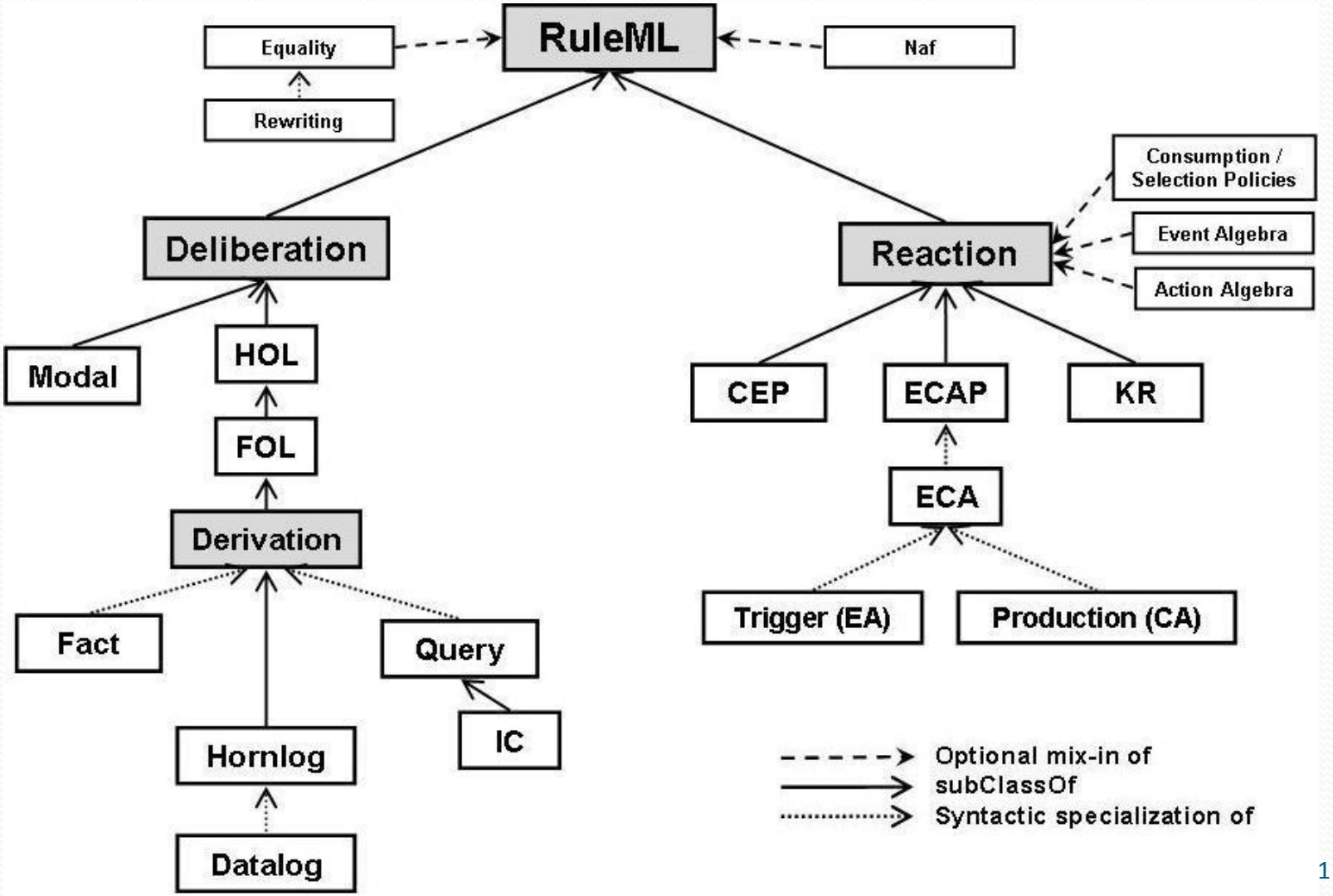
Deliberation RuleML 1.01 Specification and MYNG 1.01 Technology

Harold Boley, Tara Athan

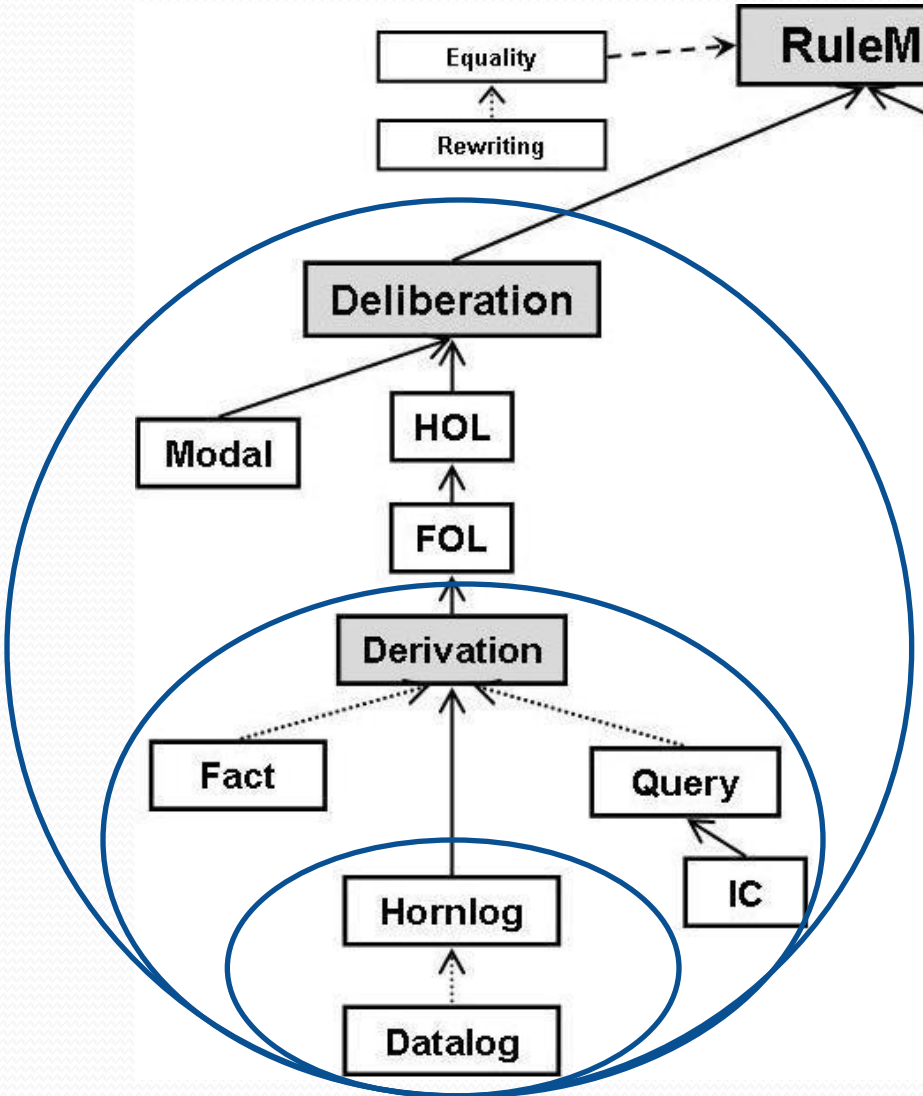
The 8th International Web Rule Symposium (RuleML 2014)

August 18-20, 2014, Prague, Czech Republic

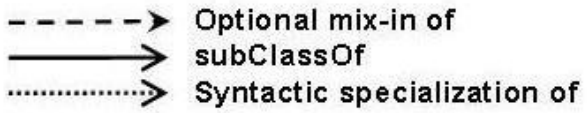
Point of Departure: RuleML 1.0 Hierarchy



Point of Departure: RuleML 1.0 Hierarchy



See: Rules, Events and Actions Keynote as well as Reaction RuleML 1.0 Demo



Finer Delib RuleML Modularization: Schema Spec

- **Deliberation RuleML 1.0:** Language lattice retains unnecessary 'hierarchical restrictions' to modularity
  - Example: Only Hornlog and 'up', not Datalog, allow **<Or>** in the **<then>** parts, so **Disjunctive *Data*log rulebases cannot be validated precisely** (only be 'underspecified' as Disjunctive *Horn*log)
- **Deliberation RuleML 1.01:** Language lattice has less hierarchical, finer-grained modularity
  - Example: allows **<Or>** in all **<then>** parts, e.g. for **precise validation of Disjunctive Datalog**

Finer Delib RuleML Modularization: Rule Instance

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-model href="http://deliberation.ruleml.org/1.01/relaxng/disdatalog_min_relaxed.rnc"?>
<RuleML xmlns="http://ruleml.org/spec">
 <Assert>
  <Forall>
   <Var>x</Var>
   <Implies>
    <if>
     <Atom>
      <Rel>integer</Rel>
      <Var>x</Var>
     </Atom>
    </if>
    <then>
     <Or>
      <Atom>
       <Rel>even</Rel>
       <Var>x</Var>
      </Atom>
      <Atom>
       <Rel>odd</Rel>
       <Var>x</Var>
      </Atom>
     </Or>
    </then>
   </Implies>
  </Forall>
 </Assert>
</RuleML>
```

disdatalog_min_relaxed.rnc

```
<then>
 <Or>
  <Atom>
   <Rel>even</Rel>
   <Var>x</Var>
  </Atom>
  <Atom>
   <Rel>odd</Rel>
   <Var>x</Var>
  </Atom>
 </Or>
</then>
```

Deliberation RuleML 1.01 Features: *Data*log Extensions Yield *Data*log$^+$

- **Existential Rules**, where the <then> part of a rule has existentially quantified variables, as needed for DL (e.g. OWL), F-logic, PSOA RuleML, Rule-Based Data Access (RBDA), etc.
- **Equality Rules**, where the <then> part of a rule is the <Equal> predicate, as needed for user-defined/'semantic' equality in logics with equality and functional logic programming (this was already allowed in RuleML 1.0)
- **Integrity Rules**, where the <then> part of a rule is falsity, as a convenient way to express negative integrity constraints

Delib RuleML 1.01 Orthogonality: *Horn*log Extensions Yield *Horn*log$^+$

- Because of modular schema design, all new features of Delib RuleML 1.01 **freely combinable**, via module inclusion,
  - with each other
  - with existing RuleML sublanguages
- Features available for other logics in Delib RuleML, including Horn logic (Hornlog RuleML 1.01), e.g. for **Hornlog$^+$** combo of
  - *Hornlog Existential Rules*
  - *Hornlog Equality Rules*
  - *Hornlog Integrity Rules*

Configure Your Own RuleML Language from Over 6 Billion via 2 MYNG Pages

- Delib RuleML 1.01 schemas customized by MYNG
- Key new **MYNG 1.01 technology** includes
  - Integration of new Relax NG schema modules – and RuleML sublanguages they define – into MYNG, e.g.
    - Datalog^{+}, Hornlog^{+}, and their many extensions
  - Improved functionality of the MYNG GUI and REST interface, e.g.
    - GUI access to automatically generated monolithic XSD schemas that are compatible with XML tools, e.g. JAXB
    - Myng-code display and URL access
- See: MYNG 1.01 Challenge Demo

RuleML Community: Getting Involved

- Talk to RuleML colleagues during RuleML 2014
- Tutorial introduction (http://ruleml.org/papers/Primer)
- RuleML MediaWiki (http://wiki.ruleml.org)
  - Includes the Wiki Issue system for users to request enhancements and report errata
- RuleML Blog & Social Mediazine (http://blog.ruleml.org)
  - Enabled Public Review of Deliberation RuleML 1.01
  - Has become a resource in its own right (courtesy to Binarypark)
- Mailing lists (http://wiki.ruleml.org/index.php/Mailing_Lists)
- Technical Groups (http://wiki.ruleml.org/index.php/Technical_Groups)
- RuleML sources hosted on Github (https://github.com/RuleML)

Follow us
- on Twitter
- on Facebook
- on LinkedIn

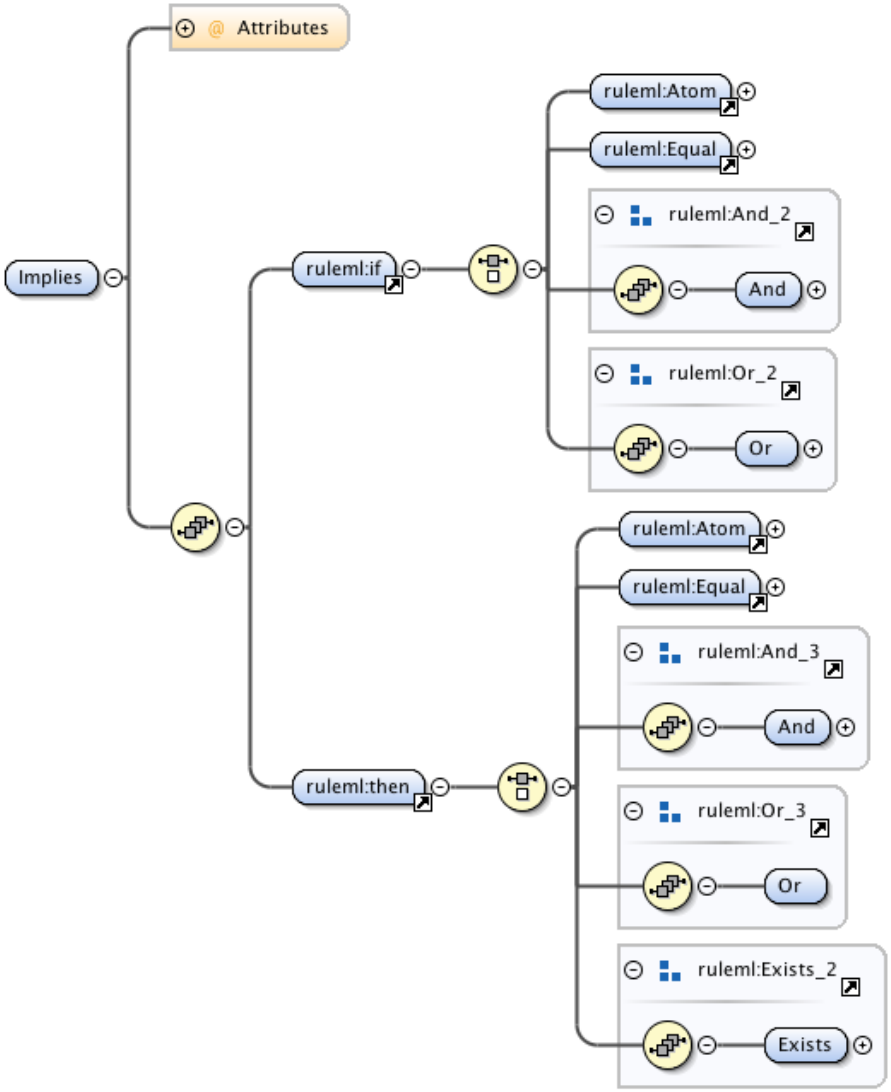
New RuleML Use Cases

- Deliberation RuleML 1.0
  - See: Geosocial SPLIS Talk
- Deliberation RuleML 1.01
  - Business Scenario Rules
  - See: Rulebase Competition 2014 Demos
    - Geospatial Rules RCC
    - Offshore Holding Analytics
    - UServ Product Derby Case Study
- Simplified RuleML 1.01 presentation syntax
  - See: RBDA/ΔForest Talk

RuleML Version Roadmap

- RuleML 1.0
  - Specification of Reaction RuleML 1.0 is released
- RuleML 1.01
  - Specification of Deliberation RuleML 1.01 is released
- RuleML 1.02
  - Deliberation RuleML 1.02 is in preparation (Timeline), with focus on improving existing features
- RuleML 1.03 is being designed, with focus on adding new features

Delib RuleML 1.01 Release on the Blog

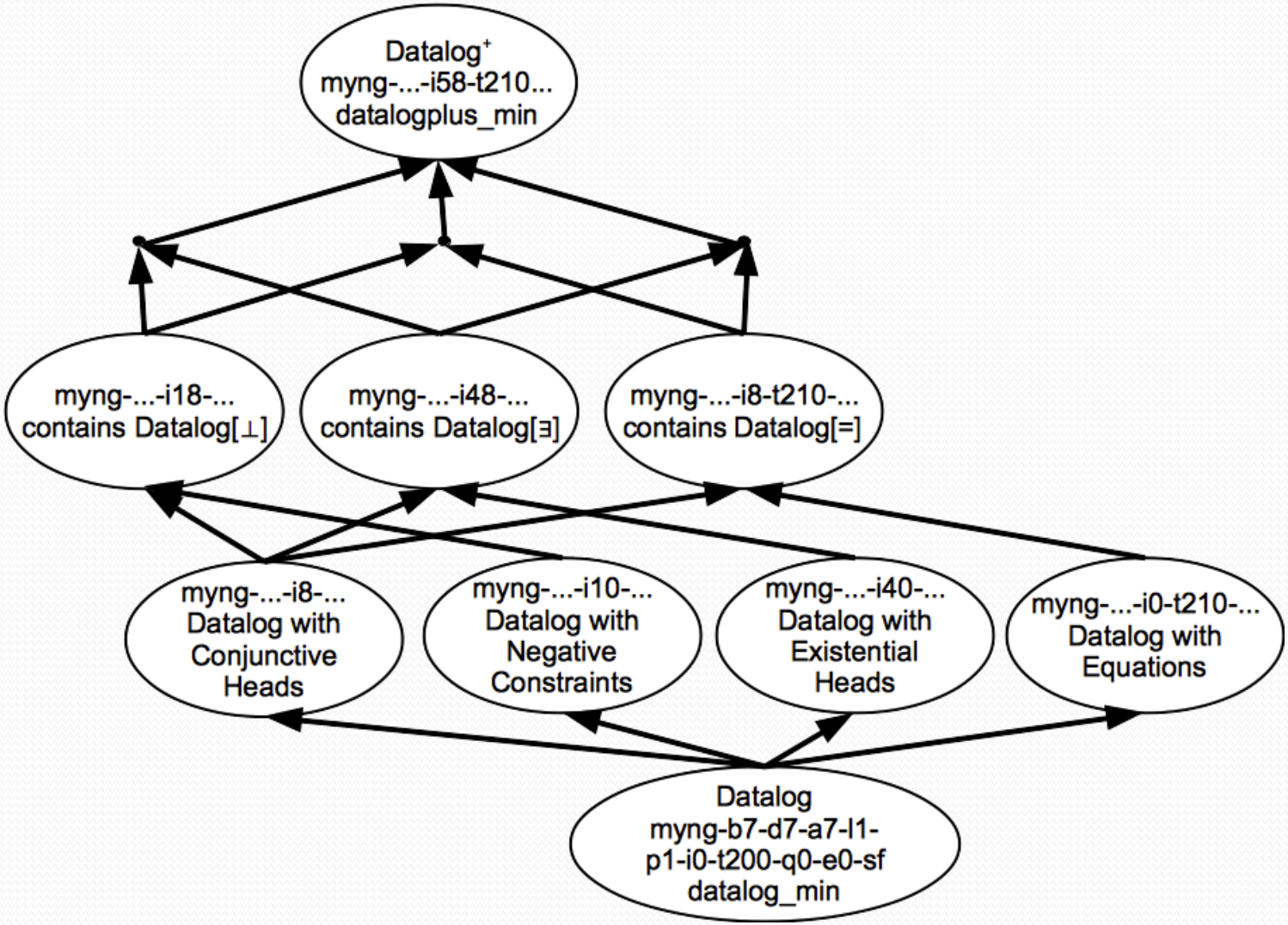


The specification of Deliberation RuleML 1.01 has been released after the Public Review and after the Steering Committee Response has addressed all comments. The full, fine-grained systematics of Delib RuleML 1.01 language features can be customized via MYNG 1.01. You can now start building Delib RuleML 1.01 Rulebases of your own, e.g. modeled on the instructive example or the entries of the Rulebase Competition 2014.

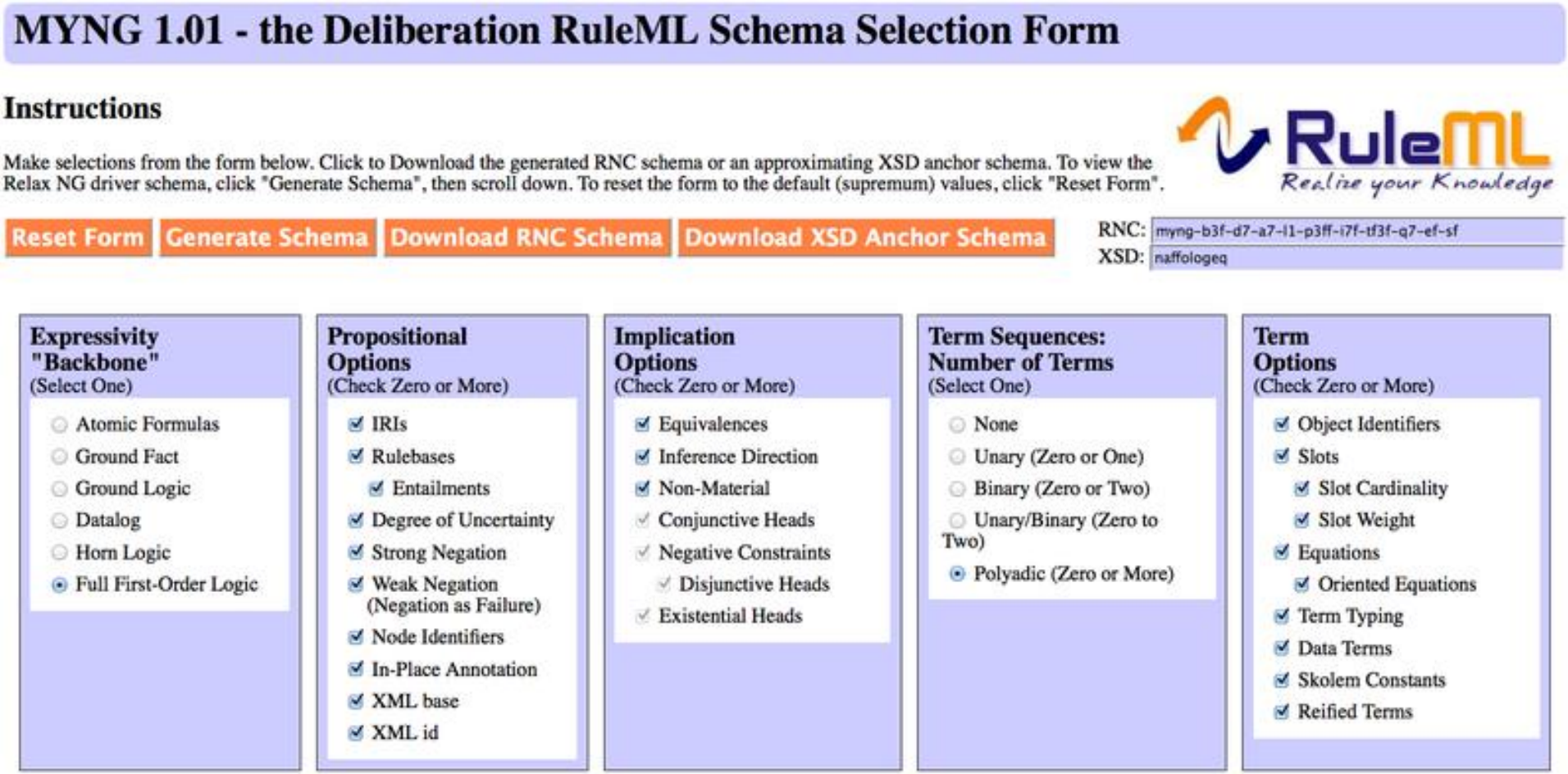
(http://blog.ruleml.org/post/2a4779af-297e-4455-b7bf-af371576a884)

Backup Slides

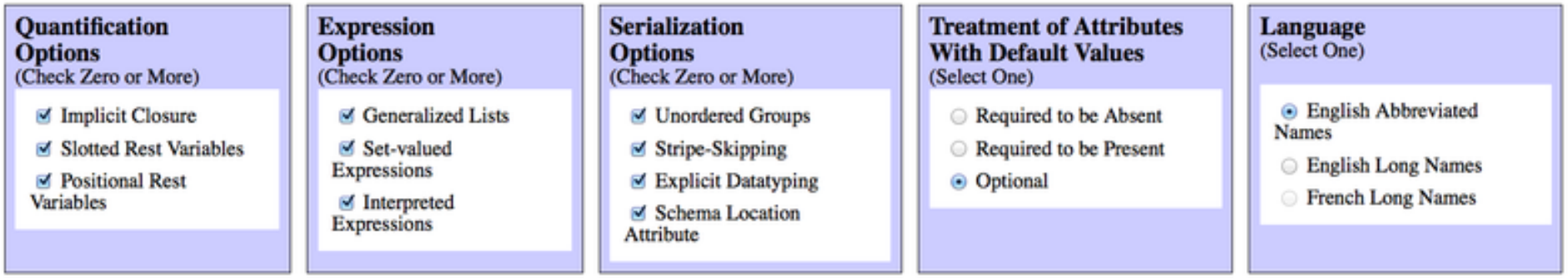
Preview: MYNG 1.01 Challenge Demo



Delib RuleML 1.01 Sublanguages Customized by MYNG 1.01 as Relax NG Schemas (1)



Delib RuleML 1.01 Sublanguages Customized by MYNG 1.01 as Relax NG Schemas (2)



Relax NG Schema URL = http://deliberation.ruleml.org/1.01/relaxng/schema_rnc.php?backbone=x3f&default=x7&termseq=x7&lng=x1&propo=x3ff&implies=x7f&terms=xf3f&quant=x7&expr=xf&serial=xf
XSD Anchor Schema URL = http://deliberation.ruleml.org/1.01/xsd/naffologeq.xsd

Usage

The RNC and XSD Schema URLs may be used directly for online validation - copy and paste as required by the validator. For a demonstration of RNC validation using the online service Validator.nu, see How to Validate with the RuleML Parameterized Relax NG Schema. Some scripts and processing instructions may require that the character "&" be replaced by "&".