One Year* Update: Using LinkML in Web of Things Specifications

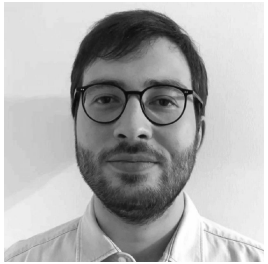
W3C TPAC 2024 Breakout
September 25 2024

Ege Korkan, Mahda Noura

Session Chairs

Ege Korkan

- Email: ege.korkan@siemens.com
- GitHub: @egekorkan



Mahda Noura

- Email: mahda.noura@siemens.com
- GitHub: @mahdanoura

Logistics

- W3C Calendar Entry | Session Proposal on GitHub
- Please all join IRC at https://irc.w3.org/?channels=%23linkml-wot
  - Then type `present+` to check in.
  - Type `q+` to raise hand
- We will take minutes there manually.
  - Scribe will be Michael McCool
- Quickly introduce yourself before speaking (now if there are not too many people)
- These slides are public. Link.
- The session is not recorded.

Participation Policies

- Antitrust and competition policy
  - W3C acts in a pro-competitive way that is ensured by this document.
- Positive Work Environment at W3C: Code of Ethics and Professional Conduct
  - Promote high standards of professional practice to ensure a positive work environment
- Health Rules
  - Masks and tests are optional. Stay in your room and attend virtually if you do not feel well.

Bringing some Context

Nature of this session:

- Brief presentation
- Discussion

Goal:

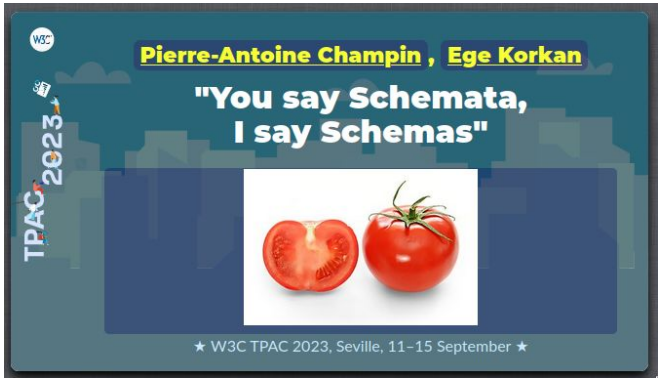
- Share experience

Required background:

- Basic understanding of JSON-LD, Ontologies, Schema languages

Have we met before?

Were you in the previous session of this?



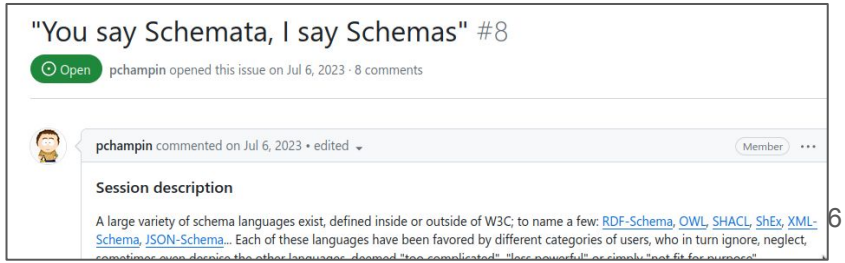
Schemata Follow-up

W3C Breakout Day
March 12, 2024

Ege Korkan, Mahda Noura

WoT Thing Description Single Source of Truth

Ege Korkan
W3C TPAC Breakouts



If not, please write "new" in IRC :)

In case you haven't been there

Resources from TPAC2023:

- Introduction from Pierre-Antoine Champin: https://perso.liris.cnrs.fr/pierre-antoine.champin/2023/TPAC-Schemata/
- Web of Things TD Use Case from Ege Korkan: https://github.com/w3c/wot/blob/main/PRESENTATIONS/2023-09-tpac/2023-09-13-Breakout-Schemata-TD_Single_Source_of_Truth.pdf
- Minutes: https://www.w3.org/2023/09/13-schemata-minutes.html
- Quick Summary

In case you haven't been there

Resources from Breakout Day 2024:

- Slides: https://docs.google.com/presentation/d/193OFcFaxD0GqrRuOggwZe5eorgL1C1Epe2cAYN3JEkk/edit?usp=sharing
- Minutes: https://www.w3.org/2024/03/12-schemata-discussion-minutes.html
- Quick Summary

Use Case of Thing Description Task Force

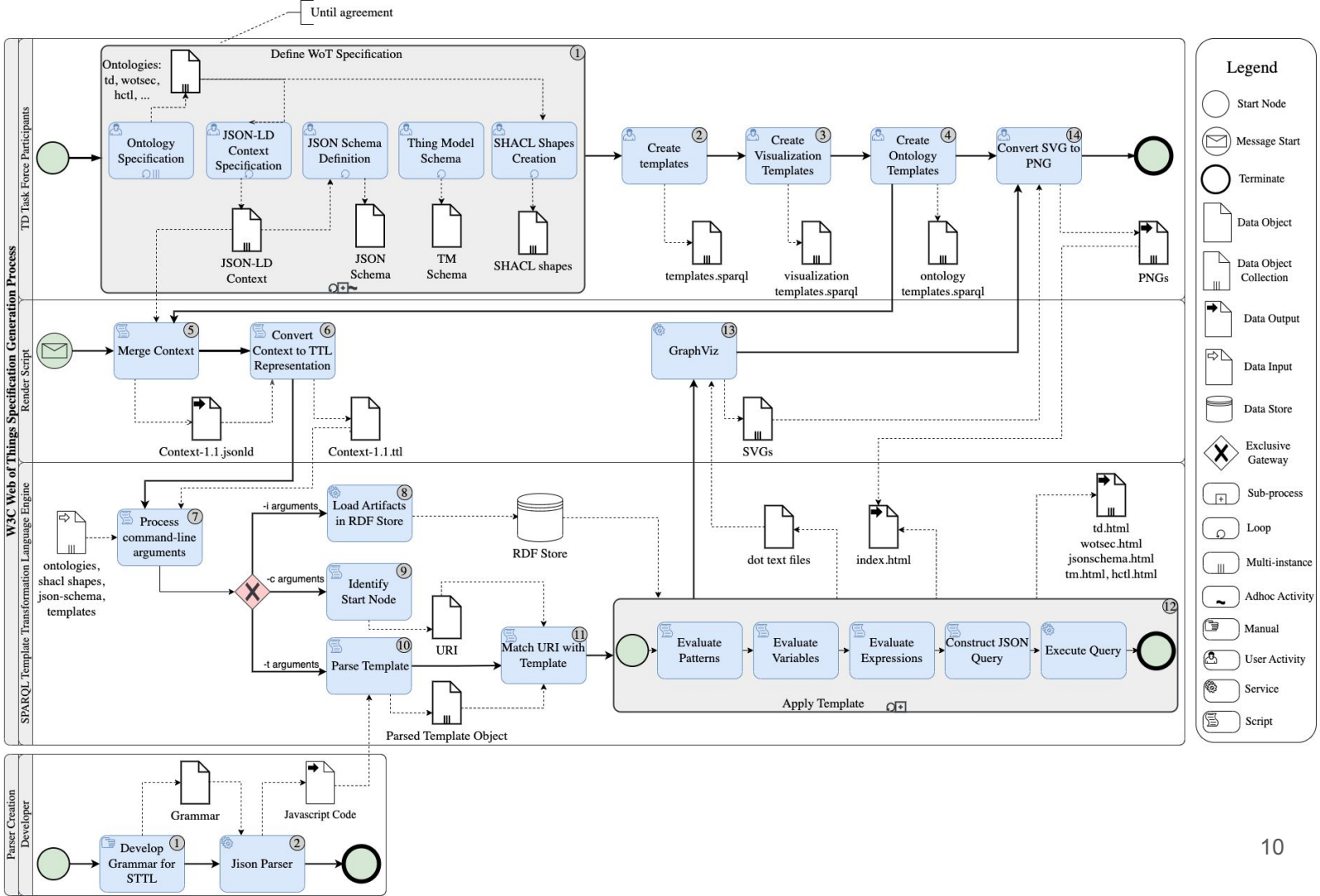
Web of Things TD Task Force needs to manage the following:

- Spec document, which contains vocabulary terms and information model
- Ontology documents
- SHACL shapes
- JSON Schema
- Type and Class Definitions (for now only TypeScript)
- Test cases
- Examples

All of them need publication procedure.

Soon, each binding in a registry will need the same and anyone should be able to do it…

What we were doing so far



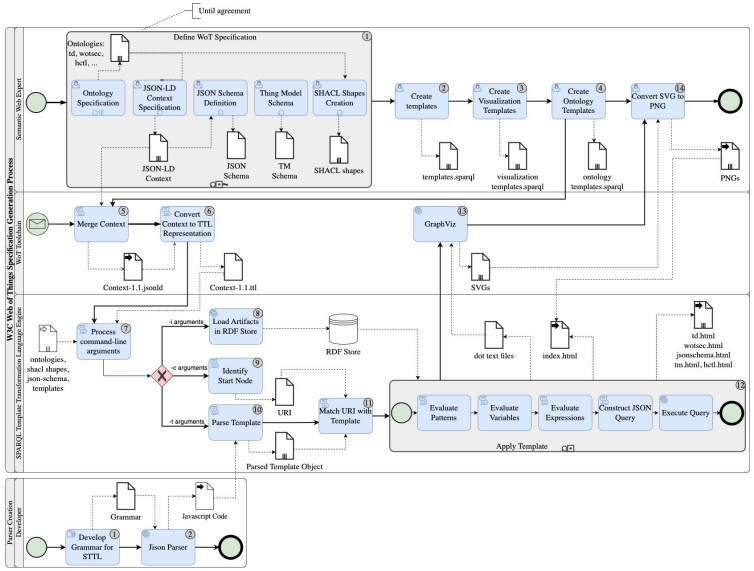
Before and Now

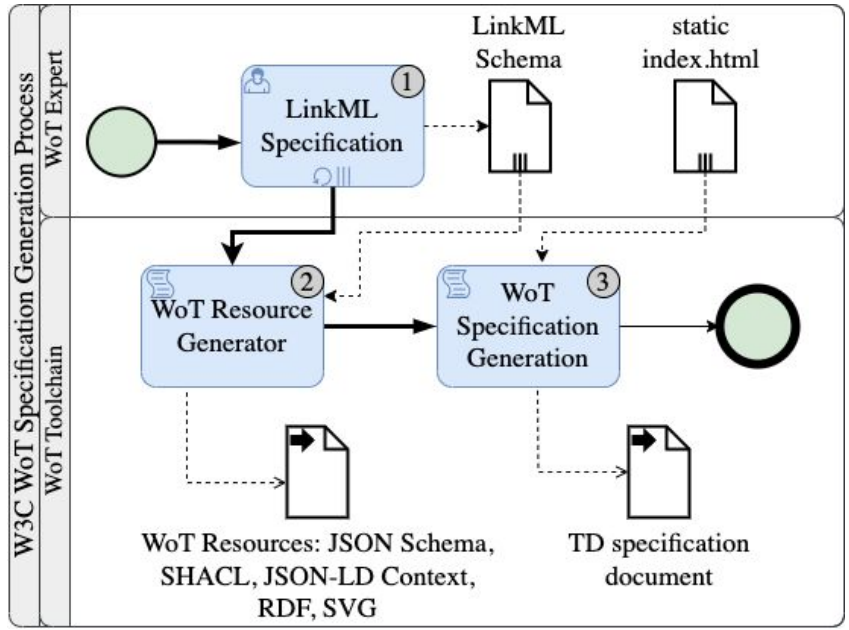
- Previously, we discussed possible technologies to use, presented an analysis
- Now we are on our path to adopting LinkML! Let's see what we have found out after 6 months…

Requirement | Tool	LinkML	TreeLDR	ESMF	A.ML	Schema Salad	SOML	WIDOCO
Language	Python	Rust	Java	Scala	Python	NG	Java
Object/Dict Support	O						
Condition Check							
Array Support	O	O	O	O	O	O	X
One of	O	O	X	O	O	X	X
Type/Type[]	X	X	X	X	X	X	X
Inheritance	O	O	O	O	O	O	X
Unknown object keys	O	X	X	O	X	X	X
Pattern Matching	O	O	O	O	X	O	X
JSON Schema Generation	O	O	O	X	X	X	X
SHACL Shapes Generation	O	X	X	X	X	X	X
Term Documentation	O	X	X	X	O	X	O
Diagram Generation	O	X	O	X	O	X	O
Prog. Lang. Gen.	O						
Extensibility	O						

Switch to Mahda

Vision





Transitioning from multiple sources to a single source-of-truth with LinkML schema
Transitioning from manually crafting WoT artifacts to full automation with LinkML generators

WoTIS - Web of Things Integration Schema

Step 1: Clone the repo: https://github.com/w3c/wot-thing-description-toolchain-tmp

Step 2: Install uv package manager

Step 3: Install the package by: `uv run wotis`

CLI Usage

```
wotis generate-wot-resources [-i] [-d] [-s] [--help]

options:
  -i, --input_schema       Path to the input schema specified as LinkML yaml.
                           [default: resources/schemas/thing_description.yaml]

  -d, --generate_docs      Boolean for local documentation generation.
  -s, --serve_docs         Boolean for serving the generated documentation.
  --help                   Show this help message and exit.
```

Note: *The HTML generation does not use the W3C style yet. Customization options will be included in the future.*

Live Demo: Putting WoTIS to Work

Realizing the Vision through Post-Processing

LinkML Generators provide a strong foundation, but…

Post-Processing is still required to ensure:

- Consistency between generated and hand-written WoT artifacts
- Customizations to meet specific domain requirements

What More Do We Need from LinkML?

- Natively support modeling JSON LD multi-language strings
- Scoped JSON-LD contexts
- Require open mappings for Class attributes
- Support for JSON-LD arrays and containers
- Best practices for modeling JSON-LD keywords
- Improved representations of URI, CURIE and URIORCURIE types beyond the string type
- Schema name collision resolution - currently multiple schemas that have the same name are merged, even though they refer to different elements
- Contribution support on LinkML code architecture to lower the barrier

Multi-language Strings

```
{
  "@context": "https://www.w3.org/2022/wot/td/v1.1",
  "title": "MyThing",
  "titles": {
    "en": "MyThing",
    "de": "MeinDing",
    "ja": "私の物",
    "zh-Hans": "我的东西",
    "zh-Hant": "我的東西"
  },
  "descriptions": {
    "en": "Human readable information.",
    "de": "Menschenlesbare Informationen.",
    "ja": "人間が読むことができる情報",
    "zh-Hans": "人们可阅读的信息",
    "zh-Hant": "人們可閱讀的資訊"
  }
}
```

```
:ThingShape a sh:NodeShape ;
    sh:targetClass td:Thing ;
    skos:definition """An abstraction of a physical or a virtual entity whose
          metadata and interfaces are described by a WoT Thing
          Description, whereas a virtual entity is the composition
          of one or more Things."""^^rdf:HTML ;
    sh:closed false ;
    sh:order 1 ;
    sh:property [
        sh:path td:title ;
        skos:definition """Provides a human-readable title (e.g., display
                a text for UI representation) based on a default
                language."""^^rdf:HTML ;
        sh:nodeKind sh:Literal ;
        sh:or ( [ sh:datatype xsd:string ] [ sh:datatype rdf:langString ] ) ;
        sh:minCount 1 ;
        sh:maxCount 1 ;
        sh:order 1 ;
    ] ;
```

Scoped JSON-LD Context

```
    "properties": {
      "@id": "td:hasPropertyAffordance",
      "@type": "@id",
      "@container": "@index",
      "@index": "name",
      "@context": {
        "td": "https://www.w3.org/2019/wot/td#",
        "jsonschema": "https://www.w3.org/2019/wot/json-
schema#",
        "wotsec": "https://www.w3.org/2019/wot/security#",
        "hctl": "https://www.w3.org/2019/wot/hypermedia#",
        "dct": "http://purl.org/dc/terms/",
        "schema": "http://schema.org/",
        "rdf": "http://www.w3.org/1999/02/22-rdf-syntax-ns#",
        "@vocab": "https://www.w3.org/2019/wot/json-schema#",
        "DataSchema": {
          "@id": "jsonschema:DataSchema"
        },
        "readOnly": {
          "@id": "jsonschema:readOnly"
        },
        "writeOnly": {
          "@id": "jsonschema:writeOnly"
        },
        "exclusiveMaximum": {
          "@id": "jsonschema:exclusiveMaximum"
        },
        "exclusiveMinimum": {
          "@id": "jsonschema:exclusiveMinimum"
        },
        "maximum": {
          "@id": "jsonschema:maximum"
        },
```

Thing Description Context Extension & Semantic Annotations

```
{
    "@context": [
        "https://www.w3.org/2022/wot/td/v1.1",
        {
            "saref": "https://w3id.org/saref#",
            "om": "http://www.ontology-of-units-of-measure.org/resource/om-2/",
            "schema": "https://schema.org"
        }
    ],
    "version": {
        "instance": "1.2.1",
        "schema:softwareVersion": "1.0.1"
    },
    "schema:serialNumber": "4CE0460D0G",
    "schema:manufacturer": {"name": "CompanyName"},
    // ...
    "@type": "saref:TemperatureSensor",
    "properties": {
        "temperature": {
            "description": "Temperature value of the weather station",
            "type": "number",
            "minimum": -32.5,
            "maximum": 55.2,
            "unit": "om:degreeCelsius",
            "forms": [...]
        },
        // ...
    },
    // ...
}
```

How This Works in Other Models:

- **JSON Schema:** Achieved by setting `"additionalProperties": true`
- **SHACL:** Managed by using `"closed": false`

Limitation in LinkML

- **LinkML** does not currently support modeling TD context extensions and allowing for external vocabulary integration.

https://github.com/linkml/linkml/issues/2238

What More Do We Need from LinkML?

- Natively support modeling JSON LD multi-language strings
- Scoped JSON-LD contexts
- Require open mappings for Class attributes
- Support for JSON-LD arrays and containers
- Best practices for modeling JSON-LD keywords
- Improved URI, CURIE, URIORCURIE types beyond string type
- Schema name collision resolution - currently multiple schemas that have the same name are merged, even though they refer to different elements
- Contribution support on LinkML code architecture to lower the barrier

Overall Experience

+ LinkML supports diverse schema definitions, suitable for simple & complex models
+ Easy-to-use LinkML generators
+ Good community engagement in issue discussions
- Lack of comprehensive documentation for complex use cases
- Provided error messages are in most cases not helpful
- High effort for correct slot selection for specific use cases
- Incomplete feature implementation often requires workaround

LinkML Long-Term Outlook

- Schema verbosity and maintenance
- Schema inflexibility sometimes results in defining intermediate classes
- RDF-like mental model still necessary, **subject-predicate-object**
- The LinkML model must be refined for individual generators can be **time-consuming** when multiple generators are incorporated
- Continued efforts to strengthen the community

Where should the discussion continue?

- LinkML Meetups (Stay tuned!)
- WoT WG (contact Ege Korkan (ege.korkan@siemens.com) to join the calls for this toolchain work)

Feedbacks and Whiteboard

Vladimir: Electrical CIM/CGMES wants to transition from Enterprise Architect to LinkML, but just starting. Discussion at https://github.com/Sveino/Spec4CIM-KG/issues/9

Check-out (to be extended in the end of the meeting)

A summary before the discussion ends:

- Main points of discussion, consensus, or disagreement?
  i. Discussion
- What are the next steps?
  i. Further work on the usage of LinkML
  ii. Reach out to DID TF
- Who is responsible for carrying them out? (Could be a person from the session, or a group where work is ongoing, a new community group, the staff, etc.)
  i. TD TF of the WoT WG