

Semantic Web API Specification

Ioana Bîrsan

Scientific Coordinator
Conf. Dr. Sabin-Corneliu Buraga

Alexandru Ioan Cuza University of Iași
Faculty of Computer Science

July 5-6, 2019

Overview

- 1 Augmenting OpenAPI Specification with Knowledge
- 2 Demo
- 3 Future Directions

Thesis Context

The Web is becoming a large repository of open data, available for:

- rich exploratory **querying**
- machine processing such as **generating visualizations**
- **combining multiple data sources**¹

The **Semantic Web** (Web 3.0) has been designed as a WWW extension that allows computing tools to search, combine and process content that is based on the meaning it has for us.

¹Jacek Kopecký, Paul Fremantle, and Rich Boakes – *A history and future of web apis*

Thesis Statement

Can we augment the existing **Web API Specifications** with **Semantic support** in order to derive both machine and human readable content in a better, more comprehensive way?

Thesis Contribution

Add support for **Semantic Augmentation** of **OpenAPI Specification**² within **Swagger Editor**³ and **Swagger UI**⁴ open source tools.

²<https://swagger.io/specification/>

³<https://swagger.io/tools/swagger-editor/>

⁴<https://swagger.io/tools/swagger-ui/>

Augmenting OpenAPI Specification With Knowledge

OpenAPI Extension Support

Extensions: **x-same-as** & **x-rdf-type**

Supported values: concepts defined in the **Schema.org vocabulary**

- **definitions/(components/schemas)-level** — equivalent to **rdfs:Class**
- **properties-level** — equivalent to **rdf:Property**

OpenAPI Extension Support

```
definitions:  
  Order:  
    type: object  
    x-same-as: 'http://schema.org/Order'  
    properties:  
      id:  
        type: integer  
        x-same-as: 'http://schema.org/identifier'  
        format: int64
```

Figure 1: Example of expansion of OAS2 with semantics.

```
components:  
  schemas:  
    Order:  
      type: object  
      x-same-as: 'http://schema.org/Order'  
      properties:  
        id:  
          type: integer  
          x-same-as: 'http://schema.org/identifier'  
          format: int64
```

Figure 2: Example of expansion of OAS3 with semantics.

Extending Swagger Editor With Semantic Support

- Implement **suggestions** and **autocomplete support** functionality for the newly added extensions
- Offer **contextual suggestions** and **autocomplete support** for concepts retrieved from **Schema.org**⁵, depending on the location where the extension is used
- Ability to **convert** and **save** an **OpenAPI definition** in **Turtle (TTL)**⁶ **format**

⁵<https://schema.org/>

⁶<https://www.w3.org/TR/turtle/>

Extending Swagger UI With Semantic Support

- Incorporate **Structured Data** in the generated OpenAPI definition documentation using **Microdata**⁷
- Retrieve existing **description** from **Schema.org** for each element that has been augmented with semantics and its inclusion in the generated OpenAPI definition documentation
- Associate elements from **OpenAPI definition** with concepts from the **Schema.org vocabulary**.

⁷<https://developer.mozilla.org/en-US/docs/Web/HTML/Microdata>

Demo

Future Directions

Incorporation of Other Ontologies

The **OpenAPI Specification** may be **extended**, enabling the user to incorporate/refer concepts from **other ontologies**.

(e.g. DBpedia⁸, Disease Ontology⁹, Dublin Core¹⁰, etc.)

⁸<http://dbpedia.org/ontology/>

⁹<http://www.disease-ontology.org/>

¹⁰<http://www.dublincore.org/specifications/dublin-core/>

Publishing resulting Ontology

The addition of a mechanism through which the **ontology resulted from the Semantic Augmentation is automatically published in a triplestore.**

(e.g. AllegroGraph¹¹, Amazon Neptune¹² Stardog¹³, etc.)



This database will entail itself to provide **API recommendations**, through the usage of **SPARQL queries**, immersed directly in the Swagger UI tool.

¹¹<https://franz.com/agraph/allegrograph/>

¹²<https://aws.amazon.com/neptune/>

¹³<https://www.stardog.com/>

Self-detection of Semantic Concepts

The creation of a mechanism by which Swagger Editor can **self-detect** what **concepts** are **defined in the created API** and **automatically create** the **associations** or **provide** the user with **suggestion** from which he can choose the most appropriate one.

Conclusions

- 1 Context.
- 2 We have created a foundational layer that enables Semantic support and augmentation of OpenAPI Specification conformant API definitions.
- 3 We extended the functionality of the Swagger Editor and Swagger UI open source tools.
- 4 Future directions.

Thank you!