

RuleML Query Answering with Personal OO jDREW Agents in Rule Responder

Benjamin Craig

Harold Boley

Fredericton, NB

National Research Council - IIT

May 15, 2008

Outline

- Rule Responder Overview
- Agents
 - Personal / Organizational / External
- Rule Engines (for Realizing Agents)
 - Prova
 - OO jDREW
- Communication Middleware (for Connecting Agents)
 - Mule ESB
 - Reaction RuleML Messages
- Symposium Planner Use Case
 - Online Demo
- 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 the virtual organization are assisted by semi-automated rule-based agents, which use rules to describe the behavioral and decision 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 some of the knowledge of its human owner

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

Organizational Agents

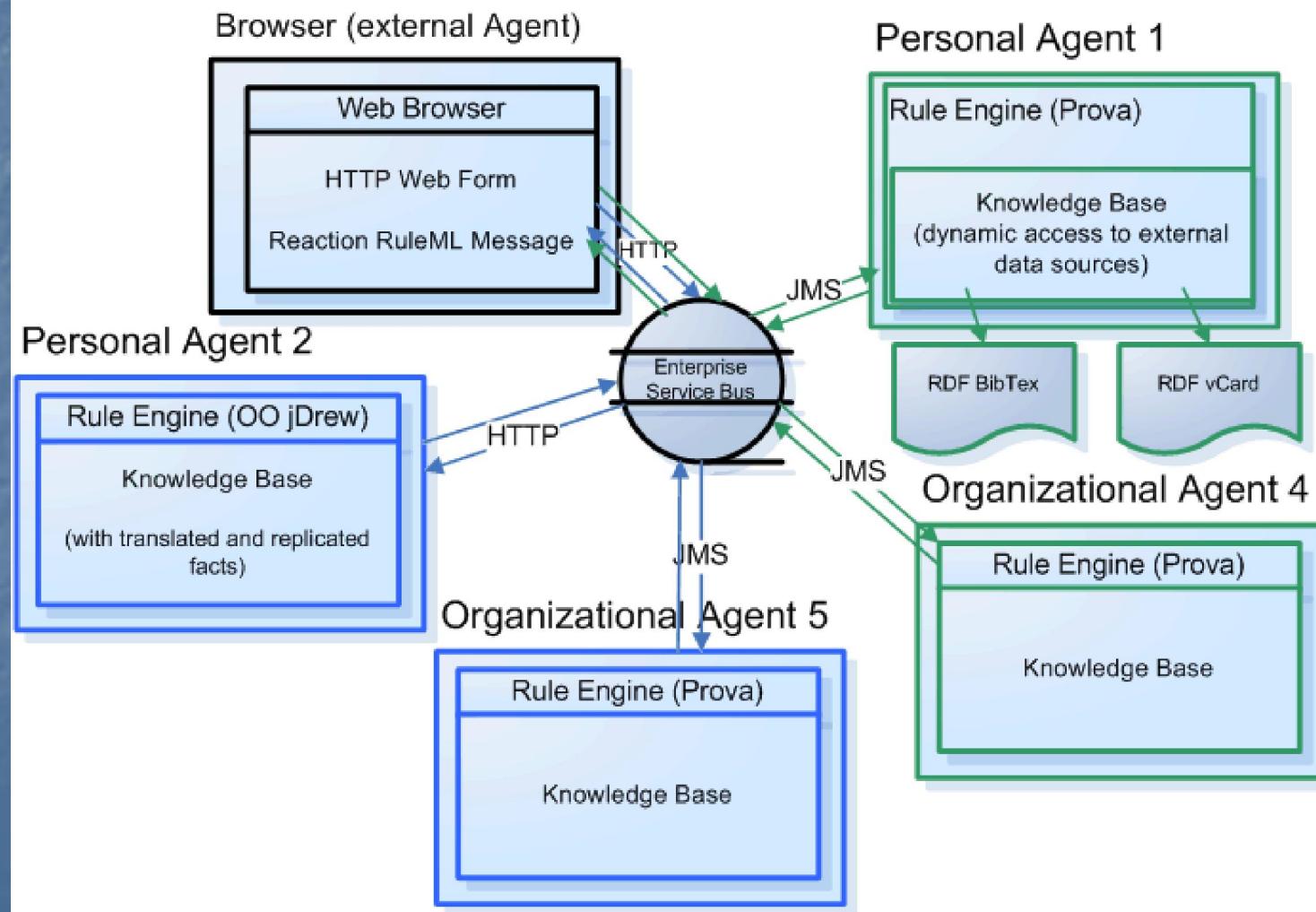
- An organizational agent represents goals and strategies shared by each member of the organization
- It contains rule sets that describe the policies, regulations, opportunities, etc. of its organization

External Agents

- External agents communicate with the public interface of organizational agents, exchanging messages that transport queries, answers, or complete rule sets
- End users, as external agents, employ a Web (HTTP) interface of Rule Responder (currently an API-like browser interface)
- Support for multiple external agents (end users) at the same time

Architecture - Overview

Use Case 4 Use Case 5



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)

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>

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

Use Case: Symposium Planner

- RuleML-20xy Symposium
 - An organizational agent acts as the single point of entry to the symposium
 - Assists with planning, preparing, and running the symposium
 - Personal agents support chairs of the symposium
 - Program Chair, Panel Chair, Publicity Chair, General 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 Panel and Publicity Chairs
- „ Organizational agent:
Supporting Symposium as a whole

Online

Personal Panel Chair Agent Knowledge Base

% Sample FOAF-extending rule in POSL syntax:

```
person(?person, ?role, ?title, ?email, ?telephone) :-  
    mailphone(?person, ?email, ?telephone),  
    role(?person, ?role),  
    title(?person, ?title).
```

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

```
mailphone(John, john@email.com, 1-555-555-5555).  
role(John, Panel Chair).  
title(John, PHD).
```

Organizational Symposium Agent Knowledge Base (Abridged)

% Sample Prova-like rule in POSL syntax:

```
getContact(?conference_part, ?info, ?contact) :-  
    person(  
        ?contact, ?role, ?title, ?email, ?telephone).
```

Sample Message to Organizational Agent

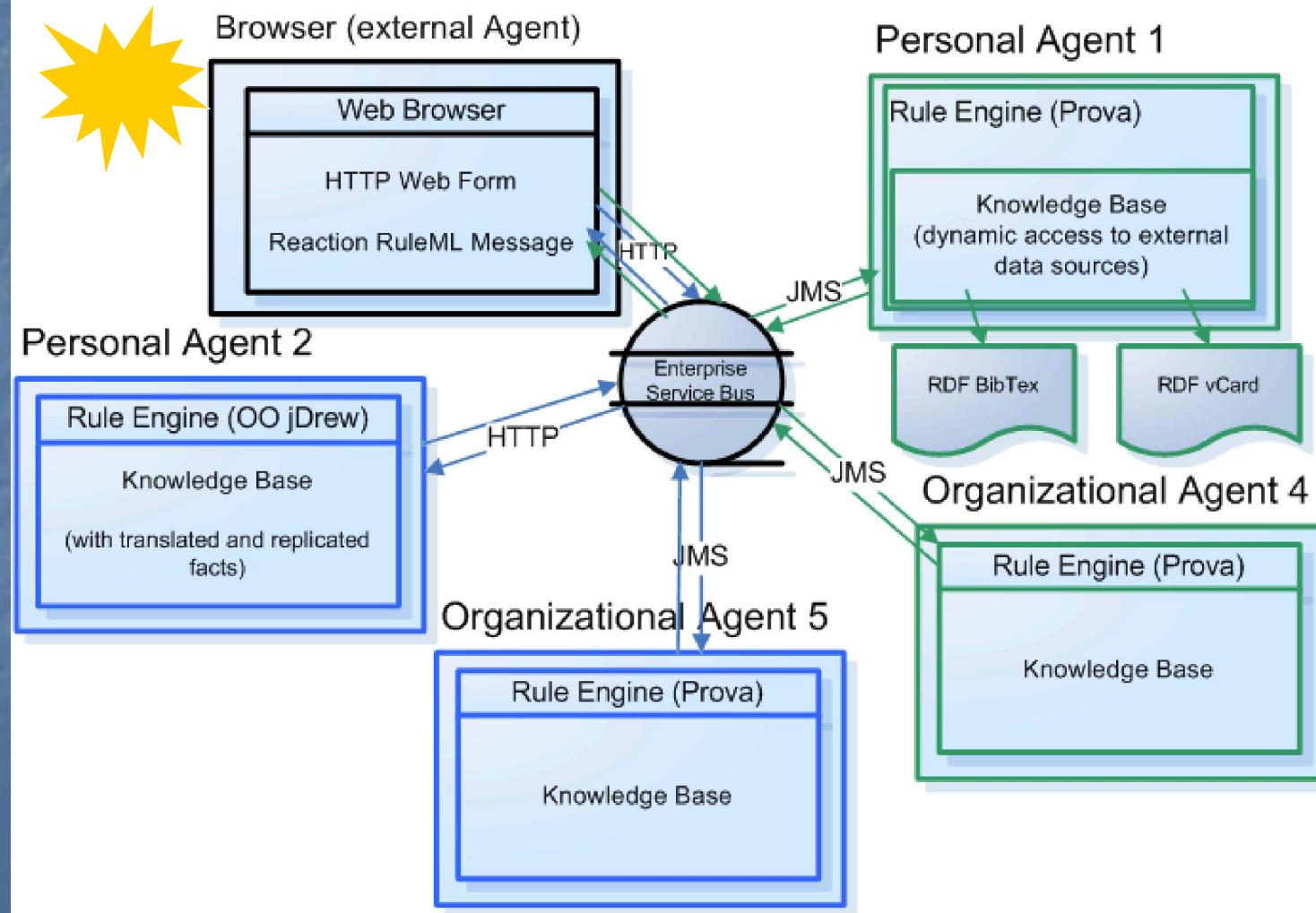
```
n <RuleML xmlns="http://www.ruleml.org/0.91/xsd"
n   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
n   xsi:schemaLocation="http://www.ruleml.org/0.91/xsd
n   http://ibis.in.tum.de/research/ReactionRuleML/0.2/rr.xsd"
n   xmlns:ruleml2007="http://ibis.in.tum.de/projects/paw#">
n
n     <Message mode="outbound" directive="query-sync">
n         <oid><Ind>RuleML-2007</Ind></oid>
n         <protocol><Ind>esb</Ind></protocol>
n         <sender><Ind>user</Ind></sender>
n         <content>
n             <Atom>
n                 <Rel>getContact</Rel>
n                 <Ind>ruleml2007_Challenge</Ind>
n                 <Ind>update</Ind>
n                 <Var>Contact</Var>
n             </Atom>
n         </content>
n     </Message>
n </RuleML>
```



Online

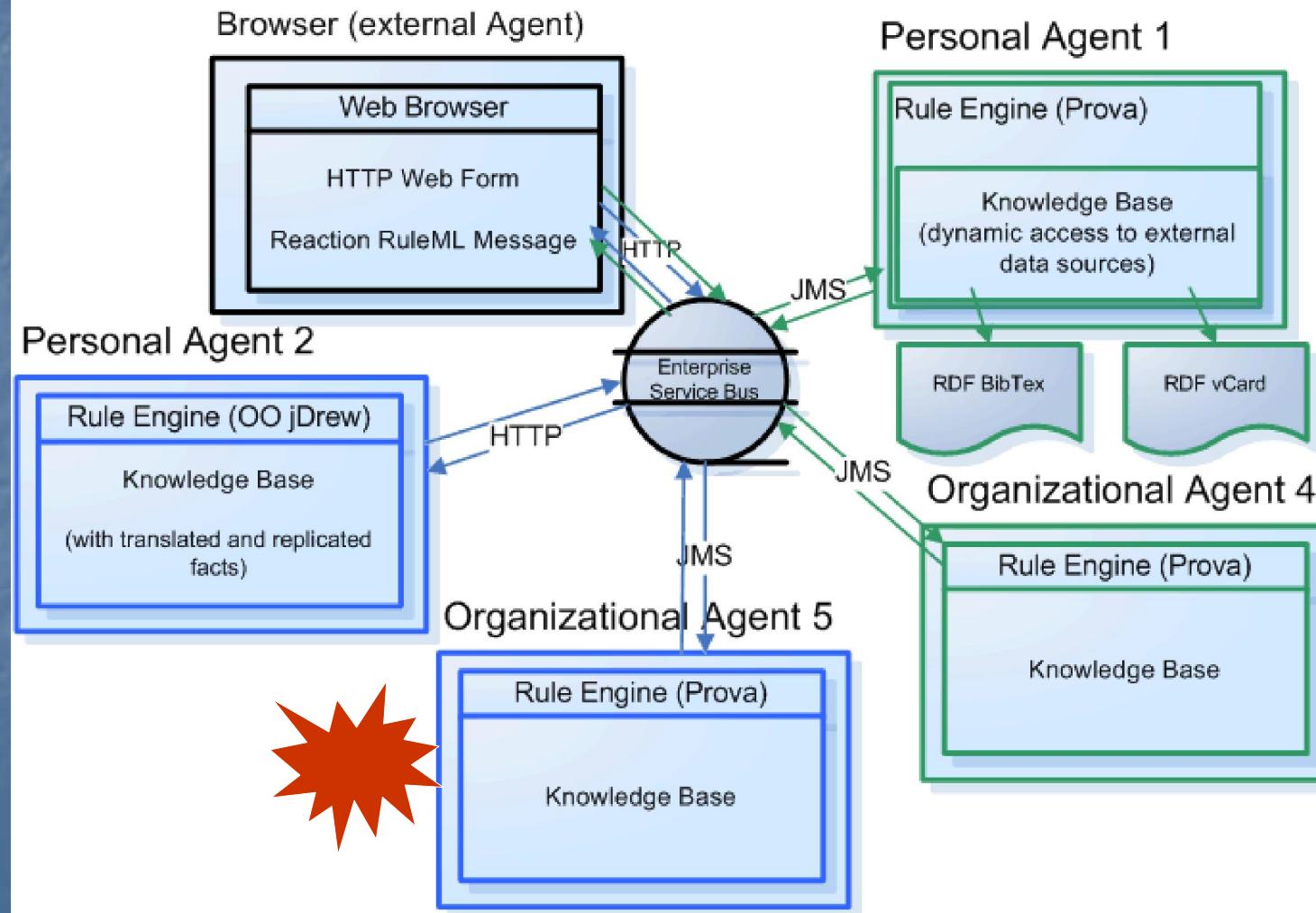
Architecture - Execution

Use Case 4 Use Case 5



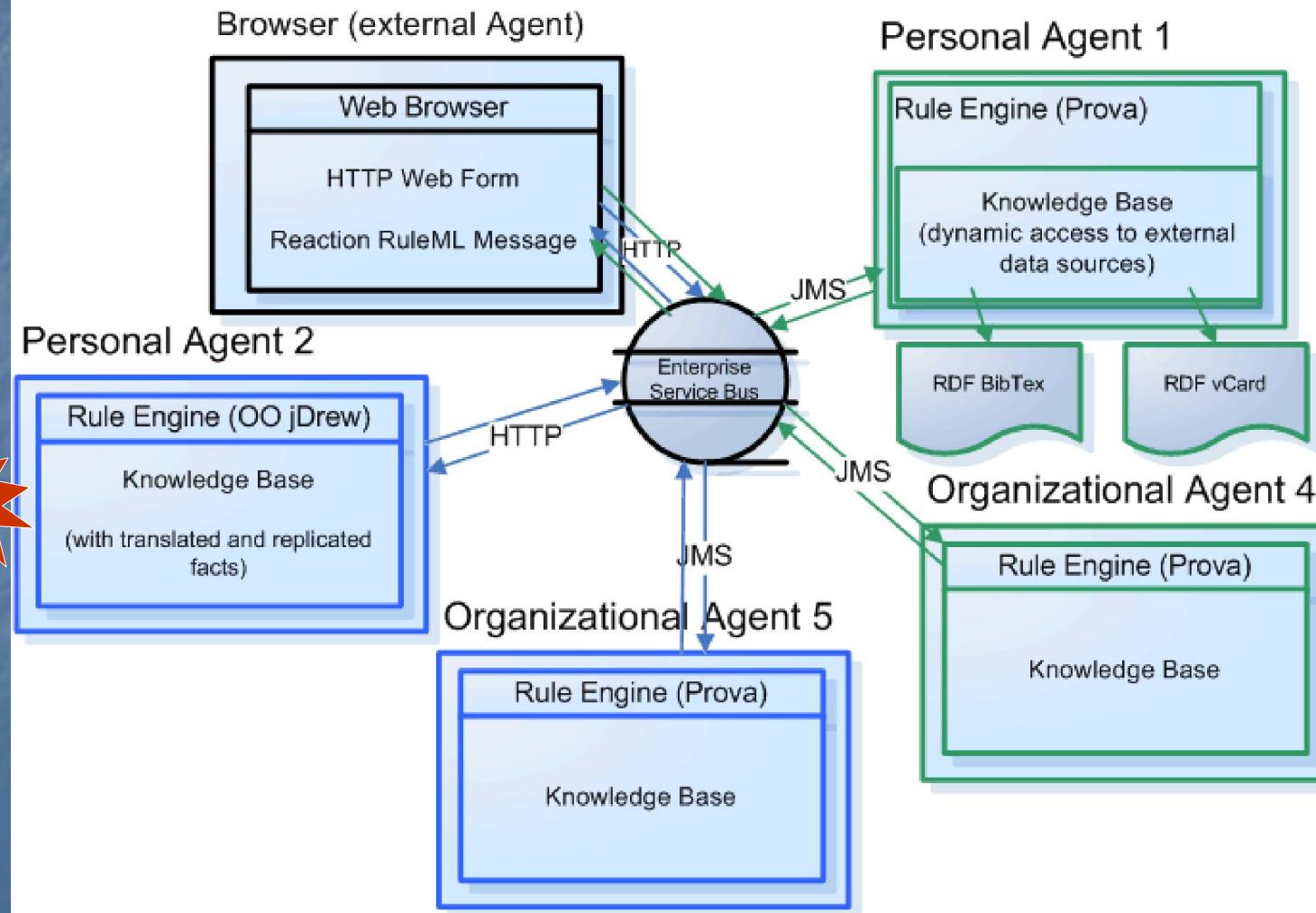
Architecture - Execution

Use Case 4 Use Case 5



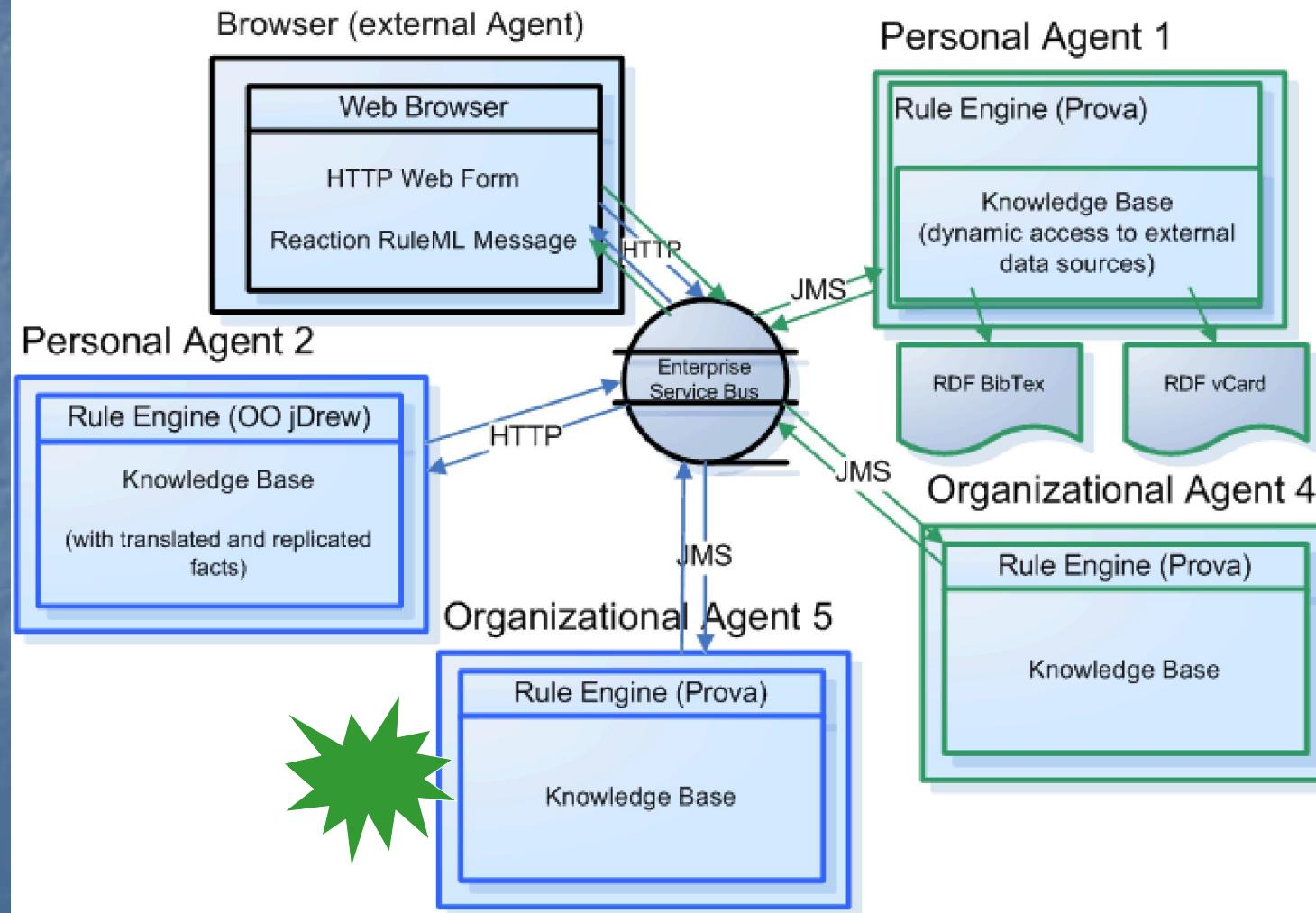
Architecture - Execution

Use Case 4 Use Case 5



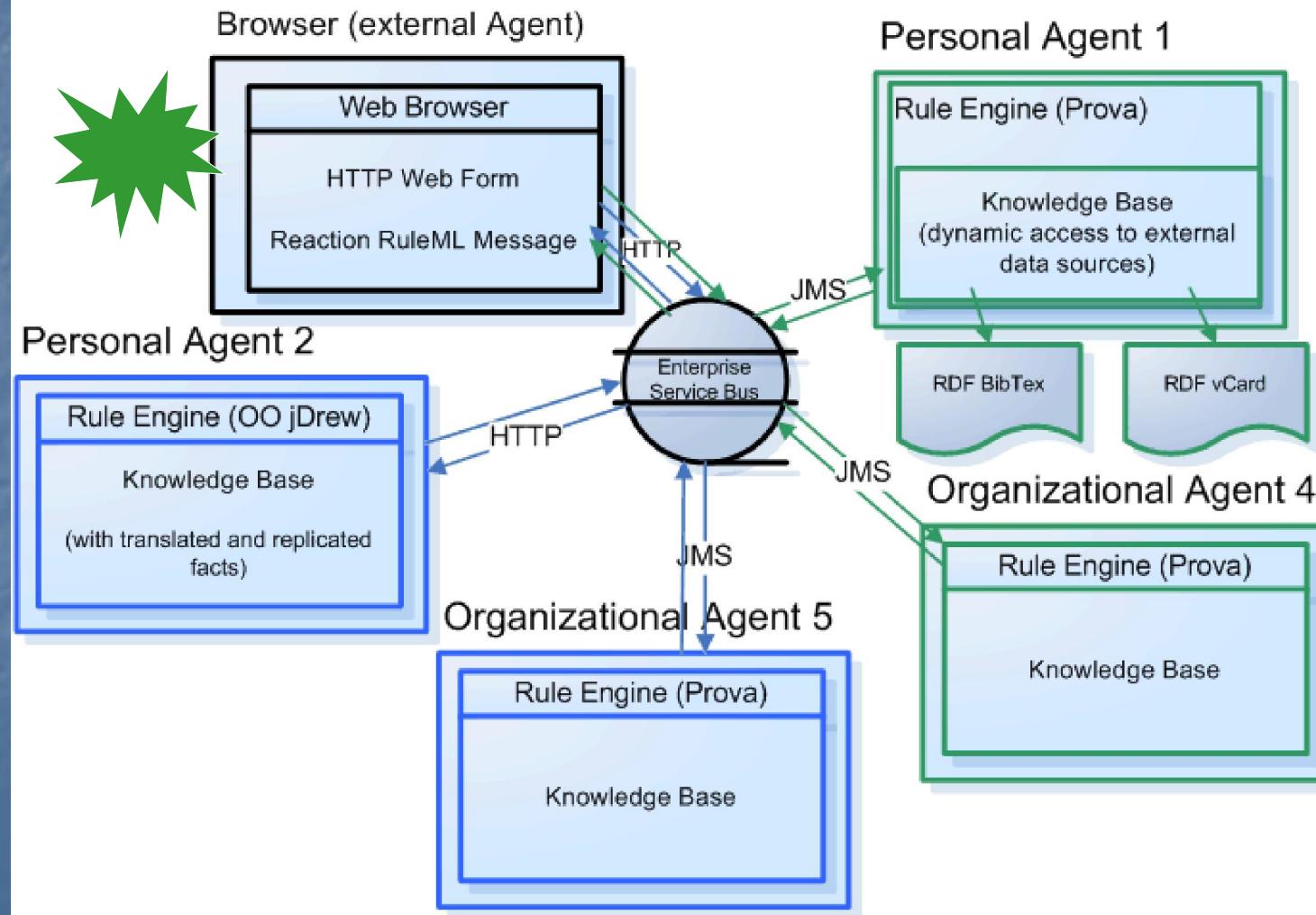
Architecture - Execution

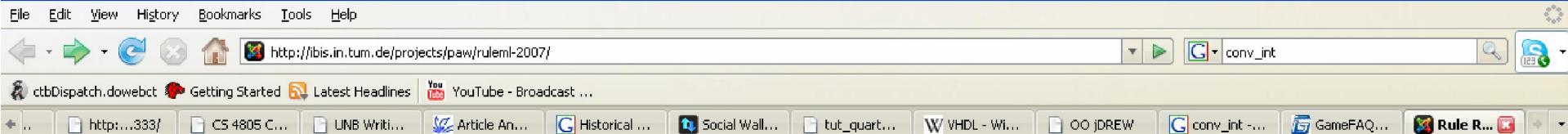
Use Case 4 Use Case 5



Architecture - Execution

Use Case 4 Use Case 5





Use this text form to send a query in [Reaction RuleML](#) in format to the RuleML-2007 Responder:

```
xmlns:ruleml2007="http://ibis.in.tum.de/projects/paw#"

<Message mode="outbound" directive="query">
    <oid>
        <Ind>RuleML-2007</Ind>
    </oid>
    <protocol>
        <Ind>esb</Ind>
    </protocol>
    <sender>
        <Ind>user</Ind>
    </sender>
    <content>
        <Atom>
            <Rel>getContact</Rel>
            <Ind>ruleml2007_Challenge</Ind>
            <Ind>update</Ind>
            <Var>Contact</Var>
        </Atom>
    </content>
</Message>
```

Description:

[RuleML-2007 Responder Use Case](#)

Rule Interface Descriptions (Signatures)

(you might copy and paste the examples in the Rule Responder form):

- ▶ [performative\(Performative\)](#) [[example](#)]
- ▶ [interface\(Query, Description\)](#) [[example](#)]
- ▶ [agent\(Agent\)](#) [[example](#)]
- ▶ [topic\(Topic\)](#) [[example](#)]
- ▶ [role\(Role\)](#) [[example](#)]
- ▶ [assigned\(Agent, Topic, Role\)](#) [[example](#)]
- ▶ [getContact\(Topic, Task, ContactInfo\)](#) [[example](#)]
- ▶ [permit\(Author, submit\(Author, Submission\)\)](#) [[example](#)]
- ▶ [submitted\(Submission\)](#) [[example](#)]
- ▶ [accepted\(Submission\)](#) [[example](#)]

```
<?xml version="1.0" encoding="UTF-8" ?>
- <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">
- <Message mode="outbound" directive="answer">
  - <oid>
    <Ind>RuleResponder@iitfrdextdev01.iit-iti.priv41</Ind>
  </oid>
  - <protocol>
    <Ind>esb</Ind>
  </protocol>
  - <sender>
    <Ind>RuleResponder</Ind>
  </sender>
  - <content>
    - <Atom>
      <Rel>getContact</Rel>
      <Ind>ruleml2007_Challenge</Ind>
      <Ind>update</Ind>
    - <Expr>
      <Fun>person</Fun>
      <Ind>johnAtemailDotcom</Ind>
      <Ind>PHD</Ind>
      <Ind>PanelChair</Ind>
      <Ind>John</Ind>
      <Ind>1555555555</Ind>
    </Expr>
    </Atom>
  </content>
</Message>
</RuleML>
```

Sample Message to Publicity Chair Agent (I)

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



```
- <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)

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



```
- <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>
- <Expr>
  <Fun>option</Fun>
- <Expr>
  <Fun>sponsor</Fun>
  <Ind>student</Ind>
</Expr>
</Expr>
- <Expr>
  <Fun>free</Fun>
  <Var>Benefits</Var>
  <Ind>registration</Ind>
- <Expr>
  <Fun>amount</Fun>
  <Ind>2</Ind>
</Expr>
</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>
- <Expr>
  <Fun>distribution</Fun>
- <Expr>
  <Fun>brochures</Fun>
- <Expr>
  <Fun>all</Fun>
  <Ind>participants</Ind>
</Expr>
</Expr>
</Expr>
- <Expr>
  <Fun>onGoing</Fun>
  <Ind>deadline</Ind>
</Expr>
</Expr>
- <Expr>
  <Fun>performative</Fun>
  <Ind>phone</Ind>
</Expr>
</Atom>
</content>
```

Conclusion

- „ Rule Responder can be used to implement a wide range of use cases that require a semi-automated decision layer
- „ The Mule middleware of Rule Responder allows platform-independent deployment of multiple running use cases simultaneously
- „ The system is reusable on all levels: Symposium Planner, Rule Responder, POSL, RuleML, OO jDREW, Prova, Mule