W3C rule interchange format

Production Rule Dialect
and relation to Core
Rule interchange

Rules

Rule system 1

Data

serialize

Application A

<XML doc>

Data model (OWL, RDF-S, XML-S, XMI, ...)

de-serial.

Rule system 2

Data

Application B

serialize

<XML doc>

<RI< doc>

de-serial.
What is the Rule Interchange Format?

- Format for interchanging rules, so they can be used across diverse systems
  - allowing rules written for one application to be published, shared, and re-used in other applications and other rule engines.
  - In a semantic preserving way (between languages with compatible semantics)
  - Encouraging interoperability
  - XML syntax
  - Compatible with relevant standards (PRR, RDF, OWL, …)

- A rule is (just another) data item
  - RIF provides a standard means to feed rules into an application (at run time)
  - Semantics to prescribe (intended) application’s behaviour
RIF Background: standards

• The early days of rule interchange
  – 1998: KIF – Knowledge Interchange Format
  – 2000: RuleML
  – 2001: SRML – Simple Rule Markup Language (Colleen and Changhai)
  – …

• 2001-2004: JSR 94 – Java rule engine API
  – Prescribes a set of fundamental rule engine operations (i.e. loading rulesets, parsing rules, adding objects to an engine, firing rules, and getting resultant objects from the engine)
  – Engine semantics are not defined, so the API and specification are very high level (JDBC API ... without SQL)
  – No underlying rule language, hence no API to introspect rules, create rulesets, provide pluggable parsers etc.

• 2003-2008: OMG PRR – Production Rule Representation
  – “A metamodel for a language that can be used with UML models for explicitly representing production rules as visible, separate and primary model elements in UML models”
    • A MOF/UML meta-model and an UML profile
    • Addresses the PIM level of MDA
  – Engine semantics are defined (forward chaining + sequential)
  – No underlying rule language: PRR Core + non-normative PRR OCL

• April 2005: W3C workshop on rule languages for interoperability
• November 2005: W3C Rule Interchange Format working group chartered
• OMG SBVR, ISO Common Logic, …
RIF Background: semantic Web

• Semantic web approach
  – interoperability requires a formal semantics

• The OWL WG approach
  – Start with something (DAML+OIL)

• Literally *hundreds* of rule system implementations
  – ISO-Prolog, CLIPS, OPS…
  – Already several “SW” rule languages
    • SWRL, RuleML, WRL, SWSL, KAON2, TRIPLE, JenaRules…
  – Everyone wants “their” system/technique/theory to be the SW rule standard
Wherefore the RIFt(s)?

• **OWL DL < OWL < FOL**
  – Original idea to add full first-order at the logic level
  – Semantic web very “open world”
• **Most back-end DBs support closed queries**
• **Many rule systems have non-FO features**
  – CWA/NAF
  – Procedural Attachment
  – Rule ordering
  – Non-monotonicity
  … can’t be layered on OWL
• **Not a strict SW layering already**
  – OWL restricted dialects (DL) not layered on (all of) RDF/S semantics
  – RDF & RDFS not layered at all
RIF Background: Business rules

• “Business Rule systems” Vendors
  – $1B/year existing market
  – 1,000’s end users
  – 1,000,000’s rules in use
  – ILOG, Fair Isaac, PegaSystem, Tibco, Corticon, Haley, …

• Database vendors
  – Oracle, IBM

• OMG PRR effort
  – Simple production rules
  – Event-condition-action
  – Vendors understand the value of standardization (see also JSR 94)
  – Interchange already a priority
  – …a common semantics?
W3C RIF working group

- W3C working group
  - Chartered Nov. 05 (for 2 years)
    - Phase 1: extensible XML rule interchange format, Horn expressiveness, semantic Web compatibility
  - Kick-off Dec. 05
    - March 06: FPWD UCR
    - March 07: FPWD Core
    - Oct. 07: FPWD BLD
  - Extended Nov. 07 (for 6 months)
    - BLD to Last Call
  - Extended May 08 (for 1 year)
    - BLD, SWC to REC
    - FLD, DTB, PRD, extensibility
- 74 participants from 35 organisations
  - IBM, HP, Oracle, ILOG, JBoss, Fair Isaac, Corticon, Tibco, MITRE…
  - NIST, OMG (esp. SBVR and PRR), RuleML…
  - Research organisations, universities…
  - And 4 invited experts
  - Chairs: Chris Welty (IBM), Christian de Sainte Marie (ILOG)
- Working in the public eye
  - Under the W3C patent policy
W3C RIF: Design issues

• Very large number of rule users/use cases and types to satisfy!!!
  – Descriptive {OMG MDA level = CIM} VS executable rules {OMG MDA level = PIM & PSM}
  – Logical (side-effect free) VS active (side-effect full) rules
  – Data-oriented (SQL triggers, PR, …) VS proof-oriented (FOL…)
    • All kinds of different data sources (DB, WM, OO, OWL…)
  – Semantic Web VS non-SW usage

• Simplicity VS coverage
• Extensibility VS compliance VS interoperability
• Executable (AST) VS human-readable syntax
• …
Superset approach
Super-set approach

• Define a super-language so expressive that any language can be translated to/from it
  – The CL and IKL approach

@deprecated: infeasible for this group, as major differences appeared irreconcilable (e.g. non-mon vs. mon)
A common core...
...and standard dialects
Core + standard dialects

- Define a core language that accounts for the intersection of all rule language capabilities
  - E.g. Horn, datalog, …

@postponed: The production rule, logic programming, and FO core is not clear
Separate families + Core

IRL

RIF PRD

Core

RIF BLD

JBoss

RLj

RLn
Separate families + Core

• Define a logic-based core and a separate production-rule core
• If there is an intersection, define the common core (which may possibly be just a syntax)

@version 0.1: BLD LC (July 08)
@version 0.0.2: PRD WD2 (December 08)
@version 0.0.2: Core WD2 (December 08)
Approach 3a

• Define a *framework* in the form of a menu of syntactic and semantic features that can be combined into dialects
  – @version 0.0.9: FLD
Design principles

• Translation paradigm
  – No intrusion in covered rule languages and rule sets

• Same semantics ⇔ same syntax
  – Share constructs across dialects wherever they agree on the semantics
  – Different constructs where semantics do not agree

• Fully striped XML (type-tagged, object-oriented, …)
  – alternating Class and role tags
  – Metadata can be attached to any class element

• Only XML schema is normative
  – Presentation syntax added for specification’s readability (examples, semantics etc)

• Principles are there so you can rest on them…until they break
BLD Overview

- **Definite Horn rules**
  - Disjunction of atoms with exactly one positive literal
- **Equality, functions, and a standard first-order semantics**
- **Syntactic features**
  - relations and frames
  - internationalized resource identifiers (IRIs) as identifiers
  - XML Schema data types and builtins
- **XML (1.0) syntax with normative XMLS definition**
- **Non-normative presentation syntax**
- **Metadata and (RDF+OWL) imports**
Symbols

• Used to identify constants, variables, functions, predicates

• "literal"^^<symspace-identifier>
  – Notable symbol spaces: xsd:string, rif:local, rif:iri
  – “Chris”^^<xsd:string>
  – “
    http://www.w3.org/1999/02/22-rdf-syntax-ns#ty
    rif:iri>
  – “Person1”^^rif:local
Rules

• IF <condition> THEN <conclusion>
  – <condition> aka rule body, antecedent
  – <conclusion> aka rule head, consequent

• BLD rule:
  – (Forall var* <conclusion> :- <condition>)
  – Conclusions may contain conjunction
  – Conditions may contain conjunction, disjunction, and existential

• Restrictions on conclusion
  – No existential, disjunction, external functions
Horn Extensions

• Functions and external calls (DTB)
• Equality (in conclusion and condition)
• Frames
  – Objects with slots and (multiple) values
    • Used to map to RDF and OWL (SWC)
  – Special syntactic treatment of class membership and subclass
• Named argument functions and predicates
  – However all arguments must be provided
Structure

• Rules occur in Groups

\[
\text{Group}( \ (\text{Forall } \ ?x \ _Q(?x) :- \ _P(?x)) \\
(\text{Forall } \ ?x \ _Q(?x) :- \ _R(?x)) \ )
\]

• Groups occur in Documents

\[
\text{Document}( \\
\text{Group}( (\text{Forall } \ ?x \ _Q(?x) :- \ _P(?x)) \\
(\text{Forall } \ ?x \ _Q(?x) :- \ _R(?x)))
\)
PRD Overview

• Production rules
  – FOR <variables> WITH <binding patterns>,
    IF <condition> THEN <actions>
  – FORALL Var* (IF patterns AND condition THEN action)
  – With an operational semantics as a labelled transition system

• Patterns and condition
  – BLD condition language
    minus logic functions and named argument terms
    plus negation
  – With a model-theoretic semantics (compatible with BLD)

• Assert, Retract, New
  – Defining a transition relation
  – Modify, Remove, Execute

• Syntactic features
  – Relations and frames: objects?
  – internationalized resource identifiers (IRIs) as identifiers
  – XML Schema data types and builtins

• Metadata
<Const type=xsd:anyURI [xml:lang=xsd:language]? >
  Any Unicode string
</Const>

<Var> any Unicode string </Var>

<External>
  <content>
    <Expr> <op> Const </op>
      <args rif:ordered="yes"> TERM* </args>?
    </Expr>
  </content>
</External>
Atomic formulas
Atomic formulas

<Atom>
  <op> Const </op>
  <args rif:ordered="yes"> TERM* </args>?
</Atom>

<External>
  <content> Atom </content>
</External>

<Equal>
  <left> TERM </left>
  <right> TERM </right>
</Equal>

<Member>
  <instance> TERM </instance>
  <class> TERM </class>
</Member>

<Subclass>
  <sub> TERM </sub>
  <super> TERM </super>
</Subclass>

<Frame> <object> TERM </object>
  <slot rif:ordered="yes"> TERM TERM </slot>*
</Frame>
Atomic formulas

- jim:owns(?c ?p)
  
  `<Atom>`
  `<op>`
  `<Const type="rif:iri">http://rif.examples.com/2008/jim#owns</Const>`
  `</op>`
  `<args rif:ordered="yes">`
  `<Var> ?c </Var>`
  `<Var> ?p </Var>`
  `</args>`
  `</Atom>`

- ?c[age -> ?a]
  
  `<Frame>`
  `<object> <Var> ?c </Var> </object>`
  `<slot rif:ordered="yes">`
  `<Const type="xs:string">`
    `child::age <!-- http://rif.examples.com/2008/jim#Chicken/age -->`
  `</Const>`
  `<Var> ?a </Var>`
  `</slot>`
  `</Frame>`
Condition formulas
Condition formulas

• Atomic formulas

<[And|Or|NmNot]>
  <formula> FORMULA </formula>*
</[And|Or|NmNot]>

<Exists>
  <declare> Var </declare>+
  <formula> FORMULA </formula>
</Exists>
• Exists Chicken(age>8)
  <Exists>
    <declare> <Var> ?c </Var> </declare>
    <declare> <Var> ?a </Var> </declare>
    <formula>
      <And>
        <formula> <Member> ?c # jim:Chicken </Member> </formula>
        <formula> <Frame> ?c[age-?>a] </Frame> </formula>
        <formula>
          <External>
            <content>
              <Atom>
                <op> <Const type="rif:iri"> op:numeric-greater-than </Const> </op>
                <args rif:ordered="yes">
                  <Var> ?a </Var>
                  <Const type="xsd:decimal"> 8 </Const>
                </args>
              </Atom>
            </content>
          </External>
        </formula>
      </And>
    </formula>
  </Exists>
Semantics of PRD conditions

- A state $S$ is a Herbrand Interpretation $IH$.
- A condition formula, $\varphi$ is satisfied under variable assignment $\sigma$ in a state $S$, written as $S \models \varphi[\sigma]$, iff $TVal_S(\varphi[\sigma]) = t$
- Let $\psi$ be a condition formula, and $\varphi$ be a set of ground formulas that satisfies $\psi$. We say that $\psi$ matches $\varphi$ with substitution $\sigma : \text{Var} \rightarrow \text{Terms}$ if and only if there is a syntactic interpretation $I$ such that for all $?x_i$ in $\text{Var}(\sigma)$, $I(?x_i) = I(\sigma(?x_i))$. 
Atomic action

<Assert>
  <target>
    [ Atom | Frame | Member ]
  </target>
</Assert>

<Retract>
  <target>
    [ Atom | Frame | TERM ]
  </target>
</Retract>

Only if new object created in the same block
Semantics of atomic actions

• Assuming a facts base $W$ that contains every true facts, the intended semantics of RIF-PRD atomic actions is completely specified by the transition relation $\rightarrow^{\text{RIF-PRD}} \subseteq W \times L \times W$, where $L$ is the set of all the ground atomic actions.

• $(w, \alpha, w') \in \rightarrow^{\text{RIF-PRD}}$ if and only if $w \in W$, $w' \in W$, $\alpha$ is a ground atomic action, and one of the following is true:
  – $\alpha$ is Assert($\phi$), where $\phi$ is a ground atomic formula, and $w' = w + \phi$;
  – $\alpha$ is Retract($\phi$), where $\phi$ is a ground atomic formula, and $w' = w - \phi$;
  – $\alpha$ is Retract($o$), where $o$ is a constant, and $w' = w - \{o[s->v] \mid \text{for all the values of terms } s \text{ and } v\} - \{o#c \mid \text{for all the values of term } c\}$. 
<Do>
  <actionVar rif:ordered="yes">
    Var
    INITIALIZATION
  </actionVar>*
  <actions rif:ordered="yes">
    ATOMIC_ACTION+
  </actions>
</Do>
Action block

- \( \text{Do((?p New(?p)) Assert(?p\#joe:Potato) Assert(…))} \)

\[
\begin{align*}
\text{<Do>}
\quad & \text{<actionVar> <Var>p</Var>}
\quad & \text{<New>}
\quad & \text{<instance><Var>p</Var></instance>}
\quad & \text{</New>}
\quad & \text{</actionVar>}
\quad & \text{<actions rif:ordered="yes">}
\quad & \text{<Assert> <target> p \# joe:Potato </target> </Assert>}
\quad & \text{<Assert>}
\quad & \text{<target> p [ child::weight -> 100 </target>}
\quad & \text{</Assert>}
\quad & \text{</actions>}
\quad & \text{</Do>}
\]
<Forall>
  <declare> Var </declare>+
  <pattern>
    FORMULA
  </pattern>*
  <formula> RULE </formula>
</Forall>

<Implies>
  <if> FORMULA </if>?
  <then>
    ACTION_BLOCK
  </then>
</Implies>
RULE

• When
  ?c Chicken(age==8)
evaluate(today()="Monday")
Then …
  <Forall>
  <declare> <Var> ?c </Var> </declare>
  <pattern>
    <And>
      <Member> ?c jim:Chicken </Member>
      <Frame> ?c.age=8 </Frame>
    </And>
  </pattern>
  <formula>
    <Implies>
      <if> <External> today()="Monday" </External>
      <then> … </then>
    </Implies>
  </formula>
</Forall>
<Group>
  <behavior>
    <ConflictResolution>
      xsd:anyURI
    </ConflictResolution>?
    <Priority> -10,000 ≤ xsd:int ≤ 10,000 </Priority>?
  </behavior>?
  <sentence> [ RULE | Group ] </sentence>*
</Group>
A **RIF-PRD production rule system** is defined as a labeled terminal transition system $PRS = \{S, A, \rightarrow PRS, T\}$, where:

- $S$ is a set of system states;
- $A$ is a set of transition labels, where each transition label is a sequence of ground RIF-PRD atomic actions;
- The transition relation $\rightarrow PRS \subseteq S \times A \times S$, is defined as follows:  
  \[ \forall (s, a, s') \in S \times A \times S, (s, a, s') \in \rightarrow PRS \text{ if and only if all of the following hold:} \]
  - $(\text{facts}(s), a, \text{facts}(s')) \in \rightarrow RIF-PRD$;
  - $a = \text{actions}(\text{picked}(s))$;
- $T \subseteq S$, a set of final system states.
Semantics of a rule set (cont’d)

• Given a rule, $r \in R$ and a ground substitution, $\sigma$, such that $\text{Var}(r) \subseteq \text{Dom}(\sigma)$, where $\text{Var}(r)$ denotes the set of the rule variables in $r$, the result, $r_i = \sigma(r)$, of the substitution of the constant $\sigma(?x)$ for each variable $?x \in \text{Var}(r)$ is a rule instance (or, simply, an instance) of $r$.

• A rule instance $r_i$ matches a state of facts $w$ iff [...]  
  \begin{itemize}
  \item rule($r_i$) is Forall $?v_1\ldots?v_n$ (p_1\ldots p_n) (r'), $n \geq 0$, $m \geq 0$, and substitution($r_i$) matches each of the condition formulas $p_i$, $0 \leq i \leq m$, to the ground condition formula that represents $w$, and the rule instance $r_i'$ matches $w$, where $r_i'$ is the instance of rule $r'$ such that substitution($r_i'$) = substitution($r_i$)
  \end{itemize}

• Given a rule set, $RS \subseteq R$, and a system state, $s$, the set, $\text{conflictSet}(RS, s)$ of all the different instances of the rules in $RS$ that match the state of the fact base, $\text{facts}(s) \in W$ is called the conflict set determined by $RS$ in $s$. 
Semantics of a rule set (cont’d)

Given a conflict set, cs, the conflict resolution strategy rif:forwardChaining is the successive application of four rules, where ri and ri’ are rule instances:

• **Refraction rule**: if $ri \in cs$ and $\text{lastPicked}(ri, s) \leq \text{recency}(ri, s)$, then $cs = cs - ri$;

• **Priority rule**: if $ri \in cs$ and $ri' \in cs$ and $\text{priority}(ri) < \text{priority}(ri')$, then $cs = cs - ri$;

• **Recency rule**: if $ri \in cs$ and $ri' \in cs$ and $\text{recency}(ri, s) > \text{recency}(ri', s)$, then $cs = cs - ri$;

• **Tie-break rule**: if $ri \in cs$, then $cs = \{ri\}$.

A system state, s, is **final** given a rule set, RS if and only if the remaining conflict set is empty after application of the **refraction rule** to all the rule instances in conflictSet(RS, s)
<CLASSELT>
  <id> Const </id>?
  <meta>
    [ Frame
     |   <And>
       <formula> Frame 
        </formula>*
     </And> ]
  </meta>?
  other CLASSELT content
</CLASSELT>

<Forall>
  <id><Const type="rif:iri">jim:CMP</Const></id>
  <meta><Frame>
    <object><Const type="rif:iri">
      jim:CMP
    </Const>
    </object>
    <slot rif:ordered="yes">dc:creator csma</slot>
    <slot rif:ordered="yes">…</slot>
    …
    </Frame>
  </meta>
  <declare> <Var> ?c </Var> </declare>
  <pattern>…</pattern>
  <formula>…</formula>
</Forall>
Core overview

• Definite Horn rules without function symbols + safeness
  – Essentially safe Datalog
  – Notice PRD rules are safe

• Condition formulas like PRD
  – minus subclass atomic formula
  – minus negation (NmNot)

• Conclusion restricted to (the assertion of) (conjunction of) atoms, frames and membership formula
  – With same restriction on membership formulas as in PRD
Semantics of Core

• Standard first order semantics
  – For that subset, equivalent to Herbrand semantics
RDF and OWL compatibility

- RDF triple \( s \, p \, o \) mapped to frame \( s'[p'\rightarrow o'] \)
  - \( s'[p'\rightarrow o'] \) is true iff \( s \, p \, o \) is in the imported RDF graph
  - Condition on data types alignment
  - Simple, RDF, RDFS, D-RDF interpretation iff vocabulary included and axioms satisfied
  - Graph/formula entailed iff satisfied in every interpretation

- OWL 2 Full compatibility is straightforward extension of RDF compatibility

- OWL 2 DL requires syntactic restrictions and semantic extension of RIF frames
  - RIF frame \( o[p\rightarrow v] \) is an OWL 2 DL frame iff \( p \) is a constant and \( v \) is a constant if \( p \) belongs to an imported ontology or \( p \) is \textit{rdf:type} and \( v \) belongs to an imported ontology
  - A variable is DL-safe if it does not occur in a DL frame such that \( p \) belongs to an imported ontology or \( p \) is \textit{rdf:type} and \( v \) belongs to an imported ontology
  - Frame \( o[p\rightarrow v] \) is interpreted as relation \( p(o, v) \) if \( p \) is not \textit{rdf:type}, and as \( o \) belonging to set \( v \) if \( p \) is \textit{rdf:type}
RIF Documents

- **BLD**: RIF basic logic dialect
  - LC July 2008
  - REC by May 2009?
- **FLD**: RIF framework for logic dialects
  - 2\textsuperscript{nd} public WD July 2008
  - LC November 2008?
- **PRD**: RIF production rule dialect
  - WD2 December 2008
  - LC May 2009?
- **DTB**: RIF data types and builtins
  - WD2 December 2008
  - LC May 2008?
- **SWC**: RIF RDF and OWL compatibility
  - LC July 2008
  - REC by May 2008?
- **UCR**: RIF use cases and requirements
  - 5th public WD December 2008
- **Test Cases**: FPWD December 2008
- **rdf:text**: FPWD December 2008 (common with OWL WG)

Credits

- BLD and FLD Editors
  - Michael Kifer (U. Stonybrook), Harold Boley (NRCC)
- PRD Editors
  - Christian de Sainte Marie (ILOG), Adrian Paschke (FUBerlin), Gary Hallmark (ORACLE)
- SWC Editor
  - Jos de Bruijn (FUB)
- DTB Editors
  - Axel Polleres (DERI Galway), Michael Kifer (U. Stonybrook), Harold Boley (NRCC)
- UCR Editors
  - Adrian Paschke (TU Dresden), David Hirtle (NRCC), Allen Ginsberg (Mitre), Paula-Lavinia Patranjan (REWERSE), Frank McCabe (Fujitsu)
- Test Cases Editors
  - Stella Mitchell (IBM), Leora Morgenstern (IBM), Adrian Paschke (FUBerlin)
- Active WG members
  - Adrian Paschke (FUBerlin), Axel Polleres (DERI), Dave Reynolds (HP), Gary Hallmark (ORACLE), Hassan Aït-Kaci (ILOG), Igor Mozetic (JFI), John Hall (OMG), Jos de Bruijn (FUB), Leora Morgenstern (IBM), Mike Dean (SRI), Stella Mitchell (IBM), Changhai Ke (ILOG)
- WG Team
  - Chris Welty (IBM), Christian de Sainte Marie (ILOG), and Sandro Hawke (W3C/MIT)
Thank you!

Questions?