RuleML Rules Lite

Harold Boley, NRC IIT e-Business
Said Tabet, Macgregor Corp

With Key Contributions from the
Joint Committee

DAML PI Meeting, Captiva Island,
FL, USA

17 Oct. 2003
Introduction

- Starting point of **RuleML Rules Lite** is Datalog [RuleML 0.8](#).
- Main elements: facts and rules (also queries)
- „Minimal“: Predicates restricted to unary/binary
  - Compatible with OWL-DL
  - Certain unary predicates also usable for typing rule variables (untyped ones being of type 'any')

- In the following:
  - Use case
  - Data model
  - Concrete syntax
  - Herbrand-like models
BinDataLog Rules Lite: Source Variants

Use case: Geography

Example: stretches rule

English:
„An object x stretches an object y if y has x as a part and x and y are co-flanked“

PR-Prolog:
stretches(?x,?y) :-
    has_as_part(?y,?x),
    co-flanked(?x,?y).

RuleML Rules Lite
Rules and Facts from PR-Prolog Source

\texttt{stretches(?x,?y) :- has\_as\_part(?y,?x), co\_flanked(?x,?y).
}

\texttt{co\_flanked(?x,?y) :- flanked\_by(?x,?z), flanked\_by(?y,?z).
}

\texttt{continent(North\_America).
country(Canada).
country(USA).
province(Ontario).
oceanpair(Oceanics).
ocean(Atlantic).
ocean(Pacific).

has\_as\_part(North\_America,Canada).
has\_as\_part(North\_America,USA).
has\_as\_part(Canada,Ontario).

area(Canada,sqkm9976140).
area(USA,sqkm9629091).

flanked\_by(North\_America,Oceanics).
flanked\_by(Canada,Oceanics).
flanked\_by(USA,Oceanics).

eastern(Oceanics,Atlantic).
western(Oceanics,Atlantic).}
The Data Model of RuleML Rules Lite Unifies the XML and RDF Models

XML and RDF employ different data models:

- XML based on, possibly attributed, \textit{left-to-right ordered, node-labeled trees}, reminiscent of parse (syntax) trees
- RDF based on \textit{directed, arc-labeled (unordered) graphs} with two kinds of nodes, resources and literals, the latter not allowing outgoing arcs

A unified data model is used in RuleML Rules Lite
BinDataLog Rules Lite: Tree Data Model

RuleML Lite rules are graphically modeled as trees, here illustrated for the stretches rule:

- **RDF graph** for OO roles: (arc-)labeled, unordered children
- **XML tree** for predicate arguments: unlabeled, ordered children

```
stretches(?x,?y) :- has_as_part(?y,?x), co-flanked(?x,?y).
```

RuleML Rules Lite
RuleML Rules Lite
Concrete Markup Syntax

Webized-only, abridged version of the XML markup of RuleML 0.8, compatible with RDF and OWL

The (always empty) 'ind' element is used – in the first or second argument position – for 'href'-naming individual constants

A new 'data' element is used – in the second argument position – for tagging various data types as content (cf. RDF literals)
Webizing Constants and Relations

- Webized individual constants:
  
  `<ind>constant</ind>` is replaced by
  `<ind href="#constant"/>

  `<ind href="uri #constant"/>` as in RuleML 0.8

- Webized relation symbols:

  `<_opr><rel>relation</rel></_opr>` is replaced by
  `<_opr href="#relation"/>

  `<_opr href="uri #relation"/>`
Uniform Webizing Syntax

Individuals as well as unary and binary predicates can be webized by augmenting, within the href's, their symbolic "#"-names by URIs, so that two possibilities result:

*1* Symbolic names: with "#"-prefixed fragment identifiers (local only, corresponding to the pre-webized situation)

*2* URIs: with "#"-separated fragment identifiers (cf. RDF)
Webizing – Use Case (I)

**PR-Prolog (with N3-like URIs and namespace declarations):**

```prolog

continent(<http://www.north_america.org#North_America>).
...
<http://www.pairs.org/ocean#oceanpair>(<w#Oceanics>).
...
<http://www.mereology.org#has_as_part>(North_America,Canada).
...
<http://www.direction.org#western>(<w#Oceanics>,Pacific).
```
Webizing – Use Case (II)

XML:

```xml
<rulebase xmlns:w="http://www.waters.org">
  <facto>
    <_opr href="#continent"/>
    <ind href="http://www.north_america.org#North_America"/>
  </facto>

  ... 

  <facto>
    <_opr href="http://www.pairs.org/ocean#oceanpair"/>
    <ind href="w#Oceanics"/>
  </facto>

  ... 

  <facto>
    <_opr href="http://www.mereology.org#has_as_part"/>
    <ind href="#North_America"/>
    <ind href="#Canada"/>
  </facto>

  ... 

  <facto>
    <_opr href="http://www.direction.org#western"/>
    <ind href="w#Oceanics"/>
    <ind href="#Pacific"/>
  </facto>
</rulebase>
```
Abridging RuleML 0.8 Tags in Rules Lite

*1* The earlier `<_opr><rel href="#relation"/>` will be abridged to `<_opr href="#relation"/>`

*2* The earlier `<_body><and>atom1...atomK</and></_body>` will be abridged to `<_body>atom1...atomK</_body>` (K = 0, 1, ...)

*3* Similarly, `<_head><and>atom1...atomK</and></_head>` could be abridged to `<_head>atom1...atomK</_head>` (for OWL Rules)

*4* Prefix form `<atom>` will be allowed (0 < N < 3);
   Postfix form `<atom>term1…termN</atom>` will be forbidden (0 < N < 3)

*5* The earlier `<fact>` will be further abridged to `<facto>`
BinDataLog RuleML Lite: Use Case Rule

A rule in RuleML Lite is marked up as an `imp` element that has two children: a `_head` role with an `atom` element augmented – before or after – by a `_body` role with 1 or more (here: 2) `atom` elements

```
<imp>
  <_head>
    <atom>
      <_opr href="#stretches"/>
      <var>x</var>
      <var>y</var>
    </atom>
  </_head>
  <_body>
    <atom>
      <_opr href="#has_as_part"/>
      <var>y</var>
      <var>x</var>
    </atom>
    <atom>
      <_opr href="#co-flanked"/>
      <var>x</var>
      <var>y</var>
    </atom>
  </_body>
</imp>
```

`stretches(?x,?y) :- has_as_part(?y,?x), co-flanked(?x,?y).`

RuleML Rules Lite
BinDataLog RuleML Lite: Use Case Facts

A fact in RuleML Lite is marked up as a `facto` element that has two or three children:

- an `_opr` role with a relation name followed by
- an argument that is an `ind` element and
- an optional argument that is an `ind` or `data` element

```xml
<rulebase>
  . . .
  <facto>
    <_opr href="#country"/>
    <ind href="#Canada"/>
  </facto>
  . . .
  <facto>
    <_opr href="#has_as_part"/>
    <ind href="#North_America"/>
    <ind href="#Canada"/>
  </facto>
  <facto>
    <_opr href="#area"/>
    <ind href="#Canada"/>
    <data>sqkm 9976140</data>
  </facto>
  . . .
</rulebase>
```
Conjunctive Retrieval Queries – as 'Negative' Integrity Constraints

PR-Prolog:
continent(x), ocean(x)

XML:
<query>
  <_body>
    <atom>
      <_opr href="#continent"/>
      <var>x</var>
    </atom>
    <atom>
      <_opr href="#ocean"/>
      <var>x</var>
    </atom>
  </_body>
</query>

A query in RuleML Lite is marked up as a query element that has one child: a _body role with 1 or more (here: 2) atom elements

RuleML Rules Lite
Inferential Queries – Step 1

**PR-Prolog:**

```
stretches(?x,North_America)
```

**XML:**

```
<query>
  <_body>
    <atom>
      <_opr href="#stretches"/>
      <var>x</var>
      <ind href="#North_America"/>
    </atom>
  </_body>
</query>
```
Inferential Queries
– Step 2

**PR-Prolog:**
has_as_part(North_America,?x), co-flanked(?x,North_America)

**XML:**
<query>
  <_body>
    <atom>
      <_opr href="#has_as_part"/>
      <ind href="#North_America"/>
      <var>x</var>
    </atom>
    <atom>
      <_opr href="#co-flanked"/>
      <var>x</var>
      <ind href="#North_America"/>
    </atom>
  </_body>
</query>
Inferential Queries – Step 3

PR-Prolog:
co-flanked(?x,North_America)                 where ?x = Canada

XML:
<query>
  <_body>
    <atom>
      <_opr href="#co-flanked"/>
      <var>x</var>
      <ind href="#North_America"/>
    </atom>
  </_body>
</query>

where <var>x</var> = <ind href="#Canada"/>
Inferential Queries – Step 4

PR-Prolog:
?x = Canada

XML:
<var>x</var> = <ind href="#Canada"/>
Inferential Queries
– Step 2’

PR-Prolog:
has_as_part(North_America,?x), co-flanked(?x,North_America)

XML:
<query>
  <_body>
    <atom>
      <_opr href="#has_as_part"/>
      <ind href="#North_America"/>
      <var>x</var>
    </atom>
    <atom>
      <_opr href="#co-flanked"/>
      <var>x</var>
      <ind href="#North_America"/>
    </atom>
  </_body>
</query>
Inferential Queries – Step 3’

**PR-Prolog:**
co-flanked(?x,North_America)  
where ?x = USA

**XML:**
<query>
  <_body>
    <atom>
      <_opr href="#co-flanked"/>
      <var>x</var>
      <ind href="#North_America"/>
    </atom>
  </_body>
</query>

where <var>x</var> = <ind href="#USA"/>
Inferential Queries – Step 4’

*PR-Prolog:*

```
?x = USA
```

*XML:*

```
<var>x</var> = <ind href="#USA"/>
```
DL-Augmented Herbrand Model

**PR-Prolog:**

continent(North_America).
country(Canada).
country(USA).
province(Ontario).
oceanpair(Oceanics).
ocean(Atlantic).
ocean(Pacific).

has_as_part(North_America,Canada).
has_as_part(Canada,Ontario).
has_as_part(North_America,Ontario).  % from OWL

area(Canada,sqkm9976140).
area(USA,sqkm9629091).

flanks(Oceanics,North_America).  % from OWL
flanks(Oceanics,Canada).  % from OWL
flanks(Oceanics,USA).  % from OWL

co-flanked(North_America,Canada).
co-flanked(Canada,North_America).
co-flanked(North_America,USA).
co-flanked(USA,North_America).
co-flanked(Canada,USA).
co-flanked(USA,Canada).
co-flanked(North_America,North_America).
co-flanked(Canada,Canada).
co-flanked(USA,USA).

stretches(Canada,North_America).
stretches(USA,North_America).

eastern(Oceanics,Atlantic).
westerne(Oceanics,Pacific).

flanked_by(North_America,Oceanics).
flanked_by(Canada,Oceanics).
flanked_by(USA,Oceanics).  % from OWL

RuleML Rules Lite
DTD for Rules Lite – Facts and Rules

<!ELEMENT rulebase ((imp | facto)*)>

<!ELEMENT imp ((_head, _body) | (_body, _head))>

<!ELEMENT facto (_opr, ind, (ind | data)?)>

<!ELEMENT _head (atom)>

<!ELEMENT _body (atom*)>

<!ELEMENT atom (_opr, (ind | var), (ind | data | var)?)>

<!ELEMENT _opr EMPTY>

<!ELEMENT ind EMPTY>

<!ELEMENT data (#PCDATA)>

<!ELEMENT var (#PCDATA)>

<!ATTLIST _opr href CDATA #REQUIRED>

<!ATTLIST ind href CDATA #REQUIRED>

<!!-- The CDATA must be either
   URIs with "#"-separated fragment identifiers (for URI naming) or
   local, "#"-prefixed fragment identifiers (for symbolic naming) -->
   RuleML Rules Lite
XML Schema (I) – Facts and Rules
XML Schema (II) – Facts and Rules
Work in Progress on Industrial Applications

- Fraud detection rules
- Mortgage loan rules
- Policies and regulations for financial services
Conclusions

• After definition of RuleML Rules Lite as an XML DTD & Schema, next step has been to modify existing parsers, generators, and XSLT translators to implement syntax for rule interchange

• Two engines running tested with discount rules (Jess, Rules.NET)

• Serialization to Java and .NET object models

• Further implementations have been envisaged for jDREW, OWL engines, Jena 2, and cwm

• Since RuleML Rules Lite semantically is a kind of RDF Rules, a pure RDF version is being considered

• Intertranslation and convergence with OWL Rules (Lite) will be important future JC work item to synergize communities

• To demonstrate the extension path to N-ary relations, ternary predicates could be added later as exemplified in http://www.daml.org/listarchive/joint-committee/1444.html

• As an alternative or complement, transition to full Horn logic could be attempted