



SWRL FOL

by Peter F. Patel-Schneider

Table of contents

- SWRL
- SWRL FOL
- SWRL FOL Abstract Syntax
- SWRL FOL Semantics
- SWRL FOL XML Syntax
- SWRL FOL XML Syntax Example
- Issues



- Adds Horn-style rules to SWRL
- Model-theoretic semantics, in style of OWL DL semantics
- XML syntax, extending OWL DL XML syntax
- RDF syntax, extending OWL RDF syntax
 - But **not** a semantic extension of RDF
- Details at www.daml.org/rules/proposal



- **Adds First-Order Logic constructs to OWL DL**
 - Propositional constructs (and, or, neg, ...)
 - Typed quantifiers (forall, exists)
- **Uses OWL (and SWRL) constructs**
 - Variables are as in SWRL
 - Predicates are as in OWL (plus SWRL built-ins)
- **Components**
 - Abstract Syntax
 - Model-theoretic Semantics
 - XML Syntax
- **Details at www.daml.org/2004/11/fol/proposal**



foformula ::= atom

| 'and(' { foformula } ')'

| 'or(' { foformula } ')'

| 'neg(' foformula ')'

| 'implies(' foformula foformula ')'

| 'equivalent(' foformula foformula ')'

| 'forall(' variable { variable } foformula ')'

| 'exists(' variable { variable } foformula ')'

- E.g., exists(l-variable(x ex:Person) or (ex:Student(l-variable(x)) ex:loves(l-variable(x) ex:Susan)))
- E.g., forall(l-variable(x ex:Person) neg (ex:Student(l-variable(x))))



- **Standard model-theoretic semantics**
- **Extension of SWRL model-theoretic semantics**
- **Uses binding mechanism from SWRL semantics**



SWRL FOL XML Syntax

- Extension of SWRL XML Syntax
- Tries to be close to FOL Rule ML syntax
 - Treatment of variables, in particular



SWRL FOL XML Syntax Example

```
<Assertion owl:name="Example">
  <owlx:Annotation>
    <owlx:Label>Example Rule for Expository Purposes</owlx:Label>
  </owlx:Annotation>
  <Forall>
    <ruleml:Var type="Person">x1</ruleml:Var>
    <ruleml:Var type="Parent">x2</ruleml:Var>
    <ruleml:Var type="Person">x3</ruleml:Var>
    <ruleml:Var type="xsd:int">x4</ruleml:Var>
    <And>
      <swrlx:individualPropertyAtom swrlx:property="hasParent">
        <ruleml:Var>x1</ruleml:Var>
        <ruleml:Var>x2</ruleml:Var>
      </swrlx:individualPropertyAtom>
      <swrlx:individualPropertyAtom swrlx:property="hasBrother">
        <ruleml:Var>x2</ruleml:Var>
        <ruleml:Var>x3</ruleml:Var>
      </swrlx:individualPropertyAtom>
      <swrlx:datatypePropertyAtom swrlx:property="hasAge">
        <ruleml:Var>x2</ruleml:Var>
        <ruleml:Var>x4</ruleml:Var>
      </swrlx:datatypePropertyAtom>
    </And>
  </Forall>
</Assertion>
```



- Adding n-ary predicates
- Adding functions
- Syntax for negation (**neg** vs **not**)