Enacting BPEL4WS Specified Workflows with Multiagent Systems

Paul Buhler¹  José M Vidal²

¹Department of Computer Science
  College of Charleston

²Department of Computer Science and Engineering
  University of South Carolina

Workshop on Web Services and Agent-Based Engineering
19 July 2004
BPEL4WS and Workflows

- The Business Process Execution Language for Web Services is a workflow language.
- XML language.
- Uses WSDL.
- Assumes SOAP.
- Widely adopted.
- Centrally and statically enacted.

GOAL
Enact BPEL4WS workflows using dynamically created multiagent systems.
BPEL4WS and Workflows

- The Business Process Execution Language for Web Services is a workflow language.
- XML language.
- Uses WSDL.
- Assumes SOAP.
- Widely adopted.
- Centrally and statically enacted.

GOAL
Enact BPEL4WS workflows using dynamically created multiagent systems.
Sample Workflow

Stock Lookup Process

1. request
2. invoke getStockQuote
3. invoke getExchngRate
4. invoke multiplyFloat
Centralized Enactment

- BPWS4J enactment engine from IBM AlphaWorks.

Activities include:
- getStockQuote
- getExchangeRate
- multiplyFloat

Process flow:
- stock lookup request
- BPEL Workflow
- BPWS4J

Buhler, Vidal
Enacting BPEL4WS Workflows with Multiagent Systems
Centralized Enactment

- BPEL Workflow
- getStockQuote
- getExchangeRate
- multiplyFloat

- stock lookup request

BPWS4J enactment engine from IBM AlphaWorks.
Centralized Enactment

- BPWS4J enactment engine from IBM AlphaWorks.

stock lookup request
Centralized Enactment

- BPWS4J enactment engine from IBM AlphaWorks.

Diagram:

- BPEL Workflow
  - getStockQuote
  - getExchangeRate
  - multiplyFloat

stock lookup request
Centralized Enactment

- BPWS4J enactment engine from IBM AlphaWorks.

Buhler, Vidal
Enacting BPEL4WS Workflows with Multiagent Systems
Centralized Enactment

BPWS4J enactment engine from IBM AlphaWorks.

- getStockQuote
- getExchangeRate
- multiplyFloat
Centralized to Decentralized Enactment

- Needed to coordinate agents.
- Used Colored Petri Nets and a native XML database.
- Agents decide who to call next: distributed control.
Colored Petri Mapping

Tokens carry data: control information.
Multiagent Workflow Enactment Example

**DESCRIPTION**
- XML database
- XML:DB API
- XPath queries
- data "distribution"

**WSAG**
- target agent
- DWA sQuote
- DWA mulFloat
- DF exRate
- getStockQuote
- mulFloat
- getExchangeRate

**Centralized Enactment**
- Xindice

---

**Distributed Enactment**

Buhler, Vidal
Enacting BPEL4WS Workflows with Multiagent Systems
Multiagent Workflow Enactment Example

**DESCRIPTION**
- Web Service Agent Gateway
- SOAP ↔ FIPA
- pubishes WSDL

**Web Service Agent Gateway (WSAG)**
- Xindice
- DWA sQuote
- DWA mulFloat
- getStockQuote
- target agent
- DWA exRate
- getExchangeRate
- mulFloat

Buhler, Vidal
Enacting BPEL4WS Workflows with Multiagent Systems
Centralized Enactment
Distributed Enactment

Multiagent Workflow Enactment Example

DESCRIPTION
- FIPA Directory Facilitator.
- JADE

DF

Xindice

description
◦ FIPA Directory Facilitator.
◦ JADE

DWA
sQuote

getStockQuote

dWA
mulFloat

mulFloat

dWA
exRate

getExchangeRate

target
agent
Multiagent Workflow Enactment Example

- **DESCRIPTION**
  - Web Services

- **WSAG**
- **Xindice**
  - **DWA sQuote**
  - **getStockQuote**
  - **DWA mulFloat**
  - **getExchangeRate**
  - **mulFloat**
  - **target agent**

- **DF**
Multiagent Workflow Enactment Example

- **DESCRIPTION**
  - interfaces WSAG to DWAs.

**Centralized Enactment**

**Distributed Enactment**

WSAG

- **Xindice**
- **DWA sQuote**
- **getStockQuote**
- **DWA mulFloat**
- **mulFloat**
- **DWA exRate**
- **DF**
- **getExchangeRate**

Buhler, Vidal

Enacting BPEL4WS Workflows with Multiagent Systems
Multiagent Workflow Enactment Example

DESCRIPTION
- Dist. Workflow Agents
- Interface to WS
- Implement control

WSAG

Xindice

DWA

sQuote

getStockQuote

target agent

DWA

mulFloat

mulFloat

DWA

exRate

getExchangeRate

DF
Multiagent Workflow Enactment Example

- Distributed Workflow
- Agent configuration
- could be automated
- need only one agent program

**DESCRIPTION**

- Distributed Workflow
- Agent configuration
- could be automated
- need only one agent program

**WSAG**

**DISTRIBUTED WORKFLOW**

- **Xindice**
- **getStockQuote**
- **mulFloat**
- **getExchangeRate**

**CENTRALIZED ENACTMENT**

- **DF**
- **target agent**
- **DWA sQuote**
- **BPEL workflow**

**MULTIAGENT WORKFLOW**

- **mulFloat**
- **DWA exRate**

Buhler, Vidal  Enacting BPEL4WS Workflows with Multiagent Systems
Multiagent Workflow Enactment Example

SOAP request

WSAG

Xindice

dTarget agent

dDWA sQuote

ggetStockQuote

dDWA mulFloat

mulFloat

dDWA exRate

ggetExchangeRate

DF
Multiagent Workflow Enactment Example
Multiagent Workflow Enactment Example

Centralized Enactment
Distributed Enactment

WSAG

target agent

store request

DWA sQuote

DWA mulFloat

getStockQuote

mulFloat

DWA exRate

getExchangeRate

Buhler, Vidal

Enacting BPEL4WS Workflows with Multiagent Systems
Multiagent Workflow Enactment Example

- **Xindice**
- **DWA sQuote**
- **getStockQuote**
- **WSAG**
- **target agent**
- **DWA mulFloat**
- **mulFloat**
- **locate partners**
- **DF**
- **DWA exRate**
- **getExchangeRate**
Multiagent Workflow Enactment Example

- **Centralized Enactment**
- **Distributed Enactment**

**Example: Xindice**
- **DF**: Target agent
- **DWA**: sQuote
- **DWA**: mulFloat
- **DWA**: exRate
- **FIPA ACL**

**WSAG**
- **getStockQuote**
- **mulFloat**
- **getExchangeRate**
Multiagent Workflow Enactment Example

get request info with XPath queries:
//agent[@role='requestor']/request//@symbol
//agent[@role='requestor']/request//@country

centralized enactment

distributed enactment

Buhler, Vidal
Enacting BPEL4WS Workflows with Multiagent Systems
Multiagent Workflow Enactment Example

- **Xindice**
  - Dynamic binding
  - SOAP invocation
  - getStockQuote

- **WSAG**
  - target agent
  - Dynamic binding
  - SOAP invocation
  - mulFloat

- **DF**
  - Dynamic binding
  - SOAP invocation
  - getExchangeRate

---

Buhler, Vidal  
Enacting BPEL4WS Workflows with Multiagent Systems
Multiagent Workflow Enactment Example

Xindice

DWA
sQuote

getStockQuote

reply

WSAG

target
agent

df

DWA
mulFloat

mulFloat

DWA
exRate

getExchangeRate

reply
Multiagent Workflow Enactment Example

- Xindice to DWA sQuote
- DWA mulFloat to DWA exRate
- DF to getExchangeRate
- target agent to getStockQuote

WSAG

Buhler, Vidal

Enacting BPEL4WS Workflows with Multiagent Systems
Multiagent Workflow Enactment Example

- **Xindice**
  - Target agent

- **DWA sQuote**
  - Get stock quote

- **DWA mulFloat**
  - Multiply floats

- **DWA exRate**
  - Get exchange rate

- **WSAG**

- **DF**
  - Locate partners

- **getStockQuote**
  - Agent target

- **mulFloat**
  - Locate partners

- **getExchangeRate**
  - Partner locate

---

**Centralized Enactment**

**Distributed Enactment**

Buhler, Vidal

Enacting BPEL4WS Workflows with Multiagent Systems
Multiagent Workflow Enactment Example

Buhler, Vidal
Enacting BPEL4WS Workflows with Multiagent Systems
Multiagent Workflow Enactment Example

- Xindice
- DWA sQuote
- DWA mulFloat
- getStockQuote
- get SOAP request message
- target agent
- DWA exRate
- getExchangeRate
- mulFloat
Multiagent Workflow Enactment Example

- Xindice
- WSAG
- DWA
- target agent
- DF
- DWA
- exRate
- DWA
- sQuote
- getStockQuote
- DWA
- mulFloat
- getExchangeRate
- dynamic binding
- SOAP invocation
Multiagent Workflow Enactment Example

WSAG
DWA
mulFloat
DWA
exRate
DF

Xindice
DWA
sQuote
getStockQuote

target agent
reply
mulFloat
getExchangeRate

Buhler, Vidal
Enacting BPEL4WS Workflows with Multiagent Systems
Multiagent Workflow Enactment Example

- Xindice
- DWA
  - sQuote
  - getStockQuote
  - store
  - response

- DWA
  - mulFloat

- DWA
  - exRate
  - getExchangeRate

- WSAG
- target agent
- DF

Buhler, Vidal Enacting BPEL4WS Workflows with Multiagent Systems
Multiagent Workflow Enactment Example
Multiagent Workflow Enactment Example

- Xindice
- DWA sQuote
- getStockQuote
- target agent
- DWA mulFloat
- mulFloat
- WSAG
- build response
- DWA exRate
- getExchangeRate
- DF
Multiagent Workflow Enactment Example

Centralized Enactment
Distributed Enactment

Multiagent Workflow Enactment Example

WSAG \rightarrow \text{reply} \rightarrow \text{target agent} \rightarrow DWA \text{exRate} \rightarrow DWA \text{mulFloat} \rightarrow DWA \text{sQuote} \rightarrow \text{getStockQuote} \rightarrow \text{Buhler, Vidal}
Multiagent Workflow Enactment Example

Centralized Enactment
Distributed Enactment

SOAP request
reply

WSAG

Xindice

DWA sQuote

getStockQuote

DWA mulFloat

mulFloat

DWA exRate

getExchangeRate

Buhler, Vidal  Enacting BPEL4WS Workflows with Multiagent Systems
Lessons Learned

- It is possible and practical to use Web services as external behaviors, allowing for generic agent code.
- Not unmarshalling SOAP responses insulates code from differences in RPC versus DOC SOAP styles.
  - XML database is natural.
  - Allows for fully stubless dynamic invocation as no class is required to hold unmarshalled response.
  - Preserves namespace information which in the future will likely have semantic importance.
- DF useful for dynamic allocation.
- Hybrid coordination model proved to be flexible.
- WSAG allows WS clients to interact with the MAS transparently and enables MAS to be used as a sub-process within another workflow enacted by a commercial workflow engine.
Lessons Learned

- It is possible and practical to use Web services as external behaviors, allowing for generic agent code.
- Not unmarshalling SOAP responses insulates code from differences in RPC versus DOC SOAP styles.
  - XML database is natural.
  - Allows for fully stubless dynamic invocation as no class is required to hold unmarshalled response.
  - Preserves namespace information which in the future will likely have semantic importance.
- DF useful for dynamic allocation.
- Hybrid coordination model proved to be flexible.
- WSAG allows WS clients to interact with the MAS transparently and enables MAS to be used as a sub-process within another workflow enacted by a commercial workflow engine.
Lessons Learned

- It is possible and practical to use Web services as external behaviors, allowing for generic agent code.
- Not unmarshalling SOAP responses insulates code from differences in RPC versus DOC SOAP styles.
  - XML database is natural.
  - Allows for fully stubless dynamic invocation as no class is required to hold unmarshalled response.
  - Preserves namespace information which in the future will likely have semantic importance.
- DF useful for dynamic allocation.
- Hybrid coordination model proved to be flexible.
- WSAG allows WS clients to interact with the MAS transparently and enables MAS to be used as a sub-process within another workflow enacted by a commercial workflow engine.
Lessons Learned

- It is possible and practical to use Web services as external behaviors, allowing for generic agent code.
- Not unmarshalling SOAP responses insulates code from differences in RPC versus DOC SOAP styles.
  - XML database is natural.
  - Allows for fully stubless dynamic invocation as no class is required to hold unmarshalled response.
  - Preserves namespace information which in the future will likely have semantic importance.
- DF useful for dynamic allocation.
- Hybrid coordination model proved to be flexible.
- WSAG allows WS clients to interact with the MAS transparently and enables MAS to be used as a sub-process within another workflow enacted by a commercial workflow engine.
Lessons Learned

▶ It is possible and practical to use Web services as external behaviors, allowing for generic agent code.
▶ Not unmarshalling SOAP responses insulates code from differences in RPC versus DOC SOAP styles.
  ▶ XML database is natural.
  ▶ Allows for fully stubless dynamic invocation as no class is required to hold unmarshalled response.
  ▶ Preserves namespace information which in the future will likely have semantic importance.
▶ DF useful for dynamic allocation.
▶ Hybrid coordination model proved to be flexible.
▶ WSAG allows WS clients to interact with the MAS transparently and enables MAS to be used as a sub-process within another workflow enacted by a commercial workflow engine.
Lessons Learned

- It is possible and practical to use Web services as external behaviors, allowing for generic agent code.
- Not unmarshalling SOAP responses insulates code from differences in RPC versus DOC SOAP styles.
  - XML database is natural.
  - Allows for fully stubless dynamic invocation as no class is required to hold unmarshalled response.
  - Preserves namespace information which in the future will likely have semantic importance.
- DF useful for dynamic allocation.
  - Hybrid coordination model proved to be flexible.
  - WSAG allows WS clients to interact with the MAS transparently and enables MAS to be used as a sub-process within another workflow enacted by a commercial workflow engine.
Lessons Learned

- It is possible and practical to use Web services as external behaviors, allowing for generic agent code.
- Not unmarshalling SOAP responses insulates code from differences in RPC versus DOC SOAP styles.
  - XML database is natural.
  - Allows for fully stubless dynamic invocation as no class is required to hold unmarshalled response.
  - Preserves namespace information which in the future will likely have semantic importance.
- DF useful for dynamic allocation.
- Hybrid coordination model proved to be flexible.
- WSAG allows WS clients to interact with the MAS transparently and enables MAS to be used as a sub-process within another workflow enacted by a commercial workflow engine.
Lessons Learned

▶ It is possible and practical to use Web services as external behaviors, allowing for generic agent code.

▶ Not unmarshalling SOAP responses insulates code from differences in RPC versus DOC SOAP styles.
  ▶ XML database is natural.
  ▶ Allows for fully stubless dynamic invocation as no class is required to hold unmarshalled response.
  ▶ Preserves namespace information which in the future will likely have semantic importance.

▶ DF useful for dynamic allocation.

▶ Hybrid coordination model proved to be flexible.

▶ WSAG allows WS clients to interact with the MAS transparently and enables MAS to be used as a sub-process within another workflow enacted by a commercial workflow engine.
Future Work

- Implement switch and pick BPEL4WS constructs.
- Aggressive dynamic allocation.
- Integrate with Semantic Discovery Services.
- Workflow as skeletons for adaptive multiagent systems?