not be afraid to respectfully challenge one another by asking questions focused on ideas not on the company or presenter.
- The final goal is not to always agree but rather gain a deeper understanding.