Allotrope Simple Models (ASM) Validation and Hands-On Demonstration

Allotrope Virtual Connect, Fall 2021

September 2021

Rethinking Scientific Data

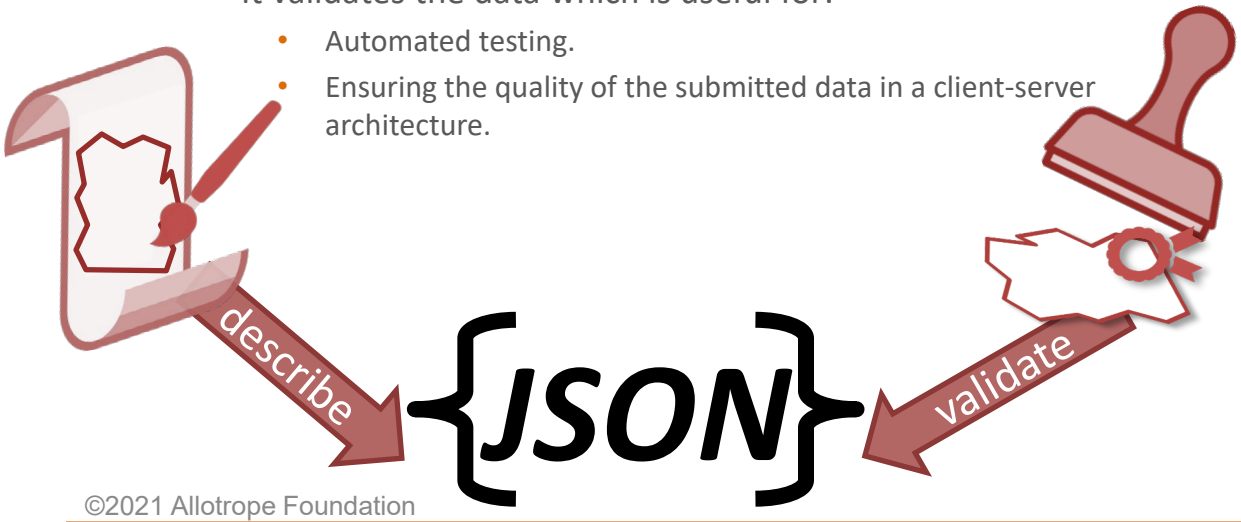
JSON Schema Primer

©2021 Allotrope Foundation

*This primer is derived from https://json-schema.org/

JSON Schema Primer

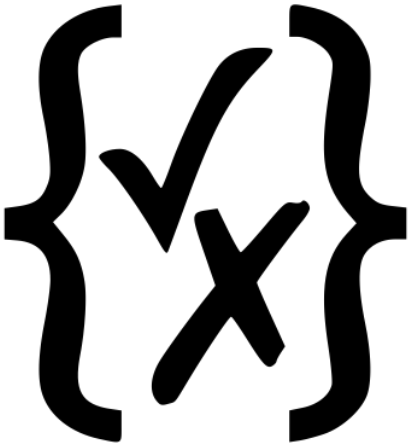
- ***JSON Schema*** is a vocabulary that allows the annotation and validation of ***JSON*** documents
  - It provides a detailed description for the data format
  - It provides clear human & machine - readable documentation.
  - It validates the data which is useful for:
    - Automated testing.
    - Ensuring the quality of the submitted data in a client-server architecture.

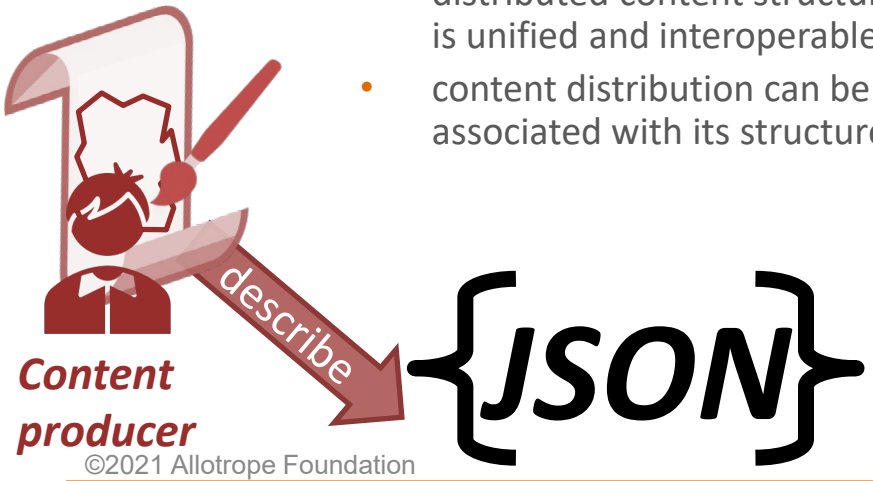


©2021 Allotrope Foundation

JSON Schema Primer

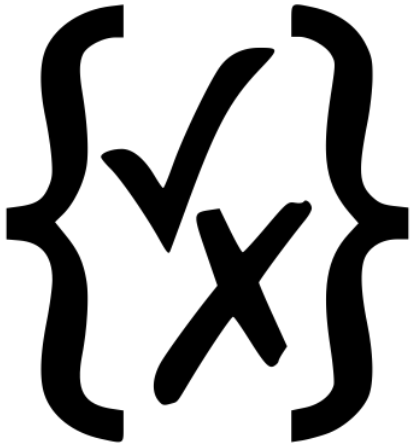
- ***JSON Schema*** is a vocabulary that allows the annotation and validation of ***JSON*** documents
  - Why do we need to formalize the description?... For content producers:
    - get clear guidelines
    - distributed content structure is unified and interoperable
    - content distribution can be associated with its structure



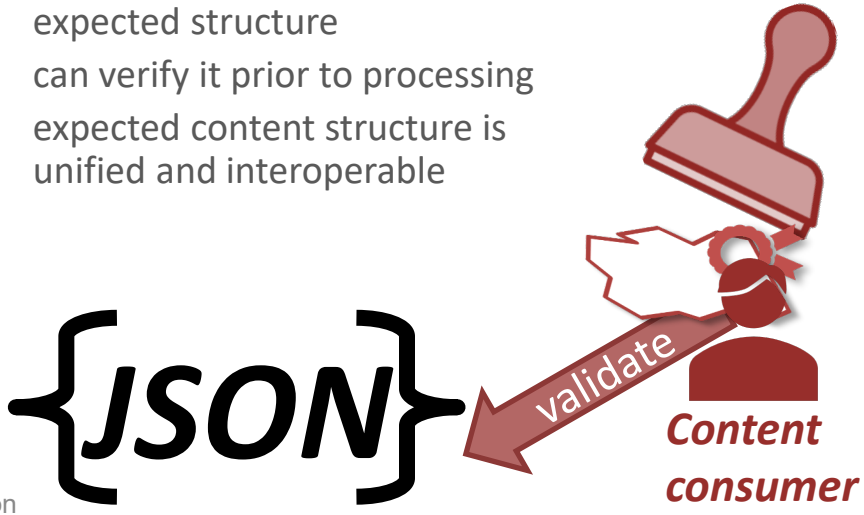


©2021 Allotrope Foundation

JSON Schema Primer



- ***JSON Schema*** is a vocabulary that allows the annotation and validation of ***JSON*** documents
  - Why do we need to formalize the validation?... For content consumers:
    - get clear guidelines on the expected structure
    - can verify it prior to processing
    - expected content structure is unified and interoperable



©2021 Allotrope Foundation

JSON Schema Primer

Some basic JSON schema definition:

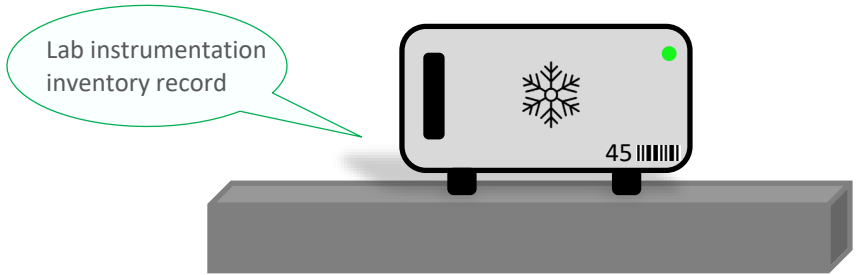
- JSON schema has properties called ***keywords***
- ***keywords*** are expressed as JSON keys
- ***Schema keywords***:
  - The schema: ***$schema*** specifies the standard version used to draft this document and it provides version control.
  - The identifier: ***$id*** defines the schema URI. It also serves as the base URI so relative URI-references in keywords within the schema can be resolved against.
- ***Annotation keywords***:
  - The title: ***title*** annotation keywords is descriptive only and do not add constraints to the validated data being validated. The intent is stated with the keyword.
  - The description: ***description*** annotation keywords is descriptive only and do not add constraints to the validated data being validated. The intent is stated with the keyword.

```
{
  "instrumentId": 45,
  "instrumentName": “bench top refrigerator”,
  "opStatus": true,
  "tags": ["research", "clinical"]
}
```

Data Instance

```
{
  "$schema": "https://json-schema.org/draft/2020-12/schema",
  "$id": "https://example.com/instrument.schema.json",
  "title": "Instrument",
  "description": "An instrument  in the laboratory",
  "type": "object"
}
```

JSON Schema



©2021 Allotrope Foundation

JSON Schema Primer

- ***Validation keywords*:**
  - The type: ***type*** validation keyword defines a constraint on the JSON data. In this case it must be a JSON Object.
  - The properties: ***properties*** validation keyword defines a constraint on the JSON data properties.
  - The required: ***required*** validation keyword defines a list where every item in the array is the name of a property in the data instance that needs to be presence.

```
{
  "instrumentId": 45,
  "instrumentName": “bench top refrigerator”,
  "opStatus": true,
  "tags": ["research", "clinical"]
}
```

```
{
  "$schema": "https://json-schema.org/draft/2020-12/schema",
  "$id": "https://example.com/instrument.schema.json",
  "title": "Instrument",
  "description": "An instrument  in the laboratory",
  "type": "object"
  "properties": {
    "instrumentId": {
      "description": "A unique identifier of a lab instrument",
      "type": "integer"
    }
  },
  "required": [ "instrumentId" ]
}
```

©2021 Allotrope Foundation

JSON Schema Primer

JSON object property definition

- ***instrumentId*** JSON key has a numeric value that uniquely identifies an instrument. Given that it is instrument identifier for an instrument, it make sense to mandate it by its listing with the ***required*** validation.
- ***instrumentName*** JSON key has a string value that names the instrument. Given that it is instrument name, it make sense to mandate it by its listing with the ***required*** validation.
- ***opStatus*** JSON key has a boolean value that indicate the instrument operational status. It make sense to mandate it by its listing with the ***required*** validation.
- The ***required*** validation keyword is a list of strings where multiple keys can be noted as required;
  - ***instrumentName*** and ***opStatus*** are added to the list.

```
{
  "instrumentId": 45,
  "instrumentName": “bench top refrigerator”,
  "opStatus": true,
  "tags": ["research", "clinical"]
}
```

```
{
  "$schema": "https://json-schema.org/draft/2020-12/schema",
  "$id": "https://example.com/instrument.schema.json",
  "title": "Instrument",
  "description": "An instrument  in the laboratory",
  "type": "object",
  "properties": {
    "instrumentId": {
      "description": "A unique identifier of a lab instrument",
      "type": "integer"
    },
    "instrumentName": {
      "description": "The instrument name",
      "type": "string"
    },
    "opStatus": {
      "description": "The instrument operational status",
      "type": "boolean"
    }
  },
  "required": ["instrumentId", "instrumentName", "opStatus"]
}
```

©2021 Allotrope Foundation

JSON Schema Primer

Additional constraints on the properties:

- It makes sense that the instrument identifier is a positive number
- We specify that value of ***instrumentId*** must be an integer greater than ***zero*** using the ***exclusiveMinimum*** validation keyword.

Other ***validation keywords*** for ***numeric*** (***number*** or ***integer***) instances:

- multipleOf
- Maximum
- exclusiveMaximum
- minimum

```
{
  "instrumentId": 45,
  "instrumentName": “bench top refrigerator”,
  "opStatus": true,
  "tags": ["research", "clinical"]
}
```

```
  "$schema": "https://json-schema.org/draft/2020-12/schema",
  "$id": "https://example.com/instrument.schema.json",
  "title": "Instrument",
  "description": "An instrument  in the laboratory",
  "type": "object",
  "properties": {
    "instrumentId": {
      "description": "A unique identifier of a lab instrument",
      "type": "integer",
      "exclusiveMinimum": 0
    },
    "instrumentName": {
      "description": "The instrument name",
      "type": "string"
    },
    "opStatus": {
      "description": "The instrument operational status",
      "type": "boolean"
    }
  },
  "required": ["instrumentId", "instrumentName", "opStatus"]
}
```

©2021 Allotrope Foundation

JSON Schema Primer

Additional constraints in the ***tags*** key:

Let's assume the following requirements on the instrumentation tagging :

- If there are ***tags*** there must be at least one ***tag***,
- ***tags*** must be unique; meaning no duplication for a single instrument.
- ***tags*** must be in a text format.
- ***tags*** are nice to have but not required to be present.

```
{
  "$schema": "https://json-schema.org/draft/2020-12/schema",
  "$id": "https://example.com/instrument.schema.json",
```

```
{
  "instrumentId": 45,
  "instrumentName": "bench top refrigerator",
  "opStatus": true,
  "tags": ["research", "clinical"]
}
```

```
      "exclusiveMinimum": 0
    },
    "instrumentName": {
      "description": "The instrument name",
      "type": "string"
    },
    "opStatus": {
      "description": "The instrument operational status",
      "type": "boolean"
    },
    "tags": {
      "description": "Tags for the instrument",
      "type": "array",
      "items": {
        "type": "string"
      },
      "minItems": 1,
      "uniqueItems": true
    }
  },
  "required": ["instrumentId", "instrumentName", "opStatus"]
}
```

©2021 Allotrope Foundation

JSON Schema Primer

Those requirements on ***tags*** translate to:

- The ***tags*** key is added with the annotations and keywords.
- The type validation keyword is ***array***.
- ***items*** validation keyword is added so we can define what appears in the array.
  - In this case: ***string*** values via the ***type*** validation keyword.
- A ***minItems*** validation keyword is used to make sure the existence of at least one item in the ***array***.
- The ***uniqueItems*** validation keyword means that each one of the ***items*** in the ***array*** must be unique.
- The ***tags*** key was not added to the ***required*** validation keyword array since it is optional.

```
{
  "instrumentId": 45,
  "instrumentName": “bench top refrigerator”,
  "opStatus": true,
  "tags": ["research", "clinical"]
}
```

```
{
  "$schema": "https://json-schema.org/draft/2020-12/schema",
  "$id": "https://example.com/instrument.schema.json",
      "exclusiveMinimum": 0
    },
    "instrumentName": {
      "description": "The instrument name",
      "type": "string"
    },
    "opStatus": {
      "description": "The instrument operational status",
      "type": "boolean"
    },
    "tags": {
      "description": "Tags for the instrument",
      "type": "array",
      "items": {
        "type": "string"
      },
      "minItems": 1,
      "uniqueItems": true
    }
  },
  "required": ["instrumentId", "instrumentName", "opStatus"]
}
```

©2021 Allotrope Foundation

JSON Schema Primer

Adding a nested JSON data structure:

- A ***dimensions*** key is added to the data instance

Therefore:

- The ***dimensions*** key is added to the schema with the ***type*** validation keyword and an ***object*** value.
- The ***properties*** validation keyword is used to define a nested data structure.
- To prevent verbosity, the ***description*** annotation keyword is omitted.
- A ***required*** validation keyword is added and is applicable to the nested dimensions key only!

```
    "instrumentName": {
      "description": "The instrument name",
      "type": "string"
    },
    "opStatus": {
```

```
{
  "instrumentId": 45,
  "instrumentName": "bench top refrigerator",
  "opStatus": true,
  "tags": ["research", "clinical"]
  "dimensions": {
    "length": 14.1,
    "width": 22.5,
    "height": 15.0
  }
```

```
    },
    "dimensions": {
      "type": "object",
      "properties": {
        "length": {
          "type": "number"
        },
        "width": {
          "type": "number"
        },
        "height": {
          "type": "number"
        }
      },
      "required": ["length", "width", "height"]
    }
  },
  "required": ["instrumentId", "instrumentName", "opStatus"]
}
```

©2021 Allotrope Foundation

JSON Schema Primer

An outside ***JSON schema*** can be referenced:

- Decomposition and modularization of a ***JSON schema:***
  - minimizes verbosity,
  - enhance readability,
  - reduce maintainability
  - increases reusability
- It is a good practice to share a ***JSON schema*** across data structures.
- For example; a reuse of a common geo location ***JSON schema*** by referencing

©2021 Allotrope Foundation

JSON Schema Primer

Reference an outside ***JSON schema***

- The geo location ***JSON Schema*** includes the min max constraints to perform a range validation:
  - ***minimum*** validation keyword
  - ***maximum*** validation keyword

©2021 Allotrope Foundation

JSON Schema Primer

The complete data instance for validation by the ***JSON schema***

```
{
  "instrumentId": 45,
  "instrumentName":  “bench top refrigerator”,
  "opStatus": true,
  "tags": ["research", "clinical"],
  "dimensions": {
    "length": 14.1,
    "width": 22.5,
    "height": 15.0
  },
  "labLocation": {
    "latitude": 67.44,
    "longitude": -55.34
  }
}
```

Data Instance

```
{
  "$id": "https://example.com/geo-location.schema.json",
  "$schema": "https://json-schema.org/draft/2020-12/schema",
  "title": "Longitude and Latitude",
  "description": "Geographical coordinates",
  "required": ["latitude", "longitude"],
  "type": "object",
  "properties": {
    "latitude": {
      "type": "number",
      "minimum": -90,
      "maximum": 90
    },
    "longitude": {
      "type": "number",
      "minimum": -180,
      "maximum": 180
    }
  }
}
```

Referenced JSON Schema

```
    },
    "opStatus": {
      "description": "The instrument operational status",
      "type": "boolean"
    },
    "tags": {
      "description": "Tags for the instrument",
    },
    "labLocation": {
      "description": "Lab coordinates of the instrument location",
      "$ref": "https://example.com/geo -location.schema.json"
    }
  },
  "required": ["instrumentId", "instrumentName", "opStatus"]
}
```

JSON Schema

©2021 Allotrope Foundation

ASM Overview

©2021 Allotrope Foundation

* This document is derived from https://json-schema.org/

The Allotrope Simple Model (ASM)

- ASM is a simple text representation of the Allotrope tabular models using JSON.
- It uses terms from the Allotrope Foundation Ontology (AFO), and it leverages the Allotrope Data Model (ADM) already defined and governed by Allotrope and SMEs.
- **Allotrope tabular models apply to domains where there is a single business object being measured and all measurements directly relate to this object.**
- ASM utilizes ***JSON Schema*** standard for validation

©2021 Allotrope Foundation

ASM JSON Model Example: Conductivity ASM

- The Conductivity Tabular Model – unique AFO Parameter prefLabel
- "The names within an object SHOULD be unique" (JSON specifications: RFC 4627)

Parameter prefLabel	Example Parameter Value	Parameter Unit Symbol
measurement identifier	413befdd	
measurement time	2015-09-24T03:47:13.001Z	
analyst	Amgentoaks1	
sample identifier	unknown-10	
equipment serial number	serial-number	
batch identifier	batch-number	
conductivity	273000	S/m
temperature	28.6	degC

©2021 Allotrope Foundation

ASM JSON Model Example: Conductivity ASM

- The Conductivity Tabular Model in an ASM JSON format: key-value pairs
- Keys are unique prefLabels in the AF Ontology (AFO).

Ref.

key

value

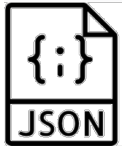
key

value

with

unit

```
{
  "$comment": "Conductivity ASM",
  "$asm.manifest": "http://purl.allotrope.org/adm/manifest/conductivity/CR/2021/09/conductivity.manifest",
  "measurement identifier": "413befdd",
  "measurement time": "2015-09-24T03:47:13.001Z",
  "analyst": "Amgentoaks1",
  "sample identifier": "unknown-10",
  "equipment serial number": "278882456",
  "batch identifier": "XYZ",
  "conductivity": {
    "value": 273000,
    "unit": "S/m"
  },
  "temperature": {
    "value": 28.6,
    "unit": "degC"
  }
}
```



©2021 Allotrope Foundation

ASM JSON Model Example: Conductivity ASM

- Each simple model JSON file MUST have a single reference to an ADM manifest (resolvable by the Allotrope PURL server)

The manifest provides:
- model identifier
- type
- ref to the ADM
- vocabulary
- ref to the JSON schema validation

Ref.

```
{
  "$comment": "Conductivity ASM",
  "$asm.manifest": "http://purl.allotrope.org/adm/manifest/conductivity/CR/2021/09/conductivity.manifest",
  "measurement identifier": "413befdd",
  "measurement time": "2015-09-24T03:47:13.001Z",
  "analyst": "Amgentoaks1"
```

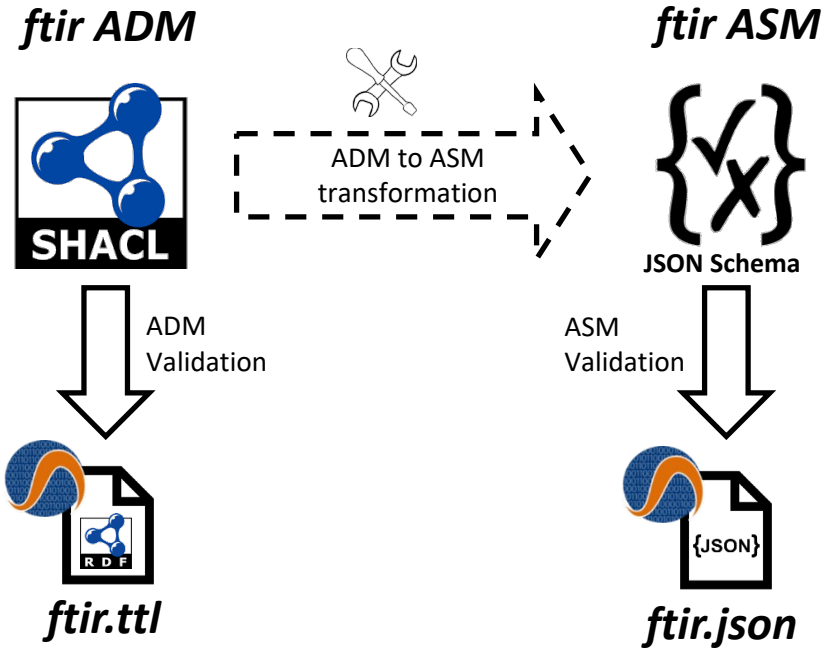
```
{
  "@id" : "http://purl.allotrope.org/manifests/conductivity/CR/2021/09/conductivity.manifest",
  "@type": "http://purl.allotrope.org/ontologies/manifest#Manifest",
  "shapes": [ "http://purl.allotrope.org/shapes/adm/conductivity/CR/2021/09/conductivity.shapes" ],
  "vocabulary": [ "http://purl.allotrope.org/voc/afo/merged/REC/2021/09/merged-and-inferred" ],
  "json-schemas": [ "http://purl.allotrope.org/json-schemas/adm/conductivity/CR/2021/09/conductivity.schema" ]
}
```

```
    "value": 28.6,
    "unit": "degC"
  }
}
```

©2021 Allotrope Foundation

ASM JSON Schema

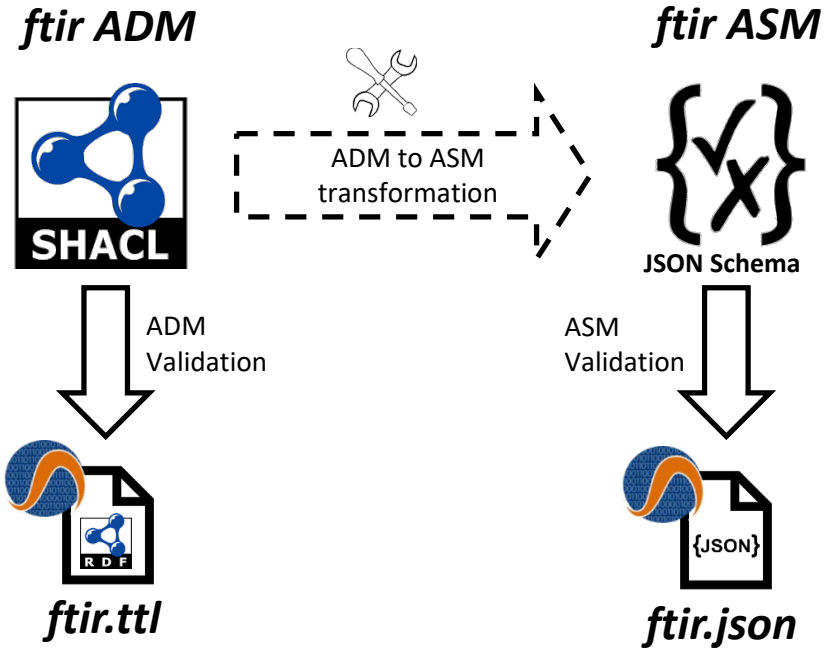
- Allotrope Simple Models are referencing ***JSON schemas*** for validation.
- These schemas are standard ***JSON schema*** following the latest specification (2020-12). https://json-schema.org/specification.html
- While ***SHACL*** (Shapes Constraint Language) is used to validate an ADM instance data, ***JSON schema*** is used to validate an ASM instance data.
- ASM JSON schema are generated using a transformation tool from ***SHACL*** to ***JSON schema***



©2021 Allotrope Foundation

ASM JSON Schema

- Unlike ***SHACL***, ***JSON schema*** is not semantically aware
- In order to maintain the ***SHACL*** semantic constraints available in the ***JSON schema*** and to help with transformations, the simple models ***JSON schema*** contain some Allotrope specific annotations, starting with a ***"$asm.*"*** prefix.
- Generic ***JSON schema*** tools can and will ignore these annotations.

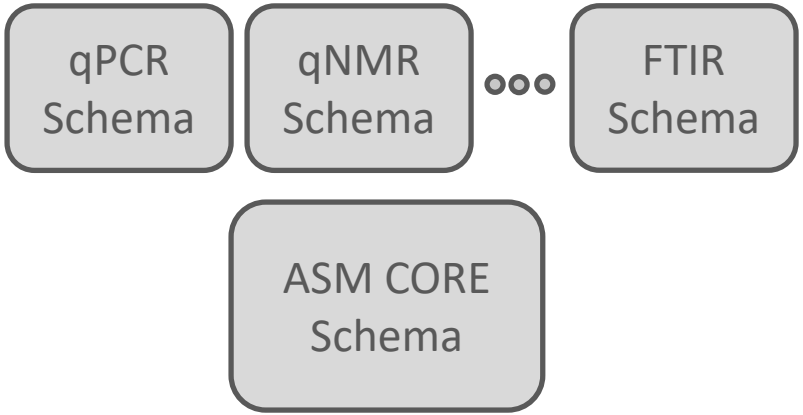


©2021 Allotrope Foundation

ASM JSON Schema

JSON schemas allow for modularization and factoring out commonly used rules by utilizing references to other ***JSON schema*** files. The simple model schemas make use of this modular approach. The ASM Schema is defined using:

- ***Core schema***: a ***JSON Schema*** that contains re-usable, domain independent rules.
  - The core schema defines value types for all possible values that may be used in tabular models.
- ***Technique specific schema***: a ***JSON Schema*** that contains the domain specific rules.
  - It references the core declarations instead of each technique defining its own
- Having the basic rules factored out in a ***core schema***, allows for later extensions without changing each ***technique specific schema***



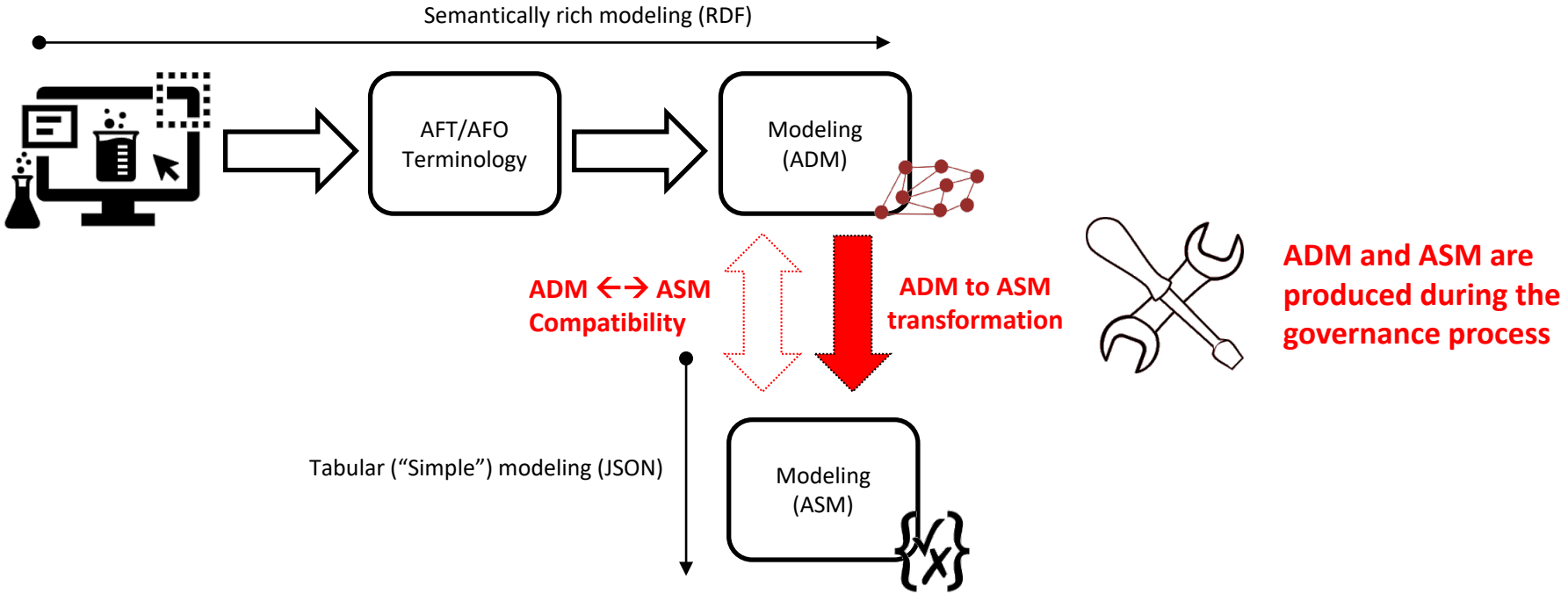
©2021 Allotrope Foundation

ASM Generation:

ADM to ASM Model Transformation Tool

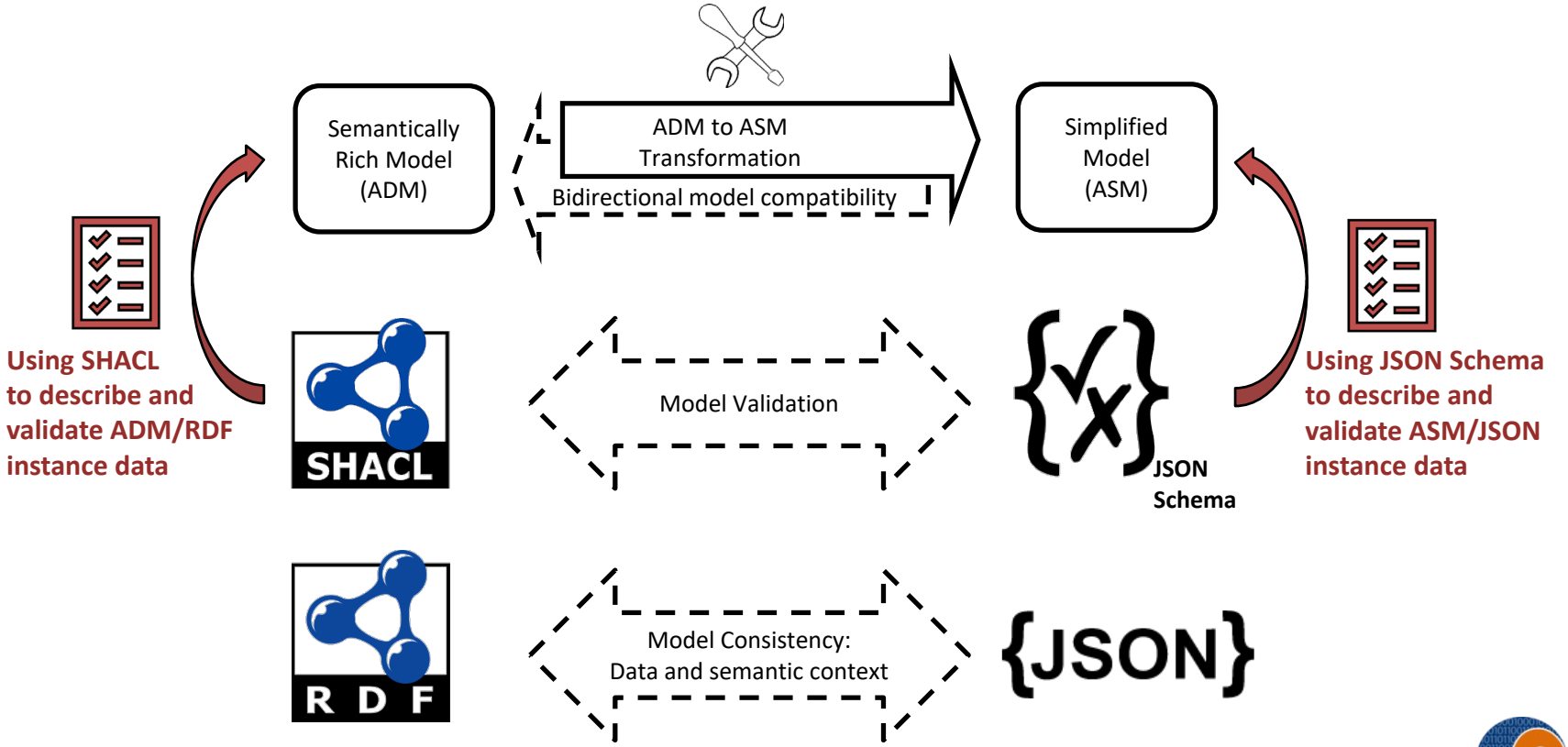
©2021 Allotrope Foundation

ADM to ASM Transformation



©2021 Allotrope Foundation

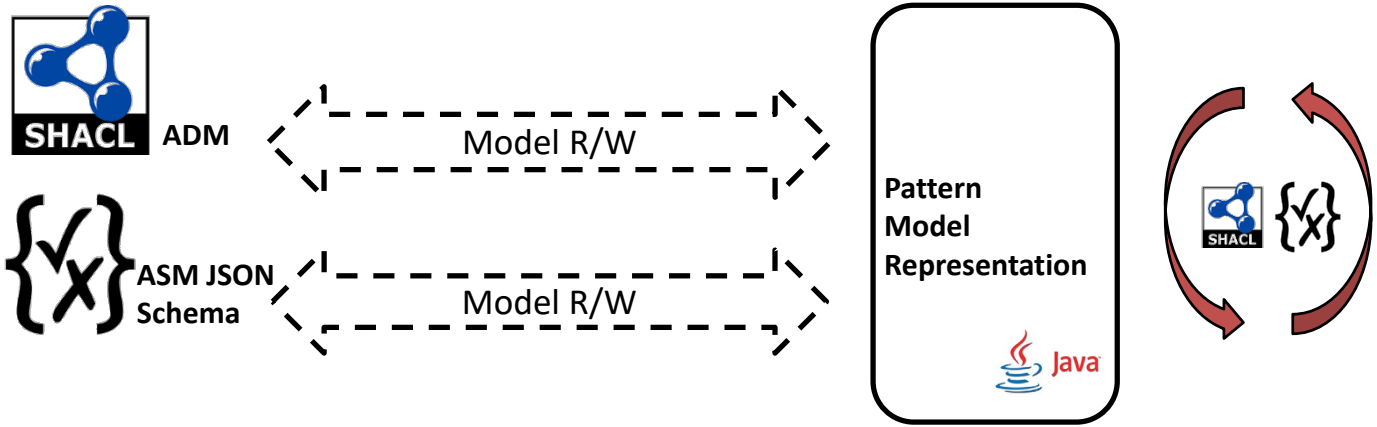
ADM ←→ ASM Semantic Compatibility

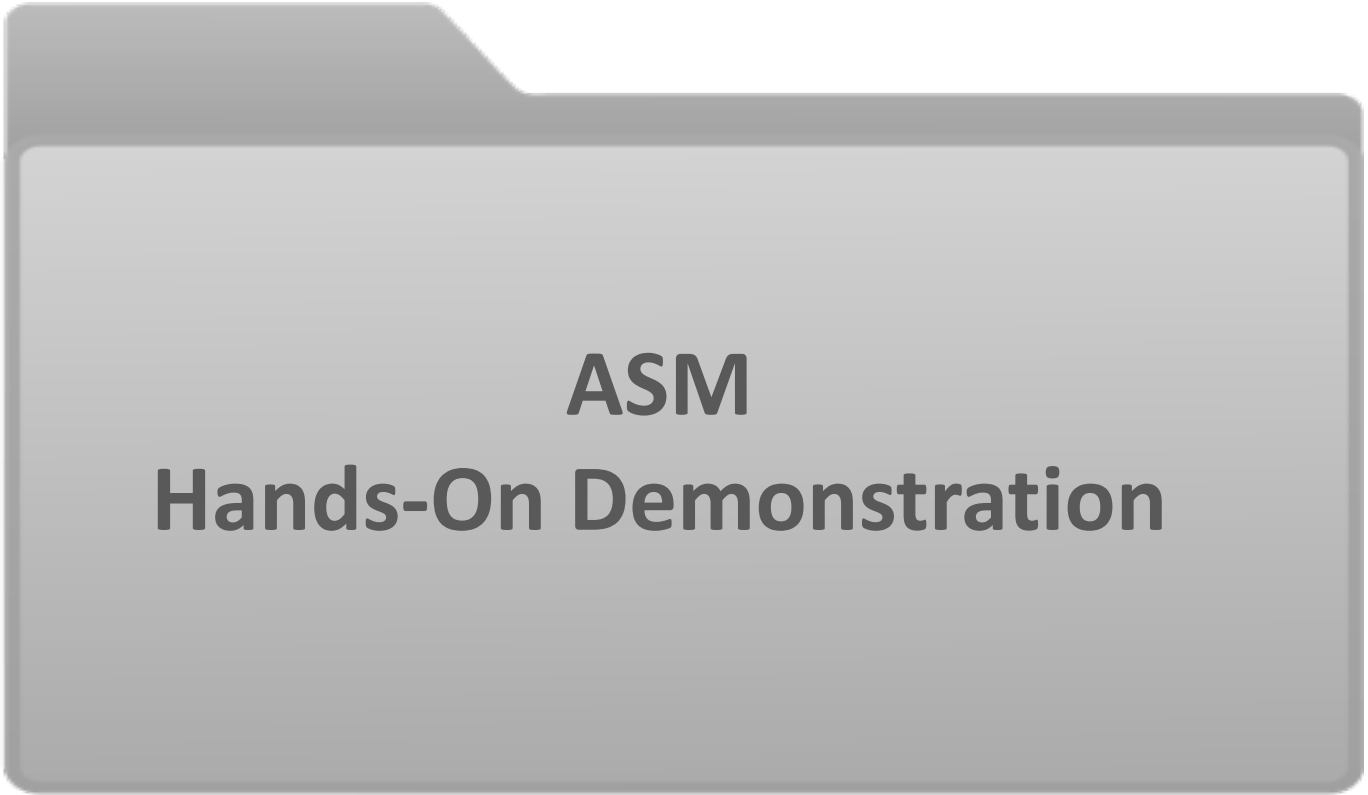


ADM to ASM Transformation Tool

Transformation Architecture:

- Written in Java
- The transformation tool has a common pattern model representation
- ***ADM*** (described by ***SHACL***) and ***ASM*** (described by ***JSON Schema***) can be read from and written into the pattern model
- It enables a decomposed, bi-directional transformation between the ADM and the ASM.

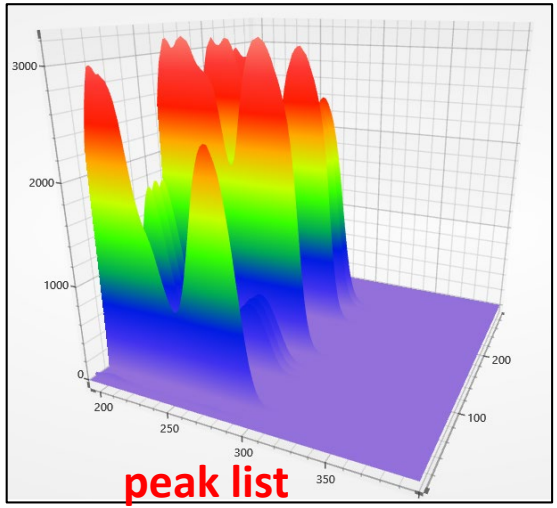




©2021 Allotrope Foundation

Aggregation Model Reminder

- The Aggregation Model is an extension of the Tabular Model
- It allows aggregation of data attributes under a unified context and/or an indexed collection of similar data pattern.
- The main characteristics of an Aggregation Model:
  - A set of key/value pairs (Table)
  - Keys are terms in the AFO providing a bridge to semantic usage.
  - The context of the key/value pairs is represented only by the container they are defined in
  - Concepts can be aggregated as:
    - **a collection** (e.g. a peak list contains peaks).
    - **facets** (attributes) of another concept (e.g. retention time and description).
- The Modeling WG is using Excel as a tool to create the Aggregation models



©2021 Allotrope Foundation

Aggregation Model Reminder

- Modeling an Aggregation model using Excel

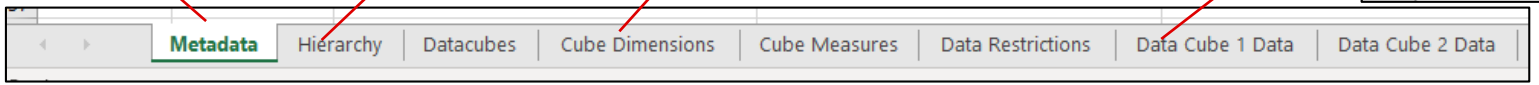
	A	B	C	D	E
1	Metadata ID	Parameter Type	Parameter Name	Parameter prefLabel	Parameter Allotrope URI
2	1	Metadata	$.system.name	device identifier	http://purl.allotrope.org/or
3	2	Metadata	$.system.serial_number	equipment serial number	http://purl.allotrope.org/or
4	3	Metadata	$.system.type	model number	
5	4	Metadata	$.run.id	measurement identifier	
6	5	Metadata	$.run.file_name	experimental data identifier	
7	6	Metadata	$.run.type	experiment type	
8	7	Metadata	$.time.measurement	measurement time	
9	8	Metadata	$.user.name	analyst	
10	9	Metadata	$.method.instrument.block_type	container type	
11	10	Metadata		plate well count	
12	11	Metadata		well volume	
13	12	Metadata	$.method.chemistry	qPCR detection chemistry	
14	13	Metadata	$.method.passive_reference	passive reference dye setting	
15	14	Metadata	$.method.processing.quantification_cycle_n	measurement method identif	
16	15	Metadata	$.method.standard_curve.ct_threshold.auto	automatic cycle threshold ena	
17	16	Metadata	$.method.standard_curve.ct_threshold	cycle threshold value setting	
18	17	Metadata	$.method.standard_curve.baseline.automati	automatic baseline determina	
19	18	Metadata	$.method.standard_curve.baseline.start	baseline determination start c	
20	19	Metadata	$.method.standard_curve.baseline.end	baseline determination end cy	
21	20	Metadata	Sample Aggregation	sample aggregate document	
22	21	Metadata	sample document	sample document	
23	22	Metadata	$.samples[*].name	sample identifier	
24	23	Metadata	$.samples[*].location.well.position	well location identifier	http://purl.allotrope.org/or

	A	B	C	D
1	Metadata ID	Parent Metadata ID	Relationship Type	Aggregation Index
2	21	20	Member of	1
3	22	21	Facet of	
4	23	21	F	
5	24	21	F	
6	25	21	F	
7	26	21	F	
8	27	21	Facet of	
9	28	21	Facet of	
10	29	21	Facet of	
11	30	21	Facet of	
12	31	21	Facet of	
13	32	21	Facet of	

	A	B	C	D	E
1	Cube ID	Data Cube Label	Measure Data Type	Measure Concept URI	Measure Concept prefLabel
2	1	Rn	xsd:double	http://purl.allotrope.org/c	normalized reporter result
3	2	Delta Rn	xsd:double	http://purl.allotrope.org/c	baseline corrected reporter result

qPCR

	A	B
1	Cycle	Rn
2	1	0.5091
3	2	0.5101
4	3	0.5127
5	4	0.5157
6	5	0.5175
7	6	0.5191
8	7	0.522
9	8	0.522
10	9	0.5208
11	10	0.5226
12	11	0.523
13	12	0.522
14	13	0.5245
15	14	0.5309
16	15	0.5325

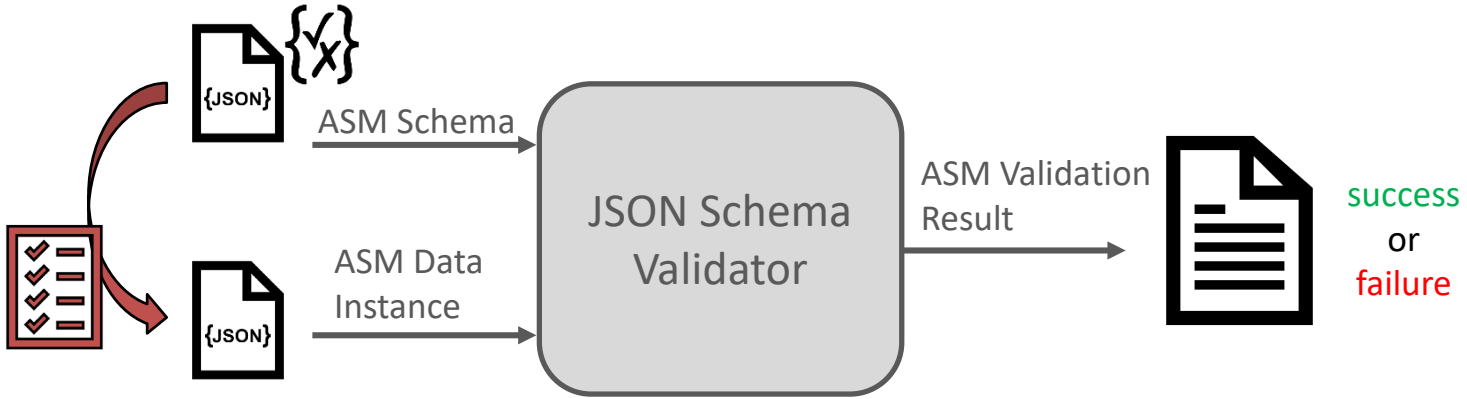


©2021 Allotrope Foundation

ASM demonstration

- ASM Tabular/Aggregation Data Instances Walkthrough
- ASM Schema Tabular/Aggregation Walkthrough
- ASM Validator in Action

*A list of many off-the-shelf validators, written in different languages, is available at https://json-schema.org/implementations.html



©2021 Allotrope Foundation

Thanks for your attention!

Allotrope Foundation Product Team

- Ben Woolford benjamin.woolford-lim@allotrope.org
- Amnon Ptashek amnon.ptashek@allotrope.org

©2021 Allotrope Foundation