

## **W3C LBD Community Group**

### **Minutes - Call 15/05/2023**

#### **Attendees:**

- Mathias Bonduel (Neanex Technologies)
- Alex Donkers (Eindhoven University of Technology)
- Philipp Hagedorn (RUB)
- Katja Breitenfelder (Fraunhofer IBP / National Academy of Science and Engineering)
- Vladimir Alexiev (Ontotext)
- Michel Böhms (TNO)
- Calin Boje (Luxembourg Institute of Science and Technology)
- Sebastian Blechmann (RWTH Aachen University)
- Peter Johansson (Jönköping University Sweden)
- Rahel Kebede (Jönköping University, Sweden)
- Hervé Pruvost (Fraunhofer IIS/EAS)
- Marin Ljuban (BIM A+)
- Michele Allori (e-Metodi srl)
- Salvatore Cataldi (BELIMO Automation AG)
- Daniel Hernández (University of Stuttgart)
- Jakob Martin (iabi.e.V)
- Klaus Linhard (iabi e.V.)
- Aaron Costin (University of Florida)
- Nataliya Keberle (Ontotext)

#### **Presentation slides**

- [GitHub](#)

#### **Date and time**

- 15/05/2023, Monday, 14:00-15:30@UTC/ 16:00-17:30@CEST/ 07:00-08:30@PDT

#### **Moderators**

1. Mathias Bonduel

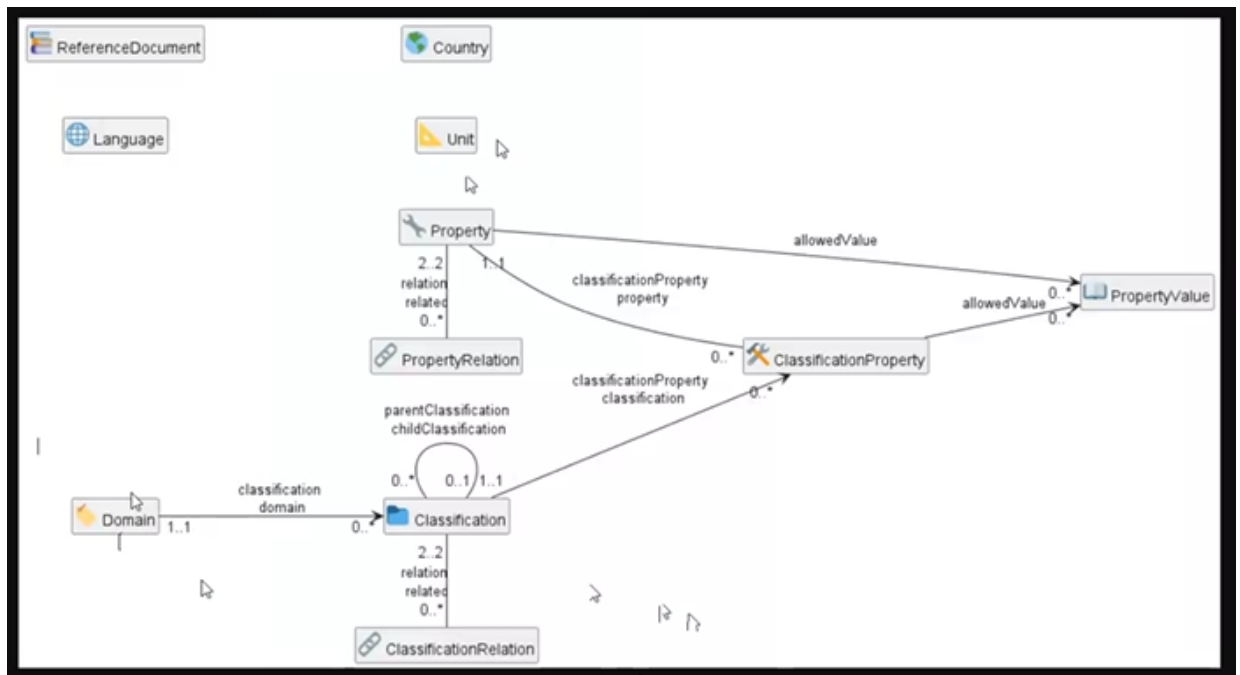
#### **Agenda**

1. Introduction of new members
2. Presentation: Vladimir Alexiev and Nataliya Keberle (Ontotext) - “semantic bSDD”
3. Discussion
4. Further topics

#### **Minutes**

1. **Introduction of new members**
  - a. xxxx
2. **Presentation: Vladimir Alexiev and Nataliya Keberle (Ontotext) - “semantic bSDD”**

- a. As part of ACCORD, Ontotext looked at bSDD.  
They did a comprehensive analysis of the bSDD, problems in bSDD and possible solutions.  
All the code is available on github. There's HTML documentation available at <https://bsdd.ontotext.com/>. Detailed findings are in [README.html](#) (see [/new/](#) for a more modern layout, but maybe slightly older), as well as source, webinar recordings, paper accepted at LDAC 2023, etc.
- b. SemanticBSDD was developed mainly by Vladimir Alexiev, Mihail Radkov and Nataliya Keberle, using Ontotext products: GraphDB and Ontotext Semantic Objects.
- c. Joint challenge with bSI submitted for the LDAC Summer School:  
[https://github.com/SSoLDAC-2023/Challenge-Data\\_Dictionary](https://github.com/SSoLDAC-2023/Challenge-Data_Dictionary)
- d. Vladimir argues that GraphQL is a promising method to obtain bSDD data, because it's more developer-friendly than the traditional REST APIs.
- e. Findings: Property is linked to a PropertyRelation, but the related Property on the other end of the relation is a string, which is why we cannot see it in the original GraphQL visualisation, and cannot obtain it in the same GraphQL query. This happens with various other entities too. Lookup tables like Country, ReferenceDocument, Units are de-facto unused since they are disconnected from the rest of the schema. Conclusion: the schema is not inter-linked enough, and the use of strings instead of things (objects) is a limitation.
- f. The semantics of ClassificationRelation and PropertyRelation are not well defined; e.g. which are symmetric, transitive, inverse, etc.
- g. Finding: Most relation entities in the original bSDD were unidirectional in GraphQL, while you would like to be able to reach the parent node.
- h. A refactored schema is proposed that should solve some of the mentioned problems.



The schema adds bidirectional relations in each case (to be able to navigate the knowledge graph in any direction).

- i. When we look at the full schema with attributes, various reconsiderations of datatypes were made. Icons were added to the schema so that the datatypes are more easily visible.
- j. We now dive into GraphQL query writing, which is shown in a visual query writing tool called GraphiQL.

- k. As this tool retrospects the schema, one can click CTRL+space to get a list of possible query inputs at any point, making query-writing easier.
- l. A new version of GraphiQL (<https://bsdd.ontotext.com/graphiql/>) is more expressive and should make query-writing even more easy. There's a long tree of possible parameters with checkboxes to see which parameters are selected (and can thus be added to the query).
- m. Finding: Property and ClassificationProperty are very similar. Their attributes are almost the same. Object modeling practices suggest to define a superclass and to only mention the differences between them.
- n. PropertyValue and ClassificationPropertyValue are exactly the same. Therefore, in the refactored schema by Ontotext, these are merged to PropertyValue.
- o. Finding: There's a mixture of singular and plural in property names. Best practice suggests to always keep them singular.
- p. Improvements in the refactored GraphQL: adds comprehensive **where** language, ordering, pagination (**limit/offset**).
- q. The refactored schema allows to query connected properties. However, it is not fully known how those properties are connected.
- r. Findings: Some units are multi-valued, while they should be single-valued.
- s. The refactored bSDD also has a SPARQL-endpoint at <https://bsdd.ontotext.com/graphdb/sparql>. Various saved queries are publicly available through this endpoint.
- t. Vladimir shows various of these queries and shows some results that are wrong. Improvements to the bSDD should be made to get rid of those errors.
- u. Suggested improvements:
  - i. It would be good to have URLs for every single object. The more strings we can turn into objects, the better.
  - ii. Equal data should be retrieved from the different API's (JSON from GraphQL, JSON from Rest API, RDF from Rest API). Currently the data that is returned in those three methods is different.
  - iii. Not just the schema, but also the quality of the data in the bSDD is important.
  - iv. Improve URL structure and consistency
    - 1. bSDD uses two different pages for the same object, where the semantic link resolves to a poorer page with the rich content. There is no reason to have two URLs for the same object.
    - 2. Next to this, there are simple improvements, such as changing dashes to slashes in <http://asdf.org/asdf/asdf-asdf> to <http://asdf.org/asdf/asdf/asdf>
- v. Modelling issues:
  - i. Unify solutions to model complex properties
  - ii. Improve modeling of dynamic properties
  - iii. Improve relations between entities: convert strings to objects
  - iv. Add more entities
    - 1. PhysicalQuantity, PropertySet, CountrySubdivision, DomainVersion, and more...
  - v. Use class inheritance
    - 1. Certain classes should have a common parent class
  - vi. Improve representation of PropertyValue
    - 1. PropertyValue and ClassificationPropertyValue are structured values with rich fields, however, most of those fields are hardly used, or stored in tables (in the description fields). This makes querying the individual values in those tables hard.
  - vii. Improve multilingual support

1. Most of the domains are unilingual. One of the promising aspects of bSDD is that it supports multilinguality, however, most of the domains are only represented using one language.
- w. The schema was also tested with GPT4. Since SOML is a simple language, GPT4 is competent enough to understand it, and to understand the structure of GraphQL queries that are generated from it in the Ontotext Platform. Then GPT4 can answer questions about the schema, **write GraphQL queries**, and interpret query results.

### 3. Discussion

- a. [Michel]: I see Property and ClassificationProperty as different things. How can they be more integrated?  
[Vladimir]: They are similar so they could inherit from a common super-class. But we haven't done it in the refactoring, we still keep Property and ClassificationProperty separate. AFAIK, the BSDD semantics doesn't describe compatibility and "inheritance" of attributes from Property to ClassificationProperty. Eg can I change dataType from "Real" to "Boolean"? That should not be allowed.
- b. [Mathias]: You developed an ontology to explain errors in the bSDD schema. Why are you creating definitions on A-box level instead of on T-box level?  
[Vladimir]: I wouldn't say we developed an ontology. We are reusing concepts from the existing bSDD schema. We improved it a bit (refactored), but in general it's very similar to the original. There's a work by Jyrki Oraskari on dynamically interpreting BSDD classes and props as an ontology: we're not doing that. Our main finding is that most fields are strings, whereas more of them should be objects (things).

### 4. Further topics

- a.

## Next Call

- 19/06/2023, Monday, 14:00-15:30@UTC/ 16:00-17:30@CEST/ 07:00-08:30@PDT

### Agenda: TBD

We are interested in getting suggestions from the community about potential agenda items and **Elevator Pitches** for the following calls. Please send your suggestions to the chairs or to [internal-lbd@w3.org](mailto:internal-lbd@w3.org), whether you have a short presentation to bootstrap the discussion, and an approximate duration you think the discussion will last.

## Previous minutes

<https://github.com/w3c-lbd-cg/lbd/tree/gh-pages/minutes>