SWSL Rules

Sketch of the proposed language

Benjamin Grosof & Michael Kifer

Presentation for Joint Committee 7/27/04

Main Features

- Logic Program based rule language
  - First order (not higher order)
- Expressiveness merges the KR features of:
  - Situated Courteous Logic Programs (SCLP)
  - HiLog and F-Logic LP
- Plus more Webizing, a few other refinements
- Designed to meet requirements of Semantic Web Services
- Initially a presentation syntax
- Plan to use (extended version of) RuleML as markup syntax

KR features

- Horn LP
- + Negation As Failure
- + Courteous Prioritization (mutex's, limited classical/strong negation, rule labels)
- + Situated Sensing, built-ins
- + Higher order syntactic sugar cf. HiLog
- + Frame syntax cf. F-Logic
- + Lloyd-Topor syntactic sugar
- + Reification (limited)
- + integrity constraints for type checking [somewhat experimental feature]

- URI's, namespaces
- Include'ing mechanism for importing of knowledge bases (e.g., rules or ontologies)

Semantic Web Services Tasks

- Targeted towards especially the "Profile" (non-executable aspects of) service descriptions
  - Discovery, advertising, matchmaking
  - Contracts, negotiation
  - Policies, e.g. in contracts or for trust (security, privacy, authorization, access control), and their Monitoring
  - Mapping of information, e.g., between output of one service to input of another, mediation between different ontologies

- Bridge to "Process Model" (executable aspects of) service descriptions which are modeled in FOL.
  - 90% overlap in syntactic constructs between LP and FOL
    - NAF, implication-connective differ

Relationship to OWL

- Includes Description Logic Programs plus some extensions

  - E.g., OWL Lite$^-$ (extension of DLP)
    - See the paper by de Bruijn, Polleres, Fensel http://www.wsmo.org/2004/d20/v0.2/20040630/
    - OWL Lite$^-$ includes the most significant features of OWL Lite

- Includes some of the features of OWL Full
  - Classes as instances

- The exact intersection of SWSL Rules and OWL is not known at this time

Futher Features being considered

- Considering Side-effectful Procedural attachments
  - Situated effectors
  - State changing operators (a la SQL bulk updates)
  - Triggers
  - Based on Transaction Logic (and possibly some non-logical)