RULES FEST

Rule Standard 101

Christian de Sainte Marie

IBM ILOG

Business rules standards are good for you

- Standardisation: an agreement to support a common interface or representation (with cooperation of the vendors)
  - Standard semantics
  - Platform neutral persistence
  - Use across different platforms
- Users benefits
  - Avoid vendor's lock-in
    - Commoditization
  - Allows cooperative development/deployment
    - Share rules/policies/documentation
  - Improves communication
    - Improves quality, traceability, maintainability
  - Preserve investment, increases ROI
    - re-use rules across tools
    - skill sets transferable
- Vendors point of view
  - Ease integration with applications, process servers, events servers, business process modeling environments, and business process management systems
  - Extend reach
  - Increase market confidence

Business rules and standards

Modeling	
▪ Describe business rules and vocabularies in a way that is understandable by non-technical business users.	▪ OMG SBVR – Semantics of Business Vocabulary and Rules ▪ Includes a metamodel and an (optional, non-normative) English like syntax
Application Design	
▪ Describe the logic of a decision in a way that is understandable by non-technical business users ▪ Integrate the business rules as decision points within an application or another model (e.g. a business process management model)	▪ OMG DMN – Decision Model and Notation – an emerging standard ▪ Integrates business rules with BPMN 2.0 based business modeling
▪ Include the rules as 1st class components in UML models	▪ OMG PRR – Production Rules Representation ▪ Associate rules with the application classes that they reference or modify
Deployment and Operation	
▪ Rule engine enablers such as serialization (interchange) formats and deployment APIs	▪ W3C RIF – Rules Interchange Format - provide a common XML serialization ▪ JSR-94 – Java Rules Engine API – provides an API to control a rule engine

Standards in the business rules life cycle

Requirements analysis

Model business rules and business vocabularies in a verifiable and communicable way

OMG SBVR defines the vocabulary and rules for documenting the semantics of business vocabularies, business facts, and business rules

Application design

Include rules as 1st class components in UML models

OMG PRR is the UML extension that permits to associate each rule with the classes they check and the classes they impact

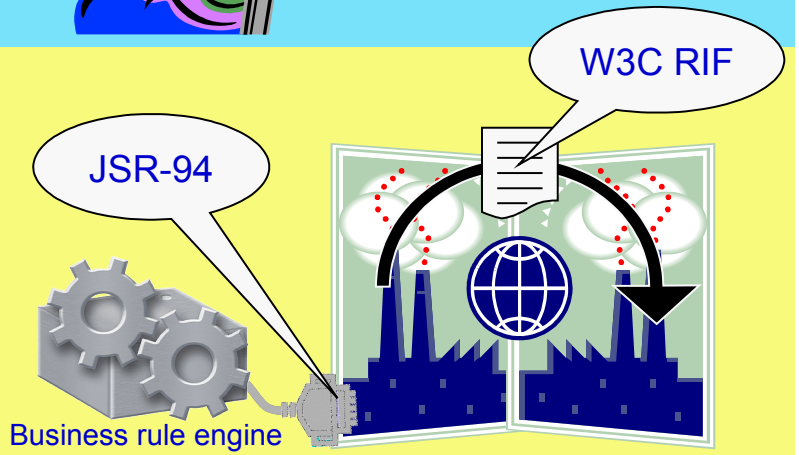
Deployment and operation

Embed rule engines in your Java applications

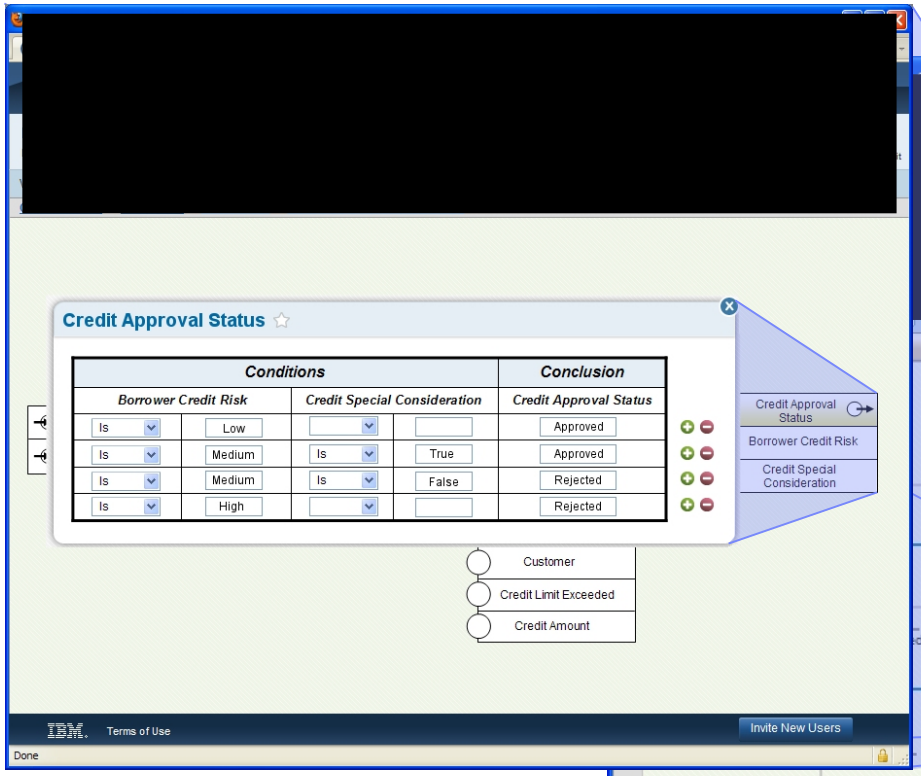
JSR-94 is the Java API for rule engines

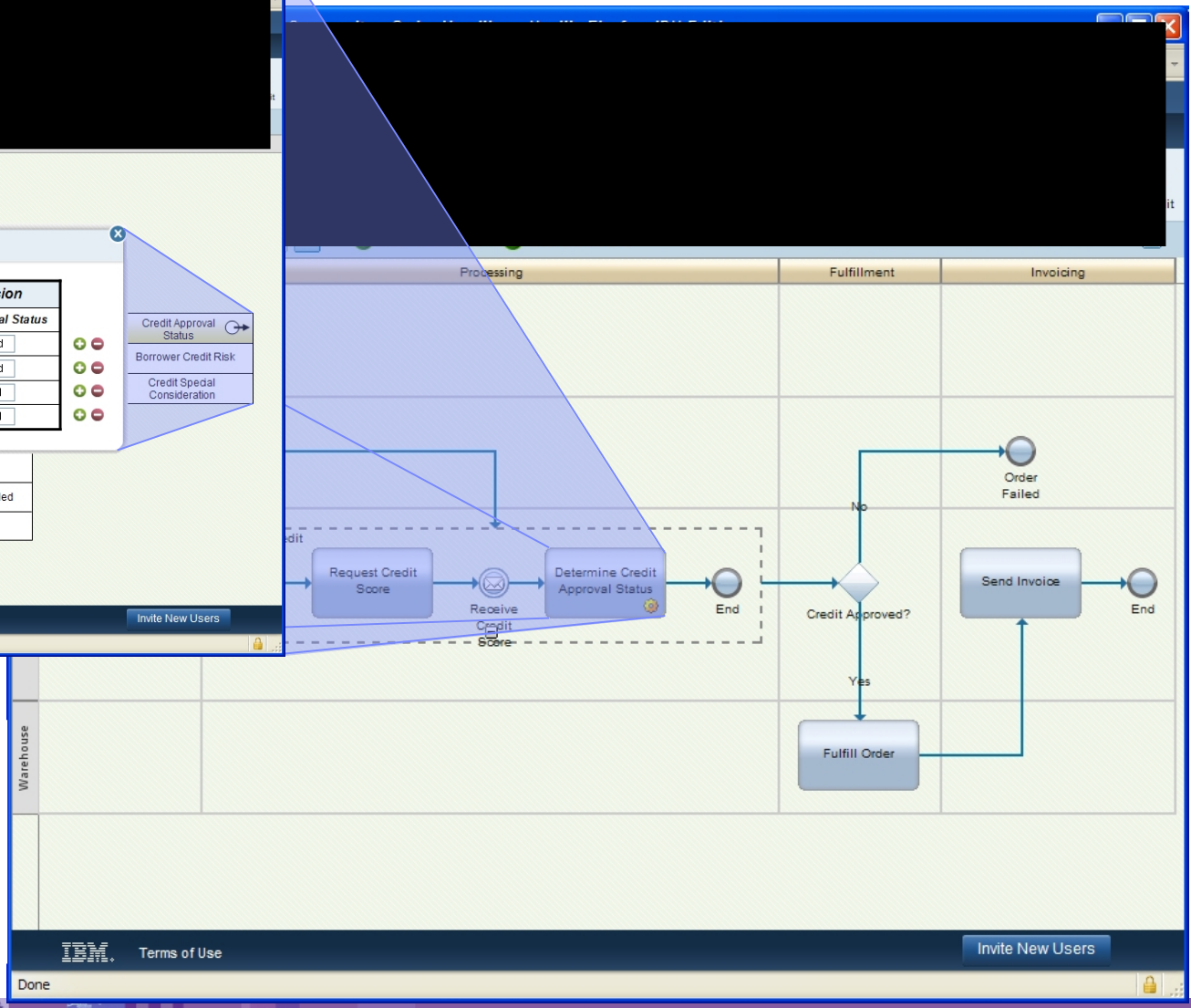
Write rules once, use them everywhere
Share rules with business partners
Publish/retrieve rules

W3C RIF is the standard XML serialization for rules



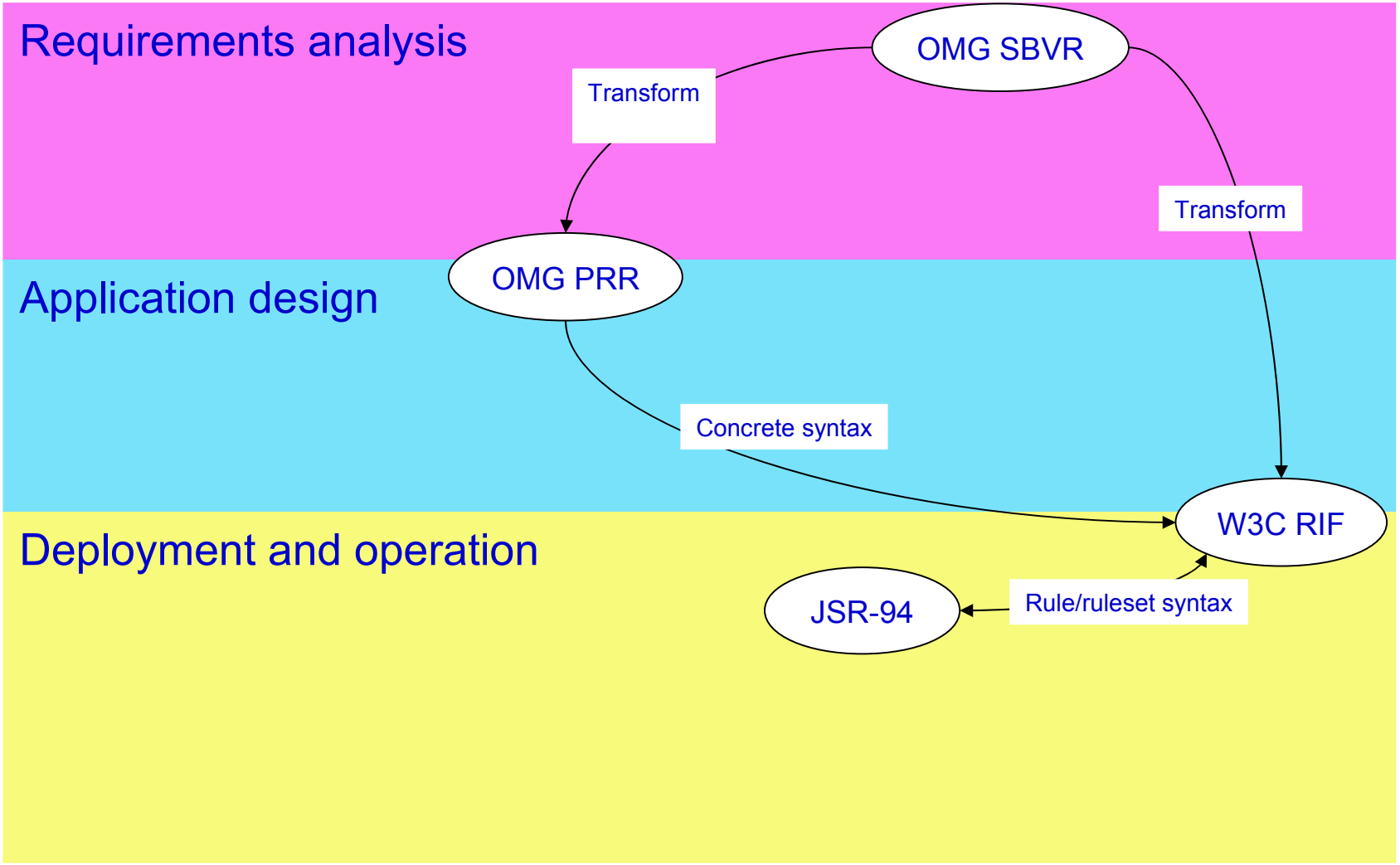
Business Rules Enhance Business Agility



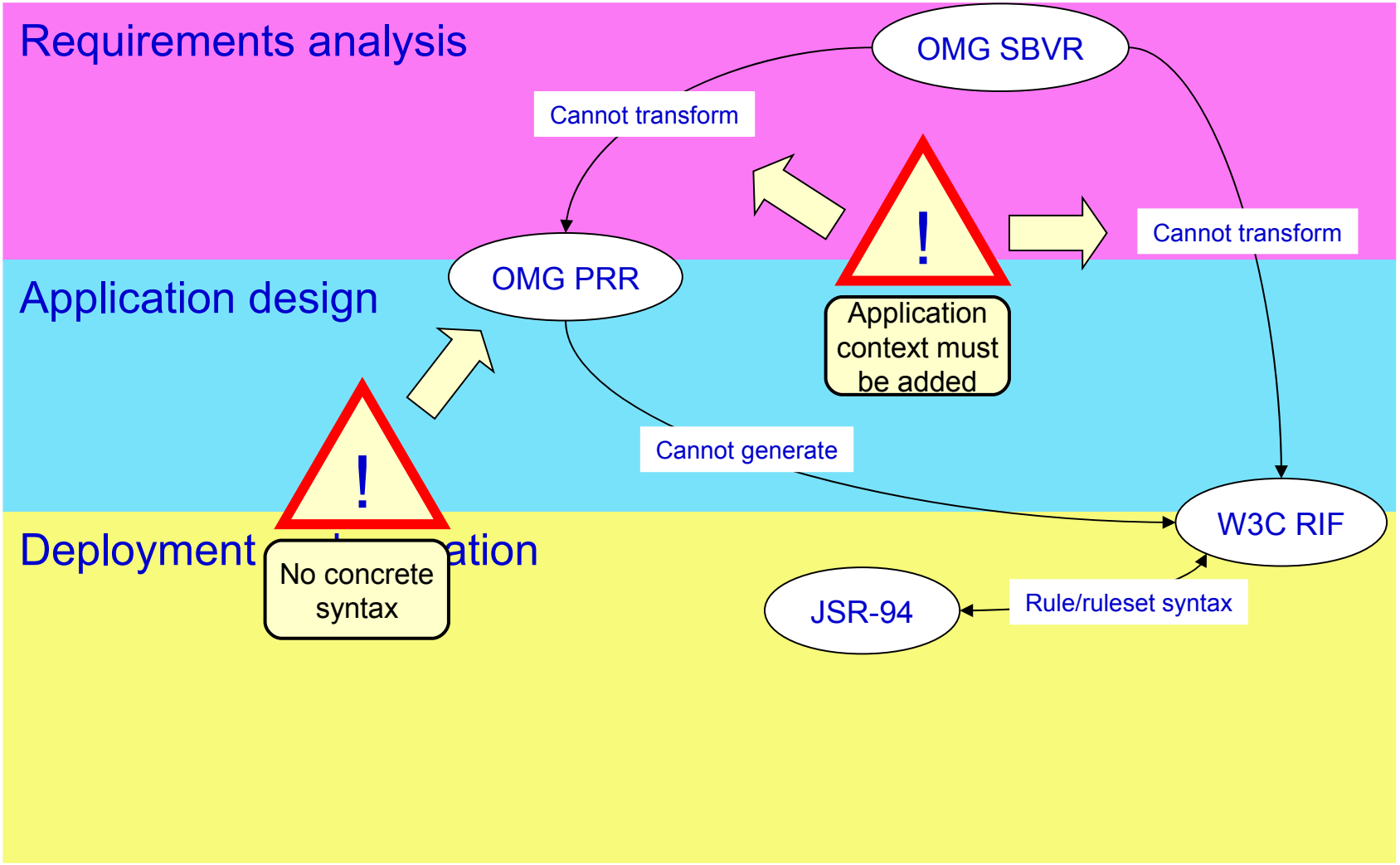


- Decisions are activities within business process
- Decisions are based on business rules

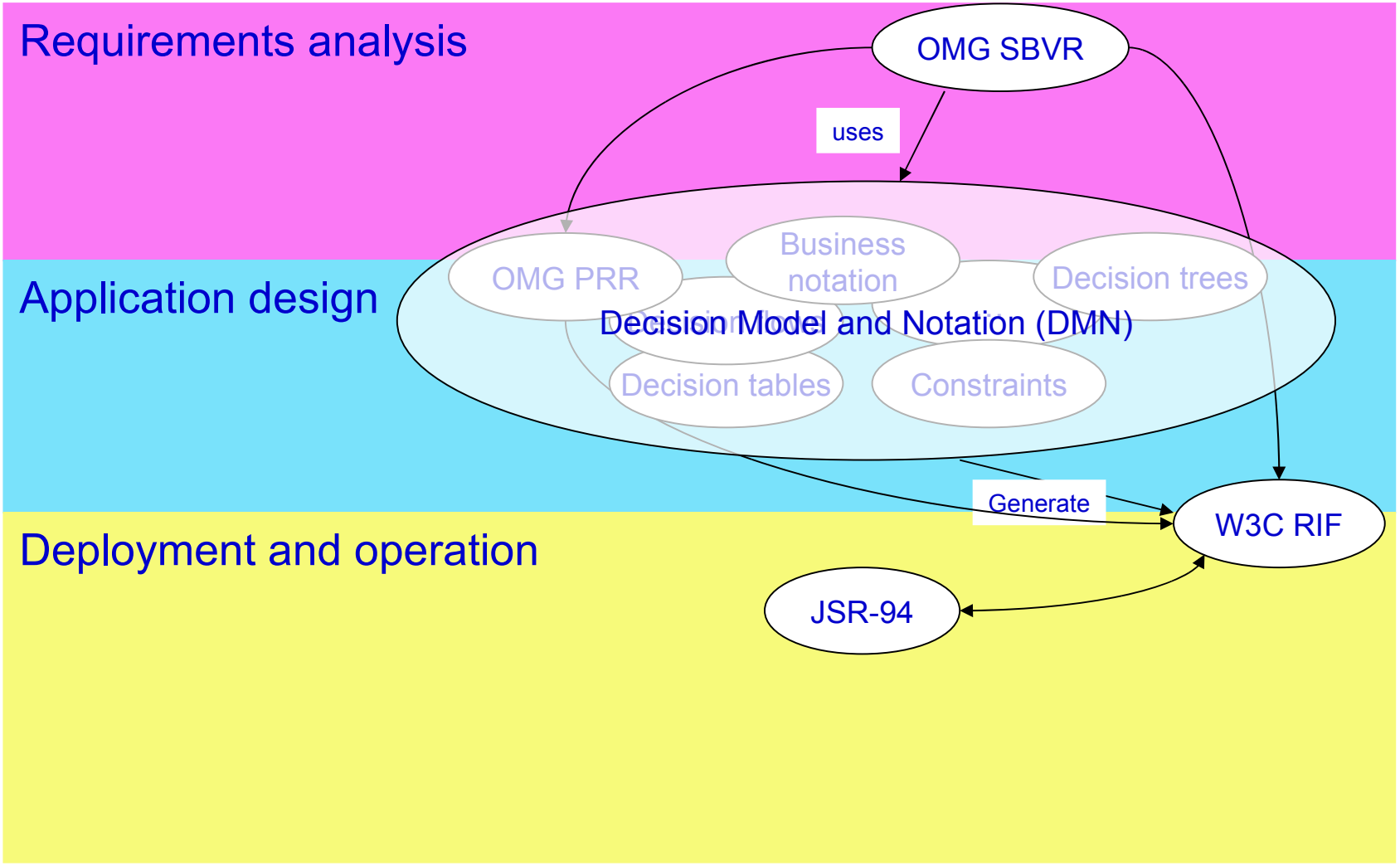
Standards in the business rules life cycle e



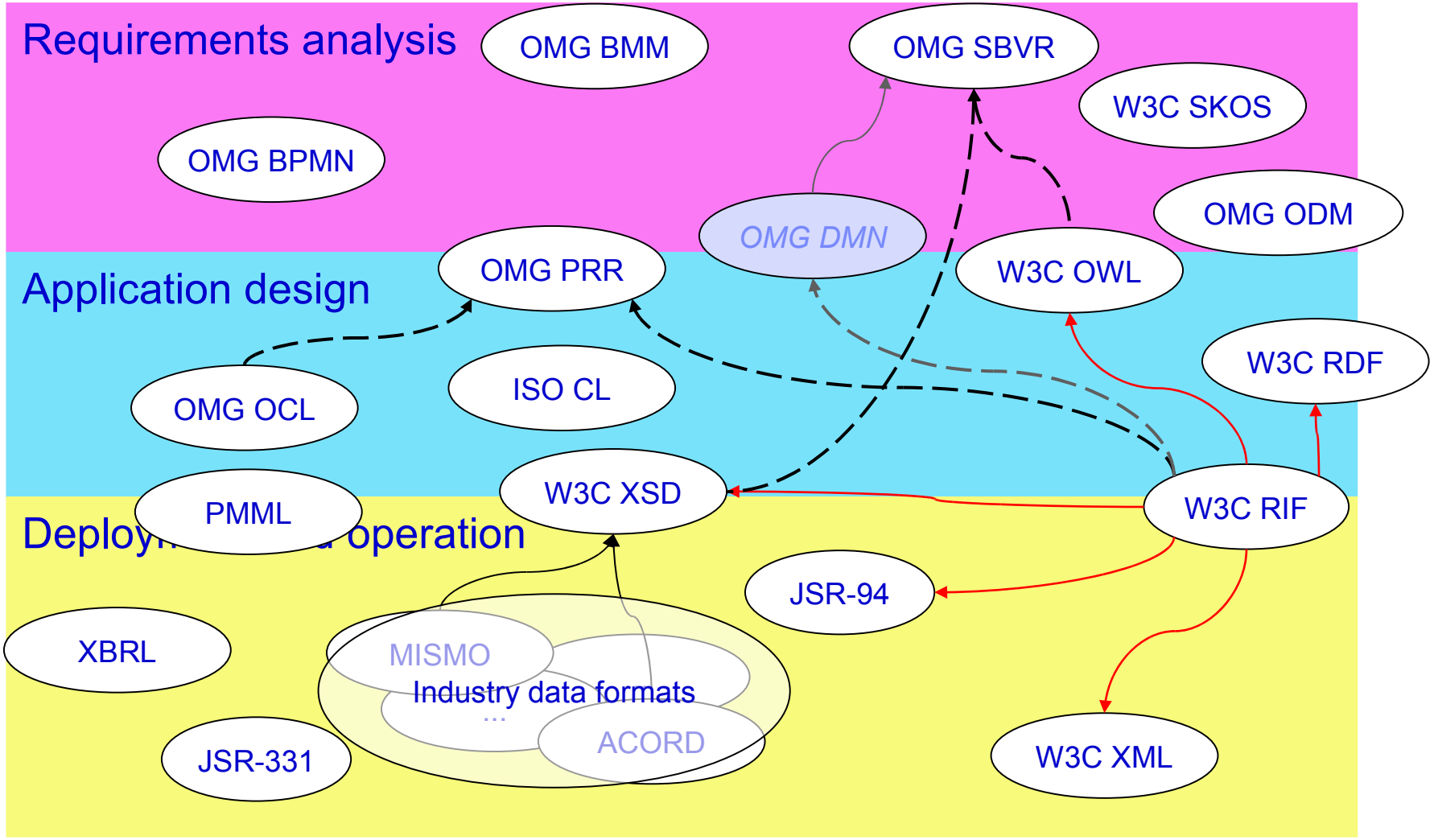
Standards in the business rules life cycle



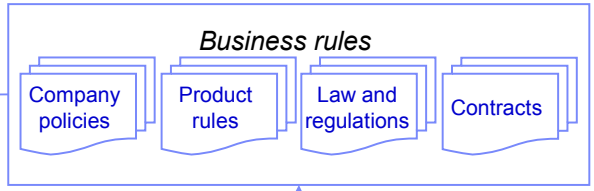
Standards in the business rules life cycle



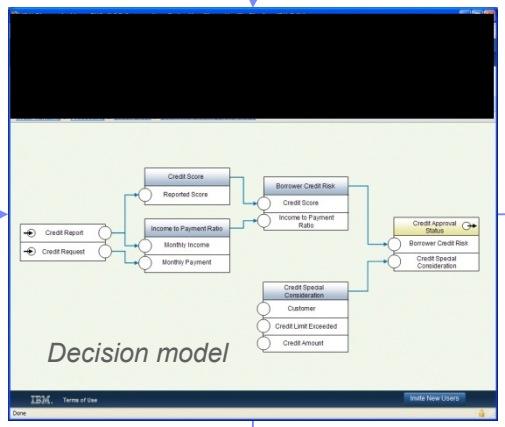
Rule standards and surrounding



All or part of the decision model is derived from documented business policies and regulations

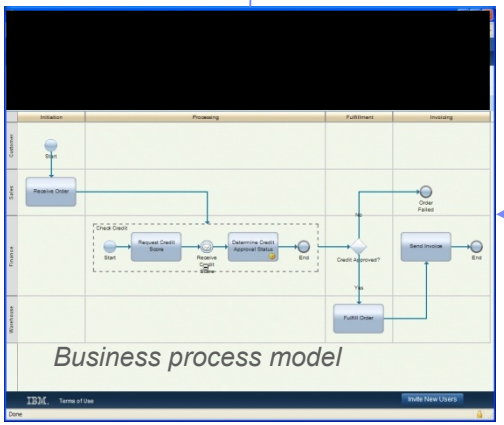


Business policies and regulations and vocabulary documentation is updated from the decision model if required

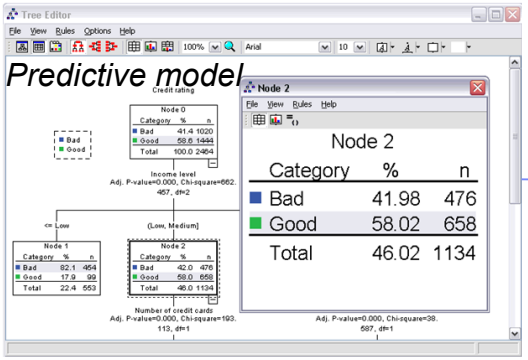


Part of the decision model can also be derived from a documented decision process

The decision model is included in the relevant business processes to document the corresponding decision activities

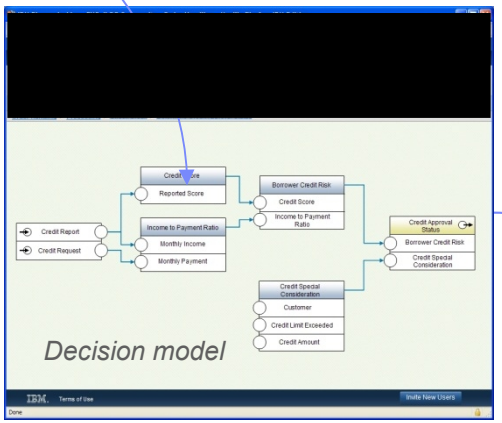


- Business policy documents, decision models and business process models are used by business people to share information across the organisation



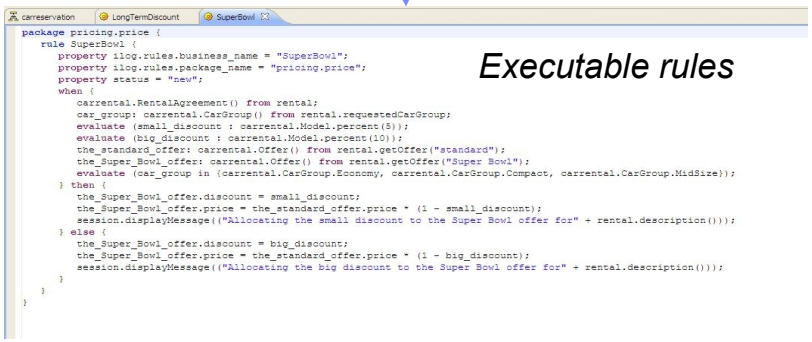
Predictive model

A decision model may reference other models, e.g. predictive analitics



Decision model

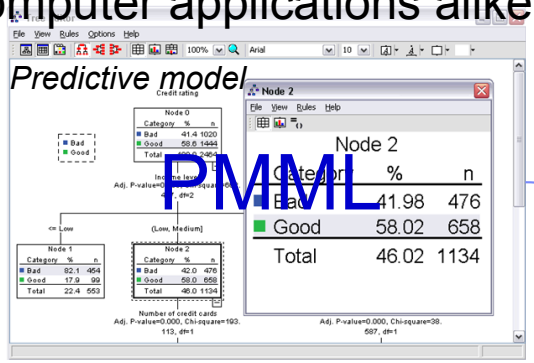
To automate the decision, a decision model is translated into a business rules application

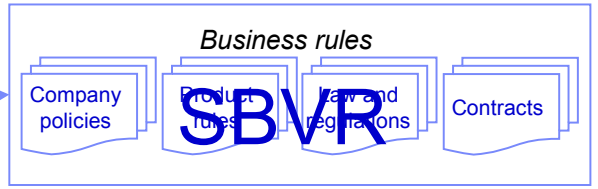


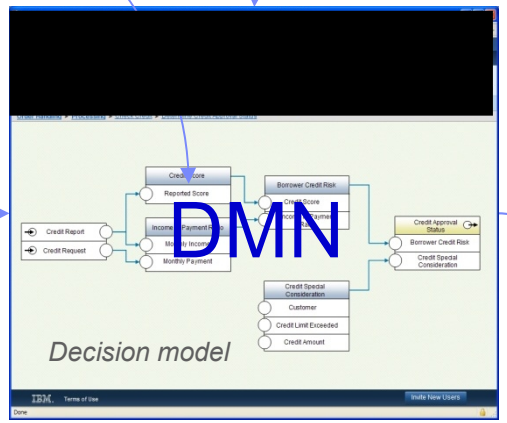
Executable rules

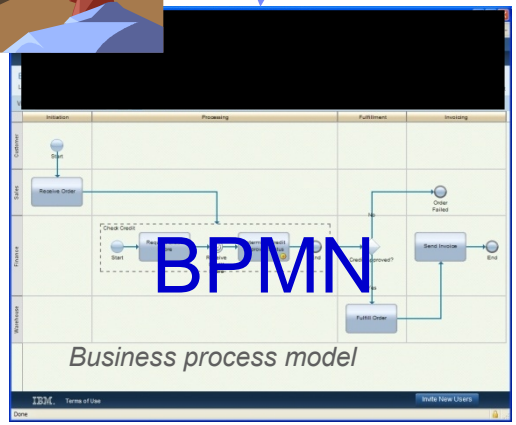
- Technical documents such as executable rules and predictive models are used to communicate information to and between computer applications

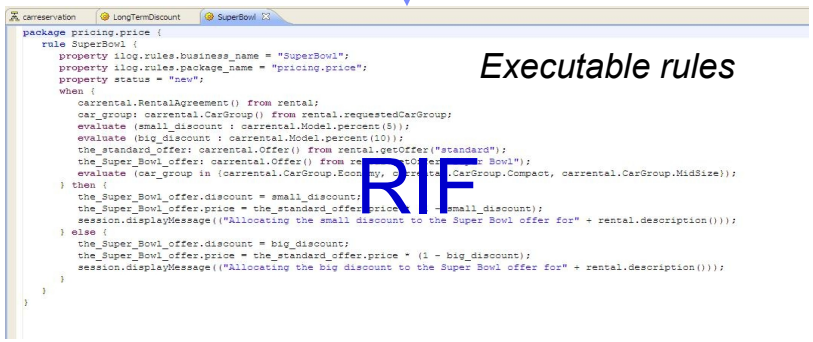
- Standards allow reliable and efficient information interchange between people and computer applications alike



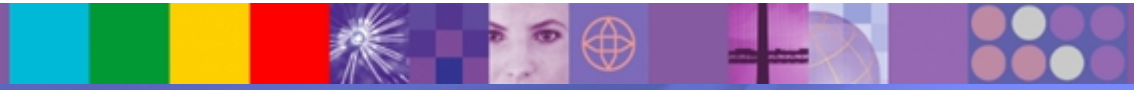








OMG Semantics of Business Vocabularies and business Rules



WebSphere software

OMG Semantics of business vocabulary and business rules (SBVR)

- Supporting the interchange of business vocabularies and rules among business people
  - "To allow business people to define the policies and rules by which they run their business in their own language, in terms of the things they deal with in the business, and to capture those rules in a way that is clear, unambiguous and readily translatable into other representations."
  - Adressing the CIM level of MDA (model driven architecture)
- Vocabulary and rules for documenting the semantics of business vocabularies, business facts and business rules
  - Conceptualized optimally for business people
  - Designed to be used for business purposes independent of information system design
  - Interpretable in predicate logic with a small extension in modal logic. This specification supports
  - Inherent extensibility
  - XMI schema for the interchange of business vocabularies and business rules
  - Linguistic analysis of text for business vocabularies and rules, with the linguistic analysis itself being outside the scope of the specification
  - Non-normative SBVR Structured English

OMG Semantics of business vocabulary and business rules (SBVR)

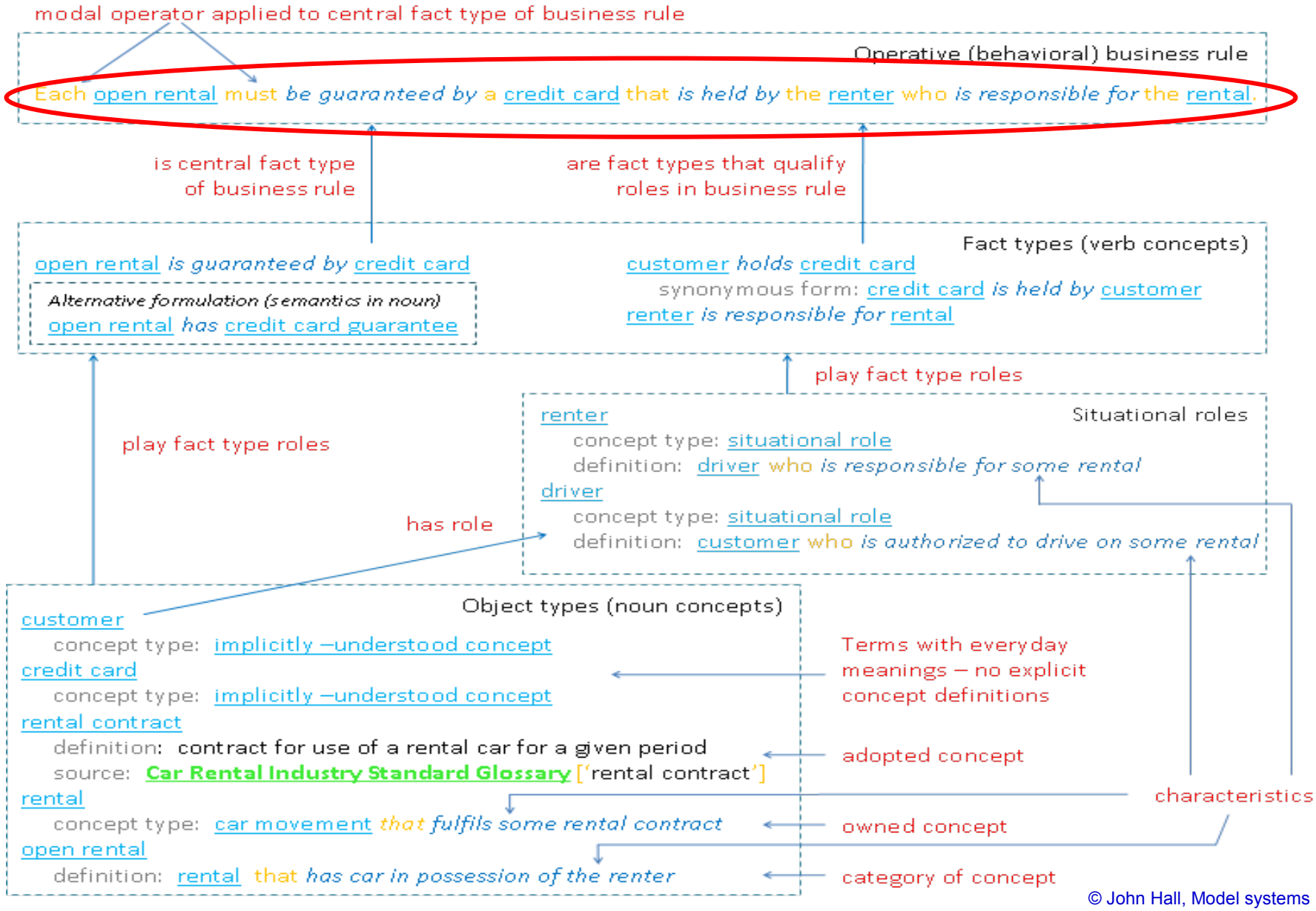
- Status and timeline
  - SBVR 1.0 an OMG adopted specification since January 2008
    - http://www.omg.org/spec/SBVR/1.0/PDF
  - SBVR 1.1 scheduled
- Contributors
  - Consultancies (Business Rule Solutions, LLC, Business Rules Group, Business Semantics Ltd, Hendryx & Associates, InConcept, KnowGravity Inc, Model Systems, PNA Group)
  - Software editors (Adaptive, Automated Reasoning Corporation, Fujitsu Ltd, Unisys Corporation, LibRT, MEGA, Hewlett-Packard Company)
  - Academics (Neumont University, Sandia National Laboratories, RuleML)
- Products
  - Unisys (now Microsoft), Collibra
  - Several implemantations by consulting companies: KnowGravity, KDM Analytics, Rules Art, InConcept, PNA
  - Depending on customer demand: TIBCO, JBoss

OMG Semantics of business vocabulary and business rules (SBVR)

- Example:

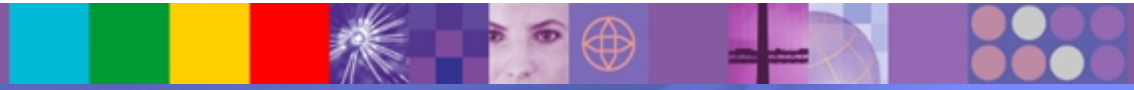
  Policy text says

  "Each open rental must be guaranteed by a credit card that is held by the renter who is responsible for the rental."



© John Hall, Model systems

OMG Decision Model Notation

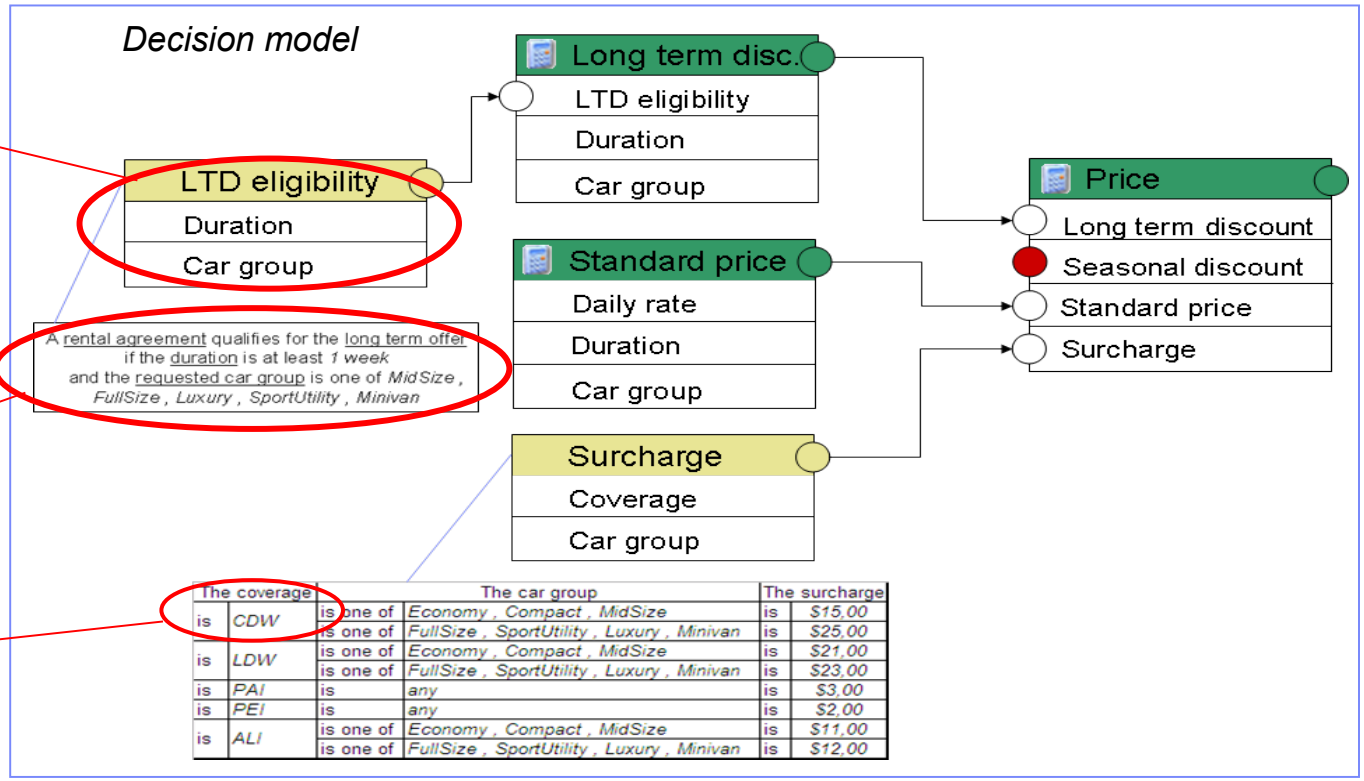


WebSphere software

Decision model could use an SBVR vocabulary

Underlying rule specified in SBVR

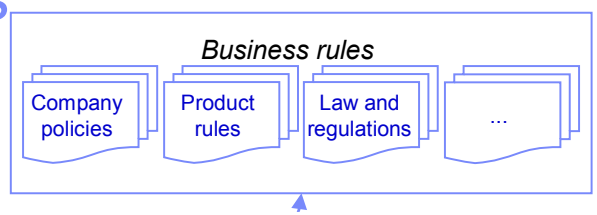
SBVR fact



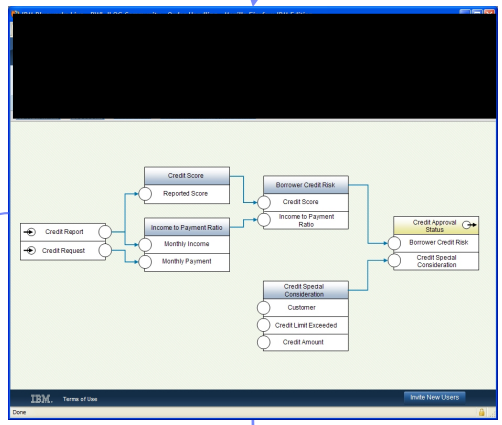
The coverage		The car group		The surcharge	
is	CDW	is one of	Economy , Compact , MidSize	is	$15,00
		is one of	FullSize , SportUtility , Luxury , Minivan	is	$25,00
is	LDW	is one of	Economy , Compact , MidSize	is	$21,00
		is one of	FullSize , SportUtility , Luxury , Minivan	is	$23,00
is	PAI	is	any	is	$3,00
is	PEI	is	any	is	$2,00
is	ALI	is one of	Economy , Compact , MidSize	is	$11,00
		is one of	FullSize , SportUtility , Luxury , Minivan	is	$12,00

Decision models and application models

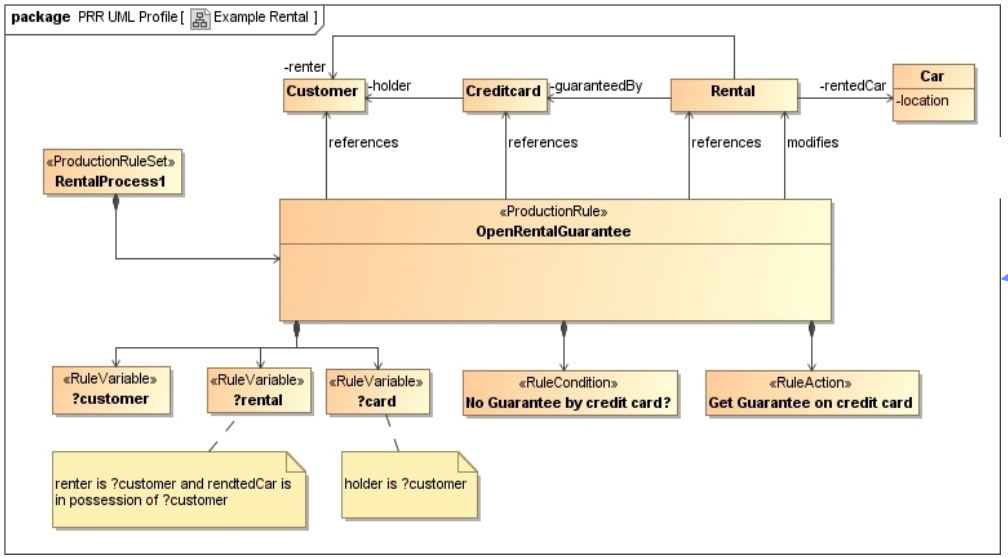
- Decision models can be translated into UML models that represent the classes that are required for the decision, and whether they are referenced or impacted by rules (e.g. using OMG PRR)



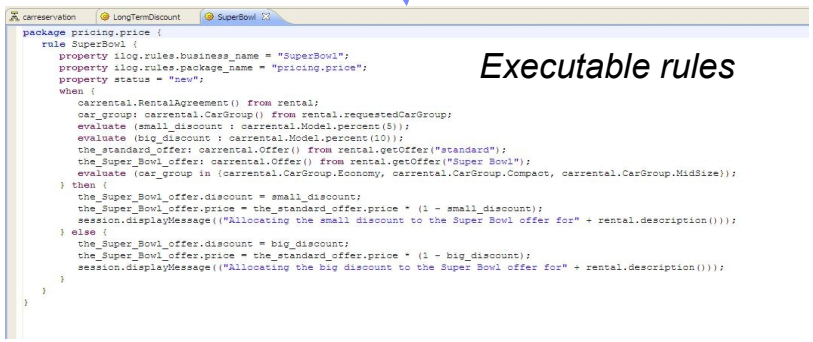
extract



extract

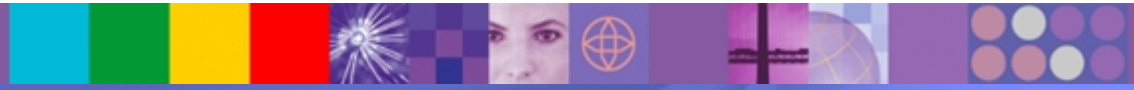


extract



Executable rules

OMG Production Rule Representation meta-model



WebSphere software

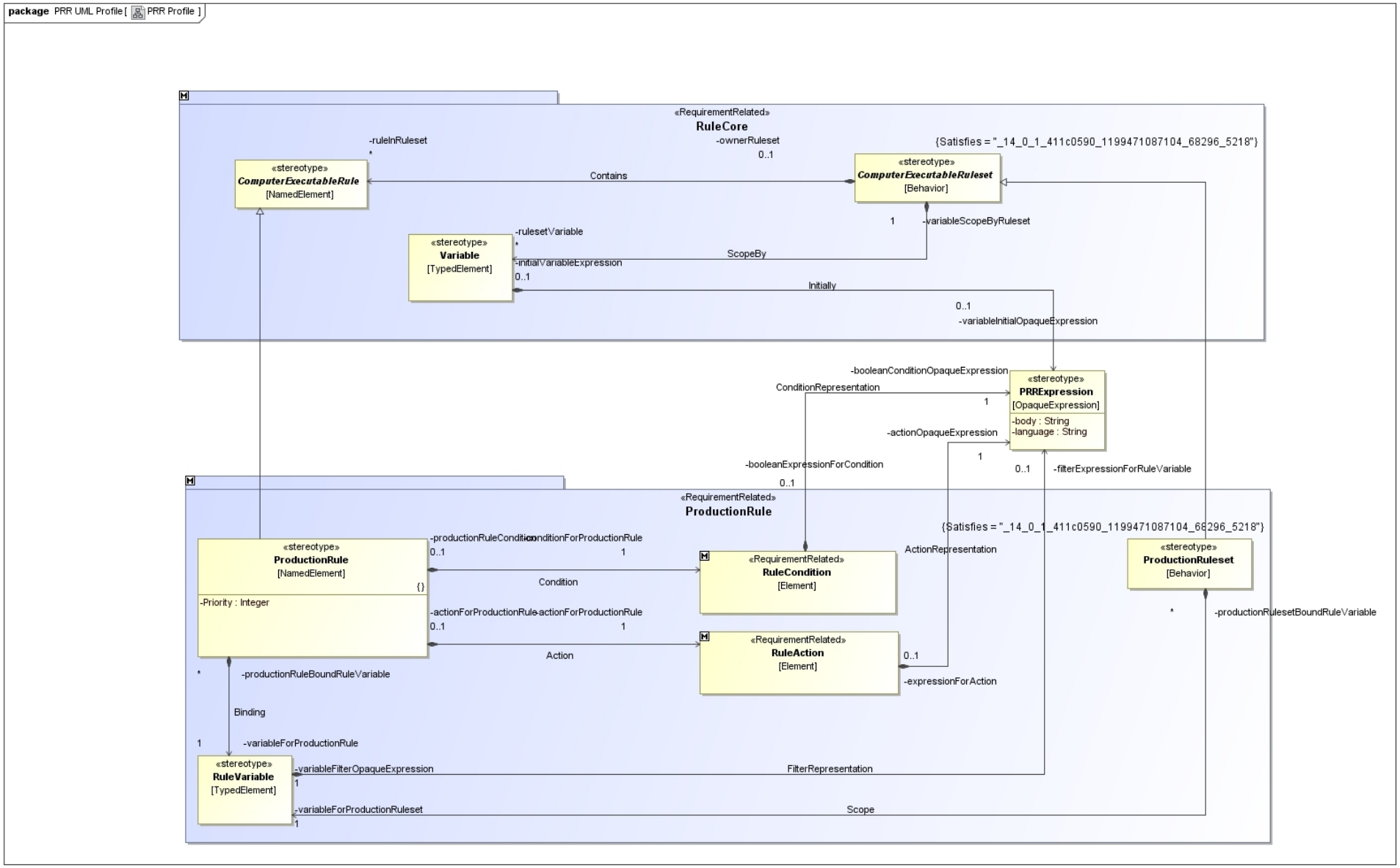
OMG Production rule representation meta-model (PRR)

- Making rules a first class citizen in UML models
  - "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"
  - Addressing the PIM level of MDA
- Metamodel for the class of rules typically used in production rule engines
  - *For <Variable> bindings, If <Condition> is satisfied then execute <Actions>*
  - Associates rules with referenced resp. modified classes
  - Associates rule components (rule variables, condition and actions) with opaque expression
    - PRR OCL: non-normative example of a syntax that is compliant with the metamodel
  - Organized by ruleset
  - Sequential and forward-chaining (Rete-type) modes
  - Provides extension points for other kinds of executable rules
  - Interchange for rule modeling via XMI
    - Import/export between UML modeling tools and between UML modeling tools and BRMSs

OMG Production rule representation meta-model (PRR)

- Status and timeline
  - Currently an OMG beta specification (RTF)
  - http://www.omg.org/docs/dtc/07-11-04.pdf
- Consortium of developers and supporters
  - Rule vendors (Fair Isaac, ILOG, TIBCO, IBM, LibRT, Pega, Corticon)
  - Academic community (RuleML.org)
  - Related vendor community (Fujitsu, IBM, NoMagic, Sandpiper Software)
- Products
  - Implemented as an UML profile by NoMagic
  - Main production rule engines are execution conformant

OMG Production Rules Representation meta-model



OMG Production rule representation meta-model (PRR)

- Policy text says

  "Each open rental must be guaranteed by a credit card that is held by the renter who is responsible for the rental."

- Represented by a set of Production rules

  including the rule

  GIVEN a customer

  GIVEN a rental owned by the customer and that has car in possession of the customer

  GIVEN a credit card that is held by the customer

  IF the rental is not guaranteed by the credit card

  THEN take a guarantee for the rental on the credit card

Object types

Central fact type

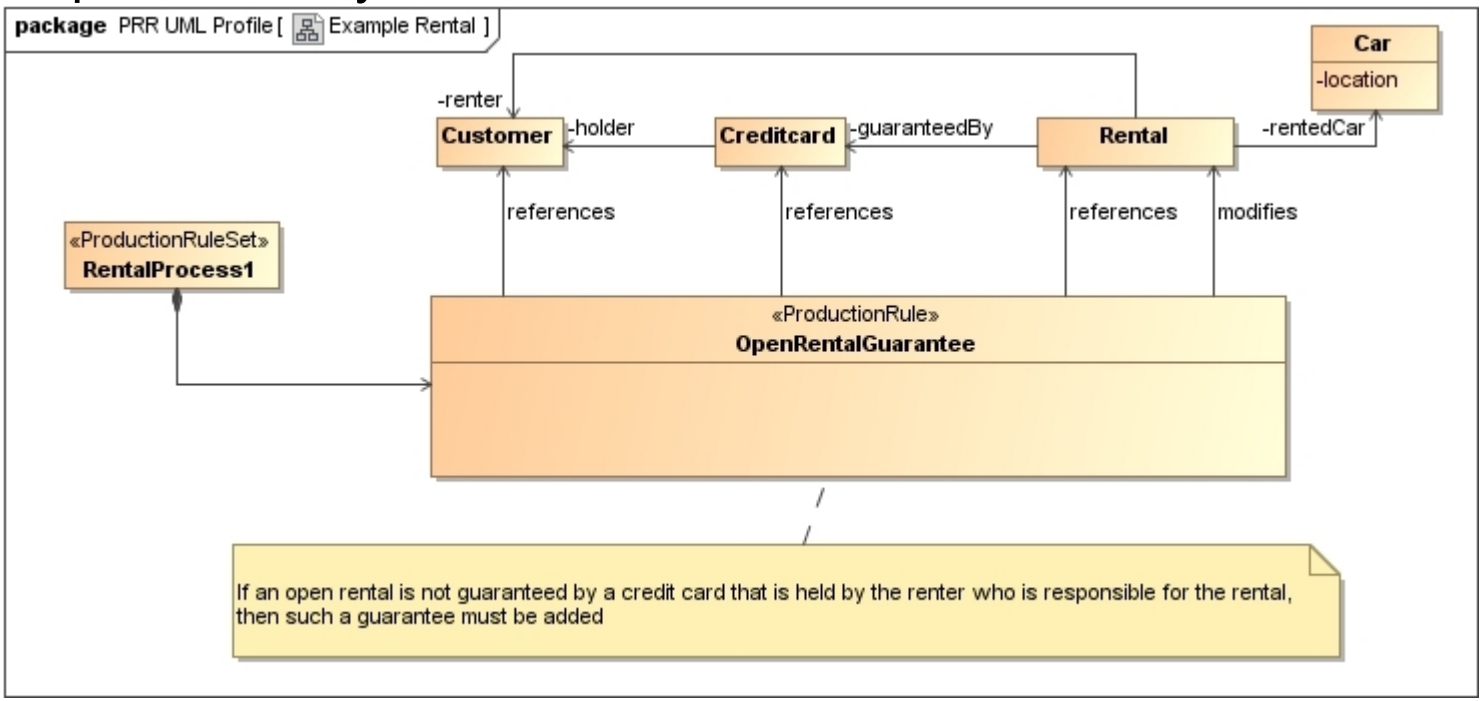
Characteristics

OMG Production rule representation meta-model (PRR)

- Policy text says

  "Each open rental must be guaranteed by a credit card that is held by the renter who is responsible for the rental."

- Represented by a set of Production rules

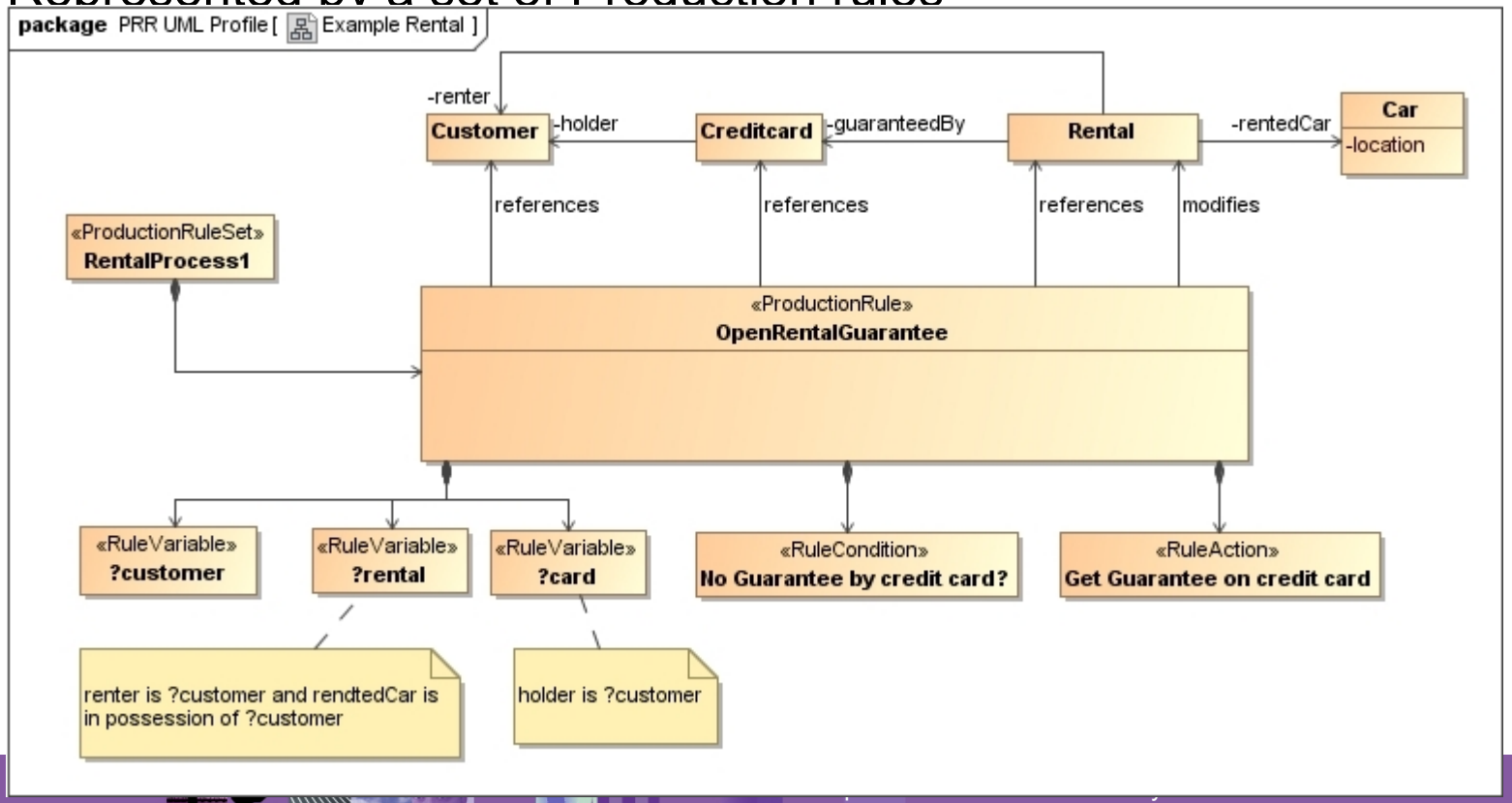


OMG Production rule representation meta-model (PRR)

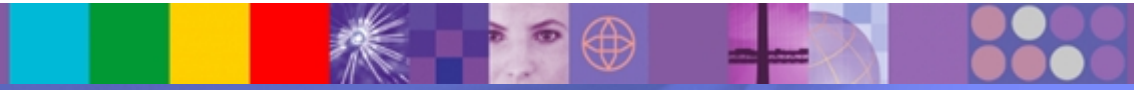
- Policy text says

  "Each open rental must be guaranteed by a credit card that is held by the renter who is responsible for the rental."

- Represented by a set of Production rules



W3C Rule Interchange Format



WebSphere software

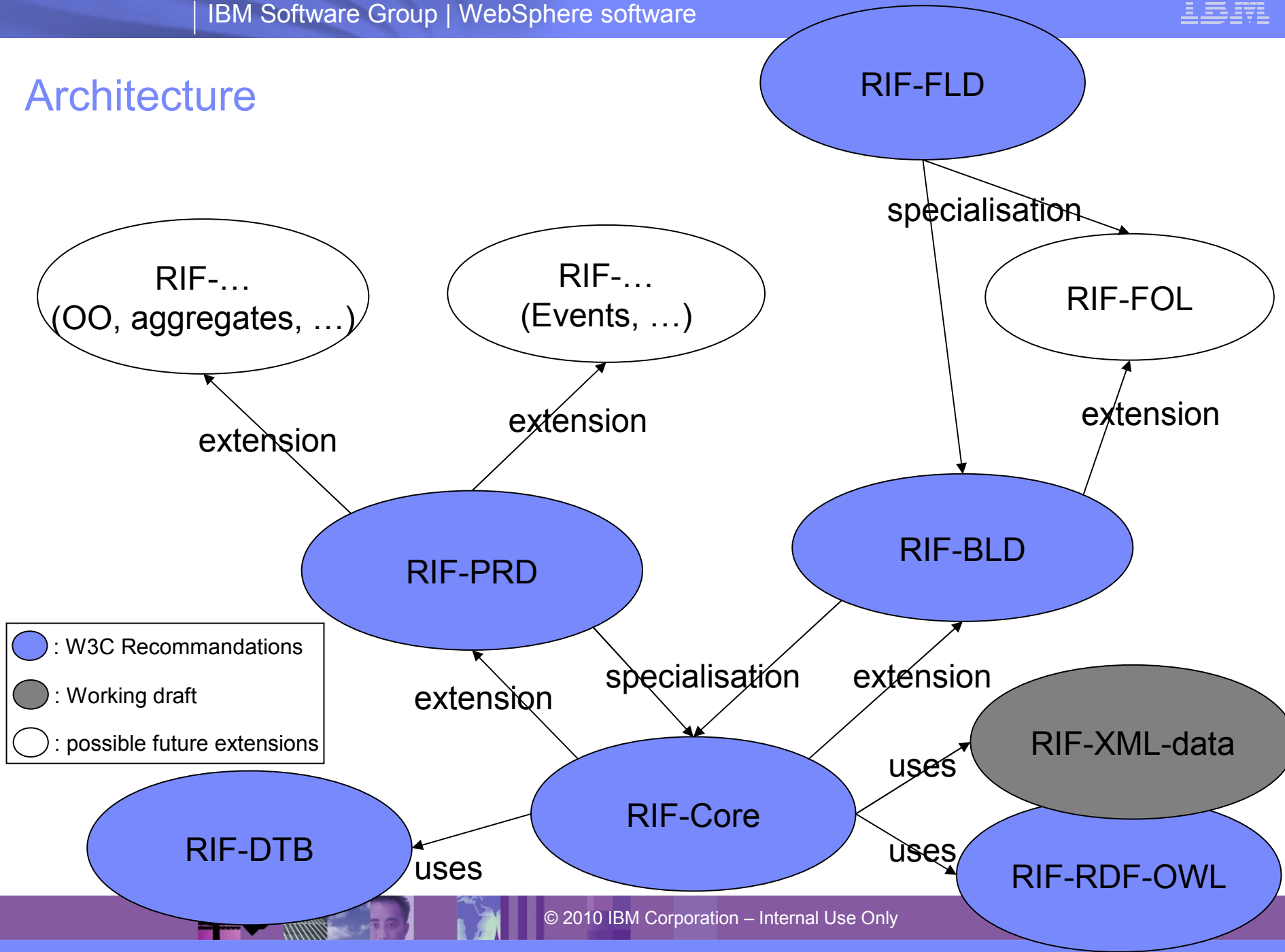
W3C rule interchange format (RIF)

- A standard XML serialization for rules
  - with standard semantics
  - allowing rules written for one application to be published, shared, and re-used in other applications and other rule engines (between rule languages with compatible semantics)
  - A rule is (just another) data item
- A set of dialects and an extensible framework, encouraging interoperability
  - A RIF-Core dialect with the expressive power of Datalog
  - Two extensions: Basic Logic Dialect and Production Rule Dialect
  - An extensive set of built-in data types, functions and operators
  - Combination with RDF, OWL and XML/XSD data sources and data models
- A semantic Web standard
- A W3C recommendation
  - An open standard, endorsed and supported by an well-established industry consortium
  - and a well-defined patent policy

W3C Rule interchange format (RIF)

- Status and timeline
  - Recommendations since June 2010
    - http://www.w3.org/2005/rules/wiki/RIF_Working_Group
- Contributors
  - Rule vendors (ILOG, Oracle, TIBCO, JBoss, Fair Isaac, Corticon, …)
  - Academics/Research (RuleML, SRI, NRCC, U. Bolzano, MITRE…)
  - Related vendors (Hewlett Packard, Sandpiper…)
  - Standard organization (OMG, NIST, …)
  - Invited experts (M. Kiefer)
- Products
  - Reference implementations: http://www.w3.org/2005/rules/wiki/Implementations
  - Commercial implementations:
    - Oracle Business Rules (part of Oracle SOA suite)
    - OntoPrise
    - Planned: IBM
    - Depending on customers demand: TIBCO, JBoss

Architecture



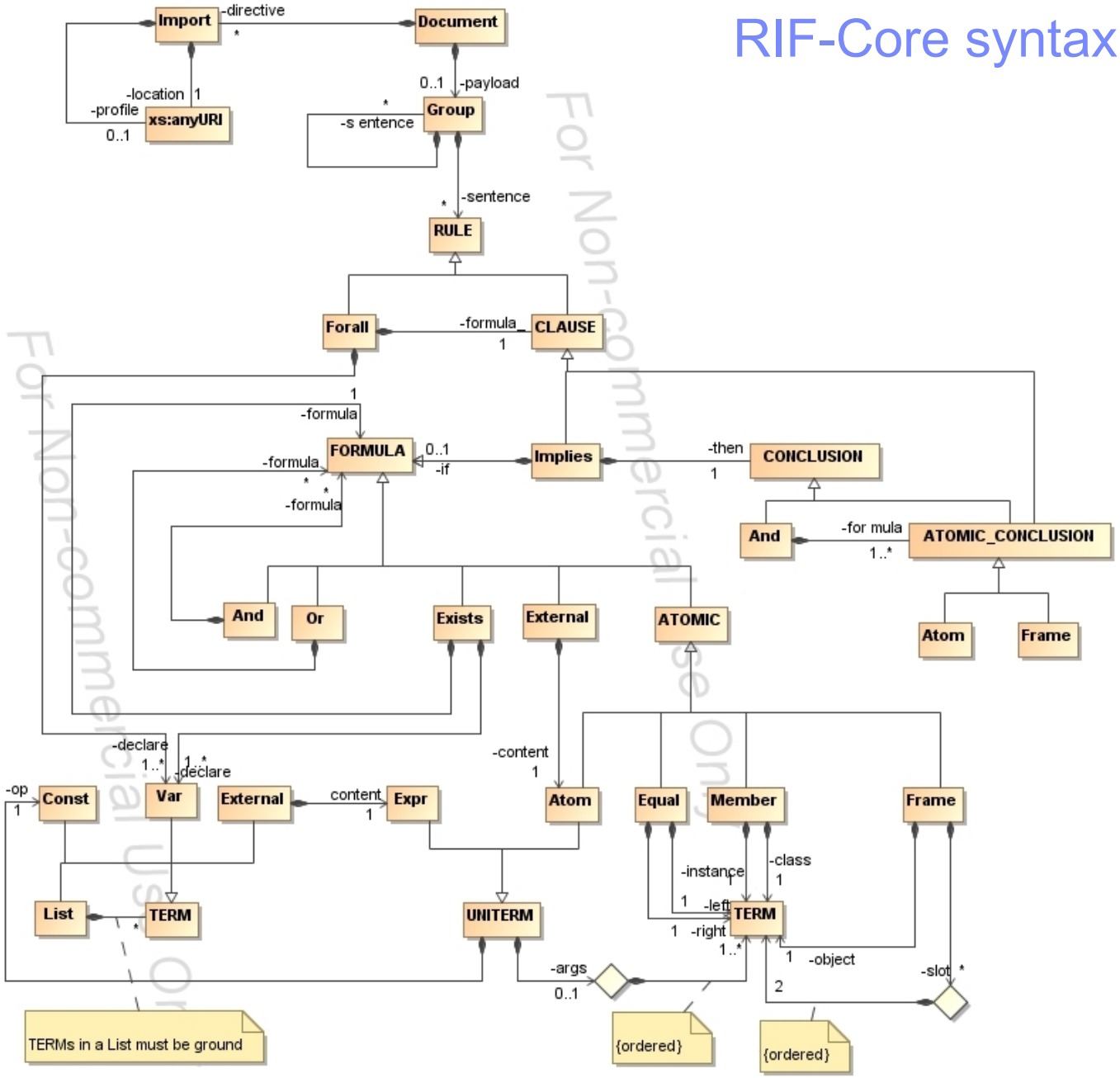
W3C RIF WG published documents

- W3C Recommendations
  - rdf:plainLiteral (common with OWL WG)
  - RIF-Core: RIF Core dialect
  - RIF-BLD: RIF basic logic dialect
  - RIF-FLD: RIF framework for logic dialects
  - RIF-PRD: RIF production rule dialect
  - RIF-DTB: RIF data types and builtins
  - RIF-RDF-OWL: RIF RDF and OWL compatibility
- REC-track specification
  - RIF-XML-data: RIF combination with XML data and schema
- Working group notes
  - RIF-UCR: RIF use cases and requirements
  - RIF-Tests: RIF Test Cases
  - RIF-Overview: Overview of RIF documentation
  - RIF-OWL2-RL: RIF implementation of OWL2 RL

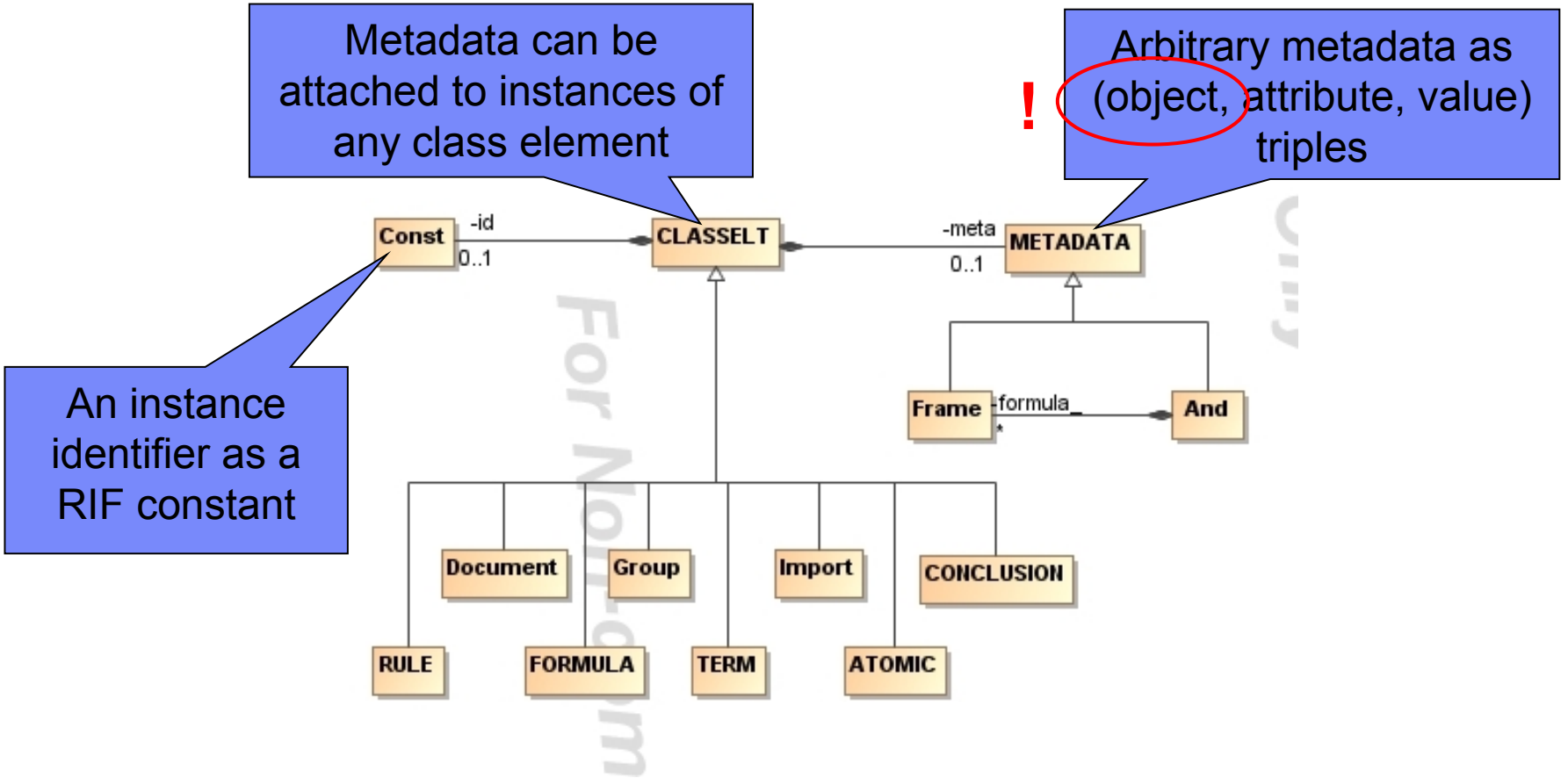
W3C RIF design principles

- Translation paradigm
  - No intrusion in covered rule languages and rule sets
- Same semantics ⇔ same syntax
  - Share constructs accross dialects wherever they agree on the semantics
  - Different constructs where semantics do not agree
- Alternating normal form XML
  - alternating <Class> and <role> tags
  - Metadata can be attached to any class element
  - Except…
- Only XML schema is normative
  - Presentation syntax for specification's readability (examples, semantics etc)

RIF-Core syntax



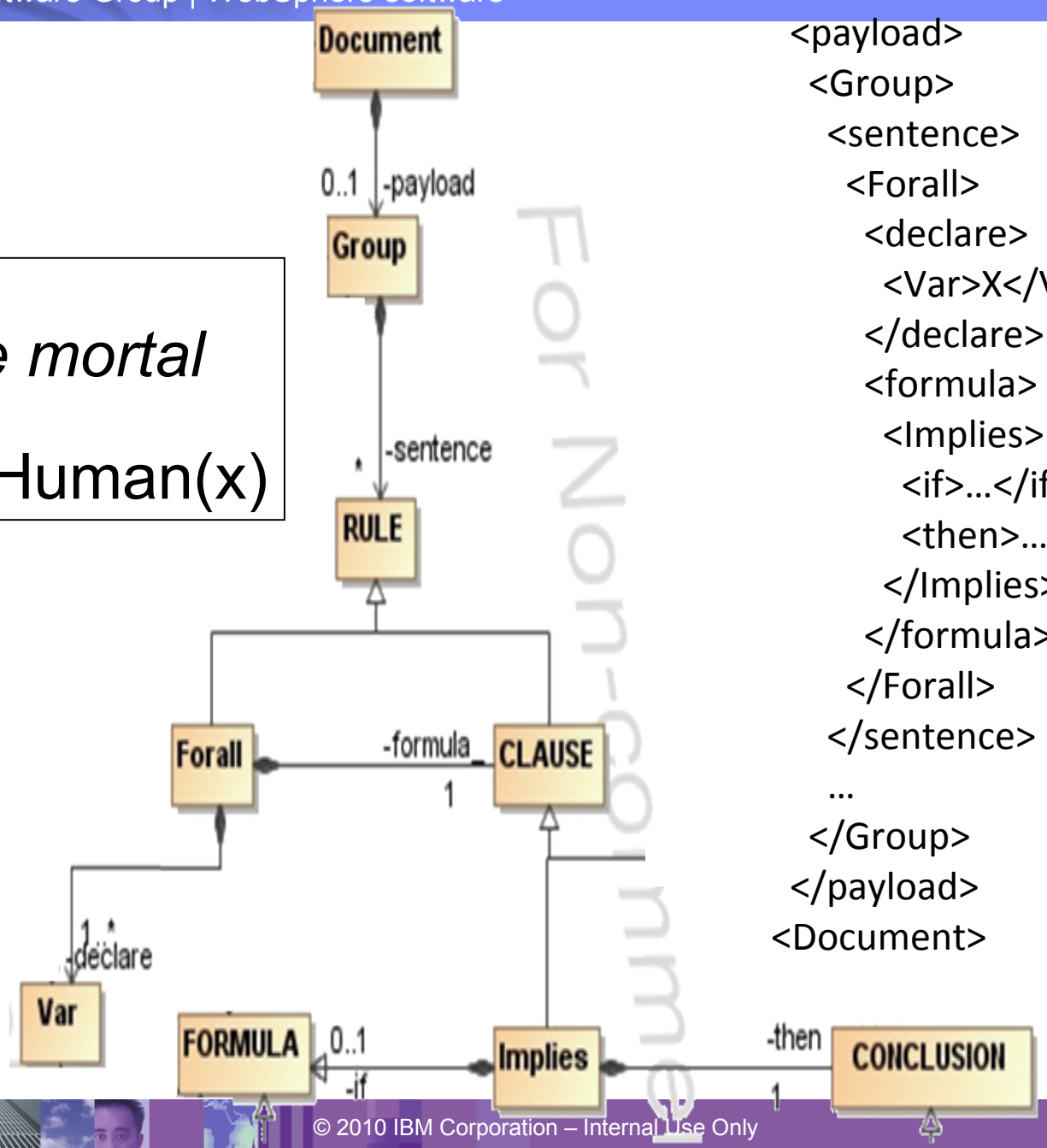
Metadata



XML syntax

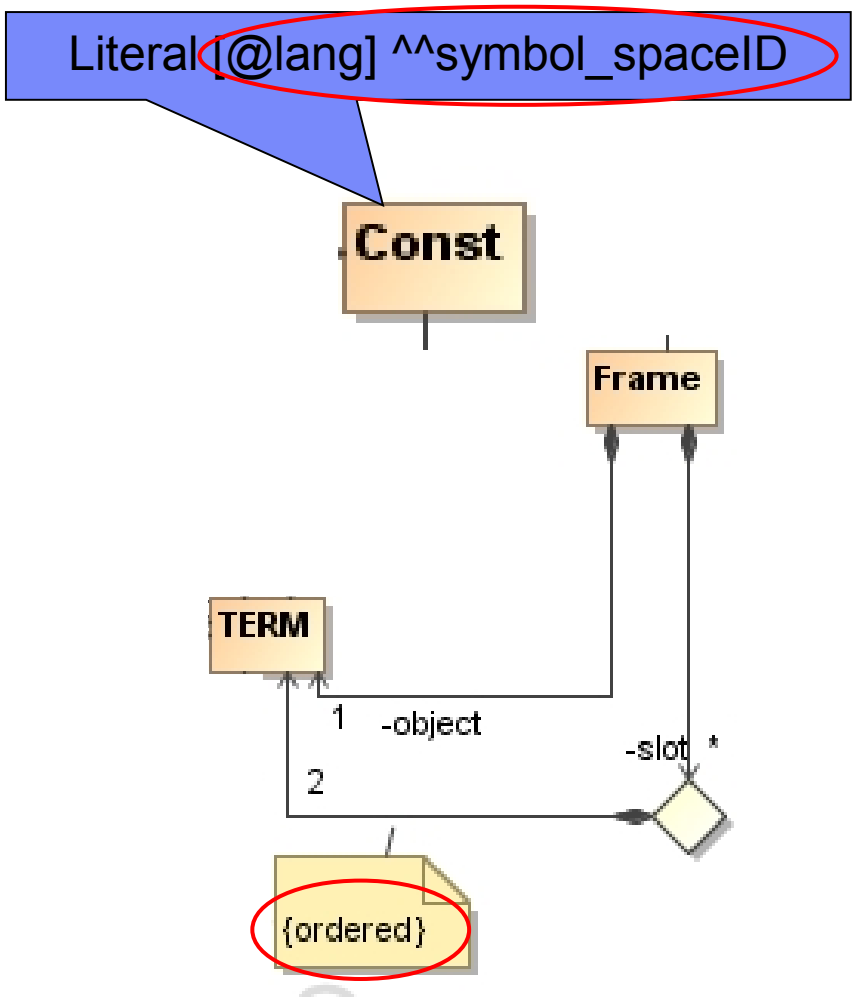
Humans are mortal

Mortal(x) :- Human(x)




```
<Document xmlns=…>
 <payload>
  <Group>
   <sentence>
    <Forall>
     <declare>
      <Var>X</Var>
     </declare>
     <formula>
      <Implies>
       <if>…</if>
       <then>…</then>
      </Implies>
     </formula>
    </Forall>
   </sentence>
   …
  </Group>
 </payload>
<Document>
```

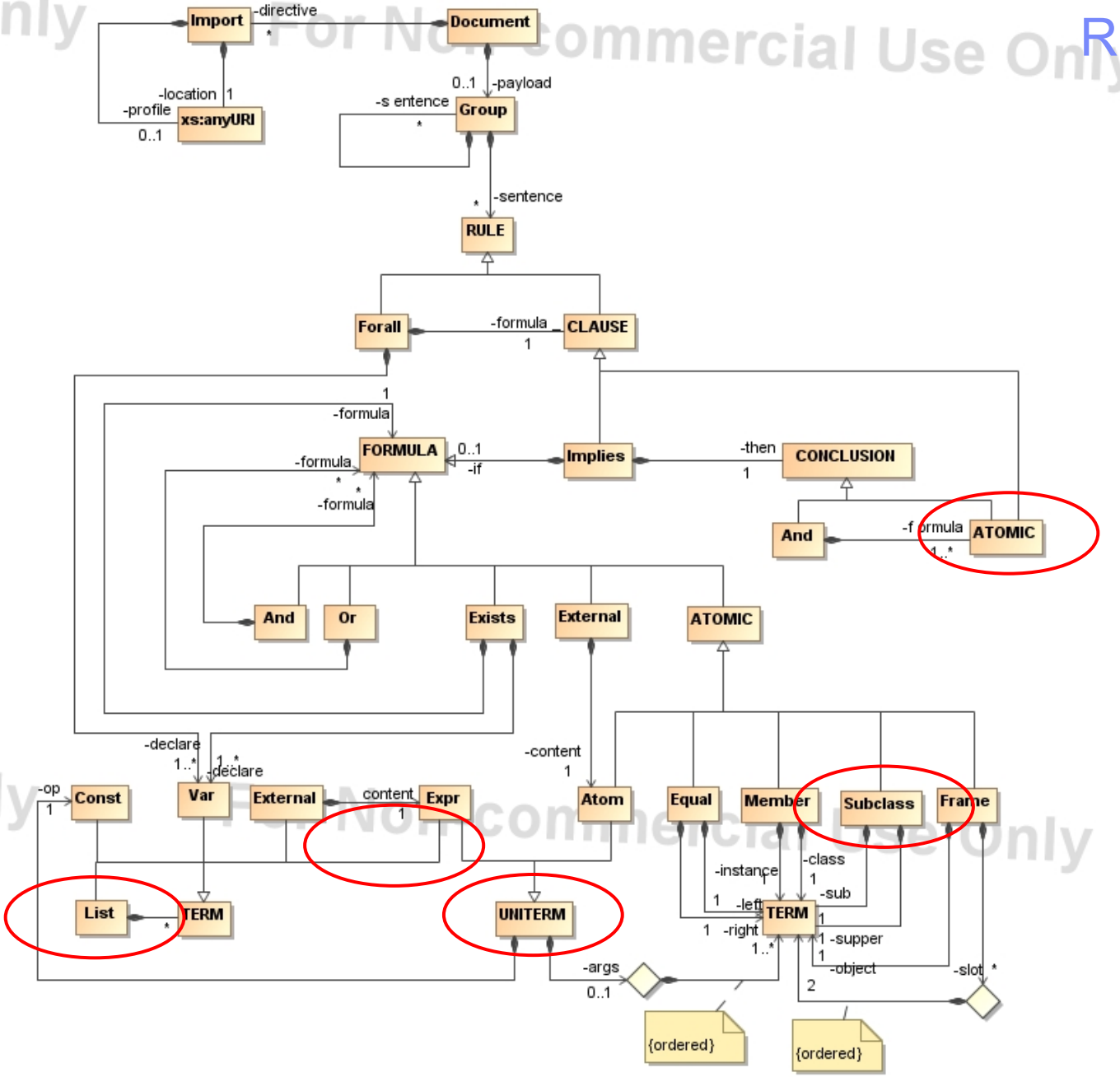
XML syntax (attributes)

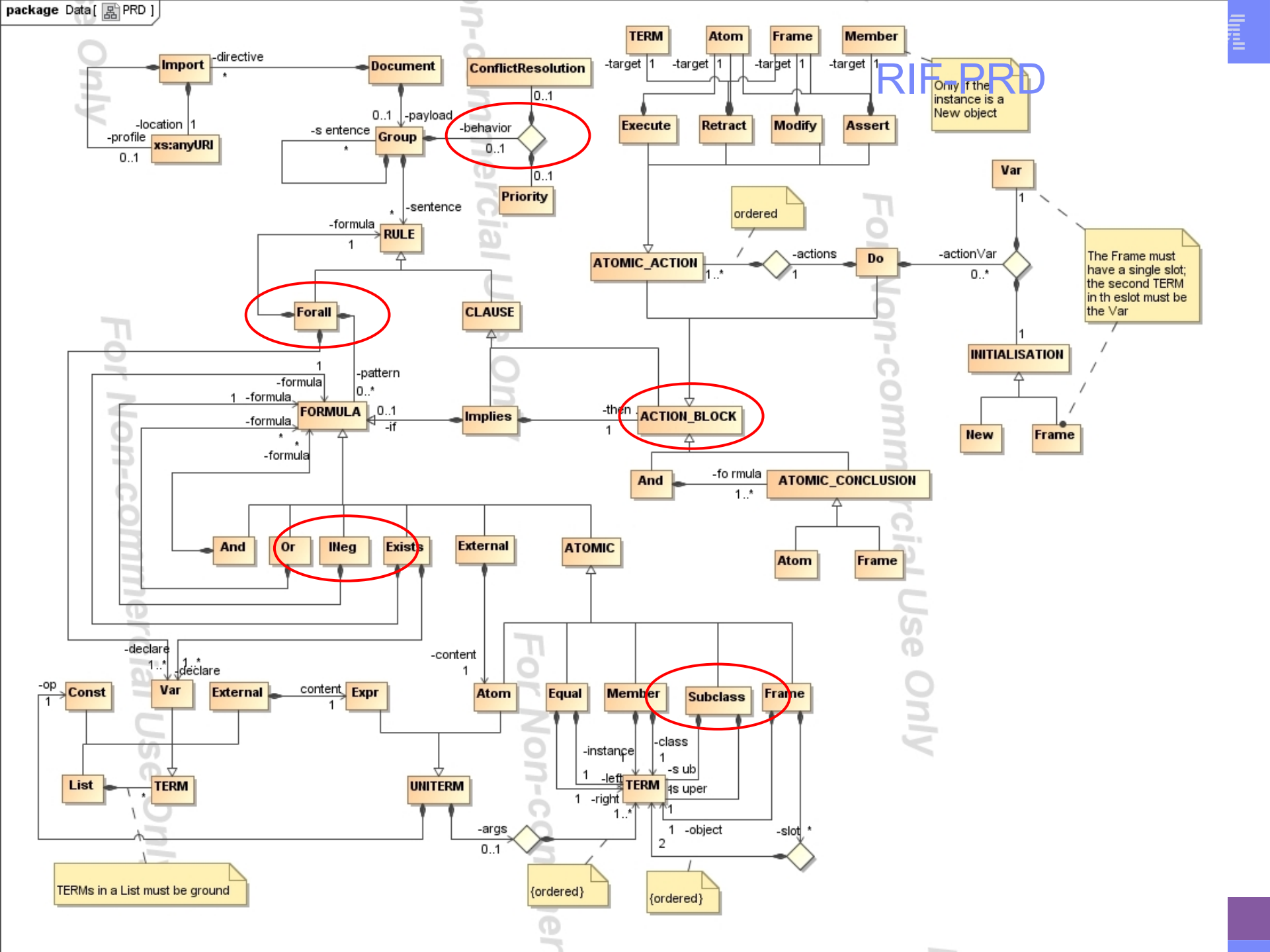


?X["human" -> ?Y]

```
<Frame>
  <object>
   <Var>?X</Var>
  </object>
  <slot ordered="yes">
   <Const type="&xs;string">
    human
   </Const>
   <Var>?Y</Var>
  </slot>
</Frame>
```

RIF-BLD


Import
-directive
*
Document
-location
1
-profile
xs:anyURI
0..1
0..1
-payload
-s entence
*
Group
*
-sentence
RULE
Forall
-formula
1
CLAUSE
1
-formula
FORMULA
0..1
-if
Implies
-then
1
CONCLUSION
-formula
*
*
-formula
And
-formula
1..*
ATOMIC
And
Or
Exists
External
ATOMIC
-declare
1..*
1..*
-declare
-content
1
-op
1
Const
Var
External
content
1
Expr
Atom
Equal
Member
Subclass
Frame
List
*
TERM
UNITERM
-instance
1
-left
1
-right
TERM
-class
1
-sub
1
1..*
1
-supper
1
-object
2
-slot
*
-args
0..1
{ordered}
{ordered}


package Data [PRD]
RIF_PRD
Import
-directive
Document
ConflictResolution
-location
-profile
xs:anyURI
-s entence
Group
-behavior
-payload
Priority
-sentence
-formula
RULE
Forall
CLAUSE
-pattern
FORMULA
Implies
-if
-then
ACTION_BLOCK
And
Or
INeg
Exists
External
ATOMIC
TERM
Atom
Frame
Member
-target
Execute
Retract
Modify
Assert
Only if the instance is a New object
ordered
ATOMIC_ACTION
-actions
Do
-actionVar
Var
The Frame must have a single slot; the second TERM in th eslot must be the Var
INITIALISATION
New
Frame
-fo rmula
ATOMIC_CONCLUSION
-declare
-op
Const
Var
content
Expr
-content
Equal
Subclass
-instance
-class
-s ub
-s uper
-left
-right
-object
-slot
-args
List
UNITERM
TERMs in a List must be ground
{ordered}
{ordered}

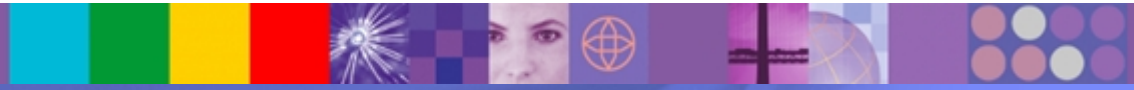
Built-in data types and externals

- All XML Schema 1.1 data types
  - Except `anyType`, `anySimpleType`, `anyAtomicType`, `normalizedString` and derived types, `NOTATION`, `precisionDecimal`, `QName`, `ENTITIES`, `IDREFS`, `NMTOKENS`
  - Plus `rdf:XMLLiteral` and `rdf:PlainLiteral`
- Symbol spaces without value spaces
  - `rif:local` and `rif:iri`
- RIF built-in functions and predicates are *external* in the sense that their semantics is defined outside of – and independently from – the rules where they are used
  - As opposed to logic functions and predicates
  - Include:
    - Data type comparison, guards, convertion and casting
    - Numeric, boolean, string, date, time and duration operators and comparison predicates
    - rdf:XMLLiteral and rdf:plainLiteral specific predicates and operators
    - Lists predicates and operators
    - Dialect-specific built-in functions
      - act:print (RIF-PRD)
- Many RIF built-in functions and predicates are adapted from *XQuery 1.0 and XPath 2.0 Functions and Operators*
  - The differences from the original XF&O include the handling of errors, the differentiation between predicates and functions, and a few other specific differences
  - If an argument value is outside of its domain, the value of the function or predicate is left unspecified

RIF semantics

- RIF-BLD: standard first-order model-theoretic semantics with functions and equality
- RIF-PRD: standard RETE-style operational semantics
- RIF-Core: essentially Datalog
  - Model-theoretic semantics of RIF-BLD, restricted to RIF-Core syntax
    - No subclass, no logic functions, no named argument UNITERMS
    - Only atoms and frames in the head
  - Operational semantics of RIF-PRD, restricted to RIF-Core syntax
    - No negation
    - Pattern-to-Condition equivalence
    - Only Asserts in the RHS: Do-to-And equivalence
    - With those restriction, the operational semantics of RIF-PRD reduces to Datalog inflationary semantics, equivalent to minimal Herbrand model semantics
- RIF Core and RIF-PRD require **safeness**
  - RIF-BLD does not

JSR94 – ava Rule Engine API



WebSphere software

JSR 94 – Java Rule Engine API

- A lightweight API to enable
  - Stateful rule engine use
    - Adding and removing objects to/from engine
    - Retrieving the results of execution
  - Stateless rule engine use
    - Passing input objects and retrieving results
  - Deploying/undeploying rulesets
  - Based on the assumption that most clients need to be able to execute a basic multiple-step rule engine cycle that consists of parsing rules, adding objects to an engine, firing rules, and getting resultant objects from the engine
- Defines a Java API for rule engines
  - Targeting the J2SE platform
  - Prescribing a set of fundamental rule engine operations
    - acquiring interfaces to an implementation.
    - Invoking rule execution sets
    - Loading rule execution sets from external resources and registering them for use
  - Out of scope
    - The syntax and file-formats of rules and rule execution sets.

JSR 94 – Java Rule Engine API

- Status
  - Final specification released August 04
    - http://jcp.org/aboutJava/communityprocess/final/jsr094/index.html
  - Integrated by SUN into the JDK
    - Javax.rules and Javax.rules.admin
- Contributors
  - ATG, BEA Systems, HNC Software, Fujitsu, IBM, ILOG (now IBM), Oracle, Sandia National Labs, Silverstream, Unisys (now IBM)
- Products
  - Widely implemented
  - 40+ Open Source and commercial Java rule engines are listed at http://www.javarules.org

Thank you!

Questions?

WebSphere software