CROSS-COMMUNITY INTEROPERATION BETWEEN THE EMERALD AND RULE RESPONDER MULTI-AGENT SYSTEMS

Kalliopi Kravari¹, Taylor Osmun², Harold Boley² and Nick Bassiliades¹

¹Dept. of Informatics, Aristotle University of Thessaloniki, Greece
²Inst. for Information Technology, NRC Canada
Overview

Motivation

Multi-Agent Systems
- EMERALD
- Rule Responder

EMERALD – Rule Responder Interoperation
- Comparison
- Gateway Architecture

SymposiumPlanner
- Interoperation Use Case

Conclusions – Future Work
Motivation

SW evolution → Intelligent Agents

A number of multi-agent systems are available but typically isolated.

EMERALD → bidirectional bridge → Rule Responder
Overview

- Motivation

- Multi-Agent Systems
  - EMERALD
  - Rule Responder

- EMERALD – Rule Responder Interoperation
  - Comparison
  - Gateway Architecture

- SymposiumPlanner
  - Interoperation Use Case

- Conclusions – Future Work
EMERALD
A Multi-Agent Knowledge-Based Framework
Rule Responder

- Open source framework
- Creating virtual organizations
- Provides the infrastructure for rule-based collaboration
- Assisting by semi-autonomous rule-based agents

OA: (global) goals/strategies
PA: assist single person
EA: Web interface/queries
CA: perform automated task
Overview

- Motivation

- Multi-Agent Systems
  - EMERALD
  - Rule Responder

- EMERALD – Rule Responder Interoperation
  - Comparison
  - Gateway Architecture

- SymposiumPlanner
  - Interoperation Use Case

- Conclusions – Future Work
# EMERALD–Rule Responder Comparison

<table>
<thead>
<tr>
<th></th>
<th>Rule Responder</th>
<th>EMERALD</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Agent technology</strong></td>
<td>Java servlets / Mule</td>
<td>Java (JADE) agents</td>
</tr>
<tr>
<td>**Interchange</td>
<td>Mule middleware</td>
<td>JADE (ACL)</td>
</tr>
<tr>
<td><strong>principles</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>RuleML</strong></td>
<td>Reaction RuleML</td>
<td>(D)R-DEVICE RuleML</td>
</tr>
<tr>
<td><strong>Agent knowledge</strong></td>
<td>Internal rule base</td>
<td>External rule base</td>
</tr>
<tr>
<td></td>
<td>Internal &amp; External data-knowledge base</td>
<td>External data-knowledge base</td>
</tr>
<tr>
<td><strong>Reasoning</strong></td>
<td>Multiple reasoning engines and instances of</td>
<td>Multiple reasoning engines</td>
</tr>
<tr>
<td></td>
<td>reasoning engines</td>
<td>(independent external services)</td>
</tr>
<tr>
<td><strong>Directory service</strong></td>
<td>NO</td>
<td>AYPS</td>
</tr>
<tr>
<td><strong>Use of Prova</strong></td>
<td>OAs always written in Prova, PAs and CAs optionally</td>
<td>A Prova Reasoner has been developed</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(one of the reasoning agents) prova 3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>not supported because it does not</td>
</tr>
<tr>
<td></td>
<td></td>
<td>support JADE yet</td>
</tr>
<tr>
<td><strong>Use</strong></td>
<td>Use cases can be obtained as instantiations of the</td>
<td>Use cases can be obtained by using</td>
</tr>
<tr>
<td></td>
<td>Rule Responder framework</td>
<td>different reasoners and different agent</td>
</tr>
<tr>
<td></td>
<td></td>
<td>behavior KBs</td>
</tr>
</tbody>
</table>
EMERALD – RR
RRP: communicating directly with OA

RR – EMERALD
CA: handles appropriate communication channel

EMERALD – RR collaboration key feature: interchange of information
Overview

- **Motivation**

- **Multi-Agent Systems**
  - EMERALD
  - Rule Responder

- **EMERALD – Rule Responder Interoperation**
  - Comparison
  - Gateway Architecture

- **SymposiumPlanner**
  - Interoperation Use Case

- **Conclusions – Future Work**
SymposiumPlanner: series of applications based on the RuleML
Symposium developed with Rule Responder

Inside the SP community each human chair (e.g. publicity chair) has a Personal Agent (PA) to assist him/her.

Each PA has a knowledge base containing the responsibilities of the position in order to answer queries relevant to the chair's role.
Interoperation Use Case

A series of use cases based on the RuleML Symposium series

SymposiumPlanner

KB

(PA)
Publicity Chair

Whether or not to sponsor RuleML-20XY Symposium

Based on its personal preferences

NOT: "What I get for this amount of money?"

BUT: "If I want these benefits, what amount of money do I have to spend?"

$5000 budget
the cheapest sponsoring level
providing at least a free registration and a demo for the company's products
Exchanged messages
# The Sponsoring Levels

## Partner’s preferences
- $5000 max
- Cheapest sponsoring level
- At least one free registration
- Demo for the company’s products

## Sponsoring levels

<table>
<thead>
<tr>
<th>Level</th>
<th>Amount</th>
<th>Benefits</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bronze</td>
<td>$500</td>
<td>Logo on website&lt;br&gt;Acknowledgement in proceedings</td>
</tr>
<tr>
<td>Silver</td>
<td>$1,000</td>
<td>Bronze level benefits + Sponsor student participants</td>
</tr>
<tr>
<td>Gold</td>
<td>$3,000</td>
<td>Silver level benefits + Logo in proceedings&lt;br&gt;Show demo&lt;br&gt;1 free registration</td>
</tr>
<tr>
<td>Platinum</td>
<td>$5,000</td>
<td>Gold level benefits + Name included in all advance publicity.&lt;br&gt;Distribution of material to all participants.&lt;br&gt;1 additional free registration</td>
</tr>
<tr>
<td>Emerald</td>
<td>$7,500</td>
<td>Platinum level benefits + 1 additional free registration</td>
</tr>
</tbody>
</table>

### Reasoning Rules

1. `possibleOffer(level->?x) := sponsorLevel(level->?x).`
2. `~possibleOffer(level->?x) := sponsorLevel(level->?x, demo->false).`
5. `makeOffer(level->?x) := possibleOffer(level->?x), sponsorLevel(level->?x, amount->?z), \( \forall y \exists x \left( possibleOffer(level->?y), ?y \in x, sponsorLevel(level->?y,amount->?w), ?w < ?z \right).`
Overview

- Motivation

- Multi-Agent Systems
  - EMERALD
  - Rule Responder

- EMERALD – Rule Responder Interoperation
  - Comparison
  - Gateway Architecture

- SymposiumPlanner
  - Interoperation Use Case

- Conclusions – Future Work
Conclusions - Future Work

Interoperation between EMERALD and Rule Responder

Bidirectional gateways
  • automated collaboration
  • declarative, knowledge-based approach

In future:
  • Bidirectional RuleML-based gateways suite
  • RuleML gateways adaptation in interoperation needs (interchange of proofs)
  • More use cases
  • Generalized gateway principles and architectures for cross-community agent interoperation
EMERALD: http://lpis.csd.auth.gr/systems/emerald

Rule Responder: http://ruleml.org/RuleResponder

SymposiumPlanner: http://ruleml.org/SymposiumPlanner

EMERALD – Rule Responder interoperation project: http://tinyurl.com/EMERALDRR

Thank you! Any Questions?