Requirements for an Expressive Semantic Web Rule Language

Michael Kifer
State University of New York at Stony Brook
The Role of a Rule Language: The RuleML, WSMO, SWSL View

- FOL++
- Rules
- OWL
- RDF(S)
- XML
- Unicode
- URI
It’s The Features, Stupid!

• Prolog was unsuccessful not because of performance, but because of features
  – Semantics: Not really declarative hence
  – Features: Fairly feature-less and low-level
What Is To Be Done

• Fix the semantics
• Add features
• Web-ize
Fixing the Semantics

• Pretty much done: systems like XSB use tabling to
  – Fix the incomplete Prolog’s search strategy
  – Implement the well-founded semantics for NAF (negation as failure)

• And they run fast!
Adding Features

• Not all features can be dropped into one language

• But the ones to be discussed are orthogonal: can be combined and have been combined (for the most part)
  – E.g., FLORA-2
  – Most of these are in SWSL-Rules
Feature Laundry List

Base: Datalog+NAF

Logical Updates

Approximate reasoning paraconsistency

Constraints

Declarative metaprogramming

Constraint programming

Frames
Adding Frames

• *F-logic* is a popular way to combine frames with rules (and, more generally, FOL)

• Several implementations:
  – FLORA-2
  – FLORID
  – Ontobroker (commercial)
  – TRIPLE (partial)

• A basis for
  – WSMO-Rules
  – SWSL-Rules
Meta-Programming

• Need second order syntax, but not semantics
• One simple solution that goes a long way: *HiLog* (has been confirmed by its rediscovery in the form of SKIF)
• Supports cleanly and tractably not only second-order syntax, but also *reification*
Logical Updates

• Prolog’s assert/retract are not logical – hard to write programs correctly

• A good solution is *Transaction Logic*:
  – Logical updates
  – Attached procedures
  – Triggers
  – Supports a variety of tasks, including planning
Approximate Reasoning

• *Annotated logic*
  – Supports:
    • Paraconsistency
    • Easy to use
    • Naturally combines with rules
  – Problem:
    • Where are all the confidence factors coming from?
Constraints

• Constraints and constraint logic programming are not new; most Prolog systems support them
Web-izing

• URIs – a matter of syntax

• Modules
  – Need labels to attach to logical theories, not just names of predicates/objects
  – An extensible integration mechanism with other theories (e.g., DL) and languages (procedural and rule-based)
  – Seems severely under-appreciated by the Web/Rules community
    • Only FLORA-2 and TRIPLE (and now WSML) got it right
Additional Niceties

• Courteous rules
  – Prioritization
  – Classical negation

• Lloyd-Topor
  – More natural expression of universal statements
Discussion