

KU LEUVEN



The SPARQL-visualizer

Software demo

LDAC Workshop 2018

Mathias Bonduel (KU Leuven & UGent)

Mads H. Rasmussen (DTU & Niras)

Pieter Pauwels (UGent)

Maarten Vergauwen (KU Leuven)

Ralf Klein (KU Leuven)

Content

- Introduction
- SPARQL-visualizer
 - Implementation details and functionality
 - Use cases
 - Hands-on introduction
- Conclusion
- Future work

Introduction

- “A collaborative ontology engineering process”

= Creation of a shared ontology that facilitates information exchange

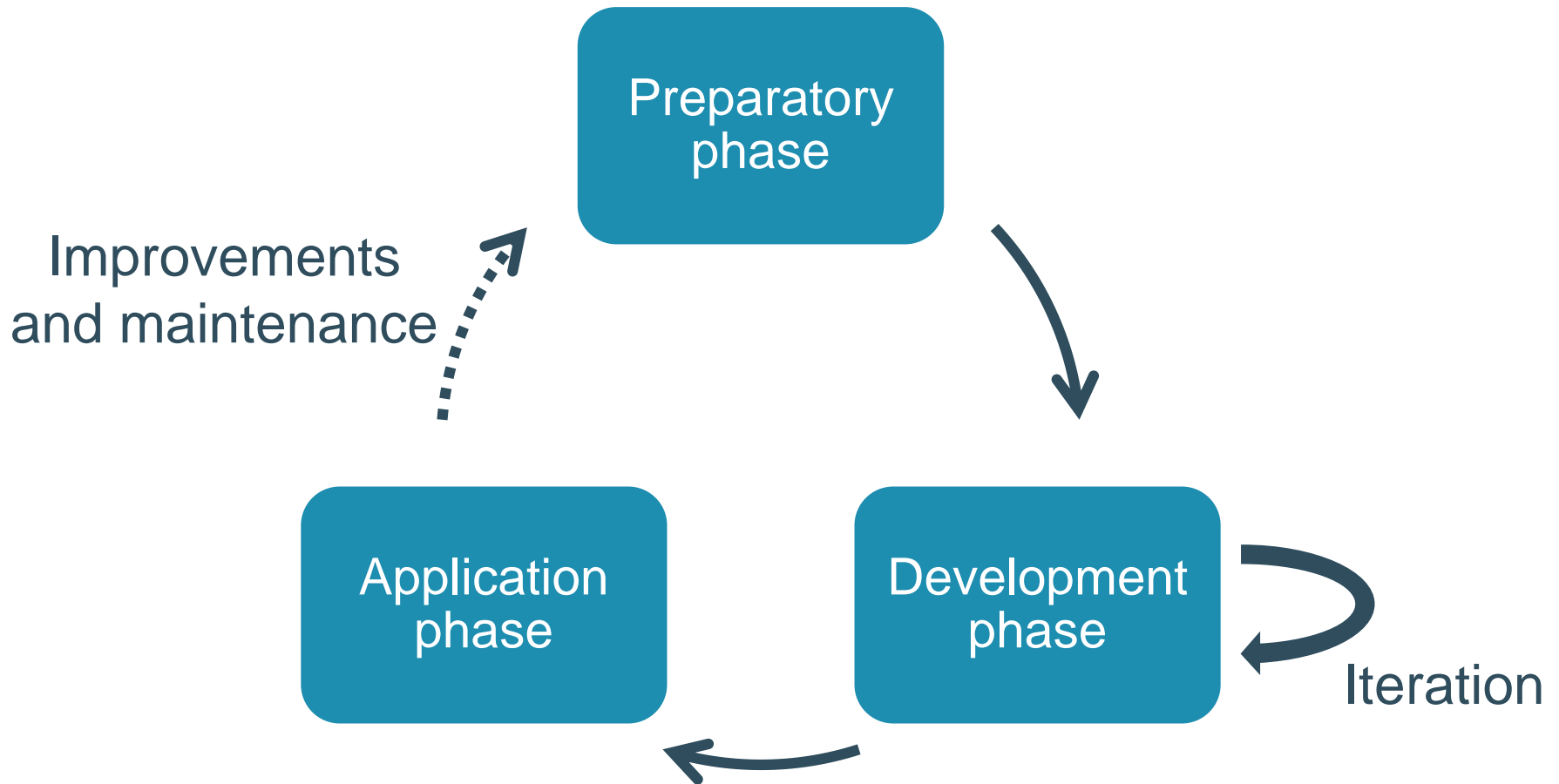
- Different methodologies for the design of new ontologies:

- On-To-Knowledge (OTK)
- DILIGENT
- NeOn
- ...



Typically a cyclic modeling approach (iterative process)

Introduction



Introduction

- Involved stakeholders:
 - Domain expert
 - Software developer
 - Ontology engineer
 - (end-users)



Different background and concerns

=> good communication during the process is paramount

Introduction

- Existing ontology engineering tools
 - OntoEdit
 - Hozo
 - WebODE
 - Protégé
 - ...
- Linked Data visualization tools
 - RDF Gravity
 - Fenfire
 - Frameworks such as Payola and LD-R
 - ...

Introduction

- Issues with:
 - Availability of the software
 - Functionality and user friendliness regarding all stakeholders
 - Combining ontologies (Tbox) and sample data (Abox) => implementation
 - No or limited Abox graph visualization
 - Sharing demo samples: documentation + triples (Abox and/or Tbox) + SPARQL

SPARQL-visualizer: implementation

- ✓ Single page application (angular)
- ✓ Open-source and available on [Github](#)
- ✓ [Online](#) and offline version (only client-side code)
- ✓ Force-directed graph visualization
- ✓ Built-in triplestore (rdfstore-js)
- ✓ External reasoner (Hylar)
- ✓ Connection to Stardog triplestore
- ✓ Samples (Abox and/or Tbox) can be shared via hyperlinks

SPARQL-visualizer: implementation

A SPARQL-visualizer | visualization

B Select dataset
Dataset
1: Simple

C Description
This is a simple way of describing requirements of a set of abstract spaces. Each space is assigned a space type as a string value, for querying for a specific type, or for updating/deleting/adding properties on a type level. The simple approach has the disadvantage of not being able to manage provenance or history of the properties, and it is nowhere stated that the `prop:area` is a requirement rather than an actual value.

D Triples
Turtle data to be queried

E Query
Query the data

F Query
Query result

The query result is a graph visualization showing the following triples:

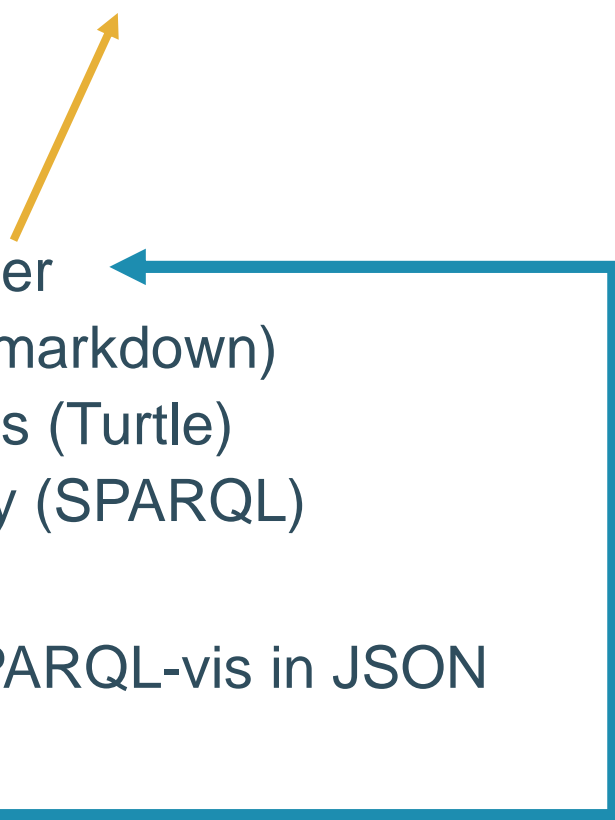
- `inst:SpaceA` (green node) `prop:area` `12 m2` (grey node)
- `inst:SpaceA` (green node) `prop:type` `Type A` (grey node)
- `inst:SpaceB` (green node) `prop:area` `12 m2` (grey node)
- `inst:SpaceB` (green node) `prop:type` `Type A` (grey node)
- `inst:SpaceA` (green node) `prop:type` `bot:Space` (red node)
- `inst:SpaceB` (green node) `prop:type` `bot:Space` (red node)

SPARQL-visualizer: use cases

- PROPS ontology (+ presentation about PROPS ontology):
 - Preparatory phase: 6 existing approaches
 - Development phase: 2 implementations
 - IFCtoLBD converter
 - Building requirements
- BOT ontology:
 - Application phase
 - Maintenance/improvement
- OPM ontology (PROPS L3):
 - Development phase

SPARQL-visualizer: hands-on

<https://madsholten.github.io/sparql-visualizer/>

1. Code editor:
 - i. New JSON file
 - ii. Component A: title + author
 2. Default SPARQL-visualizer in browser
 - i. Component C: tab description (markdown)
 - ii. Component D: tab sample triples (Turtle)
 - iii. Component E: tab sample query (SPARQL)
 3. Code editor:
 - i. Copy paste tab content from SPARQL-vis in JSON
 - ii. Component B: tab name ('title')
 - iii. Add comma (if necessary)
- 

Repeat steps for each new tab

SPARQL-visualizer: hands-on

4. Share the created sample as hyperlink
 - i. Upload JSON file on Dropbox (or Github) and create link
 - ii. Append link to *<https://madsholten.github.io/sparql-visualizer/>*

<https://madsholten.github.io/sparql-visualizer/>
+ ?file= + fileLink

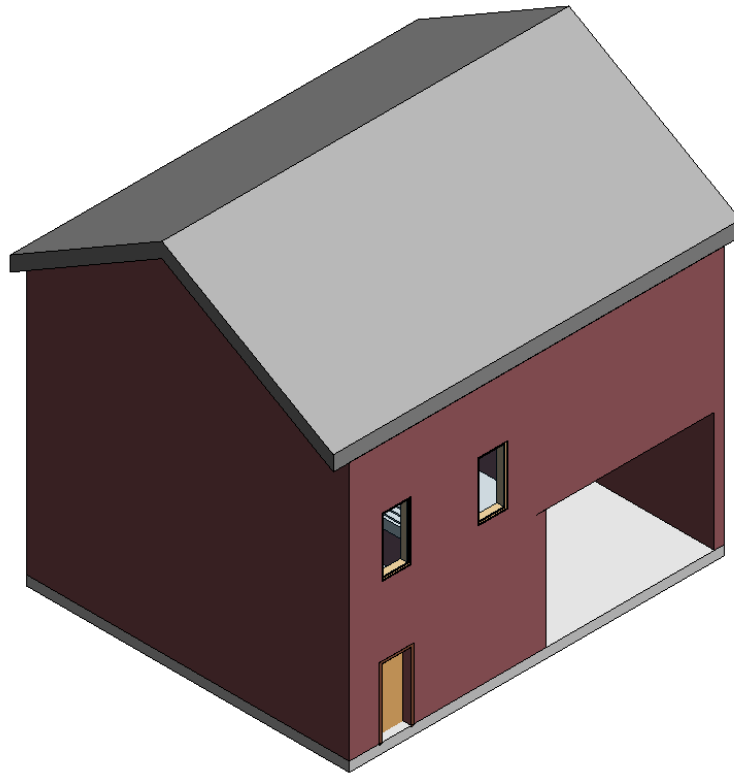
https://madsholten.github.io/sparql-visualizer/?file=https://www.dropbox.com/s/e5vmgovtg9rfsk6/LD-AC2018_spacesOverMultipleStoreys_componentA.json?dl=0

SPARQL-visualizer: hands-on

- Recommended:
 - Internet connection (online version of SPARQL-vis)
 - Cheat sheets:
 - [BOT](#) ontology (+ [docs](#))
 - [SPARQL](#) cheat sheet
 - [Markdown](#) cheat sheet
 - Code editor (e.g. Visual Studio Code)
 - Dropbox or Github account
- Optional:
 - [Stardog](#) triple store (requires Java 8)
 - Turtle validation service (e.g. [IDLab Turtle validator](#))

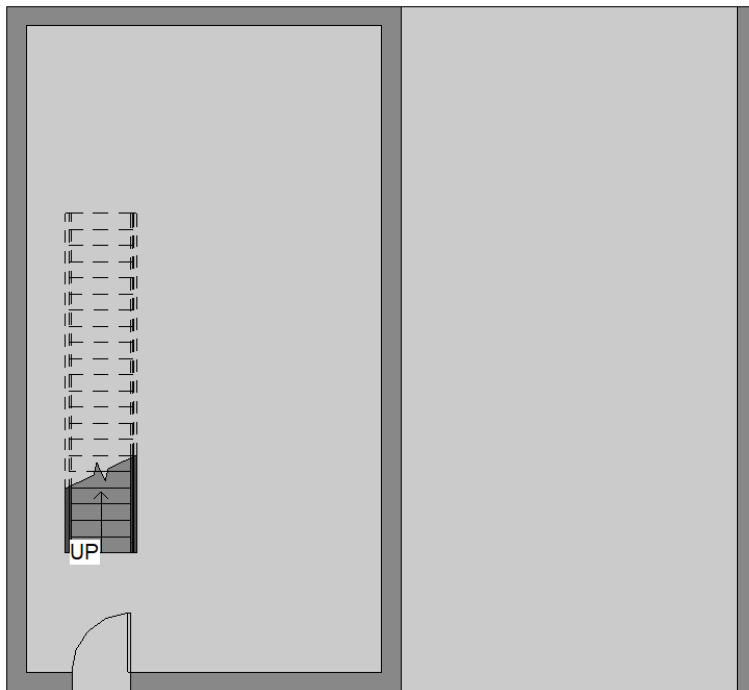
SPARQL-visualizer: hands-on

- Best modeling practices for BOT: example building

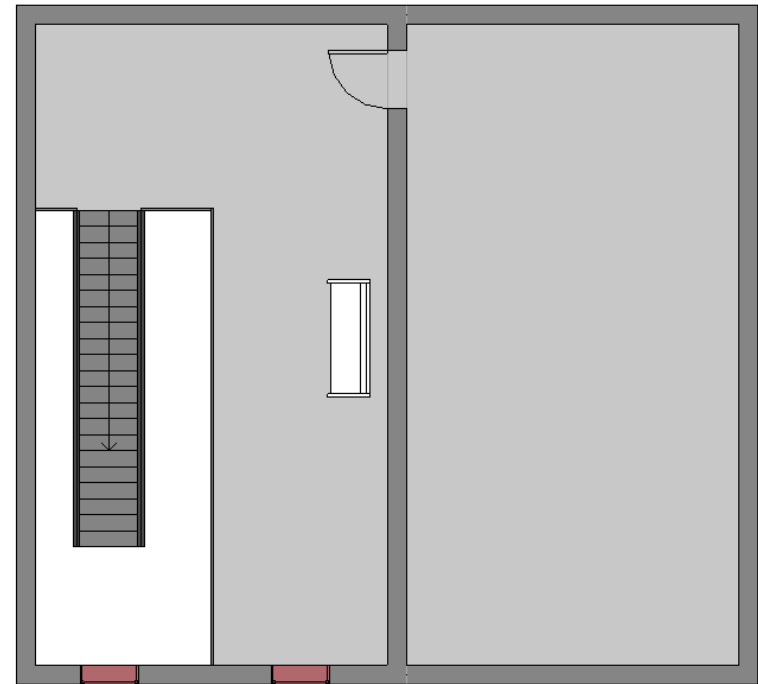


SPARQL-visualizer: hands-on

- Best modeling practices for BOT: example building



Ground floor



First floor

SPARQL-visualizer: hands-on

- Best modeling practices for BOT: stairwell space

https://madsholten.github.io/sparql-visualizer/?file=https://www.dropbox.com/s/e5vmgovtg9rfsk6/LDAC2018_spacesOverMultipleStoreys_componentA.json?dl=0

Conclusion

- Actively used during W3C LBD Telco at the 12th of March “Towards a PROPS ontology”
 - Sharing ideas effectively
 - Keeps presentation on topic
 - Lowers barrier for non-experts in Linked Data
- Plans to use it further in the PROPS subgroup and other groups
- Useful for dissemination: how to practically implement an ontology in applications (Abox + Tbox)
- Can also be useful for learning SPARQL/RDF(S)/OWL in general

Future work

- Making new samples => export complete JSON at once
- Make it possible to comment on others' examples
- Customize graph visualization:
 - color
 - linewidth
 - size
- Option to highlight inferred triples
- Support for other RDF triplestores and public SPARQL endpoints

Overview of hyperlinks

- SPARQL-visualizer
 - Github repo:
<https://github.com/MadsHolten/sparql-visualizer>
 - Online version:
<https://madsholten.github.io/sparql-visualizer/>
- W3C LBD Telco on PROPS (12th of March):
 - Presentation:
https://github.com/w3c-lbd-cg/lbd/blob/gh-pages/presentations/props/presentation_LBDcall_20180312_final.pdf
 - Minutes:
https://docs.google.com/document/d/1okoF68C_r1Uj3KqzSSXFalfZl3wE3eAeOsGLrBP1jEo/edit#
 - PROPS subgroup:
https://docs.google.com/document/d/1gl_eQSI60GuOmlwu4BgUqte_4nWgjCvx05W0-7yK9qw/edit#heading=h.q3ek0y1gzxg7
- PROPS discussion and examples:
<https://github.com/w3c-lbd-cg/props/issues/2>
- BOT examples:
<https://w3c-lbd-cg.github.io/bot/tutorial/>
- BOT: modeling stairwell space
https://madsholten.github.io/sparql-visualizer/?file=https://www.dropbox.com/s/e5vmgovtg9rfsk6/LDAC2018_spacesOverMultipleStoreys_componentA.json?dl=0
- OPM examples:
<http://www.student.dtu.dk/~mhoras/ldac2018/>

References

1. Bonduel, M.: Towards a PROPS ontology (2018), https://github.com/w3c-lbd-cg/lbd/blob/gh-pages/presentations/props/presentation_LBDcall_20180312_final.pdf, Date accessed: 2018-03-16
2. Casellas, N.: Methodologies, Tools and Languages for Ontology Design. In: Legal Ontology Engineering, vol. 3, pp. 57-107. Springer, Dordrecht (2011). <https://doi.org/10.1007/978-94-007-1497-7>
3. Chawuthai, R., Takeda, H.: RDF Graph Visualization by Interpreting Linked Data as Knowledge. In: Qi, G., Kozaki, K., Pan, J.Z., Yu, S. (eds.) Semantic Technology. JIST 2015. Lecture Notes in Computer Science. vol. 9544, pp. 23-39. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-31676-5_2
4. Guarino, N., Oberle, D., Staab, S.: What Is an Ontology? In: Staab, S., Studer, R. (eds.) Handbook on Ontologies. International Handbooks on Information Systems, pp. 1-17. Springer, Berlin, Heidelberg (2009). https://doi.org/10.1007/978-3-540-92673-3_0
5. Mizoguchi, R., Kozaki, K.: Ontology Engineering Environments. In: Staab, S., Studer, R. (eds.) Handbook on Ontologies. International Handbooks on Information Systems, pp. 315-336. Springer, Berlin, Heidelberg (2009). https://doi.org/10.1007/978-3-540-92673-3_14
6. Pauwels, P., Terkaj, W.: EXPRESS to OWL for construction industry: Towards a recommendable and usable ifcOWL ontology. Automation in Construction 63, 100-133 (2016). <https://doi.org/10.1016/j.autcon.2015.12.003>
7. Pinto, H.S., Tempich, C., Staab, S.: Ontology Engineering and Evolution in a Distributed World Using DILIGENT. In: Staab, S., Studer, R. (eds.) Handbook on Ontologies. International Handbooks on Information Systems, pp. 153-176. Springer, Berlin, Heidelberg (2009). https://doi.org/10.1007/978-3-540-92673-3_7
8. Rasmussen, M.H., Pauwels, P., Hvidd, C.A., Karlshj, J.: Proposing a Central AEC Ontology That Allows for Domain Specific Extensions. In: LC3 2017: Proceedings of the Joint Conference on Computing in Construction. pp. 237-244. Heraklion, Greece (2017)

Contact

Ing. Mathias Bonduel - PhD researcher @ FWO

Contact: mathias.bonduel@kuleuven.be

Website: <https://iiw.kuleuven.be/onderzoek/sustainable-buildings/flemishcitiesintransition>

Mads Holten Rasmussen – PhD researcher

Contact: mhoras@byg.dtu.dk

Website: <https://www.niras.dk/employees/mads-holten-rasmussen-mhra/>

Github repository:

<https://github.com/MadsHolten/sparql-visualizer>