Common Logic (ISO 24707)

Michael Grüninger

SC32 WG2 Meeting, Toronto, Ontario, Canada

November 20, 2014
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.
Comments from CD Ballot

Comments on CD ballot were discussed in Beijing, and most were resolved at the meeting.

Outstanding Issues:
- Changes to abstract syntax
- Conformance conditions
- Datatypes
- Numerical quantifiers
Abstract Syntax

- Abstract syntax extended to include schema for sequence variables
- Metamodel for specifying relationships among the syntactic categories
  - mathematical specification
  - UML
Concrete Syntaxes

Revisions for the concrete syntaxes are waiting on completion of abstract syntax:

1. Common Logic Interchange (CLIF)
2. Conceptual Graph Interchange Format (CGIF)
3. XCL
Do we specify pre-defined levels for partial syntactic conformance?

- Compact dialect (CL without sequence variables)
- CL without imports, CL without domain restrictions
Semantic Conformance

- Any CL dialect shall have a model-theoretic semantics, defined on a set of interpretations, called dialect interpretations, which assigns one of the two truth-values true or false to every sentence, phrase (except comment) or text in that dialect.

- A dialect is exactly semantically conformant when, for any syntactically legal sentence, phrase (except comment) or text $T$ in that dialect, the following two (separate) conformance conditions are true:
  - For every dialect interpretation $J$ of $T$, there exists a Common Logic interpretation $I$ of $T$ with $I(T) = J(T)$
  - For any Common Logic interpretation $I$ of $T$, there exists a dialect interpretation $J$ of $T$ with $I(T) = J(T)$
Partial Semantic Conformance

- Translation preserves entailment
- Translation preserves satisfiability
  - OWL to CL
- Translation preserves models
  - Classical FOL (e.g. TPTP) to CL
- Translation is an inclusion relation
  - Dialect without sequence variables to CL
Datatypes: Abstract Syntax

- Use "datatype" in the sense of the XSD Datatypes specification: a mapping from a lexical space, which can be represented explicitly in the syntax, to a value space, which is arbitrary.

- We need to add a syntactic category to the abstract syntax called something like "interpreted names" (following the original CLIF) as a subset of names. This syntactic category would have disjoint subsets:
  - interpreted names that denote directly, especially for elements of the lexical spaces of other datatypes (e.g. strings)
  - a syntactic construction corresponding to applying a named datatype mapping to something in its lexical space to get a name denoting the corresponding member of the datatype's value space. It would be invalid to apply a datatype mapping to something that is not in its lexical space.
Datatypes: Semantics

- The denotation of interpreted names are the same in all interpretations.
- The denotations of interpreted names of default datatypes are defined explicitly.
Numerical Quantifiers

- Rather than include numerical quantifiers (e.g. there exist exactly 9 judges on the Supreme Court) in the abstract syntax, it is possible to specify numerical quantification with axioms that use sequence variables.
Project Issues

- Submit for DIS ballot on December 30, 2014 ()
- British experts to review the document?