Making use of Ontologies: Tools, Services, and Techniques

Ontology Summit 2014

http://ontolog.cim3.net/cgi-bin/wiki.pl?OntologySummit2014_Ontology_Tools_Services_Techniques_Synthesis

Christoph Lange[1] (co-champion: Alan Rector[2])

[1] Enterprise Information Systems,
University of Bonn / Fraunhofer IAIS, Germany
[2] Medical Informatics, School of Computer Science
University of Manchester, UK

Mission Statement

The Web of Data . . .

- provides great opportunities for ontology-based *services*, but also
- puts challenges to
  - *tools* for editing and using ontologies, and to
  - *techniques* for ontological reasoning and ontology engineering.

Terminology: Web of Data, Big Data, Semantic Web

Here, we use “Web of Data” to subsume both . . .

- Big Data (w.r.t. volume, velocity, variety)
- Semantic *Web*
  - making sense of knowledge distributed over the Web
  - not just using IRIs as local symbol names

Research Questions

- How can tools and techniques scale to the Web?
- How can services benefit from tapping into the Web?
  How can they help to make Big Data manageable?

Track Structure

- two productive sessions, on 2014-01-30 and 2014-03-13, with 3 panelists each
- little further community input via the wiki and the mailing list
- one project participated in the Hackathon: Ontohub, was also presented in the 2014-01-30 session

First Session (2014-01-30)

- **TillMossakowski**: scaling an ontology **tool** suite (Hets/Ontohub) from “reasoning in the small” to the Web
- **ChrisWelty**: the potential of linking Big Data to ontological reasoning, as demonstrated by the IBM Watson natural language question answering **service**
- **AlanRector**: OWL and alternative modeling **techniques**, reviewed from the perspective of engineering knowledge-rich systems.

`http://ontolog.cim3.net/cgi-bin/wiki.pl?ConferenceCall_2014_01_30`

Second Session (2014-03-13)

- **MikeBergman**: OSF, an enterprise platform that integrates and enhances several well-known ontology **tools**
- **JoseMariaGarcia**: combining linked data technology with web **services**
- **MariaPovedaVillalon**: a **technique** for engineering linked data vocabularies, i.e. lightweight ontologies that scale to the Web

`http://ontolog.cim3.net/cgi-bin/wiki.pl?ConferenceCall_2014_03_13`

Services Enabled by Non-Standard Reasoning Techniques

Services can draw on a large knowledge pool by tapping into the Web – but are ontologies relevant for this?

- IBM Watson . . .
  - answers rich natural language questions over a broad domain of knowledge
  - gives precise answers, an accurate assessment of confidence and consumable justifications in seconds.
- Reasoning: Hypotheses → Evidence → Scoring
  - Don't build a formal ontology of the whole world, . . .
  - . . . but learn local ontologies on demand, also using informal sources, and different reasoning techniques:
    - type disambiguation using taxonomies
    - evidence scoring using machine learning

Web-wide Ontological Commitments?

- Watson limited to simple taxonomies
- Large collaboration efforts may agree on a limited set of ontologies
- Is it reasonable and feasible to turn Big Data into ontologies? – It's *difficult* in any case!
  - Manually building ontologies is labor intensive.
  - Data mining suffers from potential inconsistency, incompleteness, irrelevance of data "out there".
  - Machine learning of ontologies requires further research.
- Merely using ontologies for *annotating* big data with terms may suffice.

Should Ontologies Cover Everything?

- Traditional ontology languages assume universal knowledge.
  OWL is good for this.
- In the real world, knowledge is often contingent, accidental or particular.
  Template formalisms such as frames, UML or rules are good for this.
- Translations across formalisms not yet well understood
- RDF(S) + SPARQL usage outnumbers OWL usage
  - . . . but users are often ignorant of formal semantics.
  - Still it copes well with heterogeneous data (variety)

Is OWL still useful?

Yes!

- E.g., in the OSF, using OWL allows for
  - duplicate names
  - incomplete information (thanks to open world assumption)
  - extensibility to multiple schemas
- Lots of tools and techniques (but most date back to small, hand-made ontologies):
  - limited to single or few formalisms
  - similar to knowledge silo-ing
- Can use OWL more creatively
  e.g. take inspiration from template formalisms
- OntolOp translates between OWL and other formalisms

Beyond a Single Ontology Language

- OntoIOp supports alignments and reasoning across ontology languages.
- Not yet “big” w.r.t. volume and velocity
  . . . but w.r.t. variety
- OntoIOp retrofits linked data conformance (e.g. IRI identifiers) into pre-Web languages
- Growing tool support: Ontohub (→ Hackathon)

RDF as a Knowledge Representation Foundation

RDF is the "native language" of Linked Data:

- enforces a low ontological commitment . . .
- . . . but still allows to link to complex descriptions

E.g., the Open Semantic Framework (OSF) uses a single, internal, canonical data model (RDF and some OWL):

- representing structured, semi-structured, unstructured data
- data structures translate into web widgets; ontologies
  - inform interface displays
  - define component behaviors
  - guide visualization template selection and content

Linked Web Services

- Web services:
  1. Service provider registers service in central registry
  2. Service consumer finds service . . .
  3. . . . and communicates with it to execute it
- *Semantic* web services go beyond syntactic descriptions (e.g. WSDL) – previous state:
  - web services exchanging heavy XML messages over SOAP
  - semantics-first modeling using expressive WSMO or OWL-S ontologies
- Face the reality:
  - lightweight REST interfaces much more popular
  - describe their semantics bottom-up in a linked data style: **Linked Services** (e.g. Linked USDL lightweight ontology)

Engineering Vocabularies

"Vocabulary" = "Lightweight Ontology"
Linked Open Terms, an agile engineering technique:

1. determine the terms needed to describe your data
2. look for them in existing vocabularies (a lot exist on the Web!)
3. create your own when necessary, but link to existing ones
4. continuous evaluation

Conclusion

- Lightweight means Scalable
  - Heavyweight semantic web services have failed
  - A little RDF goes a long way
  - Even vocabularies can be engineered systematically
- Remaining Challenges
  - Visualisation
  - Scalability of reasoners
  - Requirements for ontology-based tools, services and techniques in a big data world still unclear.