

# Rule-based Semantic BPM

**Adrian Paschke**

Corporate Semantic Web (AG-CSW)  
Institute for Computer Science,  
Freie Universität Berlin  
[paschke@inf.fu-berlin.de](mailto:paschke@inf.fu-berlin.de)  
<http://www.inf.fu-berlin.de/groups/ag-csw/>

ServiceWave 2011, Poznan, Poland



# Overview

---

- Semantic Business Process Management
- Ontologies in BPM
- Rules in BPM
- Rule-based Event Driven BPM
- Summary Key Benefits of SBPM

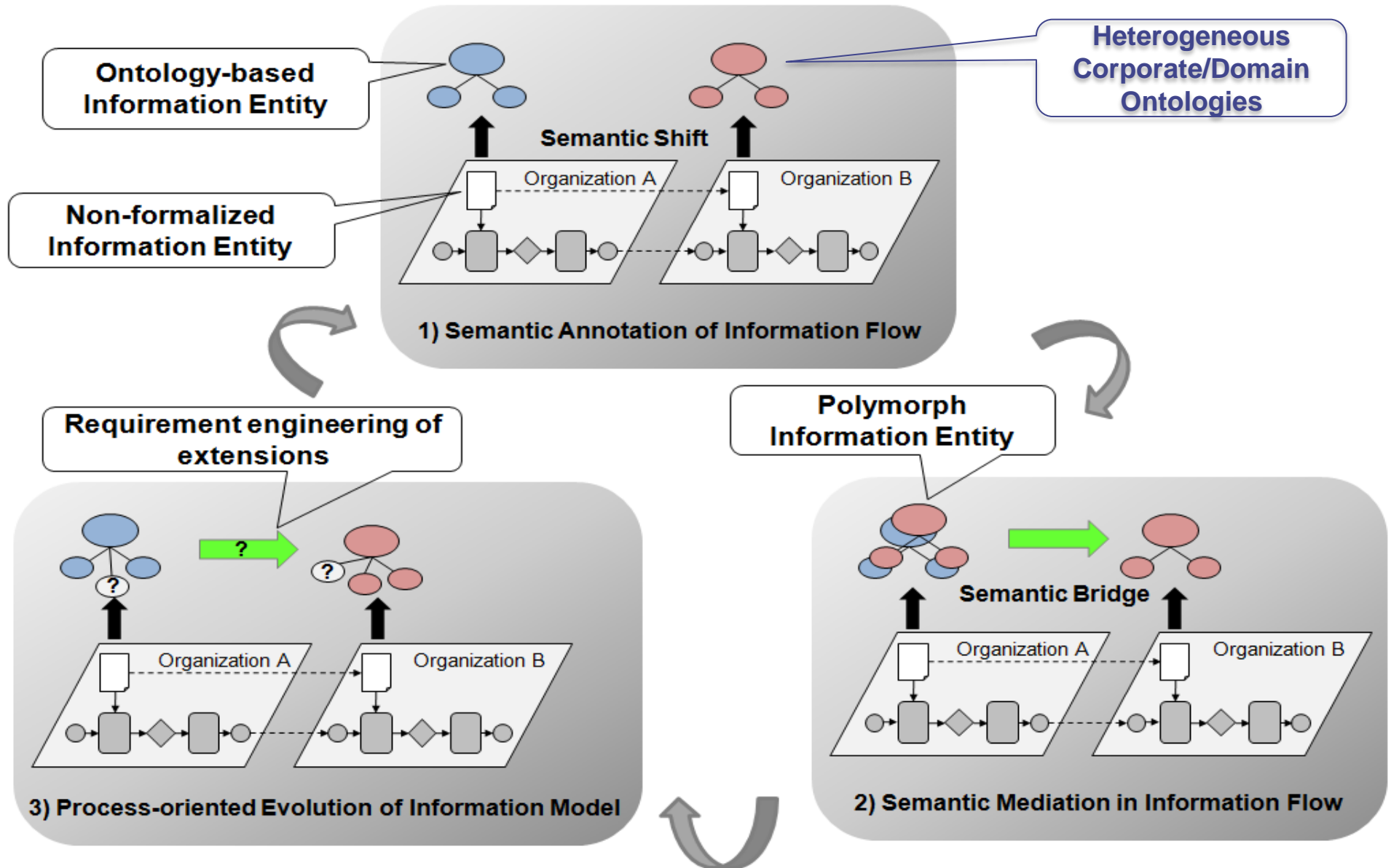
# Semantics BPM Approach

---

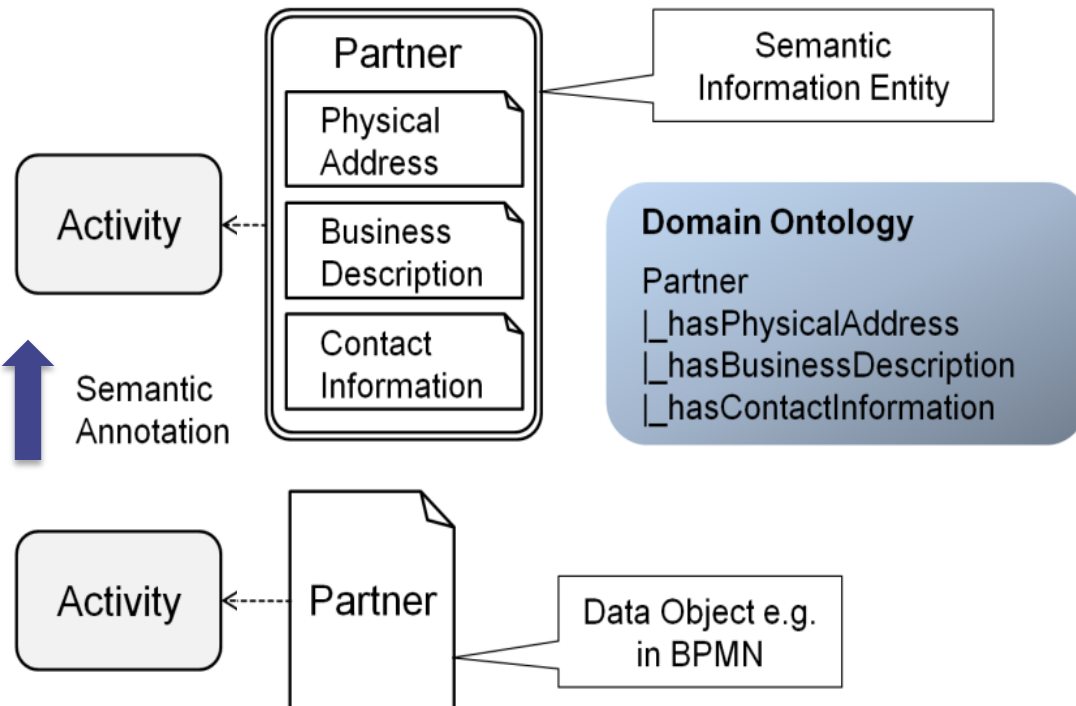
- The combination of BPM and semantic technologies leads to ***Semantic Business Process Management (SBPM)***.
- The BPM engines/tools **can semantically understand** what is happening in terms of events, situations, decision in a business process.
- The BPM engine can **know** what reactions and processes it can invoke and what events it can signal.
- Makes it possible to detect and **process semantically related (complex) events and tasks**
- Gives a **formal semantics** to concepts relevant in business processes, e.g. tense, spatio, state, ...
  - Also support interchange between domain boundaries with different vocabularies

# Ontologies + BPM

# Semantic Business Process Modeling



# Semantic Extension of Information Entities in Business Process Modeling Notation



- Utilize corporate or domain ontology concepts to define information flow on a non-technical conceptual level suitable for business process experts
- due to formal nature consistent link between the business or conceptual level and underlying technical information models can be derived
- formal domain information models are foundation for semantic mediation between heterogeneous conceptualizations used by different organizations or domains

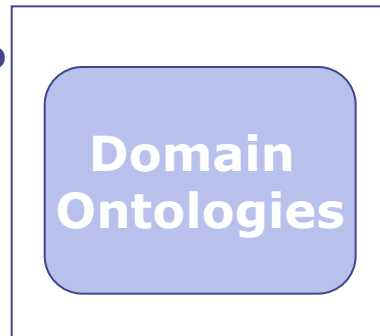
# Modular Ontology Model for SBPM

## Top Level Ontologies

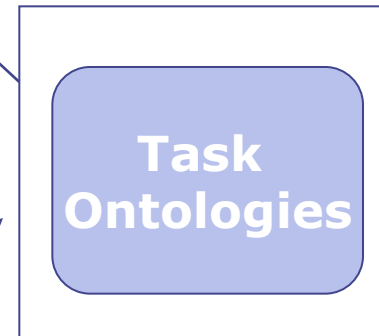
General concepts such as space, time, event and their properties and relations



vocabularies **related to specific domains** by specializing the concepts introduced in the top-level ontology



vocabularies **related to generic tasks or activities** by specializing the concepts introduced in the top-level ontology

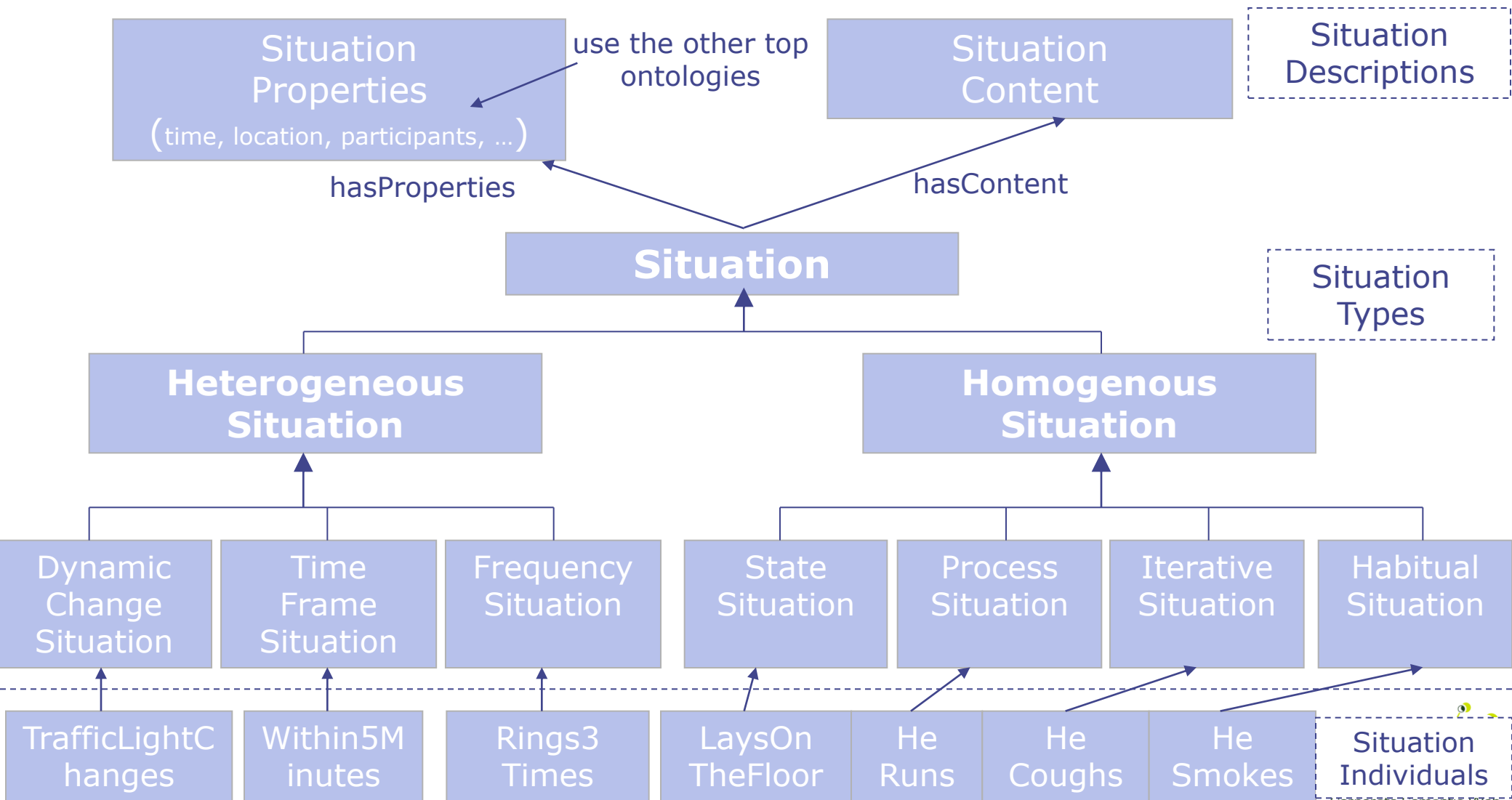


**Specific**  
user/application  
ontologies



e.g. ontologies describing roles played by domain entities while performing application activities

# Situation Top Ontology Model



# Semantic Mediation between heterogeneous Information Entities

## Instance A

Address Address  
\_hasName Name  
  \_hasGivenName ,Nils'  
  \_hasSurname ,Barnickel'  
\_hasStreet ,Kantstrasse'  
\_hasStreetNumber ,7'  
\_hasZIP ...

## Information Model A

Address  
  \_hasName  
  \_hasStreet  
  \_hasStreetNumber  
  \_hasZIP  
  
Name  
  \_hasGivenName  
  \_hasSurname

## Information Model B

PostalAddress  
  \_hasRecipient  
  \_hasStreetAddress  
  \_hasZIP

### Semantic Bridge:

(A hasName N), (N hasGivenName G), (N hasSurname S) → (A hasRecipient G+S)  
(A hasStreet S), (A hasStreetNumber N) → (A hasStreetAddress S+N)

## Instance A

Address Address  
\_hasName Name  
  \_hasGivenName ,Nils'  
  \_hasSurname ,Barnickel'  
\_hasStreet ,Kantstr.'  
\_hasStreetNumber ,7'  
\_hasZIP ...  
\_hasRecipient ,Nils Barnickel'  
\_hasStreetAddress ,Kantstr. 7'

### Classification

## Instance A

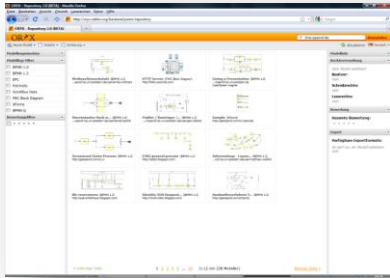
Address Address/PostalAddress  
\_hasName Name  
  \_hasGivenName ,Nils'  
  \_hasSurname ,Barnickel'  
\_hasStreet ,Kantstr.'  
\_hasStreetNumber ,7'  
\_hasZIP ...  
\_hasRecipient ,Nils Barnickel'  
\_hasStreetAddress ,Kantstr. 7'

- Mapping heterogeneous semantic sub-graphs via rule-based bridges
- polymorph classification preserving object identity

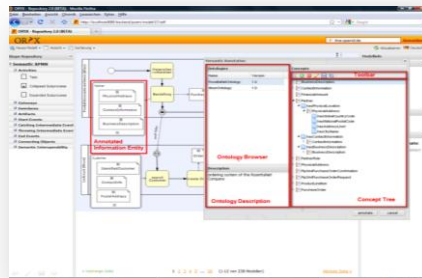
# Mediated Business Process Modeling Tool

## Semantic Extension of Oryx

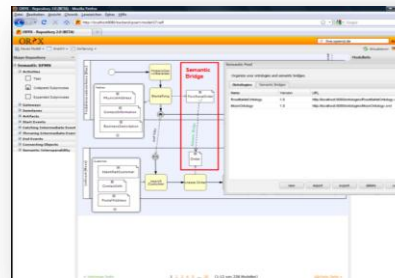
---



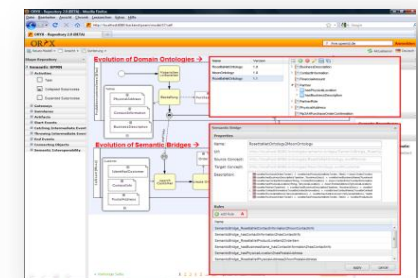
**Oryx  
Process Editor**



**Ontology-based  
Information Entity  
Annotation**

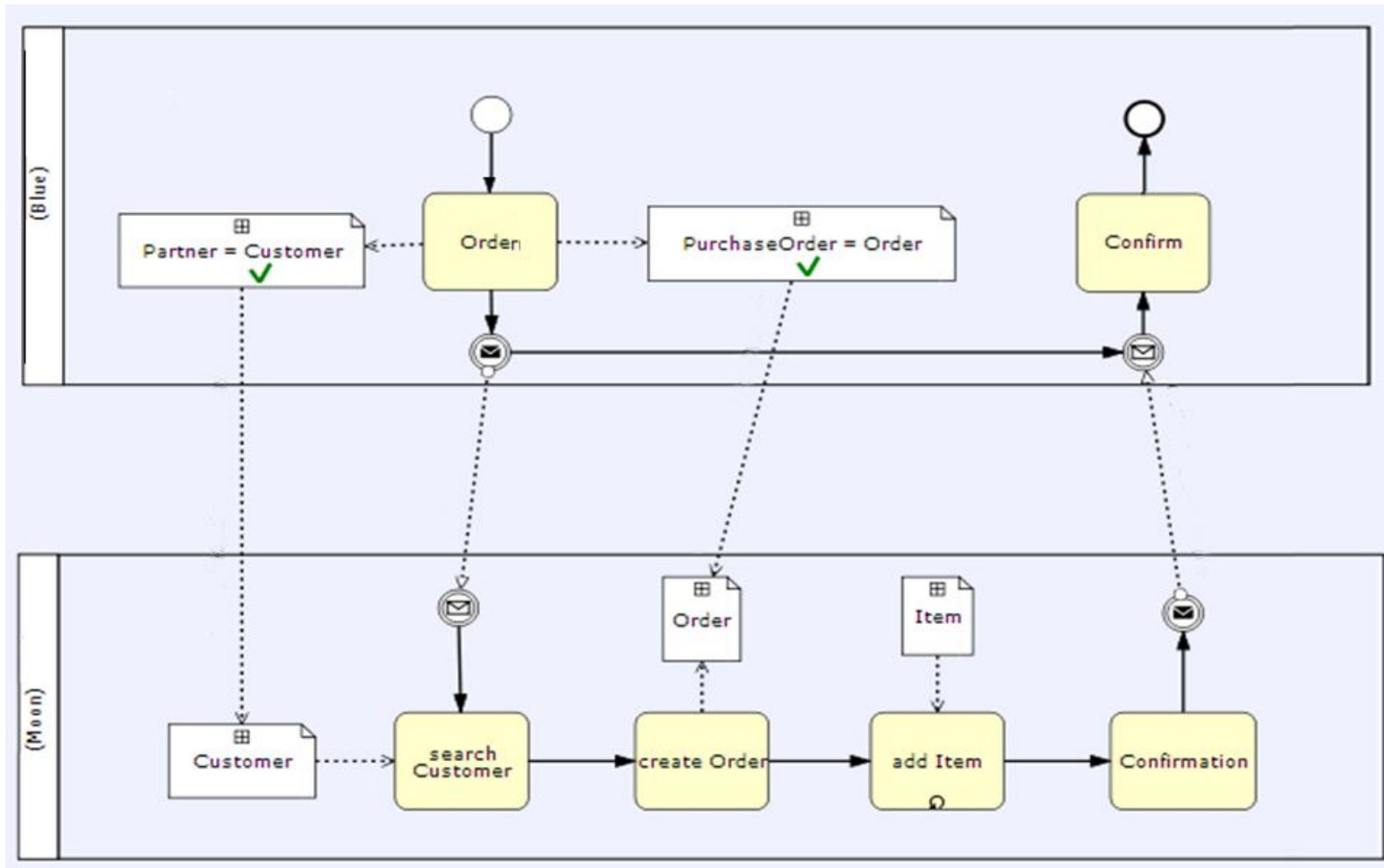


**Cross-organizational  
Semantic Mediation**



**Evolution of  
Information Models  
And Semantic Bridges**

# Example Mediated Business Process



# Semantic CEP: Ontologies (cont.)

---

- Better understanding of **situations (states)**
  - e.g., *a process is executing when it has been started and not ended*
  - e.g. *a plane begins flying when it takes off*
  - e.g. *a plane is no longer flying after it lands*
- Better understanding of the **relationships** between events  
e.g., temporal, spatial, causal, .., relations between events, states, activities, processes
  - e.g., *a service is unavailable when the service response time is longer than X seconds and the service is not in maintenance state*
  - e.g. *a landing starts when a plane approaches*
  - e.g. *during landing mobile phones must be switched off*
- Data becomes **meaningful information / declarative knowledge** while conforming to an underlying **formal semantics**
  - e.g., **automated semantic mediation** between different heterogeneous domains and abstraction levels

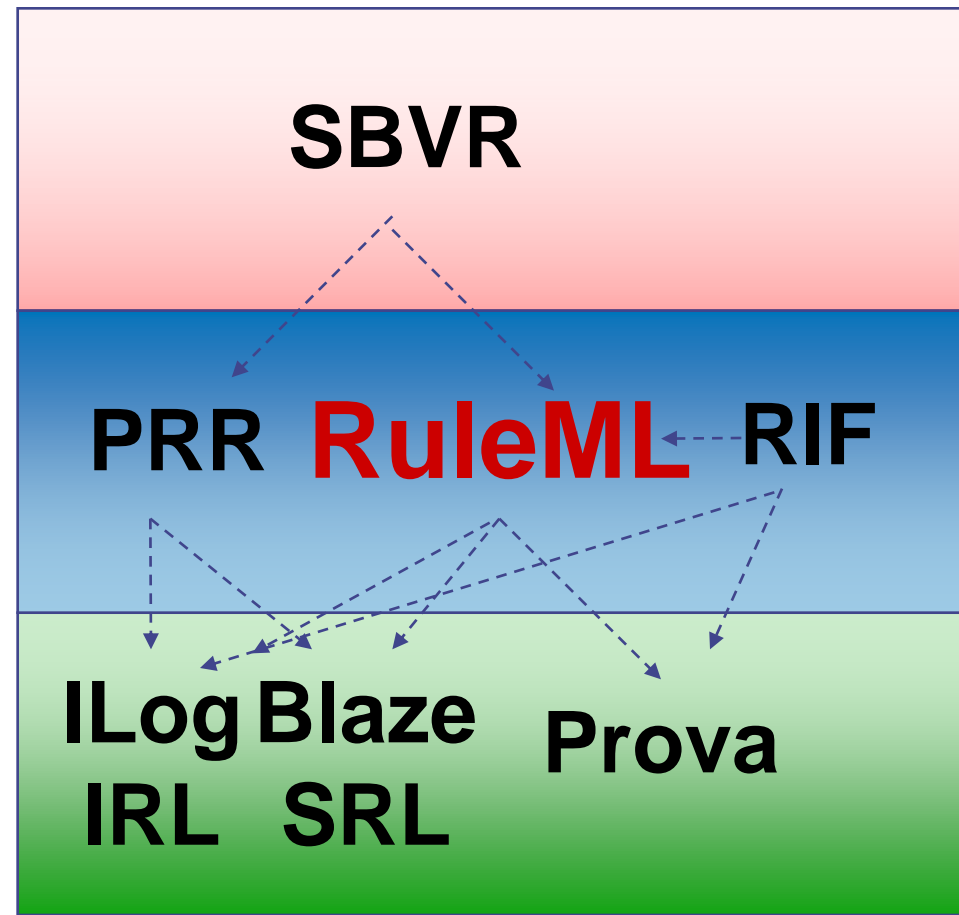
# Rules + BPM

# Rules Technology

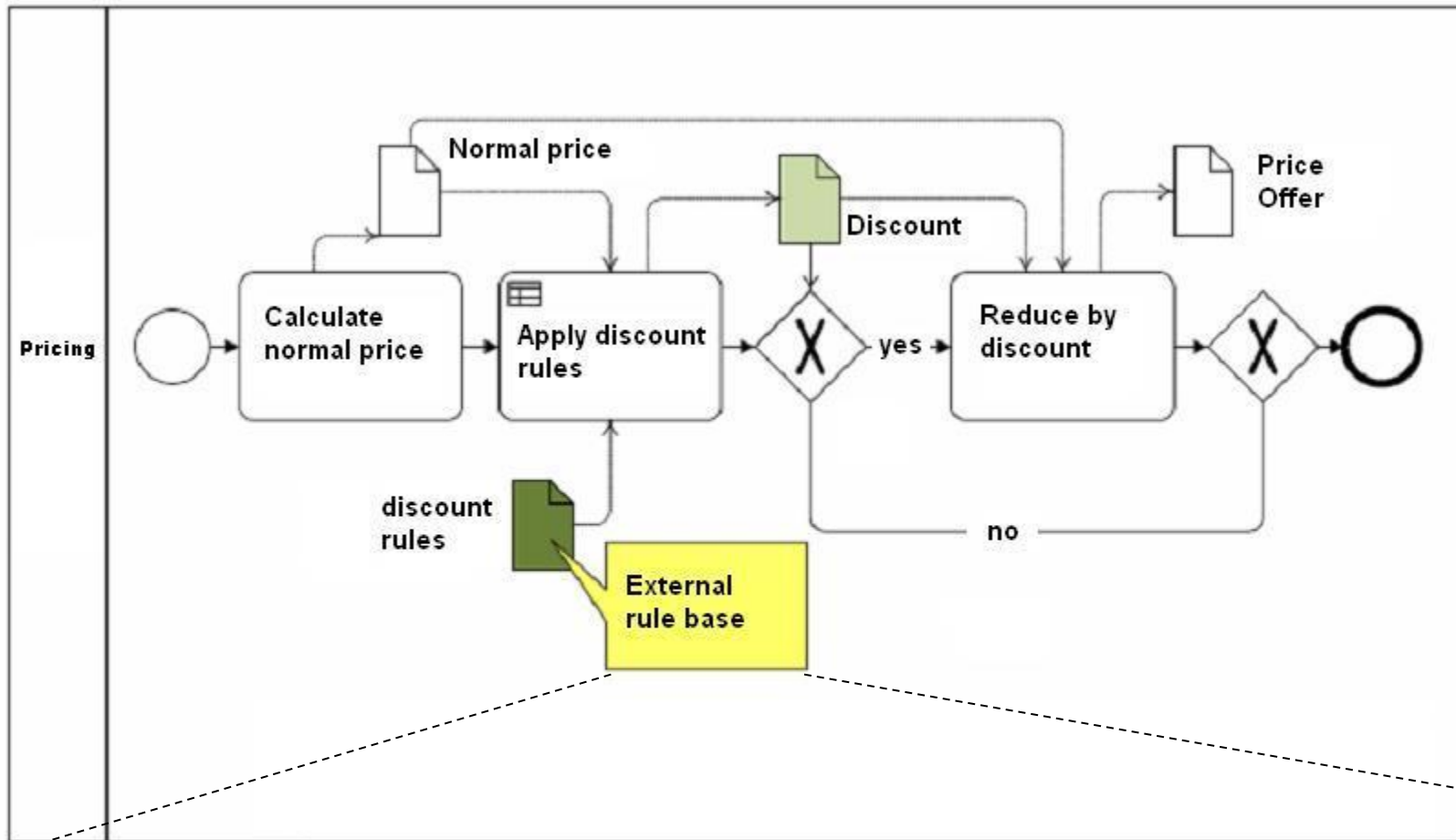
---

*Users employ rules to express **what** they want, the responsibility to interpret this and to decide on **how** to do it is delegated to an interpreter*

*Represent knowledge in a way that is understandable by 'the business', but also executable by rule engines, thus bridging the gap between business and technology*

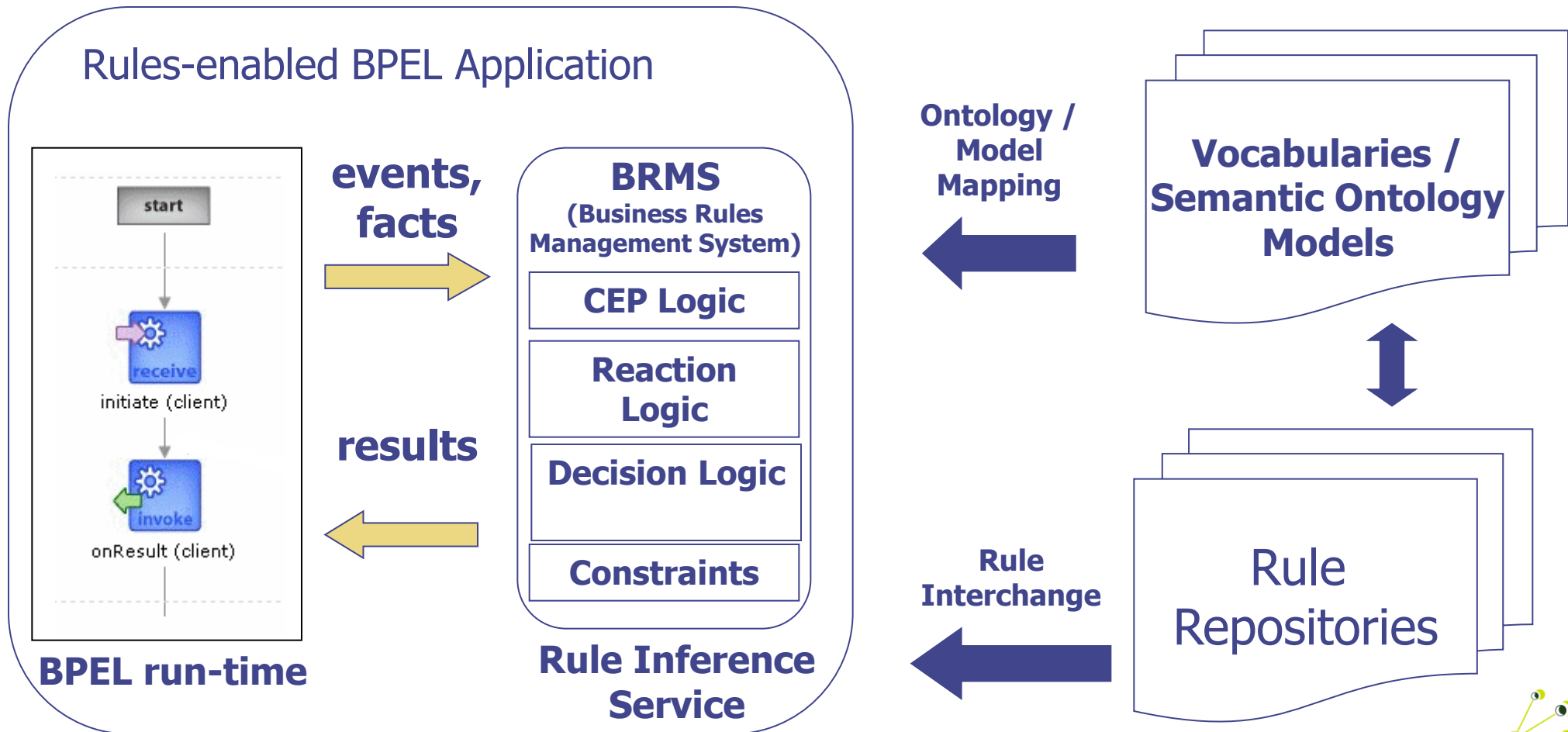


# SBPM: Heterogeneous Integration of Rules in BPMN 2.0



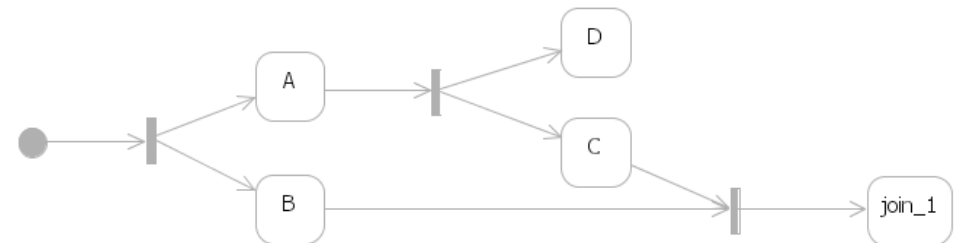
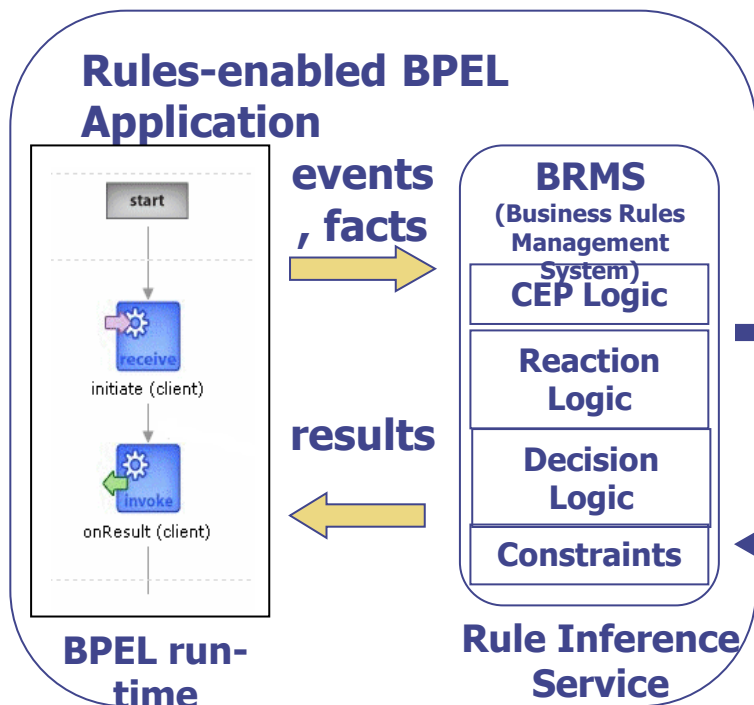
```
discount(_customer, _product, 5.0 percent) :- premium(_customer), regular(_product).  
discount(_customer, _product, 7.5 percent) :- premium(_customer), luxury(_product).  
premium(_customer) :- spending(_customer, min 5000 euro, previous year).
```

# Rule-based BPEL+ (Semantic BPEL)



# Orchestrated BPEL + Choreography Rule Workflow

- Rules can be used to implement **choreography workflows** as subprocesses in the orchestration BPEL flow
- Workflows might span several **communicating (messaging) rule inference services**



```

% receive query and delegate it to another party
rcvMsg(CID,esb,Requester,acl_query-ref,Query):-
    responsibleRole(Agent,Query),
    sendMsg(Sub-CID,esb,Agent,acl_query-ref,Query),
    rcvMsg(Sub-CID,esb,Agent,acl_inform-ref,Answer),
    ... (other goals)...
    sendMsg(CID,esb,Requester,acl_inform-ref,Answer).
  
```

Prova rule engine <http://prova.ws>

# Prova: Messaging Reaction Rules

- Send a message

*sendMsg*(XID, Protocol, Agent, Performative, [Predicate|Args]|Context)

- Receive a message

*rcvMsg*(XID, Protocol, Agent, Performative, [Predicate|Args]|Context)

- Receive multiple messages

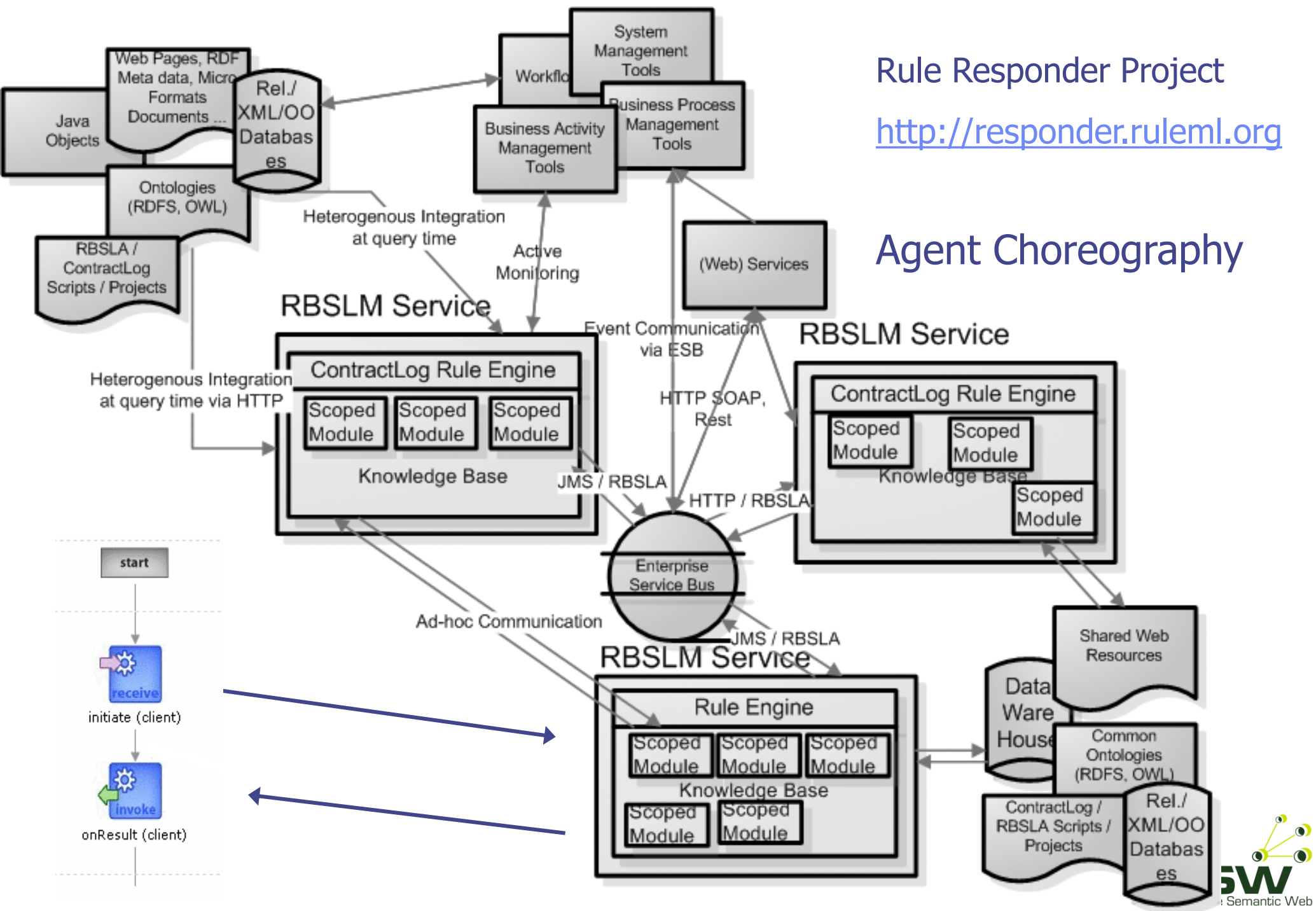
*rcvMult*(XID, Protocol, Agent, Performative, [Predicate|Args]|Context)

## Description:

- *XID* is the conversation identifier
- *Protocol*: protocol e.g. self, jms, esb etc.
- *Agent*: denotes the target or sender of the message
- *Performative*: pragmatic context, e.g. FIPA Agent Communication
- *[Predicate|Args]* or *Predicate(Arg<sub>1</sub>, ..., Arg<sub>n</sub>)*: Message payload

Rule Responder Project  
<http://responder.ruleml.org>

# Agent Choreography



# Semantic BPM: Rules (cont.)

---

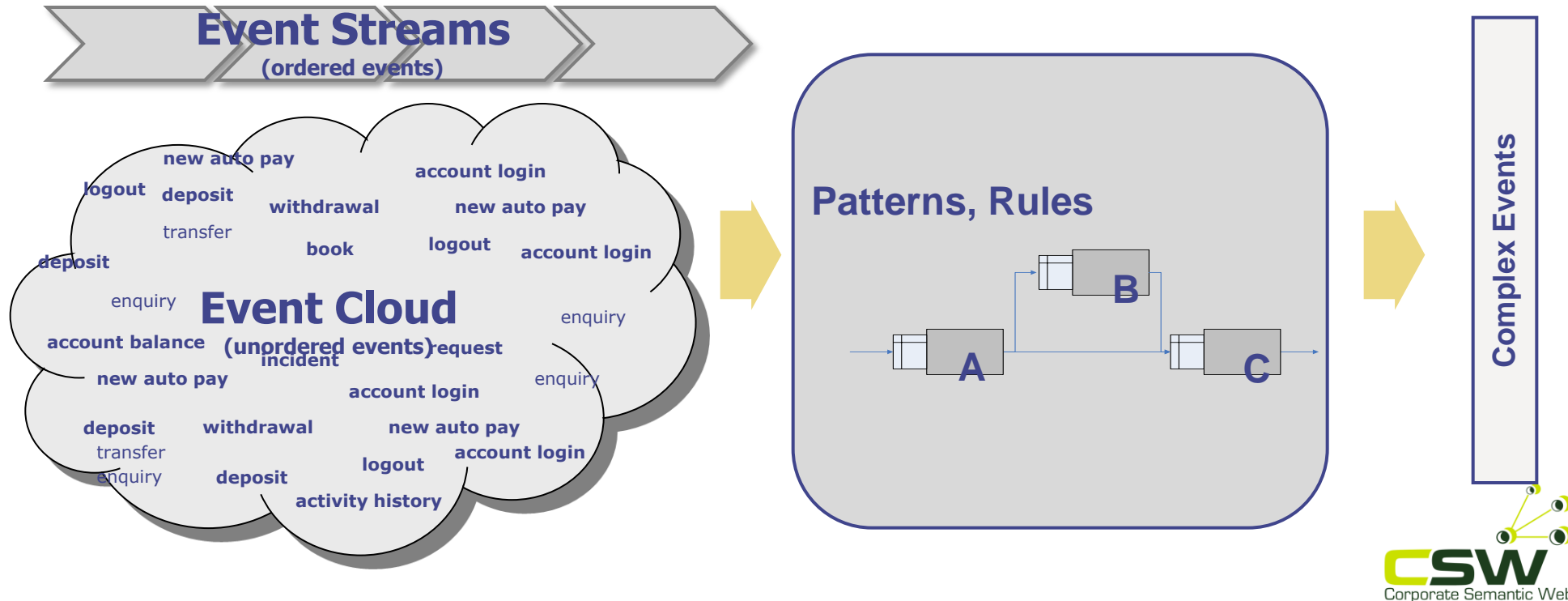
*Rules engines can be invoked as a Semantic Decision / Inference Service from a BPM process.*

- Dynamic processing
  - Intelligent routing
  - Validation of policies within process
  - Constraint checks
- Ad-hoc Workflow
  - Policy based task assignment
  - Various escalation policies
  - Load balancing of tasks
- Business Activity Monitoring
  - Alerts based on certain policies and complex event processing (rule-based CEP)
  - Dynamic processing based KPI reasoning

# **Event-Driven Semantic BPM**

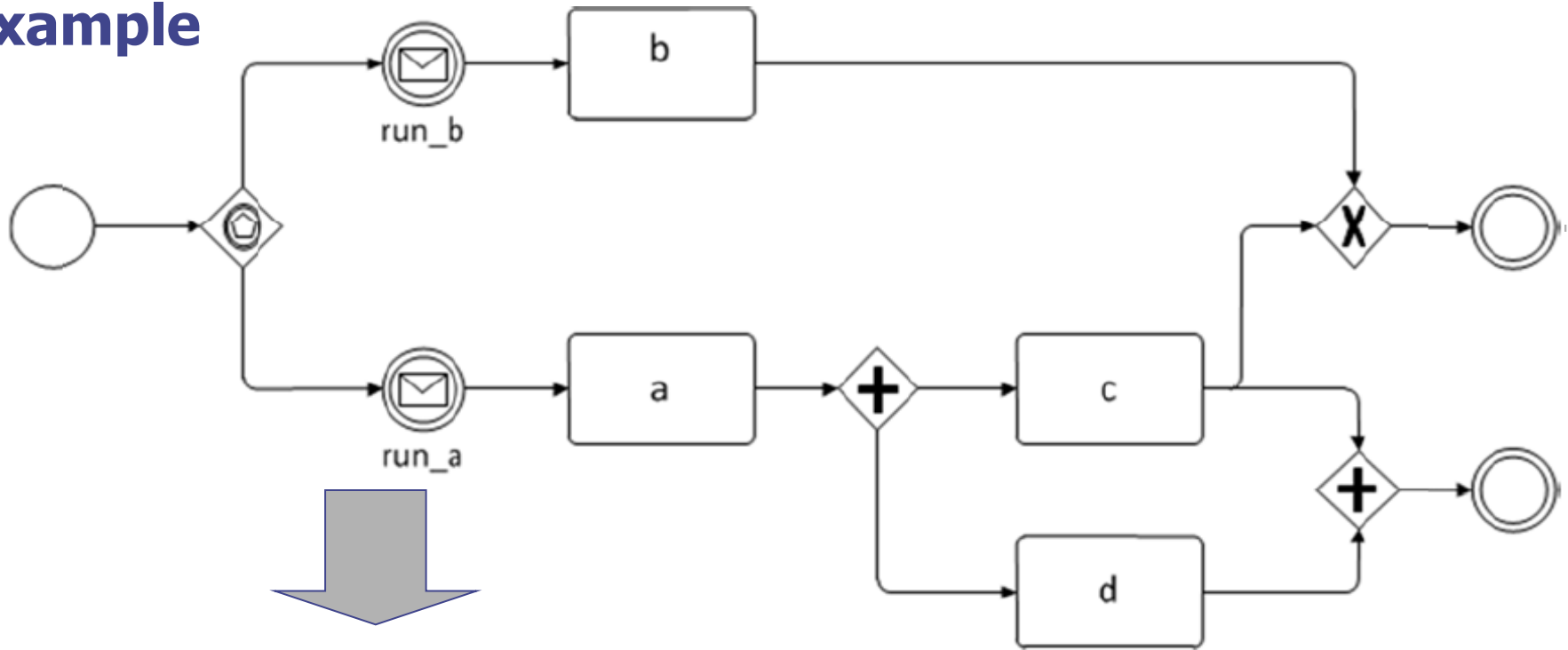
# Complex Event Processing

- CEP is about complex event detection and reaction
  - Efficient (near real-time) **processing** of large numbers of events
  - **Detection, prediction** and **exploitation** of relevant complex events
  - Situation awareness, **track & trace, sense & respond**





# Example



fork (CID) :-

**@group (ab)**

rcvMsg (CID, Protocol, From, command, **run\_A**), ... .

fork (CID) :-

**@group (ab)**

rcvMsg (CID, Protocol, From, command, **run\_B**), ... .

fork (CID) :-

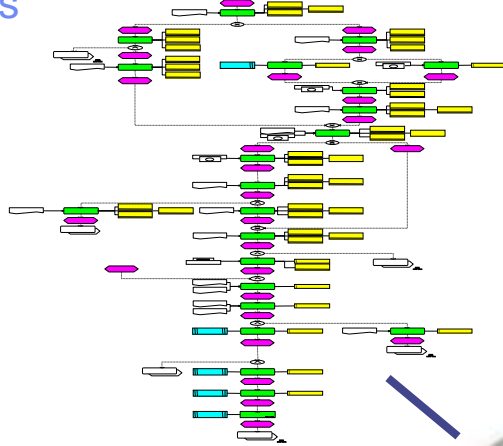
**@or (ab)**

rcvMsg (CID, Protocol, From, or, \_) .

# Example Scenarios ...

## Real Time Enterprise

Reactions to threats and opportunities according to events in business transactions



## BAM, Agile ITSM and BPM

Monitor and detect IT service, SLA, and business behavior exceptions from observed events



## Semantic Systems

**Detect  
Decide  
Respond**

Valuable Information at the Right Time to the Right Recipient



**Dynamic  
Information  
Dissemination**

Enterprise Decision Management

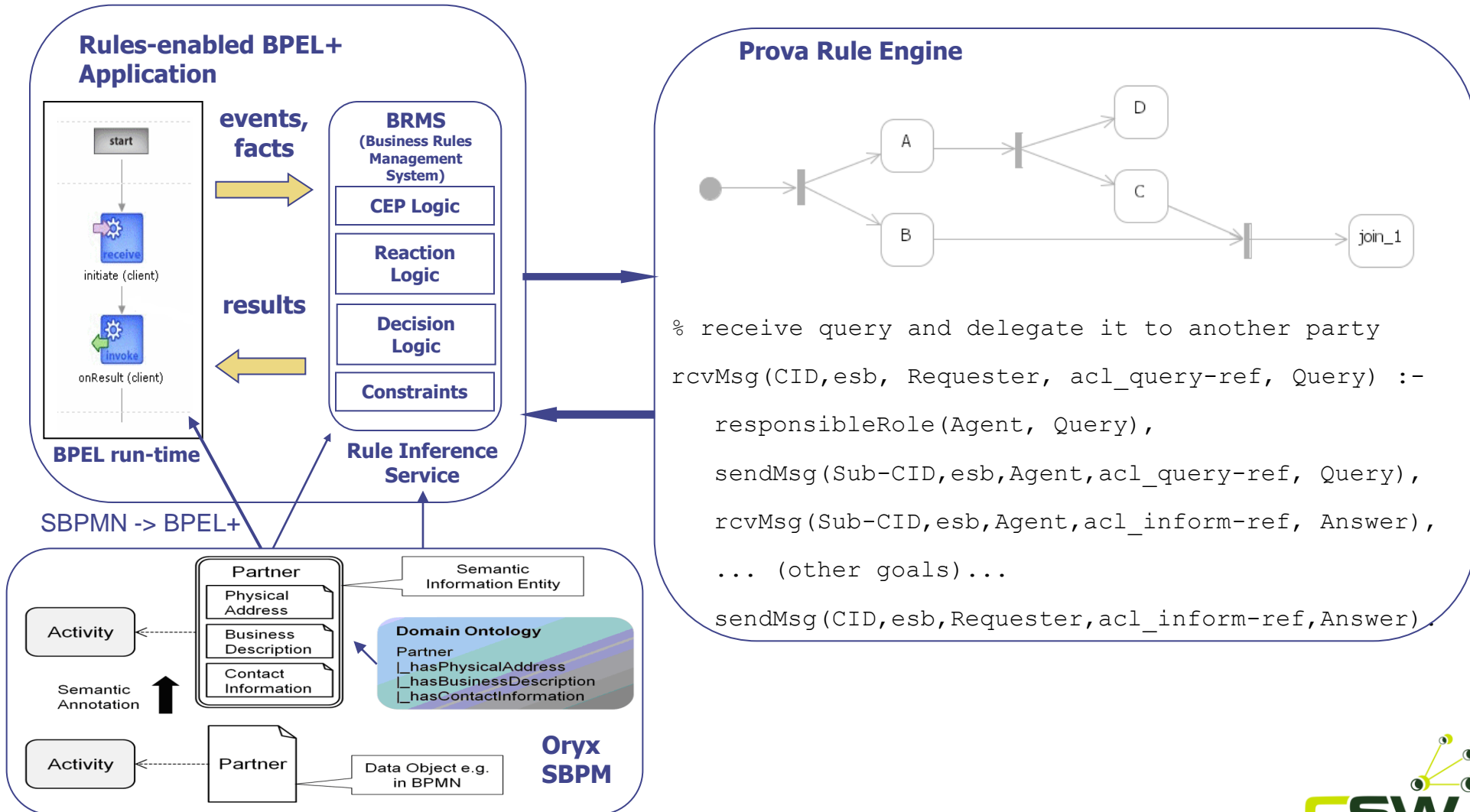
Real-time Decision Guidance and Expert Decision Support

**Real Time Decision  
Management**



# Summary

# Summary: Semantic BPM Approach



# Summary – Key Benefits of SBPM

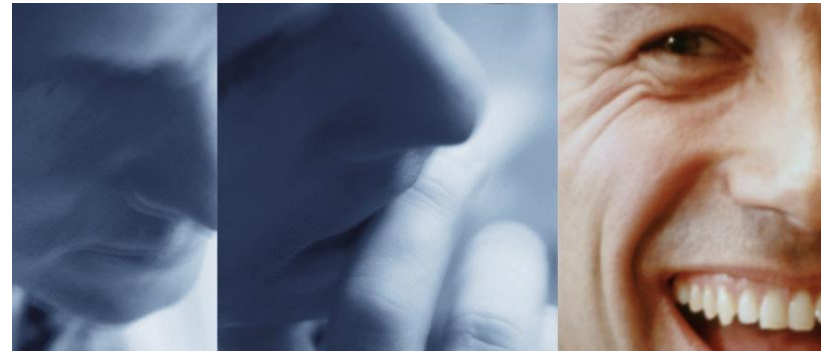
---

- Complementary technologies: semantic technologies + BPM technologies
- BPMN-BPEL for orchestration of services, systems, people & partners
- Rules focus on decision making and policies
- Rules can be used to integrate choreography sub-workflows in orchestrated BPEL processes
- Declarative specification of constantly changing business policies and regulations
- Agent Model enables business users to participate in business processes
- Modify and apply new rules without redeploying processes

# Summary for Semantic Event-Driven BPM

---

- Each (graphical) syntax concept in a BPM model is using an **ontology concept that references the semantics** of it
- Event processing engines **understand** what is happening in terms of (top-level and domain) ontologies
- Reaction rule engines **know** what (re)actions and processes they can invoke and what events they can signal
- Makes it possible to detect and **process semantically related (complex) events**
- Needs a **formal semantics** to concepts relevant in event processing, e.g. time, spatio, state, action behavior ...
  - Also support interchange across domain boundaries with different vocabularies



# Questions?