Objectives of the Common Logic Ontology Repository Project

Michael Grüninger

Semantic Technologies Lab
University of Toronto
(with Notes from Cameron Ross, Kojeware Corporation)

September 10, 2010

Ontological Engineering

Establish a testbed for integration and evaluation techniques

- Relationships between ontologies
  - ▶ Relative interpretation
  - ▶ Definable equivalence
- Ontology verification (FOIS 2010)
  - ▶ Reducibility
  - ▶ Using theorem provers and model construction software for ontology verification and semantic integration.

Ontological Engineering

Support the design and reuse of modular ontologies

- Criteria for modularity
- What are impediments to reusability?
- Ontology design for reuse
- Constructing ontologies by assembling/reusing existing modules

Common Logic

Develop techniques for representing ontologies using CL

- How do we assist the user in writing axioms?
- CL Ontology Design Assistants (Hashemi 2009)
- CL module/named text management

Common Logic

Promote ISO 24707 for constructing ontologies using first-order logic

- There is a lack of substantial robust CL ontologies for a wide range of generic and domain-specific concepts
- Applications of CL ontologies – decision support, semantic integration, search

Common Logic

Develop best practices for ontological engineering using CL

`http://code.google.com/p/colore/wiki/ClifBestPractices`

- Employ modular design principles
- Avoid unnamed texts
- Name things using URNs

Repositories

Establish an open repository of CL-based ontologies

- How is a repository more than a library?
- What is the role of reasoning services in an ontology repository?

Enable proliferation and interoperation of CL-based repositories

Independent CL repositories should also be able to join a federated system of other CL repositories.

Software Functionality

Develop APIs and services for accessing CL-based repositories

Focus on services specific to CL.

Identify services that are generic for any language.

Support both web-based and non-web-based applications

How do name things in a manner that supports both environments?

Enable and promote an ecosystem of both open source and commercial developments focused on CL technology

- editors
- translators
- reasoning engines