



# Production Rule Standards

OMG PRR Presentation

**Christian de Sainte Marie**  
ILOG S.A.

**James Taylor**  
Fair Isaac Corporation

June, 2007



Changing the rules of business



# AGENDA



- Introduction to Production Rules
- Standards Work
- PRR 1.0, including changes since SanDiego/March07
- Vote to Vote
- FTF Chartering
- PRR Roadmap and PRR v2 RFP



# Introduction to Production Rules

# Problems in Business Automation

- Software is in maintenance once deployed, but business never stops changing or adapting
- Solutions include
  - Model driven engineering (e.g. eXecutable UML)  
BUT behaviour models are still “programmed” in Action Semantics etc!
  - CASE / RAD tools  
BUT “programmers” still script the solution + few standards
  - Business Rules  
The subject of this presentation

# Business Rules Approach

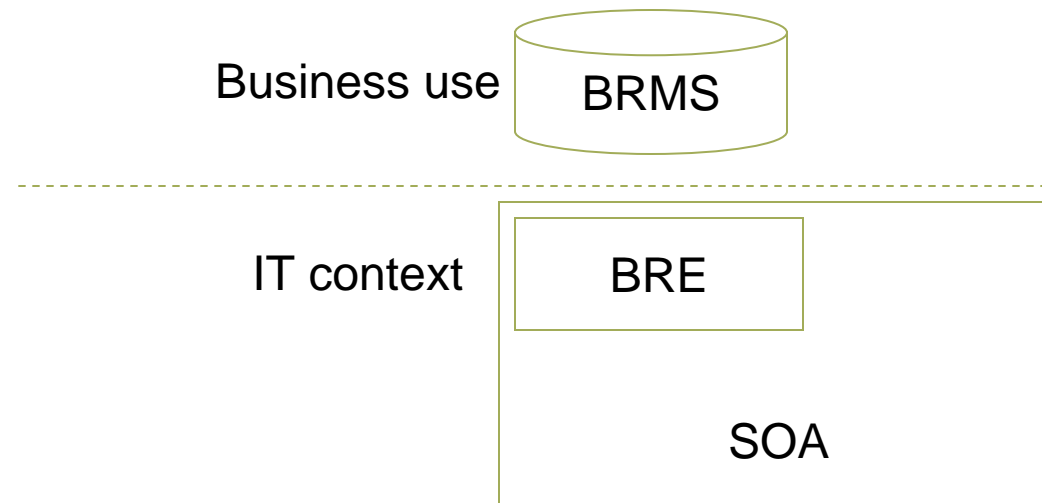
- Define behaviour as declarative logic statements
- Applies ACROSS the software development lifecycle
  - Software Requirements - Use Cases + Lists of rules
  - Decision services - lists of rules orchestrated by ruleflows
  - etc
- Conforms to model-driven approach
  - Rules can be presented using business terms/vocabulary
  - Rule change process can be embedded in “business change” workflow
  - Easy mapping to production rules that can be efficiently executed by a “Business Rule Engine” (BRE)

# Business Rules = Flexible + Extensible

- Users can be presented with PR defined in business terms + appropriate user interfaces
  - Agility through business control / business rule management (via BRMS)
  - Familiar vocabulary / Domain Specific Language
- Extensible to multiple user paradigms
  - Decision Tables
  - Decision Trees
- Extensible to multiple IT architectures
  - Rules in BPM
  - Rules in EDA and CEP
  - Decision Services in SOA
  - Rule-driven UIs

# What are Business Rule Engines?


























- Rules executed by a BRE (and managed in a BRMS) are:
  - Declarative and atomic
  - Contained in rulesets for execution-context
  - Executed sequentially or with high performance algorithms (such as Rete)
  - Test against and/or update data and object models
  - Deployed by being grouped into rule or decision services



# Why are BREs significant to OMG?

- Fit to OMG MDA™
  - Rules (behavior) is a 1<sup>st</sup> class entity versus “objects”
  - Separation of concerns: business logic NOT best “encapsulated” in objects (although system logic could be – per OOP)
- Significant usage of rules already in OMG
  - QVT, OCL, already specify some rule types for MDA/UML usage
  - BPM is a significant specifier of rules – many BPMS vendors link to a BRMS
- End-user recognition
  - In v large organizations, **8% often** and **27% occasionally** use a rule engine (IDC, 2005)
  - **24% of developers are using** and **39% considering** a BRE to enhance agility (IntelligentEnterprise.com, 2007)

# BREs & BMI: BPMS usage of BRMS/BREs

BPMS Vendors	Partner with BRMS Vendors
 	 
	   
	 
 	
	 
 	 
	 

# BRE End Users Include



# Books about using Business Rule Engines

- [Smart Enough Systems: How to Deliver Competitive Advantage by Automating Hidden Decisions](#) by James Taylor with Neil Raden
- [Business Rule Revolution: Running Business the Right Way](#) by Barbara Von Halle et al
- [Business Rules Applied: Building Better Systems Using the Business Rules Approach](#) by Barbara Von Halle
- [Business Rules Management and Service Oriented Architecture: A Pattern Language](#) by Ian Graham
- [Principles of the Business Rule Approach](#) by Ronald G. Ross
- [Business Process Management With a Business Rules Approach: Implementing the Service Oriented Architecture](#) by Tom Debevoise
- [Business Rules and Information Systems: Aligning IT with Business Goals](#) by Tony Morgan



# Standards Work

# What standards exist for BREs?

- No standard approach for production rule representation
- Vendors have their own representations
  - Fair Isaac - Blaze Advisor - SRL
  - Ilog - JRules – IRL
  - TIBCO – BusinessEvents
  - LibRT - RBML
  - ...
- Additional, domain-specific data standards include rules
  - ACORD SPX insurance validation rules
  - xBRL reporting rules
  - MISMO mortgage processing rules
  - ...

# Existing Standard: Java Rule Engine Execution

- JSR-94 Java API for Rule Engines
- JCP (Java Community Process)
- 1<sup>st</sup> standards body to address rule API standards
- JSR-94 proposed as a cross-vendor rule execution API
  - Targeting the J2SE platform
  - Prescribing a set of fundamental rule engine operations
- Integrated by SUN into the JDK
- Widely supported by most Java (PR) rule engines



# Proposed OMG Standard: Production Rule Representation

- OMG – via Business Rules Working Group (now BMI)
- 1st standards body to address rule representation standards
- Production Rule Representation proposed as a cross-vendor rule modeling representation
- Consortium of developers and supporters
  - Rule vendors (including Fair Isaac, Ilog, LibRT, IBM, Pega, Corticon, TIBCO)
  - Academic community (RuleML.org)
  - Related vendor community (Fujitsu, IBM)



# Proposed W3C Standard: Rule Interchange Format

- W3C – via its Semantic Web group
- 1st standards body to address rule standards for interchange
- W3C Rule Interchange Format
  - RIF CORE proposed as a cross-rule-type rule interchange representation; dialects to extend RIF Core for e.g. rule-type-specific rule interchange
  - Consortium of developers and researchers:
    - Rule vendors (Fair Isaac, Ilog, Oracle, ...)
    - Research community (RuleML.org, DERI, REVERSE...)
    - End-users

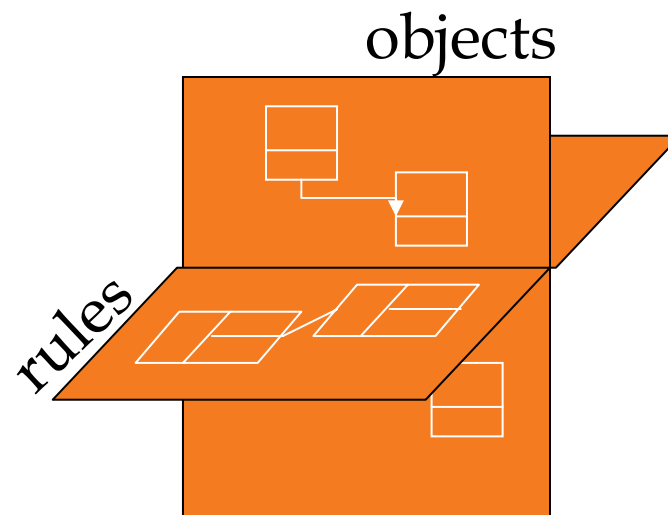




# The Production Rule Representation

# What is OMG PRR?

- Formal model for production rules
- Uses OMG MOF, defined in UML
- Extends UML so production rules are 1st class citizens alongside objects
- Vendor-neutral UML-friendly rule representation
- Intended to be defined via tools, not business users!



# PRR 1.0 Goals

- Metamodel for production rules
- 2 rule “semantics” (types) considered:
  - Forward chaining inference rules (e.g. Rete-model)  
For commonly-used production rule engines
  - Sequentially processed procedural rules (e.g. scripts)  
For tools that separate out simple business logic as non-inference production rules
- Import / export rules between UML tools and BRMSs

# PRR 1.0 issues

- To implement a Production Rule metamodel, need to define a general rule representation / metamodel to allow for other rule types (ECA, constraint, etc)
- PRR must also be compatible with non-UML / non-OMG representations
  - Expression (condition + action) language used outside of UML
  - Object models mapped from UML class diagrams
- PRR therefore defined at 2 levels
  - PRR core: general rule + production rule model
  - PRR OCL: PRR core + extended OCL expression language (non-normative)

# OMG, MDA and PRR

## Model Driven Architecture (MDA)

Computation  
Independent  
Models (CIM)

Business  
Models

BMM

SBVR

Platform  
Independent  
Models (PIM)

UML  
Models

ODM

PRR  
Production Rule  
Representation

BPDM  
+/-  
BPMN

Platform  
Specific  
Models (PSM)

UML  
Models  
e.g. J2EE specific

Vendor-specific rule languages  
PR-RIF

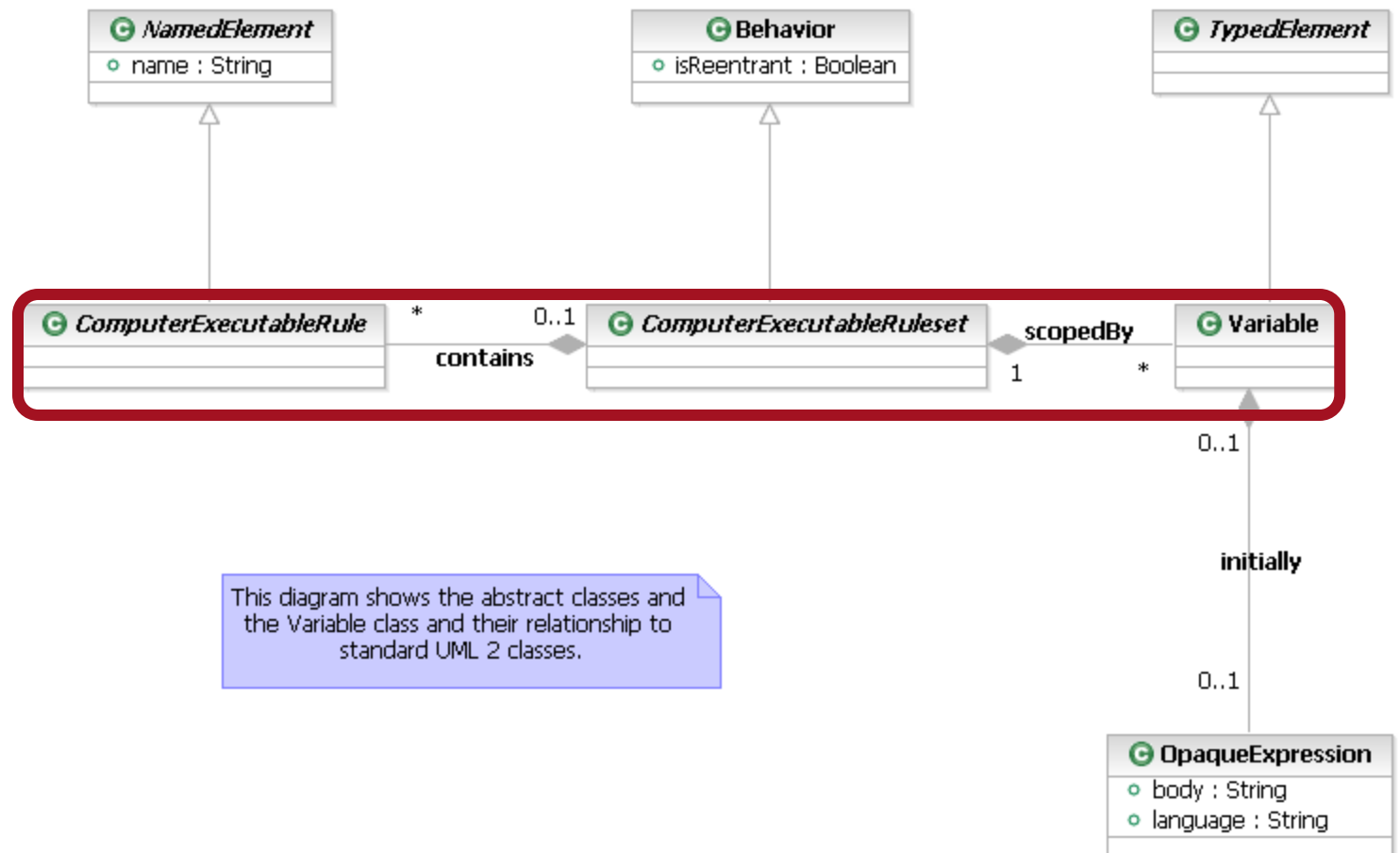
Code / Engines



**PRR 1.0, including changes  
since SanDiego/March07**

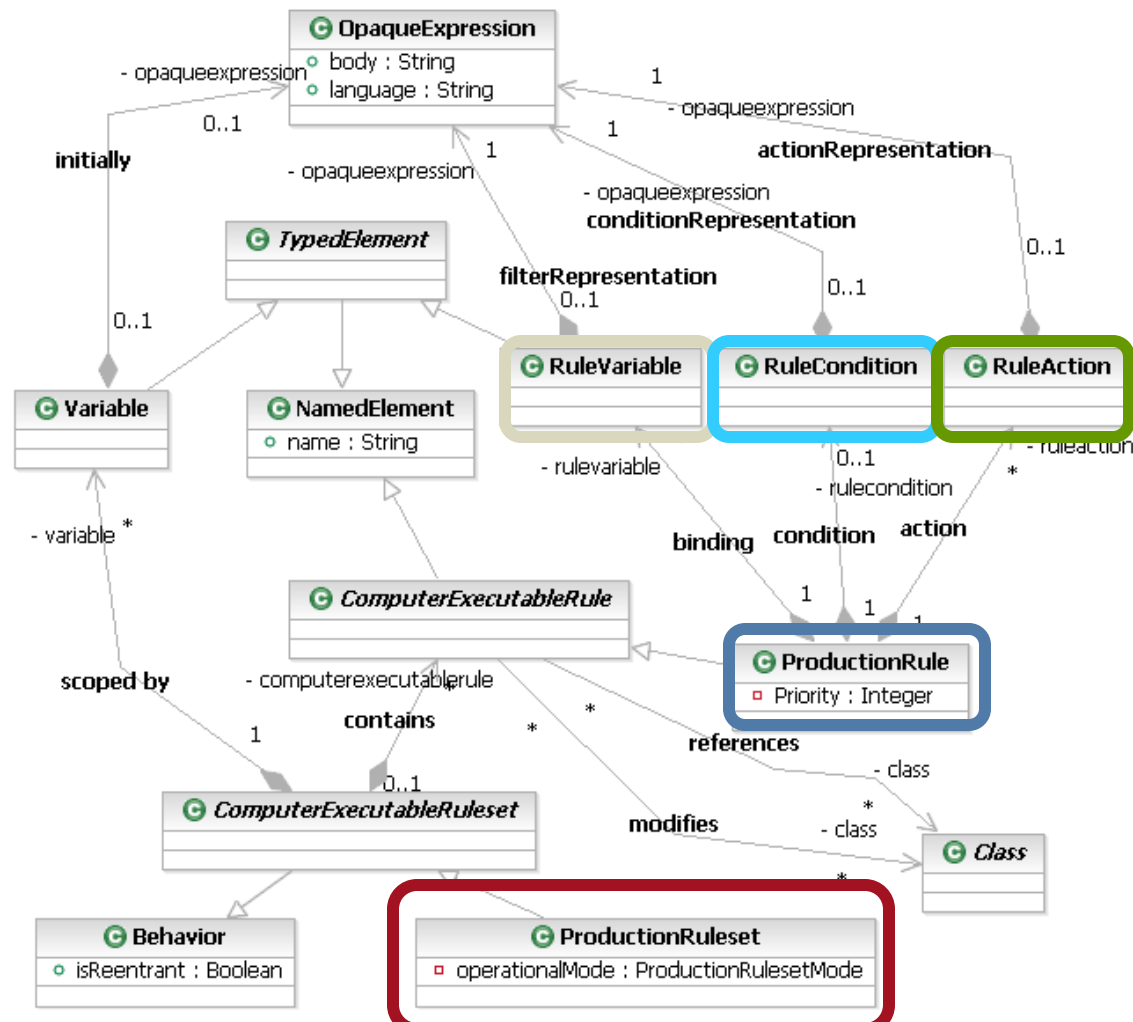
# Generic metamodel supporting PRR

- ComputerExecutableRule, ComputerExecutableRuleset & Variable defined as UML extensions to support production rules (and others as needed)

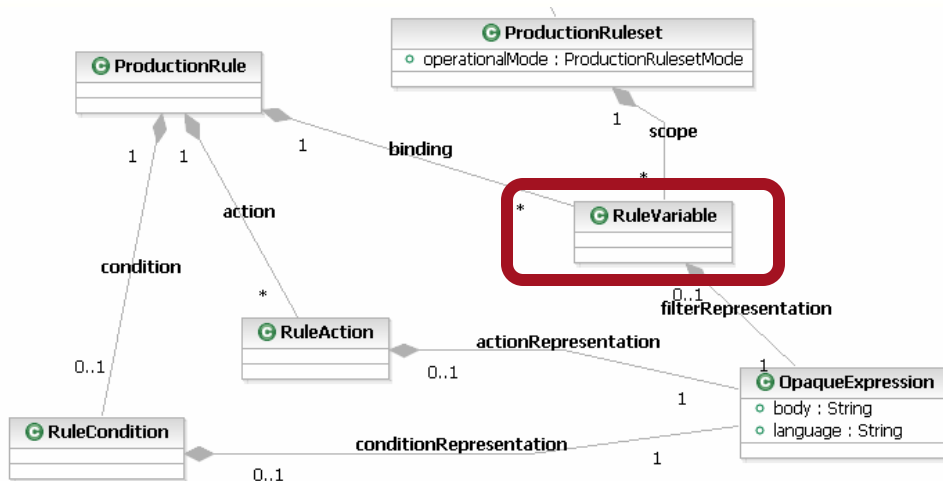


# PRR metamodel from common BRMS uses

- **Ruleset** = collection of Rule
- **Rule** is (for **RuleVariables**) if **<Condition>** then **<Actions>**



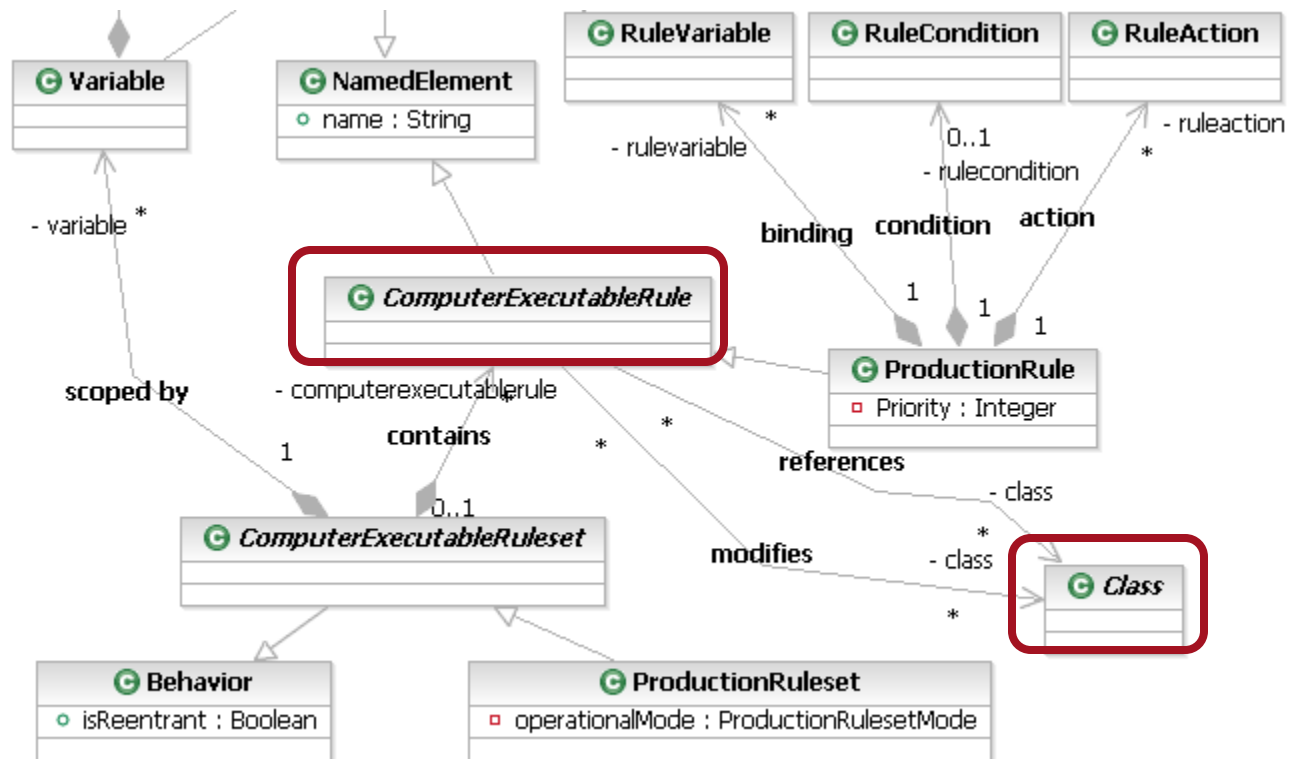
# RuleVariable definition



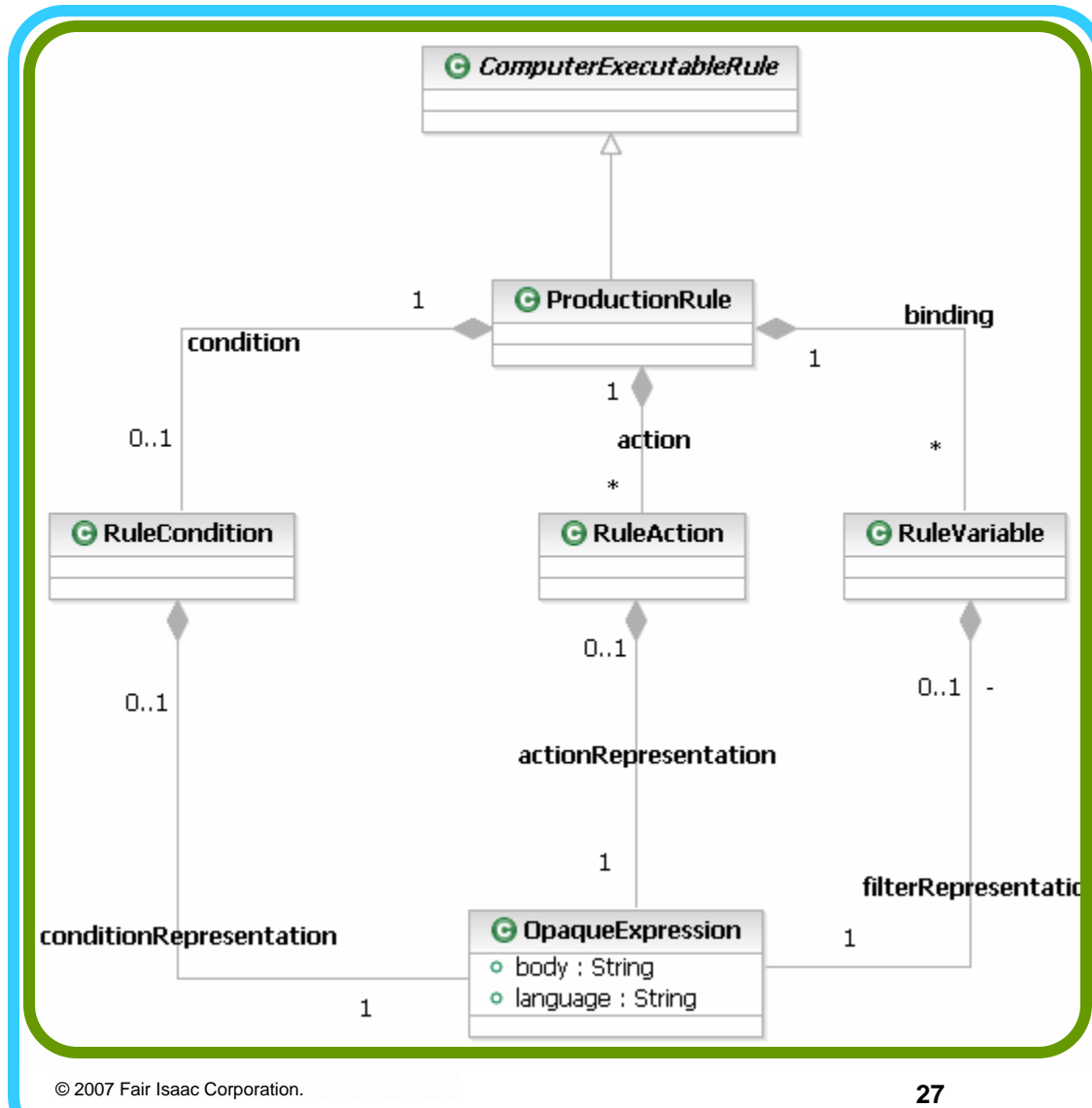
- In production rules, efficient rule execution (e.g. using Rete) requires concept of **RuleVariable**
  - Matches to a range of objects in scope / in a collection with an optional filter (cf unscoped variables / UML invariants)
  - Defined per rule (cf JRules) or per ruleset (cf Blaze Advisor)
  - Allows for powerful rule statements and succinct rule definitions
- Effect:
    - Rule is effectively duplicated at runtime for each combination of RuleVariables (ie tuple)
    - RuleVariables can be changed by rule firings
  - This is called a pattern in Blaze Advisor, a variable in JRules

# References to Class

- ComputerExecutableRule references Class
- ComputerExecutableRule modified Class



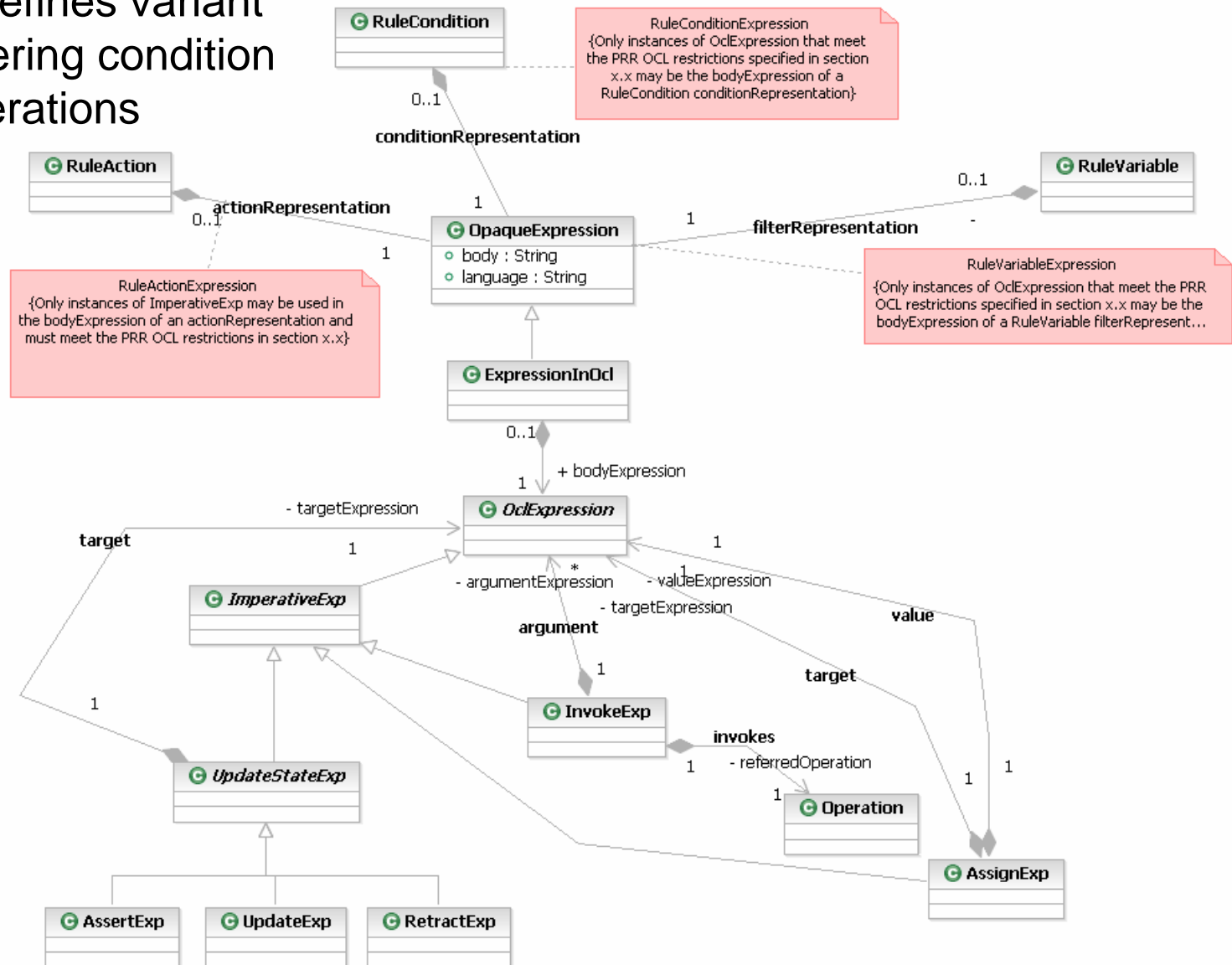
# 2 Different Conformance Levels for PRR



PRR Core	PRR OCL
Generic metamodel	OMG-specific subclass
Expression language a plug-in	OCL2 customized (includes actions)
General production rule model	Per RFP requirements

# PPR OCL extensions – Non Normative

- PRR OCL defines variant of OCL covering condition + action operations



# PRR Summary

- PRR provides a standard metamodel for production rules as used in popular rule engines for business automation
- PRR is constrained to the types of rules executed by rule engines
- Other rule “types” such as decision tables, decision trees etc will be for later versions
- Implications:
  - UML modelling tools can become “rule-aware”
  - UML tools and BRMSs can cooperate on the rule development lifecycle
  - Maturity for BREs – no excuses not to use

# Errata

- Updated model to link ComputerExecutableRule class to Class class
  - modifies Class[\*]  
The classes modified by the rule, e.g. within the OpaqueExpression for associated RuleAction. PRR Core only.
  - referenced class[\*]  
The classes referred to by the rule, e.g. within the OpaqueExpression for associated RuleCondition. PRR Core only
- Made PRR OCL non-normative
- XMI file has been added and submitted
- Added definition of scope:RuleVariable[\*] shown in metamodel but not described:

# Why do the vendors involved think this is of value?

- Set a baseline for PR BRE vendors
- Help define / relate PR to W3C RIF and others like PMML
  - W3C RIF for runtime rule persistence and interchange
  - OMG PRR for design time production rule persistence and interchange
- Continue to challenge UML vendors to keep providing value
  - BPM is here to stay
  - PR BRE are also beyond the hype curve
- PRR allows rules as a 1st class citizen of UML models at every stage of application modeling
  - Recent new entries include Oracle (JESS), Microsoft, TIBCO
  - PR market is \$170M+ (2005, IDC)
  - 1,000's end users
  - 1,000,000's rules in use



# Vote to Vote and FTF

# Summary

- Rule standards are now in development by large community
- Better communication among vendors, academics
- Better understanding of rules and rule uses
- OMG PRR and W3C RIF being kept in sync by members
- Timing:
  - OMG PRR submitted
  - W3C RIF in development / active development of Phase 1 (2007)
  - JSR94 completed (2004)
- Next Steps

# Questions

- Should FTF continue with PRR OCL?
  - Likely to result in far more issues than PRR Core
  - Not clear that this is the most useful place to spend energy
  - Would it be more useful to have an adopted PRR Core that could be:
    - Extended to handle decisions/ruleflow
    - Integrated with Use Cases
    - Integrated with Business Processes
    - Mapped to SBVR
    - Mapped to object models

# References & Q&A

- **OMG** = <http://www.omg.org>
  - **OMG BMI** = <http://bmi.omg.org/>  
(includes the OMG Business Rules Working Group)
  - **OMG PRR details [OMG only]** =  
[http://www.omg.org/techprocess/meetings/schedule/Prod. Rule Representation\\_RFP.html](http://www.omg.org/techprocess/meetings/schedule/Prod.Rule_Representation_RFP.html)
- **W3C** = <http://www.w3.org>
  - **W3C RIF** = <http://www.w3.org/2005/rules/wg>
- **JSR-94** = <http://www.jcp.org/en/jsr/detail?id=94>

# Proposed Charter for PRR FTF

**TC Meeting Date:** 25 June 2007  
**Presenter:** James Taylor  
**Group email:** TBA (eg prrftf@omg.org)  
**WIP page (URL):** TBA

## ● Adopted Specification:

- bmi/2007-03-06.
- {need to check ISO format (see pas/2003-03-01 for template)}

## ● Members:

- List: TBA e.g.
  - Christian de Sainte Marie, ILOG
  - Paul Vincent, TIBCO
  - Donald Chapin, Business Semantics
- Chair
  - James Taylor, Fair Isaac

## ● Deadlines:

- Draft Adopted Specification: {day month, year} for Brussels TC meeting ? June 4, 2007
  - Must fall soon after FTF charter, well before the Comments deadline and at least 21 calendar days before the publication date for the Final Adopted Specification Publication Date.
- Final Adopted Specification Publication: {day month, year} August 6<sup>th</sup> 2007?
  - Must fall at least 28 calendar days before the Comments Due deadline.
- Comments Due: {day month, year} September 24<sup>th</sup> 2007?
  - Must be after the Final Adopted Specification Publication Date.
- Report Deadline: {day month, year} November 16<sup>th</sup>, 2007?
  - This is when the FTF expires. Must be no more than one year after the date of FTF chartering, or alternatively one year from the recommendation deadline of a previous FTF for this specification (if there was one). Recommended to be set to one week after the TC Meeting at which the adoption poll is initiated.
  - ***The FTF Report must adhere to the Three-Week Rule.***



# Roadmap

# PRR version 2 RFP

- Extend PRR to cover the 2 most usual rule representations after if-then rules
  - Decision Tables
  - Decision Trees
- Can run concurrently with PRR 1.0 FTF

# PRR version 3 RFI

- What next for PRR / PIM Rule Models?
- Deeper PR support
  - Ruleflows (UML Activity Diagram profile for ruleset orchestration)?
  - W3C RIF syntax for rule conditions and actions?
  - UML profile for rule representations in class diagrams?
- More rule type support
  - ECA type rules for EDA?
  - Formal logics / ontology / JENA-type rules?
  - Backward chaining?
- Rule management
  - Rule templates?
  - SBVR extensions for behavioral rules?



$$\left\{ \begin{aligned} R_3 &< C_{\text{sur}}(g_1^{-1}(B))n^{(p+q+r)/2} \leq C_{\text{vol}}(g_1^{-1}(B))n^{p+q} \\ \text{ed}(u, F^T) &= \frac{\exp(-nu^*Tu^*/2)}{(2\pi/n)^{(p+q+r)/2} \det Z} \left(1 + \sum_{n=1}^{\infty} O(\ln(u^*n))\right) \text{ed}(u, F^T) \end{aligned} \right.$$

**Thank You**

**Christian de Sainte Marie**  
[csdm@ilog.fr](mailto:csdm@ilog.fr)

**James Taylor**  
[jamestaylor@fairisaac.com](mailto:jamestaylor@fairisaac.com)



Changing the rules of business

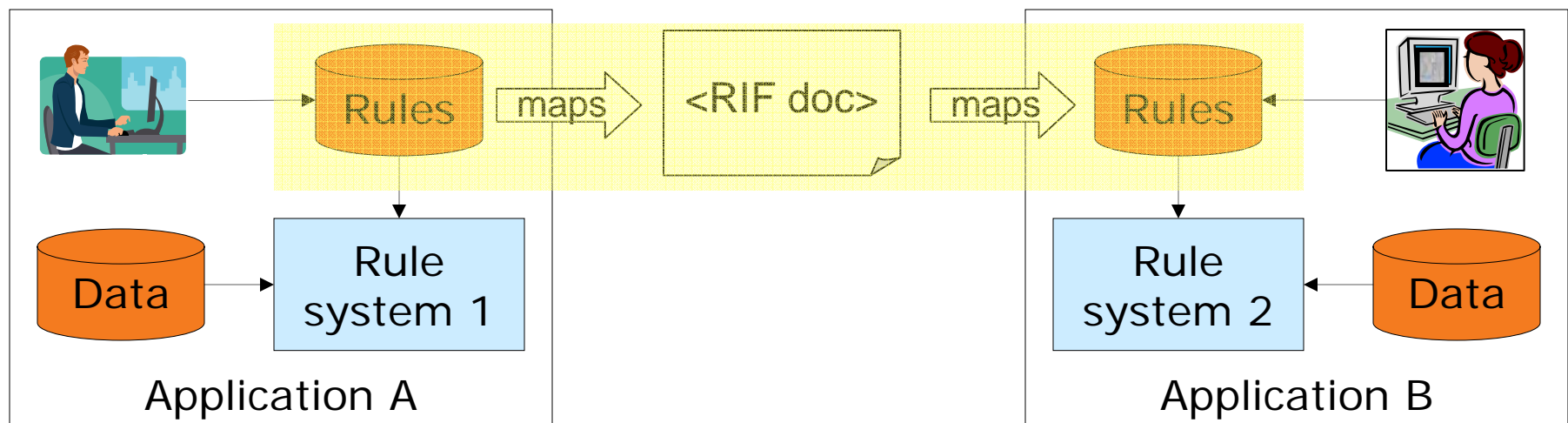




**Appendix:  
PRR Relationship to  
W3C Rule Interchange Format**

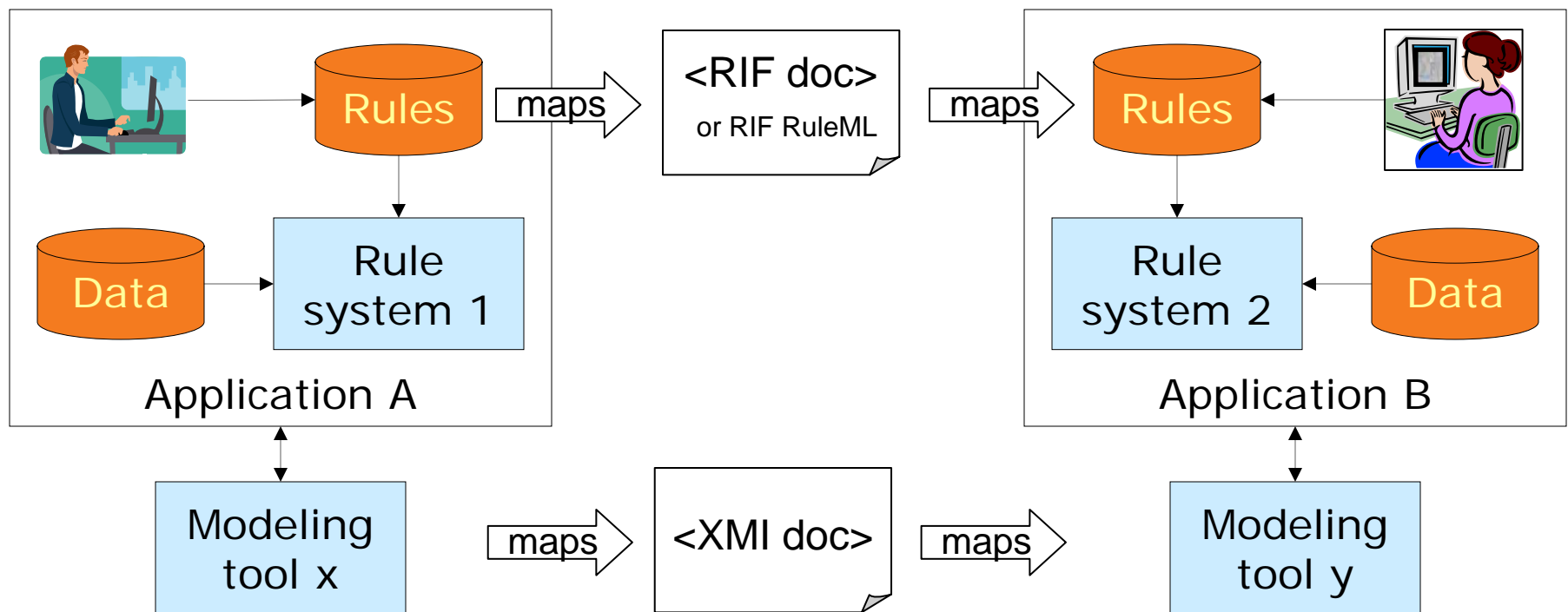
# What is RIF?

- Format for interchanging rules: allowing use across diverse systems
  - Rules written for one application can be published, shared, and re-used in other applications and other rule engines.
  - Part of W3C's wider goal of enabling the sharing of information in forms suited to machine processing
  - Includes “semantic web rule languages”



# Comparing PRR and RIF

- PRR: a rule is (just another) model entity
  - PRR provides a way to include rules into the model of an application at design time
- RIF: a rule is (just another) data item
  - RIF provides a standard means to feed rules into an application (at run time)



# W3C RIF Summary

- RIF provides a standard XML format to interchange rules
- Covers many rule types + possibility of interchange
- Wide scope
- Implications:
  - Intra/inter-organisational interchange of executable (run-time) policies e.g. run-time rule interchange for e-commerce contracts
  - Standardized rules for interchange with XML documents
  - Common rule language for domain-specific vocabularies
  - More opportunities for rule management across multiple rule types
  - Opportunities for governance / regulatory compliance automation across industries
  - Ubiquitous rules

# RIF vs PRR

	<b>PRR</b>	<b>RIF</b>
<i>Focus</i>	Production rule focus (metamodel may be extended to other rule types later)	Horn rule focus in Phase 1 (logic rules); Phase 2 covers PR etc
<i>Primary Output</i>	Metamodel	Language to use as an interchange format
<i>Overlapping contributors</i>	ILOG, Fair Isaac, TIBCO, RuleML, IBM	
<i>Standards body</i>	OMG	W3C
<i>Cross-fertilization</i>	PRR will use RIF condition language in future version	RIF will support PRR