Ontology Repositories Make a World of Difference

Michael Grüninger

Semantic Technologies Lab
University of Toronto

July 11, 2013

Origins

- Ontologies (as we know them today) arose from the Sharable Reusable Knowledge Bases project.
  - 20th anniversary of Gruber's paper "A translation approach to portable ontologies"
- However, people still treat ontologies like toothbrushes – they don't want to share or reuse one from anyone else.

Obstacles to Sharing and Reuse?

1. Ad hoc ontology representation languages or languages without explicit semantics
2. Different ontology representation languages and logics
3. Insufficient axiomatizations of ontologies
4. Inability to compare ontologies

Implemented Ontologies

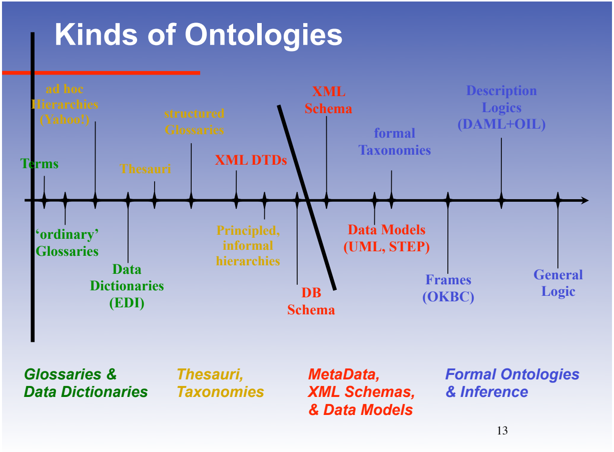
Workshop on Implemented Ontologies, ECAI 1994 Amsterdam.

- Wide variety of languages
- Many ontologies were odd combinations of logic and "creative" implementations techniques

Sadly, the idea of submitting your ontologies with your paper did not become a best practice ...

Ontology Spectrum

Are Ontologies Expert Systems All Over Again?, AAAI 1999, Orlando.



Common Logic

- Common Logic (published as "ISO/IEC 24707:2007 — Information technology Common Logic : a framework for a family of logic-based languages") is a language based on first-order logic, but extending it in several ways that ease the formulation of complex ontologies that are definable in first-order logic.
- Common Logic addresses the question
  Which languages are syntactic variants but semantically equivalent?

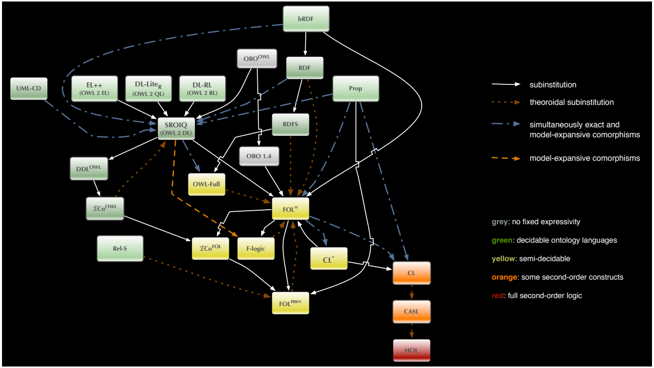
OntoIOp

The Distributed Ontology Language (DOL) provides a meta-language that provides support for

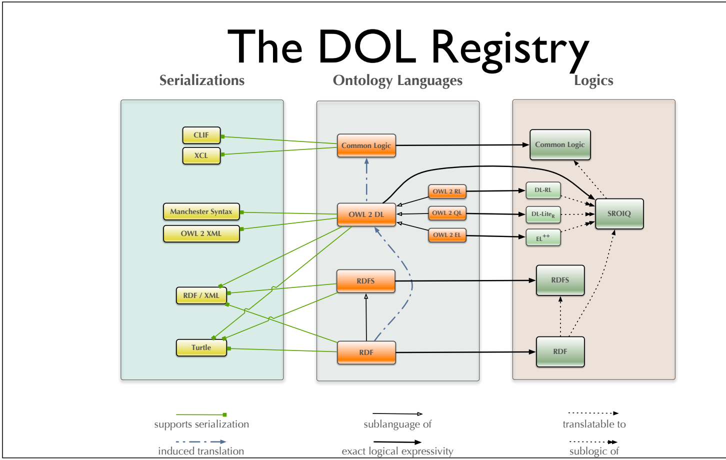
- Support multiple logics through translation
- formal and informal links: e.g. logic translations, theory interpretations, and ontology alignment as first-class citizens
- general structuring principles and modularity concepts (inherited from algebraic specification and institution theory)
- logically heterogeneous ontologies
- annotation, metadata, and documentation

See [Mossakowski, Kutz, and Lange 2012]

Language Translations in DOL



DOL Registry



OntoHub

An open ontology repository for DOL conforming ontologies
`http://ontohub.org`

Plurality of Ontology Representation Languages

With a repository such as OntoHub, a diversity of ontology representation languages and logics is not an obstacle to ontology sharing and reuse.

Challenges

1. Relationships between languages with different expressiveness
   - What are maximal translations of one language into a less expressive language?
2. Implementation of seamless translators
3. Formal specification of semantics for all ontology representation languages

One Ring to Bring Them All and In the Darkness Bind Them

The temptation to organize all ontologies as specializations of a standard upper ontology.

Upper Ontology Summit 2006: SUMO, Cyc, ISO 15926. DOLCE, BFO, PSL

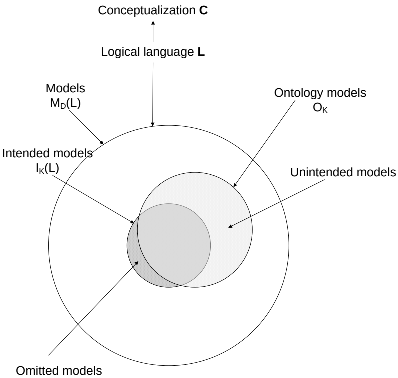
- How do we agree on the upper ontology?
- What properties should an upper ontology satisfy?
- How do we evaluate an upper ontology?
- What if an ontology disagrees with the upper ontology?

Axiomatizations of Ontologies

- A major obstacle to ontology sharing and reuse appears with ontologies in which the intended semantics is captured primarily by the documentation rather than the axiomatization.
- The problem is not disagreement about multiple ontologies, but disagreement about the intended models of the ontologies

Ontology Verification

With verification, we want to characterize the models of an ontology up to isomorphism and determine whether or not these models are equivalent to the intended models of the ontology. (Diagram from Guarino (2009))



Why Care About Ontology Verification?

Semantic Integration

- Software systems are semantically integrated if their sets of intended models are equivalent.
- Guarantee that the inferences made with exchanged sentences are equivalent to the inferences made with respect to the system's intended models

Why Care About Ontology Verification?

Decision Support Systems

- Verification $\Rightarrow$
  any inferences drawn by a reasoning engine using the ontology are actually entailed by the ontology's intended models.
- If an ontology's axiomatization has unintended models, then it is possible to find sentences that are entailed by the intended models, but which are not provable from the axioms of the ontology.

Representation Theorems

Representation theorems show that there is a one-to-one correspondence between the set of models of the ontology and the corresponding class of mathematical structures.

Definition

A class of structures $\mathfrak{M}$ can be represented by a class of structures $\mathfrak{N}$ iff there is a bijection $\varphi : \mathfrak{M} \rightarrow \mathfrak{N}$ such that for any $\mathcal{M} \in \mathfrak{M}$, $\mathcal{M}$ is definable in $\varphi(\mathcal{M})$ and $\varphi(\mathcal{M})$ is definable in $\mathcal{M}$.

Ontology Verification: Model Theory

The necessary direction of a representation theorem (i.e. if a structure is intended, then it is a model of the ontology's axiomatization) can be stated as

$$\mathcal{M} \in \mathfrak{M}^{intended} \Rightarrow \mathcal{M} \in Mod(T_{onto})$$

The sufficient direction of a representation theorem (any model of the ontology's axiomatization is also an intended structure) can be stated as

$$\mathcal{M} \in Mod(T_{onto}) \Rightarrow \mathcal{M} \in \mathfrak{M}^{intended}$$

This is Too Hard!

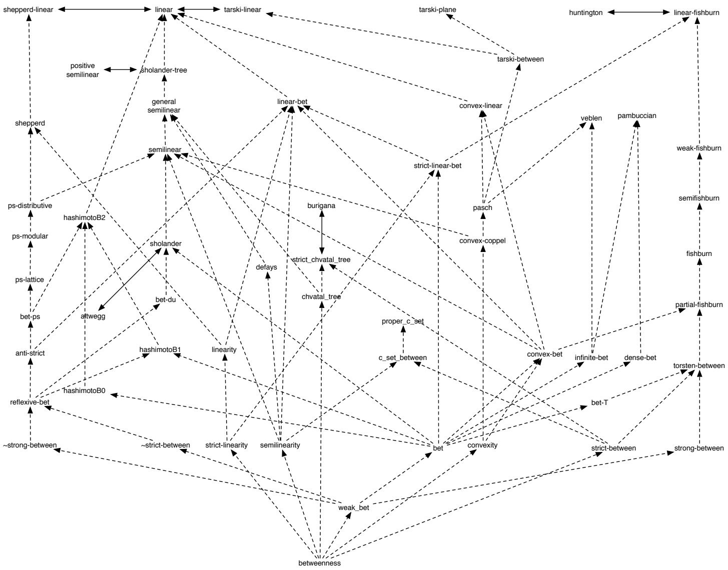
- Relationships between first-order ontologies within a repository can be used to support ontology verification.
- The fundamental insight is that we can use the relationships between ontologies to assist us in the characterization of the models of the ontologies.
- The objective is the construction of the models of one ontology from the models of another ontology by exploiting the relationships between these ontologies and their modules in the repository.

COLORE

- The COLORE (Common Logic Ontology Repository) project is building an open repository of ontologies specified using Common Logic (ISO 24707).
  - Testbed for ontology evaluation and integration techniques, and that can support the design, evaluation, and application of ontologies in rst-order logic.

http://code.google.com/p/colore/source/browse/trunk/

Betweenness



Interpretability

Specify mappings between models by specifying the relationships between modules in the repository.

- Interpretable – mapping between theories that preserves theorems
- Faithfully interpretable – mapping between theories that preserves models
- Definable equivalence – theories that are interpretable into each other.

Interpretability

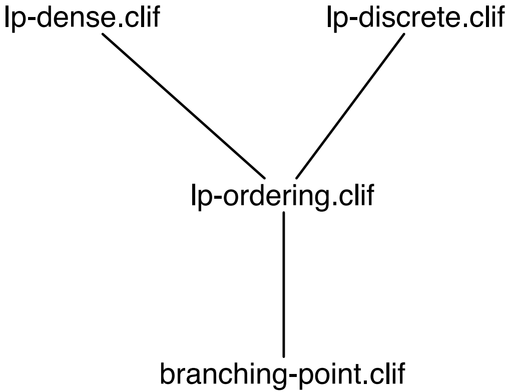


Figure: Some timepoint ontologies in COLORE.

Interpretability

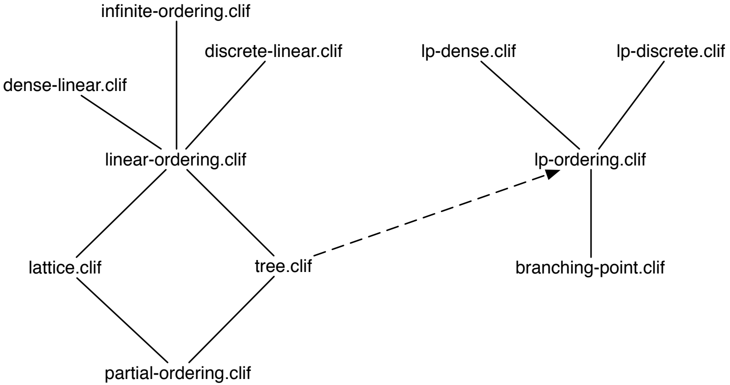


Figure: Relationship to ontologies for linear orderings in COLORE.

Interpretability

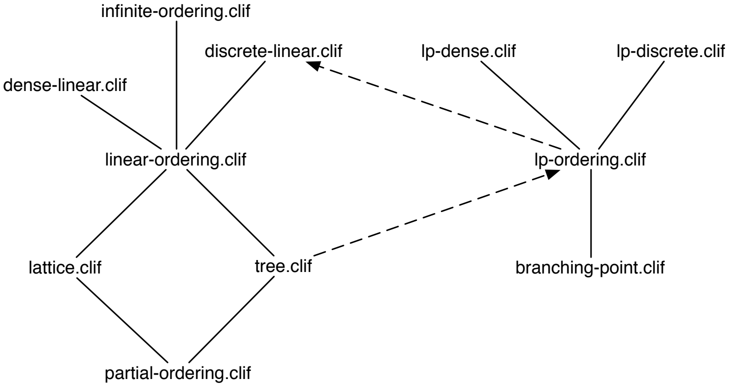


Figure: Relationship to ontologies for linear orderings in COLORE.

Interpretability

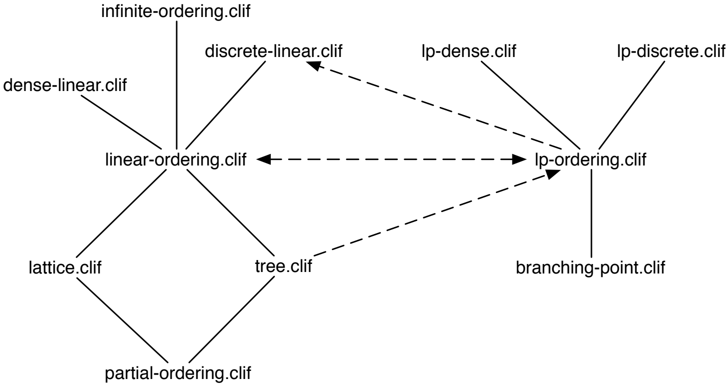


Figure: Relationship to ontologies for linear orderings in COLORE.

Ontology Verification: Reasoning

The necessary direction of the representation theorem is equivalent to the following reasoning task:

$$T_{onto} \cup \Delta \models T_1 \cup \ldots \cup T_n \qquad \textbf{(Rep-1)}$$

The sufficient direction of the representation theorem is equivalent to the following reasoning task:

$$T_1 \cup \ldots \cup T_n \cup \Pi \models T_{onto} \qquad \textbf{(Rep-2)}$$

Using COLORE

- Ontology relationships
- Ontology design
- Ontology recognition
- Ontology design patterns
- Modularization

Ontology Relationships

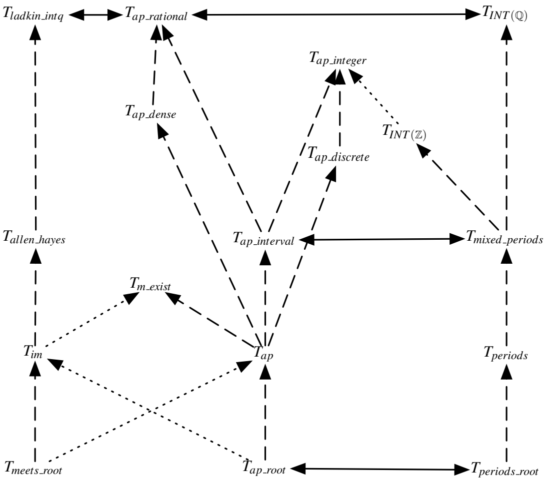


Figure: Time interval ontologies in COLORE.

Ontology Design

- Trunk theories capture the basic ontological commitments in the hierarchy
- We can use complete extensions of theories to explore the set of possible trunk theories for sets of new axioms that make "ontological" sense.

Ontology Design

- Maximal Horn subtheories of an ontology
- Definability of classes of structures in different description logics

Ontology Recognition

Which is the right ontology for me?

- Using a set of examples specified as a set of models, we can search through a hierarchy in the repository to find the ontology whose models are equivalent to the examples.

Reuse and Verification

The core ontologies used in the verification of other generic ontologies can be reused across multiple ontologies.

Mereology

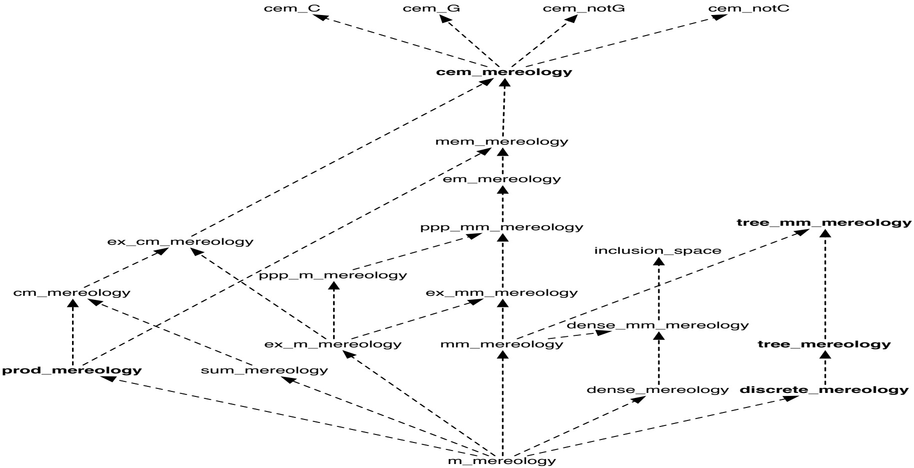


Figure: Ontologies in $\mathbb{H}^{mereology}$: the hierarchy of mereologies.

Orderings

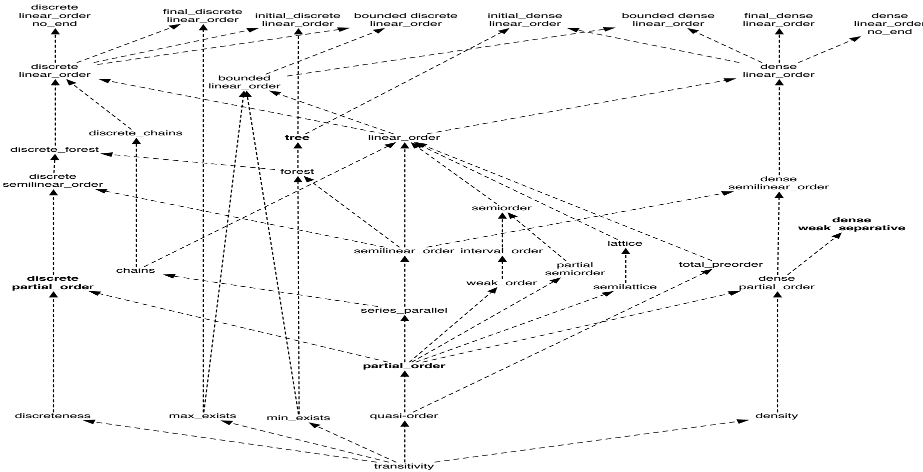


Figure: Ontologies in $\mathbb{H}^{ordering}$: the core hierarchy of orderings.

Subposet Hierarchy

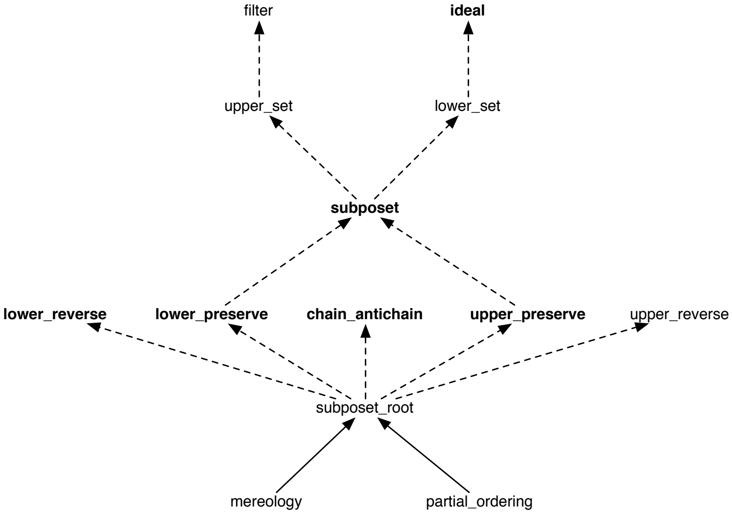
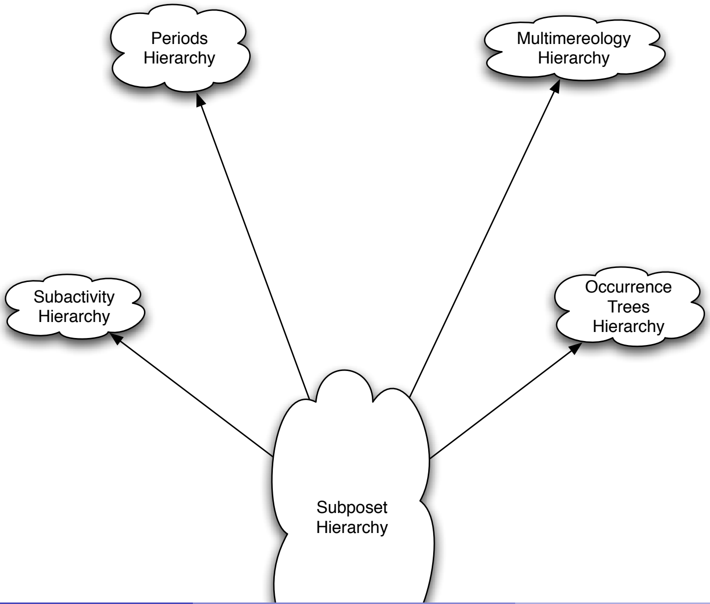


Figure:

Ontology Patterns and Reducibility



Modularization

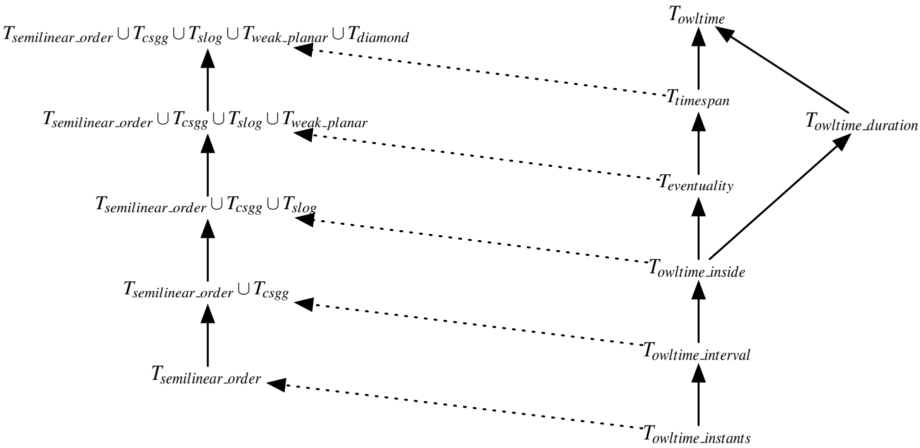


Figure: Modularization of OWL-Time.

Practicality?

A similar approach was implemented in the Ontolingua system, and did not scale from a practical point of view. Why will COLORE work?

- Earlier approaches have not explicitly used metatheoretic relationships between ontologies.
- Methodology of ontology verification

Summary

Diversity without divergence

- The barriers to ontology sharing and reuse rest on ambiguity in cases where there is a lack of formal semantics or unintended models.
- Relationships between ontology representation languages and relationships among ontologies within a repository can be used to support ontology design and evaluation.