Modular Schema Development for Wikibase

Pascal Hitzler
Data Semantics Laboratory (DaSe Lab)
Kansas State University

http://www.daselab.org

Collaborators:
Cogan Shimizu, Andrew Eells, Lu Zhou
Seila Gonzalez, Alicia Sheill, Catherine Foley, Dean Rehberger
DaSeLab, Kansas State University
MATRIX, Michigan State University
Knowledge Graphs vs. Linked (Open) Data

Knowledge Graphs are like linked data, with
- linking de-emphasized
- openness de-emphasized
- renewed consideration of schema quality
- tighter and central control
- clear transition from academia to industry

Goal is to produce a flexible, extendable, end-user friendly but in the end rather tightly controlled repository of integrated, re-useable data.
Knowledge graphs are critical to many enterprises today. They provide the structured data and factual knowledge that drive many products and make them more intelligent and "magical."

In general, a knowledge graph describes objects of interest and connections between them. For example, a knowledge graph may have nodes for a movie, the actors in this movie, the director, and so on. Each node may have properties such as an actor's name and age. There may be nodes for multiple movies involving a particular actor. The user can then traverse the knowledge graph to collect information on all the movies in which the actor appeared or, if applicable, directed.

Many practical implementations impose constraints on the links in knowledge graphs by defining a schema or ontology. For example, a link from a movie to its director must connect an object of type Movie to an object of type Person. In some cases the links themselves might have their own properties: a link connecting an actor and a movie might have the name of the specific role the actor played.
Wikidata

Started 2012 at Wikimedia Deutschland, with funding by Google, Yandex, Allen Institute for AI.

Crowdsourced knowledge graph, like Wikipedia is a crowdsourced encyclopedia.

Wikibase Software, Mediawiki interface.

Constrains graph shapes wrt. context information, references, datatype use. Otherwise crowdsourced.

Wikibase can be set up locally, used for your own knowledge graph. • powerful tool support for management, including interface
what does this look like in RDF?
Enslaved | Peoples of the Historic Slave Trade

Building a Linked Open Data Platform for the study and exploration of the historical slave trade.

Learn More
enslaved.org process

1. Quality Ontology Design.
2. Realization of ontology-based schema in Wikibase.
3. Knowledge graph construction and interaction through Wikibase as engine.
4. Additional front-end (simplified view)

(4) https://enslaved.org/
(3) https://lod.enslaved.org/

Focus of this talk: Going from (1) to (2)

>53M RDF triples from Wikibase export
### Statements

**instance of**
- Person
  - 0 references

**hasName**
- Sannom
  - 1 reference

**hasEventName**
- Sale of unnamed enslaved persons by Mathurin Guerin and sons (1817-6-7)
  - 1 reference

**hasEventType**
- Sale
  - 1 reference

**date**
- 7 June 1817 (Gregorian)
  - 1 reference

**hasSex**
- Male
  - 1 reference

**hasPlace**
- St. James
  - 1 reference

**hasPersonStatus**
- Enslaved Person
  - 1 reference
Ontology

- Modular ontology modeling process to obtain a high-quality knowledge graph schema. [Shimizu et al, 2021]

- Focus on Persons, Events, Places.

- Developed using an ontology design patterns approach.

- Primary concern: quality of the ontology for later re-use and/or extension.

- In particular: We ignored Wikibase constraints.
Enslaved Ontology [JWS 2020]
Ontology into Wikibase: Issues

• The Wikibase realization is similar but not identical to the ontology.
• The RDF export is based on an ontology (graph schema) that is inferred from the Wikibase realization.

• How difficult is it to map between these two ontologies (and thus, between these to graphs?)

• We used this as basis for an OAEI “complex ontology alignment” benchmark, published at CIKM 2020
  – This realistic benchmark is currently way beyond capabilities of automated ontology alignment systems.
Enslaved benchmark example mapping rules

\[\text{enslaved} : \text{Person}(x) \land \text{enslaved} : \text{hasRaceRecord}(x, y) \land \text{enslaved} : \text{RaceRecord}(y) \land \text{enslaved} : \text{asText}(y, z) \iff \]
\[\text{ed} : Q410(\text{Person})(x) \land \text{ep} : P32(\text{hasRaceColor})(x, y) \land \text{wikibase} : \text{Statement}(y) \land \text{eps} : P32(\text{hasRaceColor})(y, z)\]

\[\text{enslaved} : \text{Person}(w) \land \text{enslaved} : \text{hasAgeRecord}(w, x) \land \text{enslaved} : \text{AgeRecord}(x) \land \text{enslaved} : \text{hasAgeValue}(x, z) \iff \]
\[\text{ed} : Q410(\text{Person})(w) \land \text{ep} : P42(\text{hasAge})(w, x) \land \text{wikibase} : \text{Statement}(x) \land \text{eps} : P42(\text{hasAge})(x, y) \land \]
\[\text{ed} : Q424(\text{AgeRecord})(y) \land \text{edt} : P3(\text{hasAgeValue})(y, z)\]

\[\text{enslaved} : \text{Person}(x) \land \text{enslaved} : \text{hasInterAgentRelationshipRecord}(x, y) \land \text{enslaved} : \text{InterAgentRelationshipRecord}(y) \land \]
\[\text{enslaved} : \text{hasInterAgentRelationshipType}(y, z) \land \text{enslaved} : \text{InterAgentRelationshipTypes}(z) \iff \]
\[\text{ed} : Q410(\text{Person})(x) \land \text{ep} : P39(\text{hasInterAgentRelationshipTypeTo})(x, y) \land \text{wikibase} : \text{Statement}(y) \land \]
\[\text{eps} : P39(\text{hasInterAgentRelationshipTypeTo})(y, z) \land \text{ed} : Q463(\text{InteragentRelationship})(z)\]

\[\text{enslaved} : \text{Person}(x) \land \text{enslaved} : \text{hasParticipantRoleRecord}(x, y) \land \text{enslaved} : \text{ParticipantRoleRecord}(y) \land \]
\[\text{enslaved} : \text{roleProvidedBy}(y, z) \land \text{enslaved} : \text{Event}(z) \iff \]
\[\text{ed} : Q410(\text{Person})(x) \land \text{ep} : P17(\text{hasParticipantRole})(x, y) \land \text{wikibase} : \text{Statement}(y) \land \]
\[\text{epq} : P19(\text{roleProvidedBy})(y, z) \land \text{ed} : Q238(\text{Event})(z)\]

\[\text{enslaved} : \text{Person}(w) \land \text{enslaved} : \text{hasNameRecord}(w, x) \land \text{enslaved} : \text{NameRecord}(x) \land \]
\[\text{enslaved} : \text{hasPreferredNameVariant}(x, y) \land \text{enslaved} : \text{NameVariant}(y) \land \text{enslaved} : \text{fullNameAsString}(y, z) \iff \]
\[\text{ed} : Q410(\text{Person})(w) \land \text{ep} : P20(\text{hasName})(w, x) \land \text{wikibase} : \text{Statement}(x) \land \text{eps} : P20(\text{hasName})(x, z)\]
Ontology modeling for Wikibse

• Provide “patterns” (Ontology design patterns) that bridge the gap between modular ontology modeling and Wikibase use for the corresponding graph.

• The patterns shall give a traditional modeling look-and-feel,
• while at the same time restricting the modeling to seamless transfer to Wikibase.

• Modeling with domain experts done mostly via schema diagrams
  – technique taken from Modular Ontology Modeling
  – there is always an underlying (OWL) axiomatization of course.
Wikibase ODPs: dateTime
Wikibase ODPs: decimal

Diagram of Wikibase ODPs showing relationships and properties.
Wikibase ODPs: datatype properties
Enslaved redone
Enslaved redone
Enslaved redone
Enslaved redone
Enslaved redone
Forthcoming?

Certainly:

• Axiomatization, ShEx

Possibly:

• a variety of tool support
Contribution Summary

• With the patterns we now have, we could have saved a lot of time earlier.

• Patterns and approach should be adoptable by anybody who is interested in bridging
  – traditional ontology modeling and
  – use of Wikibase as graph platform.

• Perhaps it can even help stimulate discussions on recommended patterns for Wikibase?
Thanks!

Collaborators:

Cogan Shimizu, Andrew Eells, Lu Zhou  
DaSeLab, Kansas State University

Seila Gonzalez, Alicia Sheill, Catherine Foley, Dean Rehberger  
MATRIX, Michigan State University
References


References


