

Common Logic for the Common User

Michael Grüninger

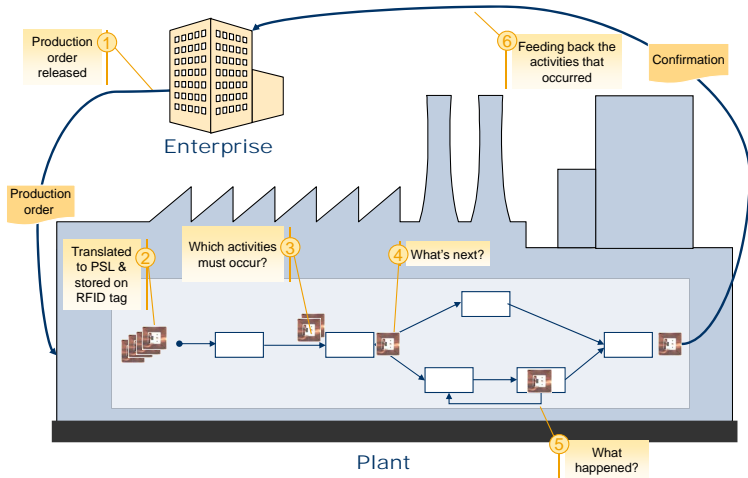
Semantic Technologies Lab
University of Toronto

OASIS Symposium/ISWC2011

Semantic Integration



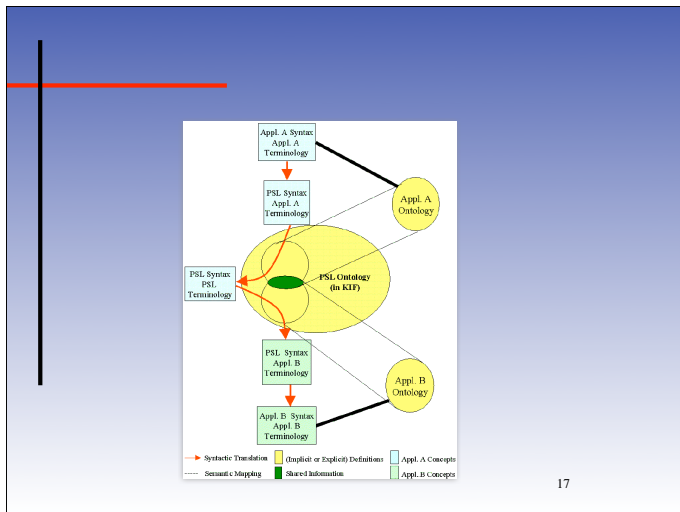
Decision Support



What Is Common Logic?

- Common Logic is a family of logic languages designed for use in the representation and interchange of knowledge among disparate computer systems. e.g. the specification of expressions that are the input or output to automated reasoning systems such as theorem provers and constraint satisfaction systems.
- Combines KIF (Knowledge Interchange Format) and CG (Conceptual Graphs).

Semantic Integration Revisited



17

Using Translation Definitions

ILOG Schedule:

```
(forall(occ1 occ2 d a)
  (iff      (endsAfterStart occ1 occ2 d a)
            (exists(s1 s2)
              (and      (leaf_occ s1 occ1)
                        (root_occ s2 occ2)
                        (soo_precedess2 s1 a)
                        (=d (delay s1 s2)))))))
```

SAP:

```
(forall(o1 o2 a)
  (iff      (next_op o1 o2 a)
            (exists(a1)
              (and      (subactivity a1 a)
                        (simple_sequencea1)
                        (soo_precedes o1 o2 a1)
                        (soo_precedes o1 o2 a))))))
```

Decision Support Revisited

Which subactivities can possibly occur next after an occurrence of fabrication?

```
(exists (a)
  (forall (s1)
    (if (occurrence_of s1 fabrication)
      (exists (a2 s2)
        (and (subactivity a2 a)
              (occurrence_of s2 a2)
              (next_subocc s1 s2 a))))))
```

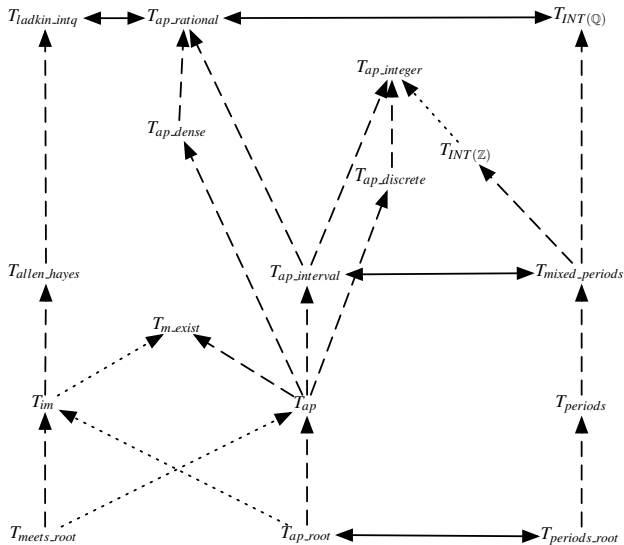
COLORE

- 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.

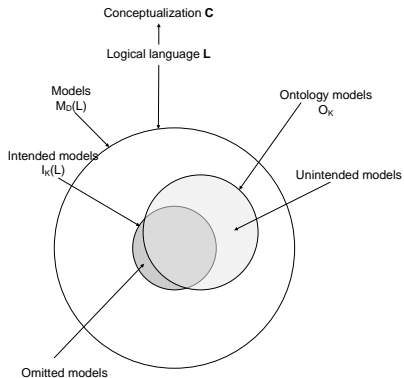
<http://stl.mie.utoronto.ca/colore/>

<http://code.google.com/p/colore>

Relationships between Ontologies



Verification of Ontologies



Consistency is not enough ...

The following set of sentences is consistent

```
(forall (x)
  (not (supervises x x)))
```

```
(forall (x y)
  (if (and (ceo x)
           (employee y))
      (supervises x y)))
```

```
(forall (x)
  (if (ceo x)
      (employee x)))
```

but it has an unintended consequence.

Why Care About Ontology Verification?

Semantic Integration

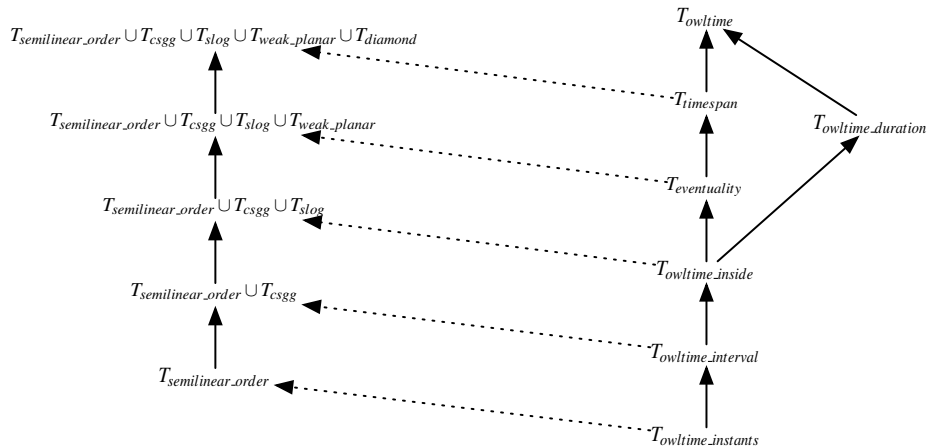
- Software systems are semantically integrated if the sets of intended models for their ontologies are equivalent.
- Guarantee that the inferences made with exchanged sentences are equivalent to the inferences made with respect to the system's intended models

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.

COLORE and Ontology Evaluation



- MACLEOD, A Common Logic Environment for Ontology Development
 - A proposed development environment to guide and facilitate the development of ontologies with this lifecycle
 - Assisted documentation to reduce workload and aid decision-making
 - Track potential impact of revisions on verification results

Thoughts

- Design and evaluation of good ontologies to support semantic integration and decision support.
- Focus on the applications and build the ontologies that are needed for these applications.