

Personal Agents in the Rule Responder Architecture

Benjamin Craig
Harold Boley

Institute for Information Technology
National Research Council, Canada
Fredericton, NB, Canada

RuleML-2008
Orlando Florida
October 30-31, 2008

Outline

- Rule Responder Overview
- Agents
 - Personal / Organizational / External
- Infrastructure
 - Reaction RuleML Messages
 - Message Performatives
 - Agent Communication Protocols
 - Mule ESB (Communication Middleware)
- Rule Engines (for Realizing Agents)
 - Prova
 - OO jDREW
- Symposium Planner Use Case
 - Query Delegation/Answering
 - Shared Knowledge between Pas
 - Ontology Description
- Future Work and Conclusion

Overview of Rule Responder (I)

- Rule Responder is an experimental multi-agent system for collaborative teams and virtual communities on the Web
- Supports rule-based collaboration between the distributed members of such a virtual organization
- Members of each virtual organization are assisted by semi-automated rule-based agents, which use rules to describe the decision and behavioral logic

Overview of Rule Responder (II)

- Uses languages and engines of the RuleML family for rule serialization, based on logic and XML:
 - Hornlog RuleML: Reasoning
 - Reaction RuleML: Interaction
- Implemented on top of a Mule-based Service Oriented Architecture (SOA)

Personal Agents

- A personal agent assists a single person of an organization, (semi-autonomously) acting on his/her behalf
- It contains a FOAF*-like **fact** profile plus FOAF-extending **rules** to encode selected knowledge of its human owner

* The Friend of a Friend (FOAF) project: <http://www.foaf-project.org>

Organizational Agents

- An organizational agent represents goals and strategies shared by each member of the organization
- It contains rule sets that describe the policies, regulations, opportunities, etc. of its organization

External Agents

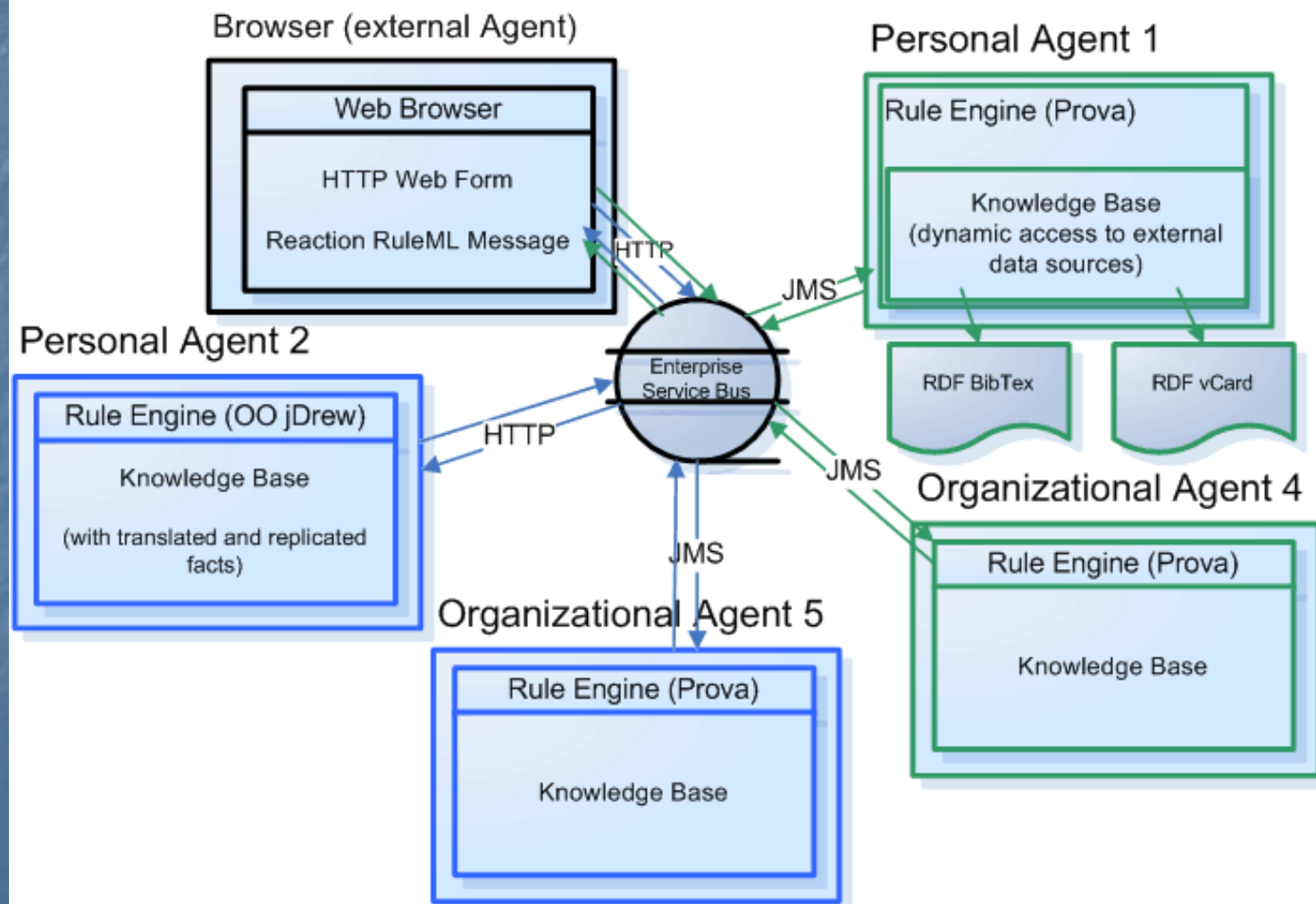
- External agents exchange messages with (the public interface of) organizational agents, asking queries, receiving answers, or interchanging complete rule sets
- End users, as external agents, employ a Web (HTTP) interface of Rule Responder (currently an API-like browser interface)
- Support for simultaneous external agents:
Currently, end users (B2C)
Ultimately, other organizations (B2B)

Rule Responder as a Multi-Agent Infrastructure

- Realizes a System of OAs, PAs, and EAs
- Built on the Mule ESB
- The OAs and PAs are realized each with an instance of a Rule Engine
- Combines the ideas of multi-agent systems, distributed rule management systems, as well as service-oriented and event-driven architectures

Infrastructure - Overview

Use Case 4 Use Case 5



Translation Between PAs' Native Languages and OA's Interchange Language

- Each rule engine can use its own rule language
- Agents require an interchange language so that they can understand each other
- Rule Responder uses Reaction RuleML as its interchange language
- Translation is done with an XSLT stylesheet

Reaction RuleML

- Reaction RuleML is a branch of the RuleML family that supports actions and events
- When two agents need to communicate, each others' Reaction RuleML messages are sent through the ESB
- Carries RuleML queries, answers, and rule bases to/from agents

Example Reaction RuleML Message

- `<RuleML xmlns="http://www.ruleml.org/0.91/xsd"`
- `xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"`
- `xsi:schemaLocation="http://www.ruleml.org/0.91/xsd`
- `http://ibis.in.tum.de/research/ReactionRuleML/0.2/rr.xsd"`
- `xmlns:ruleml2007="http://ibis.in.tum.de/projects/paw#">`
- `<Message mode="outbound" directive="query-sync">`
- `<oid> <Ind>RuleML-2008</Ind> </oid>`
- `<protocol> <Ind>esb</Ind> </protocol>`
- `<sender> <Ind>User</Ind> </sender>`
- `<content>`
- `<Atom>`
- `<Rel>getContact</Rel>`
- `<Ind>ruleml2008_PanelChair</Ind>`
- `<Ind>update</Ind>`
- `<Var>Contact</Var>`
- `</Atom>`
- `</content>`
- `</Message>`
- `</RuleML>`

Message Performatives

- The attribute `directive="..."` corresponds to the pragmatic performative
- Specify message exchange/interaction protocols
- Rule Responder Performatives
 - In tradition of KQML and FIPA-ACL
 - Currently implemented: Query and Answer
 - Retract and Update in collaboration with RIF-PRD

Agent Communication Protocols

WSDL-like protocols

- In-Only
 - Message is sent to agent₁ from agent₂; then agent₁ executes performative
- Request-Response
 - Performs above in-only; then agent₁ sends response back to agent₂
- Request-Response-Acknowledge
 - Does Request-Response; then agent₂ sends a response back to agent₁
- Workflows
 - Generalizes the above protocols to allow arbitrary compositions of agent messages

Communication Middleware

- **Mule Enterprise Service Bus (ESB)**
 - Mule* is used to create communication end points at each personal and organizational agent of Rule Responder
 - Mule supports various transport protocols (e.g. HTTP, JMS, SOAP)
 - Rule Responder currently uses HTTP and JMS as transport protocols

* **Mule – The open source SOA infrastructure:**
<http://mulesource.com>

Rule Engines

- Prova: Prolog + Java
- OO jDREW: Object Oriented
java Deductive Reasoning Engine for the Web

Prova

- Prova is mainly used to realize the organizational agents of Rule Responder
- It implements Reaction RuleML for agent interaction (event-condition-action rules)

OO jDREW

- OO jDREW is used to realize the personal agents of Rule Responder
- It implements Hornlog RuleML for agent reasoning (Horn logic rules)
- Supports rules in two formats:
 - POSL: Positional Slotted presentation syntax
 - RuleML: XML interchange syntax
(can be generated from POSL)

Use Case: Symposium Planner

- RuleML-20xy Symposia
 - An organizational agent acts as the single point of entry to **assist** with the symposium:
 - Currently, query answering about the symposium
 - Ultimately, preparing and running the symposium
 - Personal agents have supported symposium chairs since 2007 (deployed as Q&A in 2008)
 - General Chair, Program Chair, Panel Chair, Publicity Chair, etc.

Online Use Case Demo

- Rule Responder:
<http://responder.ruleml.org>
- RuleML-2007/RuleML-2008 Symposia:
<http://ibis.in.tum.de/projects/paw/ruleml-2007>
<http://ibis.in.tum.de/projects/paw/ruleml-2008>
- Personal agents:
Supporting all Chairs
- Organizational agent:
Supporting Symposium as a whole

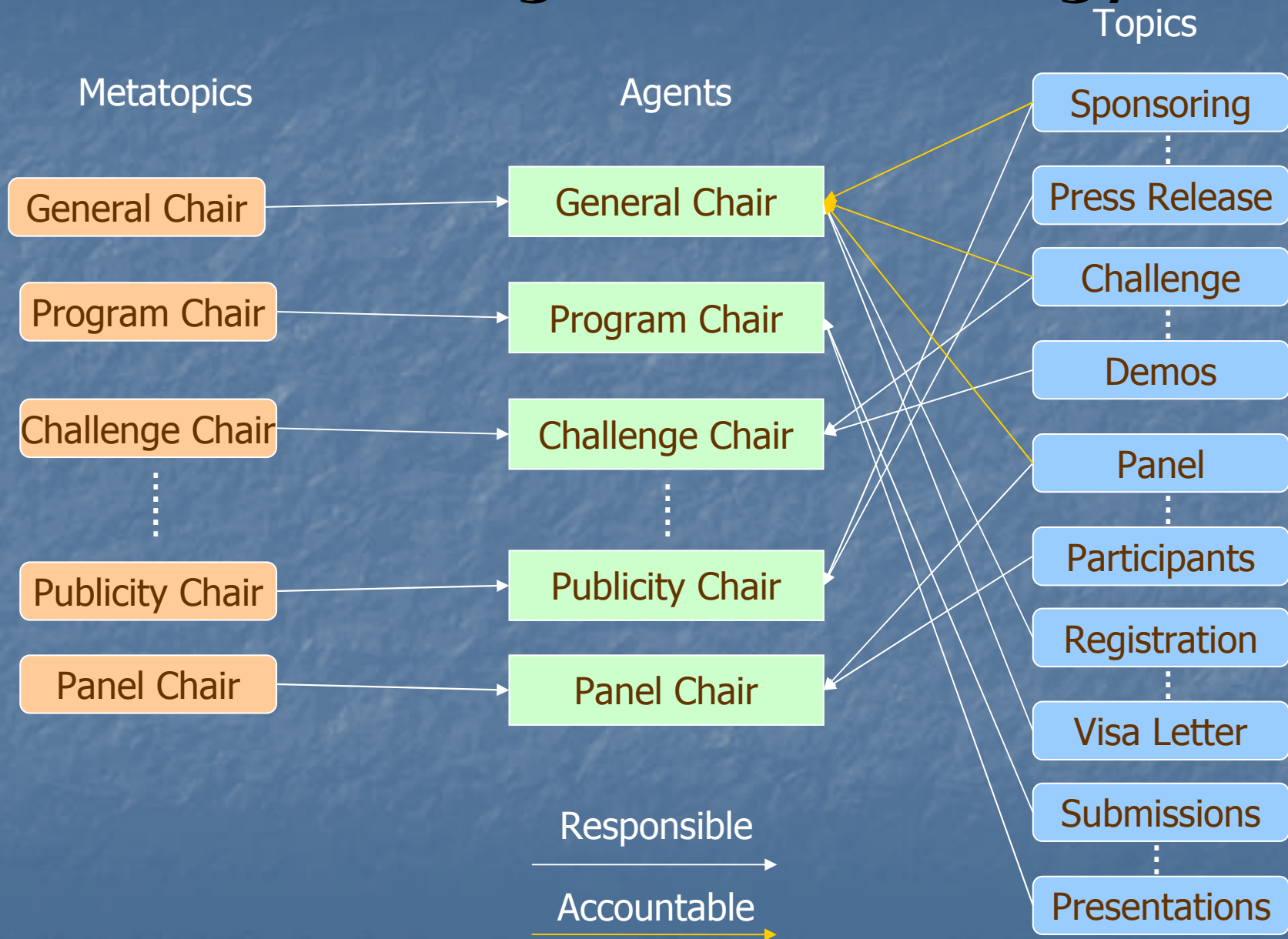


Online

Query Delegation

- Query delegation to personal agents is done by the organizational agent
- Tasks for the symposium organization are managed via a role assignment matrix
- Is defined here by an OWL Lite Ontology (alternatives: RDFS, RuleML, ...)
- Assigns (meta)topics to agents within the virtual organization: ... *see next slide* ...

Role Assignment Ontology



Multiple Query Answers by PAs

- Some queries have more than one answer
- The PA will send the answers one at a time to the OA (interleaved backtracking and transmission)
- When no more answers are computed, an end-of-transmission message is sent back

Shared Knowledge Between Personal Agents

- Rules can be shared among personal agents
- Rules that apply to all PAs can be moved up to the OA level
- ... *see next slide* ...

Organizational Symposium Agent Knowledge Base

% Sample Prova-like rule (in POSL syntax)
stored in the OA:

```
getContact(?topic, ?request, ?contact) :-
```

% Uses the topic and request to delegate the
following query to appropriate PA

```
person(  
    ?contact, ?role, ?title, ?email, ?telephone).
```

Personal Panel Chair Agent Knowledge Base

% Sample FOAF-like facts used by the OA rule:

% Example fact stored in the Panel Chair's PA

```
person(John, PanelChair, PHD,  
      john@email.com, 1-555-555-5555)
```

% Example fact stored in the Publicity Chair's PA


```
person(Tracy, PublicityChair, PHD,  
      tracy@email.com, 1-444-444-4444)
```

% Sample query in RuleML syntax:

... see next slide ...

Sample Message to Organizational Agent

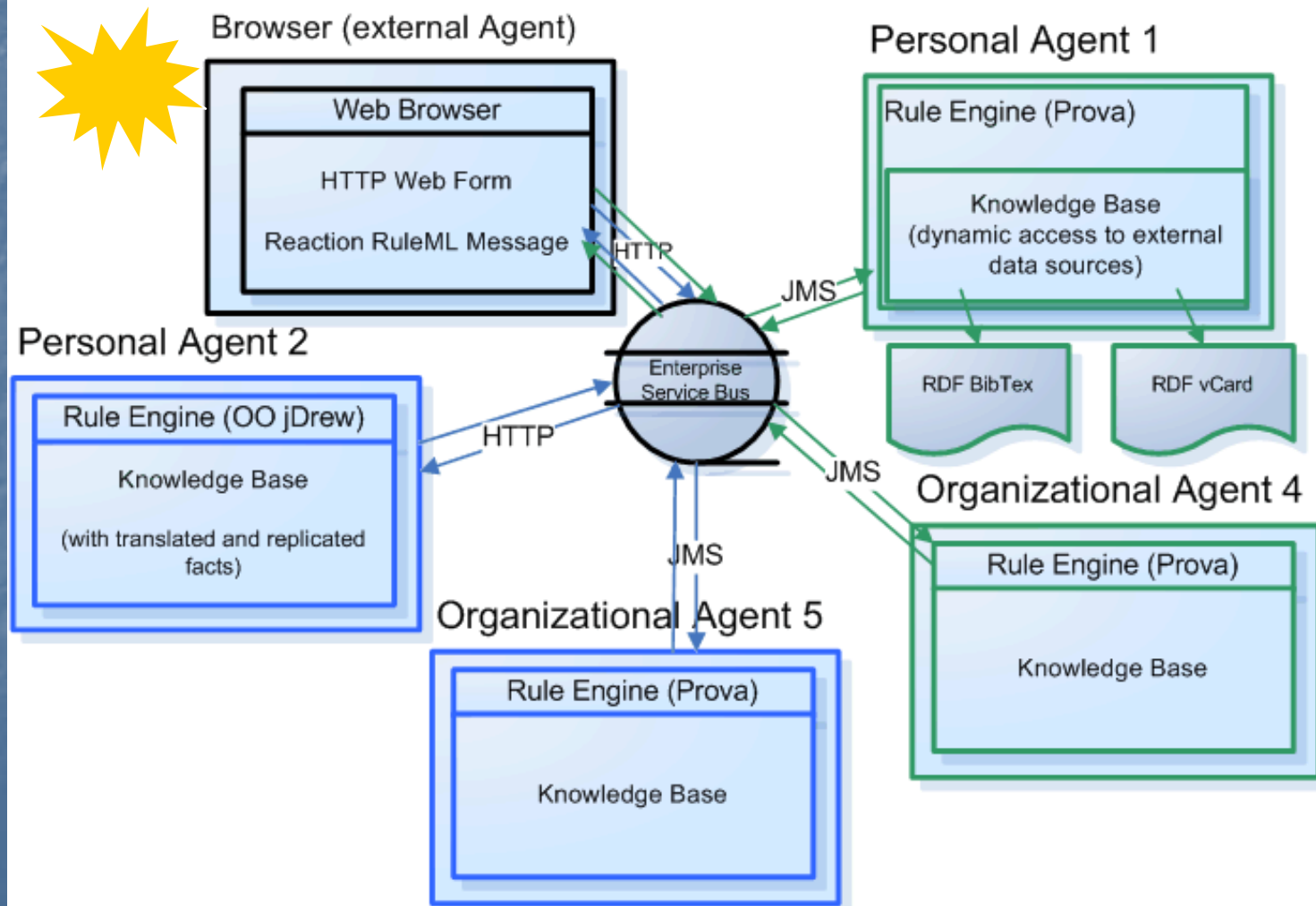
```
<RuleML xmlns="http://www.ruleml.org/0.91/xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ruleml.org/0.91/xsd
http://ibis.in.tum.de/research/ReactionRuleML/0.2/rr.xsd"
xmlns:ruleml2007="http://ibis.in.tum.de/projects/paw#">
  <Message mode="outbound" directive="query-sync">
    <oid>
      <Ind>RuleML-2008</Ind>
    </oid>
    <protocol>
      <Ind>esb</Ind>
    </protocol>
    <sender>
      <Ind>User</Ind>
    </sender>
    <content>
      <Atom>
        <Rel>getContact</Rel>
        <Ind>ruleml2008_PanelChair</Ind>
        <Ind>update</Ind>
        <Var>Contact</Var>
      </Atom>
    </content>
  </Message>
</RuleML>
```



<http://www.ruleml.org/RuleML-2008/RuleResponder>
Query Selection: Panel Chair Contact

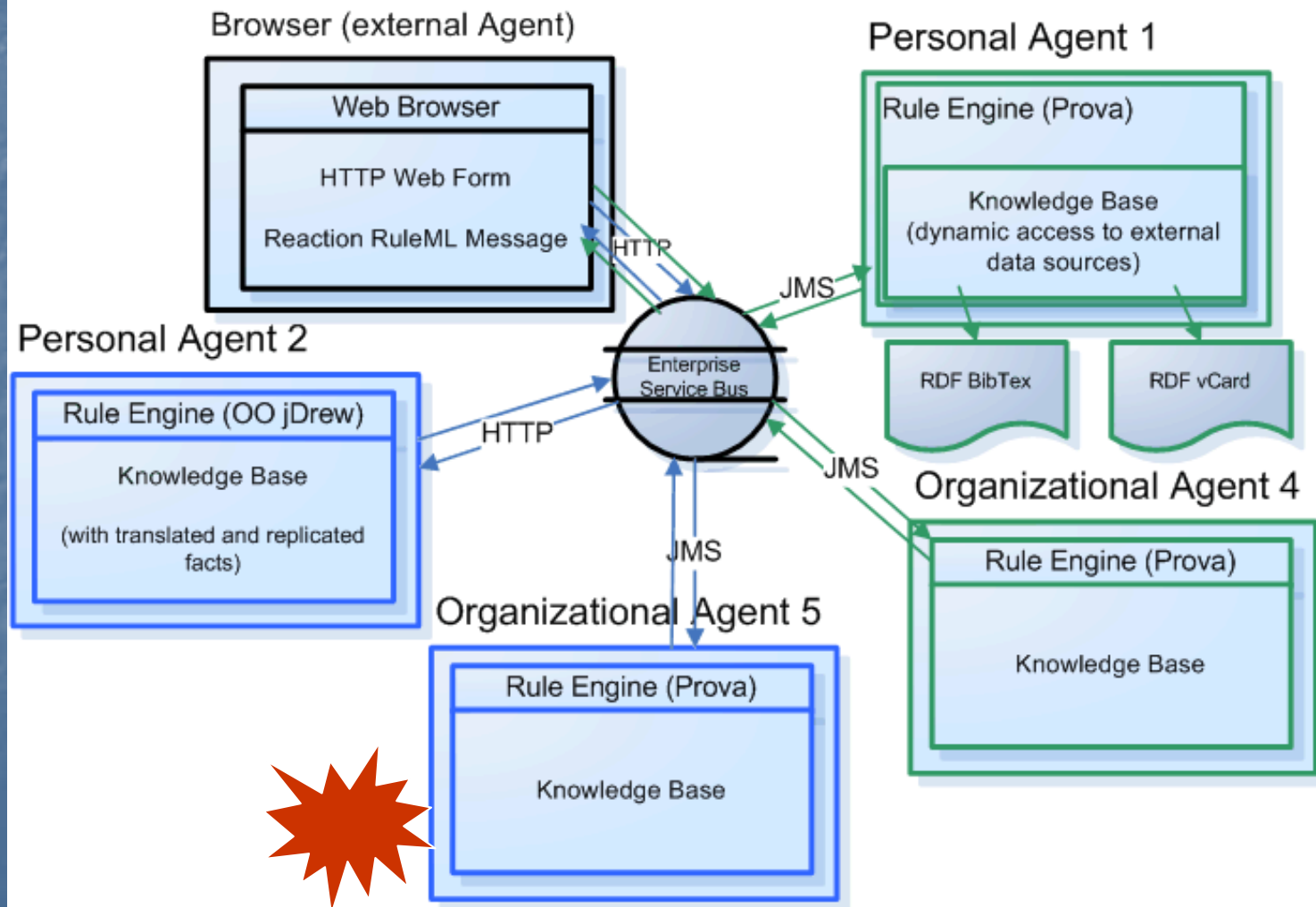
Architecture - Execution

Use Case 4 Use Case 5



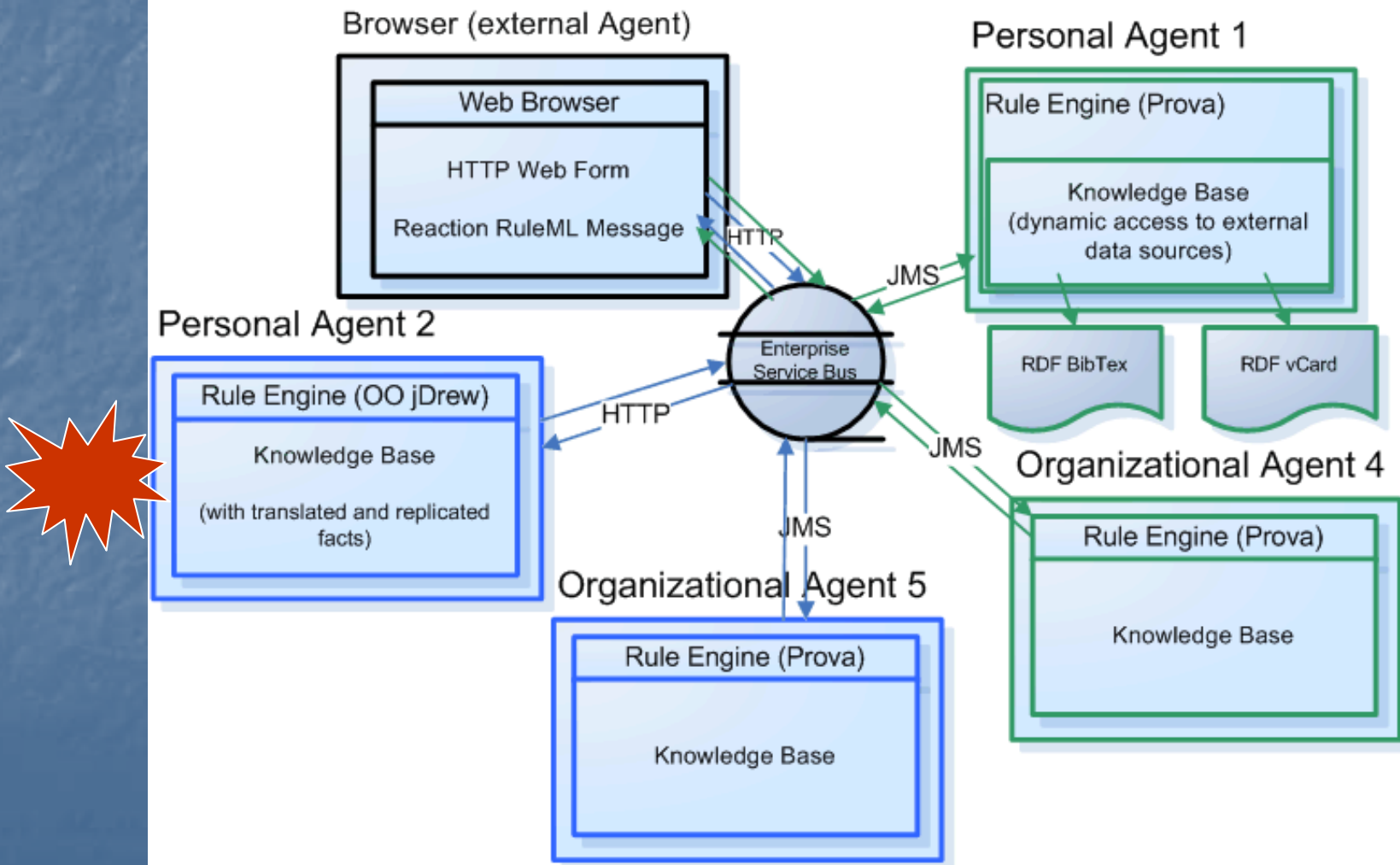
Architecture - Execution

Use Case 4 Use Case 5



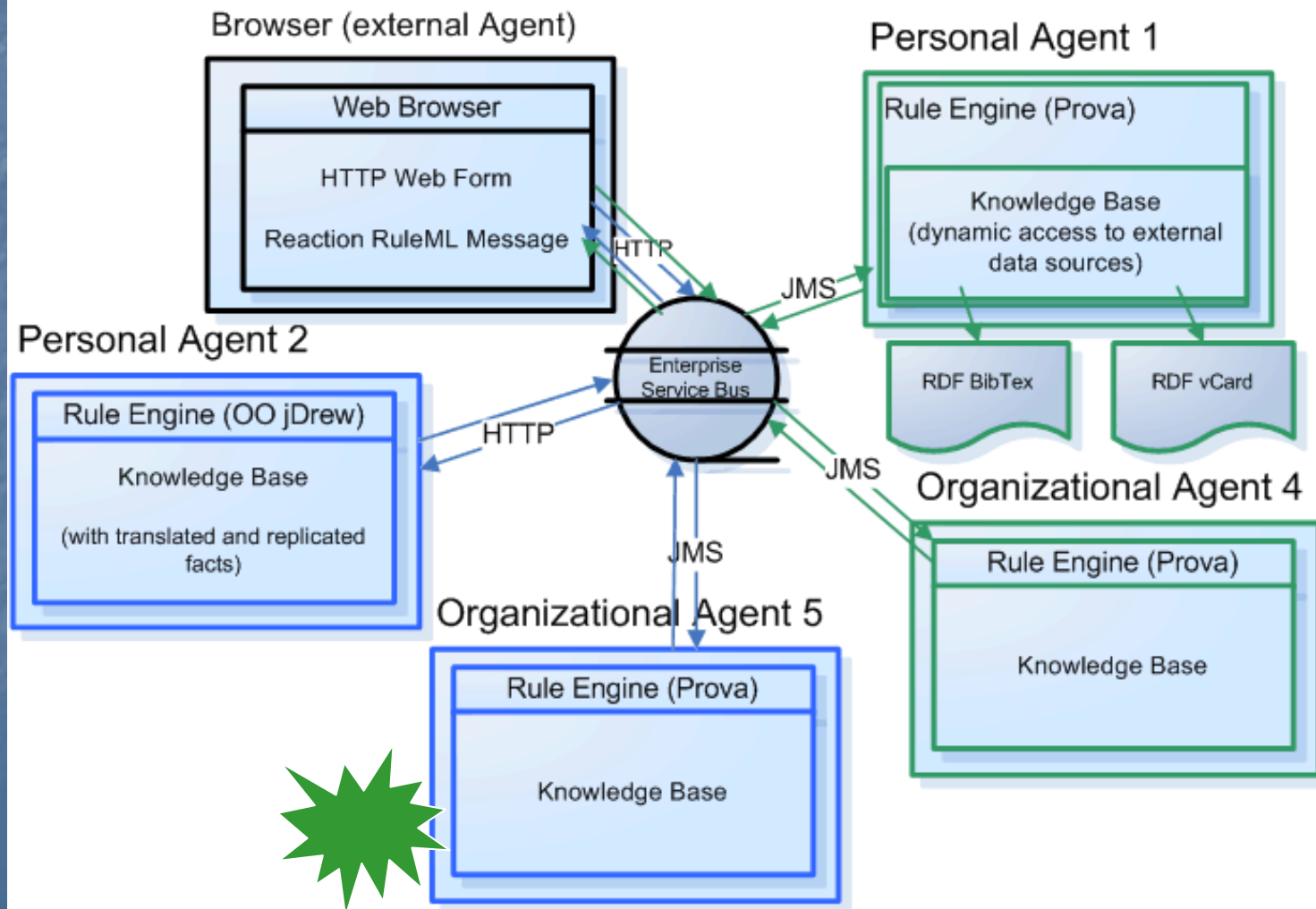
Architecture - Execution

Use Case 4 Use Case 5



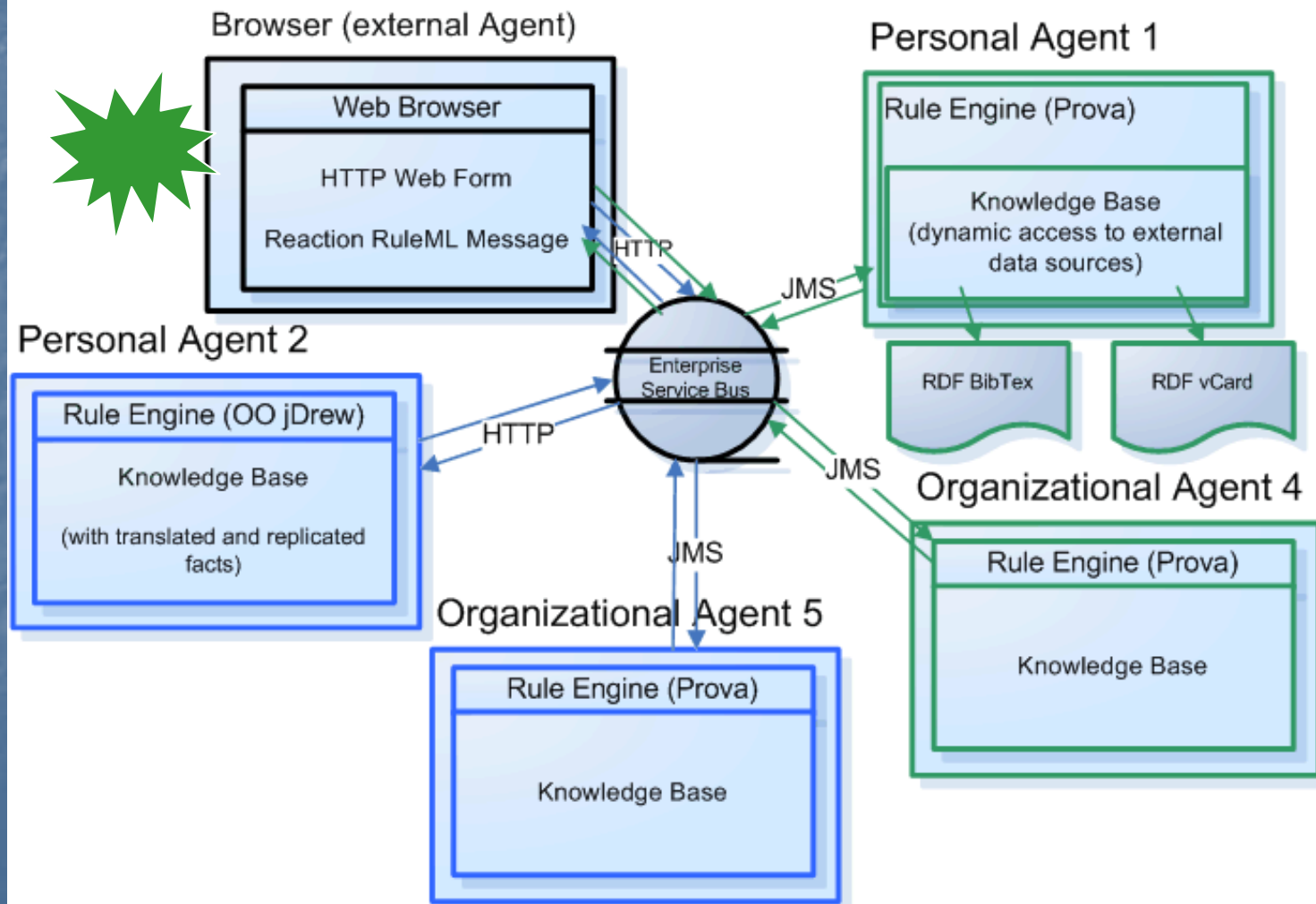
Architecture - Execution

Use Case 4 Use Case 5



Architecture - Execution

Use Case 4 Use Case 5



Latest News

Updated - June 20, 2008

Calls for lightning talks, highlight talks and fast abstracts opened.

[read more news](#)

The Symposium

- [Who Will Attend](#)
- [Highlights](#)
- [Keynote Speakers](#)
- [Venue](#)
- [Business Rule Forum](#)

Authors

- [Objectives](#)
- [Topics](#)
- [Open Calls](#)
- [RuleML-2008 Challenge](#)
- [Submission Guidelines](#)
- [Call for Papers \(PDF\)](#)
- [Important Dates](#)

Organisation

- [Organising Committee](#)
- [Program Committee](#)
- [Sponsorship](#)

Rule Responder: RuleML-2008 Experimental Demo

DEMO ARCHITECTURE DESCRIPTION INTENTION ONTOLOGY

Use this text form to send a query in [Reaction RuleML](#) format to the [RuleML-2008 Organizational Agent](#):

```
xmlns:ruleml2007="http://ibis.in.tum.de/projects/paw#">
  <Message mode="outbound" directive="query-sync">
    <oid>
      <Ind>RuleML-2008</Ind>
    </oid>
    <protocol>
      <Ind>esb</Ind>
    </protocol>
    <sender>
      <Ind>User</Ind>
    </sender>
    <content>
      <Atom>
        <Rel>getContact</Rel>
        <Ind>ruleml2008\_PublicityChair</Ind>
        <Ind>update</Ind>
        <Var>Contact</Var>
      </Atom>
    </content>
  </Message>
</RuleML>
```

Send Message

Query Selection

Use the drop down box to select which query you want to send to the [RuleML-2008 Organizational Agent](#):

Publicity Chair Contact

```

<?xml version="1.0" encoding="UTF-8" ?>
- <RuleML xmlns="http://www.ruleml.org/0.91/xsd" xmlns:xsi="http://www.w3.org/2001/XMLSchema-insta
- <Message mode="outbound" directive="answer">
  + <oid>
  + <protocol>
  + <sender>
  - <content>
    - <Atom>
      <Rel>person</Rel>
      + <Expr>
      + <Expr>
      + <Expr>
      + <Expr>
      - <Expr>
        <Fun>foafname</Fun>
        - <Expr>
          <Fun>firstName</Fun>
          <Var>N@@82_Title</Var>
          <Ind>tracy</Ind>
          </Expr>
        - <Expr>
          <Fun>lastName</Fun>
          <Var>N@@82_Title</Var>
          <Ind>bost</Ind>
          </Expr>
        </Expr>
      </Atom>
    </content>
  </Message>
- <Message mode="outbound" directive="answer">
  + <oid>
  + <protocol>
  + <sender>
  - <content>
    - <Atom>
      <Rel>person</Rel>
      + <Expr>
      + <Expr>
      + <Expr>
      + <Expr>
      - <Expr>
        <Fun>foafname</Fun>
        - <Expr>
          <Fun>firstName</Fun>
          <Var>N@@82_Title</Var>
          <Ind>matthias</Ind>
          </Expr>
        - <Expr>
          <Fun>lastName</Fun>
          <Var>N@@82_Title</Var>
          <Ind>nickles</Ind>
          </Expr>
        </Expr>
      </Atom>
    </content>
  </Message>
</RuleML>

```

Sample Message to Publicity Chair Agent (I)

```
■ <content>
■   <Atom>
■     <Rel>sponsor</Rel>
■     <Expr>
■       <Fun>contact</Fun>
■       <Ind>Mark</Ind>
■       <Ind>JBoss</Ind>
■     </Expr>
■     <Ind type="integer">500</Ind>
■     <Expr>
■       <Fun>results</Fun>
■       <Var>Level</Var>
■       <Var>Benefits</Var>
■       <Var>DeadlineResults</Var>
■     </Expr>
■     <Expr>
■       <Fun>performative</Fun>
■       <Var>Action</Var>
■     </Expr>
■   </Atom>
■ </content>
```



Online

<http://www.ruleml.org/RuleML-2008/RuleResponder>
Query Selection: Publicity Chair Sponsoring

```
- <content>
- <Atom>
  <Rel>sponsor</Rel>
  - <Expr>
    <Fun>contact</Fun>
    <Ind>Mark</Ind>
    <Ind>JBoss</Ind>
  </Expr>
  <Ind type="integer">500</Ind>
  - <Expr>
    <Fun>results</Fun>
    <Ind>bronze</Ind>
  - <Expr>
    <Fun>benefits</Fun>
  - <Expr>
    <Fun>logo</Fun>
  - <Expr>
    <Fun>on</Fun>
    <Ind>site</Ind>
  </Expr>
  </Expr>
  - <Expr>
    <Fun>acknowledgement</Fun>
  - <Expr>
    <Fun>in</Fun>
    <Ind>proceedings</Ind>
  </Expr>
  </Expr>
  - <Expr>
    <Fun>onGoing</Fun>
    <Ind>deadline</Ind>
  </Expr>
  </Expr>
  - <Expr>
    <Fun>performative</Fun>
    <Ind>email</Ind>
  </Expr>
</Atom>
</content>
```

Sample Message to Publicity Chair Agent (II)

```
■ <content>
■   <Atom>
■     <Rel>sponsor</Rel>
■     <Expr>
■       <Fun>contact</Fun>
■       <Ind>Mary</Ind>
■       <Ind>Super</Ind>
■     </Expr>
■     <Ind type="integer">5000</Ind>
■     <Expr>
■       <Fun>results</Fun>
■       <Var>Level</Var>
■       <Var>Benefits</Var>
■       <Var>DeadlineResults</Var>
■     </Expr>
■     <Expr>
■       <Fun>performative</Fun>
■       <Var>Action</Var>
■     </Expr>
■   </Atom>
■ </content>
```



Online

<http://www.ruleml.org/RuleML-2008/RuleResponder>
Query Selection: Publicity Chair Sponsoring (edit)

```

- <content>
- <Atom>
  <Rel>sponsor</Rel>
+ <Expr>
  <Ind type="integer">5000</Ind>
- <Expr>
  <Fun>results</Fun>
  <Ind>platinum</Ind>
- <Expr>
  <Fun>benefits</Fun>
+ <Expr>
- <Expr>
  <Fun>acknowledgement</Fun>
- <Expr>
  <Fun>in</Fun>
  <Ind>proceedings</Ind>
</Expr>
- <Expr>
  <Fun>option</Fun>
- <Expr>
  <Fun>sponsor</Fun>
  <Ind>student</Ind>
</Expr>
- <Expr>
  <Fun>free</Fun>
  <Var>Benefits</Var>
  <Ind>registration</Ind>
- <Expr>
  <Fun>amount</Fun>
  <Ind>2</Ind>
</Expr>
- <Expr>
  <Fun>logo</Fun>

```

```

- <Expr>
  <Fun>in</Fun>
  <Ind>proceedings</Ind>
</Expr>
- <Expr>
  <Fun>option</Fun>
  <Var>Benefits</Var>
  <Ind>demo</Ind>
</Expr>
- <Expr>
  <Fun>name</Fun>
- <Expr>
  <Fun>all</Fun>
- <Expr>
  <Fun>advance</Fun>
  <Ind>publicity</Ind>
</Expr>
- <Expr>
  <Fun>distribution</Fun>
- <Expr>
  <Fun>brochures</Fun>
- <Expr>
  <Fun>all</Fun>
  <Ind>participants</Ind>
</Expr>
- <Expr>
  <Fun>onGoing</Fun>
  <Ind>deadline</Ind>
</Expr>
- <Expr>
  <Fun>performative</Fun>
  <Ind>phone</Ind>
</Atom>
</content>

```

Conclusion (I)

- Rule Responder was implemented & tested for various use cases (<http://responder.ruleml.org>) and deployed for RuleML-2008 [Q&A](#)
- Its organizational agents delegate external queries to topic-assigned personal agents
- It couples rule engines [OO jDREW](#) & [Prova](#) via Mule middleware and [RuleML 0.91](#) XML interchange format

Conclusion (II)

- Without a Reaction Rule Dialect, RIF could not be used for behavioral Responder logic
- Current system is reusable on all levels: Symposium Planner, Rule Responder, POSL, RuleML, OO jDREW, Prova, Mule
- RuleML Techn. Group with [Adrian Paschke](#), [Alexander Kozlenkov](#) and [Nick Bassiliades](#): Looking for more 'partner engines' (mainly [Flora-2](#)) for use case, e.g. on [RuleML FOAF](#)

Future Work (I)

- Communication Between Personal Agent and Expert Owner
 - The PA at some point may need to interact with its expert owner
 - The formal interaction between PAs and their owners is email (SMTP)
 - The interaction language of these emails is Reaction RuleML
- Query Decomposition
 - Each premise of a rule can be delegated to different PAs, followed by Integration

Future Work (II)

- Centralized, Hierarchical (Distributed), and Networked (Distributed) Query Answering
- Centralized and Distributed Knowledge Maintenance
 - How to keep your rules updated
- Distributed: Fault Tolerance
 - Alternative agents when an agent stops working
- Communication Overhead vs. Centralized Processing