Model Driven Dynamic Composition of Web Services Flow for Business Process Integration

Liang-Jie Zhang, Jen-Yao Chung
IBM T. J. Watson Research Center

April, 2003
Agenda

- Introduction

- Adaptive Web Services Composition Framework
  - Major Components

- Research prototype
  - Web based composition portal
  - Eclipse composition toolkit

- Summary
Why Need Dynamic Web Service Composition?

- A business process includes many "stateful," long-running interactions involving two or more parties. Creating business processes on demand is important for meeting changing customer requirements.

- Companies offering new business processes composed by Web Services is a direct and effective way in which an organization can control costs, improve profits, and enhance overall responsiveness to changing market dynamics. (outsourcing)

- Selection and composition of services in a drag-and-drop manner is time-consuming manual labor. Dynamically composing the flow of Web services via discovery and selection instead of manual coding is imperative for developers.
1. Automatically generate business process flow skeleton

2. Dynamically bind the tasks to specific Web services provided by multiple service providers

3. Adaptively tune the constructed business process flow on a business process engine.
Business Process Flow Languages

- **BPEL4WS**: Business Process Execution for Web Services
  - IBM, BEA, Microsoft
  - Convergence of WSFL and XLANG

- **WSCl**: Web Service Choreography Interface (message-flow oriented interface)
  - BEA Systems, Intalio, SAP AG, and Sun Microsystems
Problems to be solved

- Lack **uniformed representation** to capture customers’ requirements, preferences, QoS, Web Services features, event-action mappings as well as the relationships among Web Services

- Lack an **automation mechanism** to generate search script to dynamically find appropriate Web Services for performing a specific task from UDDI registries with thousands or tons of Web Services records
Problems to be solved (cont.)

- Lack **effective service selection mechanism** to automatically construct an optimal business process using available Web Services

- Lack **efficient tooling** to support dynamic adaptation of Web Services flow to different modeling languages
Agenda

- Introduction

**Adaptive Web Services Composition Framework**
- Major Components

- Research prototype
  - Web based composition portal
  - Eclipse composition toolkit

- Summary
A business process consists of multiple Web services which maybe provided by different service providers.

A framework should enable dynamic composition of Web services flow based on customer requirements (Business Process On Demand).

Web Services Outsourcing Manager (WSOM) is an example dynamic composition framework.
Adaptive Web Services Flow Composition Framework

- Major Components:
  - Requirement annotation document to describe the customer’s requirement, preferences, QoS and relationships
  - Generating Search Script based on the requirement annotation document for Leveraging advanced Web Services discovery engine: First level service selection
Adaptive Web Services Flow Composition Framework (cont.)

- Using optimization algorithm for service selection (aka. service matchmaking) to configure the optimal Web services flow to satisfy customer requirements (e.g. lowest cost, shortest time): **Second level service selection**

- Final Output: WSFL or other Web Services business process languages (BPEL4WS)
Features of the Composition Framework

- Fully automate the end-to-end composition of business processes by using existing Web services.

- Bridge the gap between the informal, subjective customer requirements and the objective, machine-readable flow language, such as BPEL4WS, WSFL, and XLANG.

- Automate the generation of scripts for searching for Web services; and it automates the process of selecting Web services by using a "pluggable" optimization framework.

- Monitor and tune the composed business process so that it adapts to the new requirements at run time.
Agenda

- Introduction

- Adaptive Web Services Composition Framework
  - Major Components

- Research prototype
  - Web based composition portal
  - Eclipse composition toolkit

- Summary
GUI based tool (e.g. HoloSofx Workbench) could be used to create a workflow template.
Internal Representation of a Flow Template

OMG WS Workshop
Example: Travel Planning Process Skeleton

The model contains the following nine activities:

1. Start
2. Purchase a game ticket (Match Ticket)
3. Purchase transportation ticket (Transport Ticket)
4. Book a hotel room (Hotel)
5. Shop for the trip (Shop)
6. Collect news about the game (News Collection)
7. Reserve a pickup service (Pickup)
8. Make a sightseeing plan (Travel)
9. Make a restaurant reservation (Food)
10. Watch the game (Match)
11. End
Generate Requirement Annotation Document

- Two major parts in Requirement Annotation Document
  - Flow rules
    - Parallel services
    - Sequential services
  - Customer requirements
    - Preferences (e.g. QoS)
    - Business rules
    - Relationship links
    - Event-action mappings
Example: WSOM Web Portal - Screens

Web Service Outsourcing Manager
Dynamic Web Service Flow Composition Framework

Introduction

A business user wants a number of activities in his/her business process. The activities can be implemented or realized by Web services. With the activities in mind, the user can create a process model considering their execution order. Web Services Outsourcing Manager (WSOM) is a software framework providing a mathematical model for dynamic business process flow composition using existing Web services to meet the business users' requirements and the preference. This Web interface to the WSOM framework enables a business user.

1. Define a Business Process Model by specifying the preferences and relationships activities.
2. Define Business Rules associated with activities.
3. Generate an XML Requirement Document specifying the defined process model with the rules.
4. Run the engine and find a Business Solution that fulfills the requirements of the activities.
5. Generate an XML Solution Document specifying the business solution.

Demo Scenario

In this demo, we present a business process scenario for a soccer fan planning a trip for watching a game. To show the capabilities of the WSOM framework, this demo uses a particular process model as shown below:
Example: Process Model

2. Process Model

In this step, the user creates a business process model by defining each business activity in the model with:

1. Zero or more following activities,
2. Business requirements for various factors, e.g., cost, time, benefit, and quality, and

To create an entire process model, the user needs to repetitively use this step for each and every activity.

Select the target service

<table>
<thead>
<tr>
<th>Service</th>
</tr>
</thead>
<tbody>
<tr>
<td>Match Ticket</td>
</tr>
</tbody>
</table>

Select one or more following services

<table>
<thead>
<tr>
<th>Service</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start</td>
</tr>
<tr>
<td>Match Ticket</td>
</tr>
<tr>
<td>Transport Ticket</td>
</tr>
<tr>
<td>Hotel</td>
</tr>
<tr>
<td>Shopping</td>
</tr>
<tr>
<td>News Collection</td>
</tr>
<tr>
<td>Pickup</td>
</tr>
<tr>
<td>Travel</td>
</tr>
<tr>
<td>Food</td>
</tr>
<tr>
<td>Match</td>
</tr>
<tr>
<td>End</td>
</tr>
</tbody>
</table>

Requirements for target service

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cost ($)</td>
<td>100</td>
</tr>
<tr>
<td>Time (Days)</td>
<td>5</td>
</tr>
<tr>
<td>Discount ($)</td>
<td>10</td>
</tr>
<tr>
<td>Quality (0 - 1.0)</td>
<td>0.8</td>
</tr>
</tbody>
</table>
Example: Business Rules Associated with Activities

Web Service Outsourcing Manager
Dynamic Web Service Flow Composition Framework

3. Business Rules

In this step, the user specifies various business rules associated with business activities. The user can define a business rule by specifying:

1. Business rule type,
2. Business activity,
3. Business rule priority, and
4. Consideration factor and its maximum allowable value.

To create an entire set of necessary business rules, the user needs to repetitively use this step.

<table>
<thead>
<tr>
<th>Select a rule type</th>
<th>Condition-Service</th>
</tr>
</thead>
<tbody>
<tr>
<td>Select a service cluster</td>
<td>Match Ticket</td>
</tr>
<tr>
<td>Rule priority (The bigger, the higher)</td>
<td></td>
</tr>
<tr>
<td>Condition (Maximum allowable)</td>
<td>Cost ($)</td>
</tr>
</tbody>
</table>
Example: Requirement Annotation Document Creation

Web Service Outsourcing Manager
Dynamic Web Service Flow Composition Framework

4. XML Requirement Document

Your XML Requirement Document (in BPOL) is created. You can download the Requirement Document. Now you are ready to generate the optimal Business Solution. (This will take a few minutes. Please be patient.)
Step 5: Create Business Solution

Web Service Outsourcing Manager

Dynamic Web Service Flow Composition Framework

1. Home
2. Process Model
3. Business Rules
4. Requirement Document
5. Business Solution
6. Solution Document

5. Business Solution

Your Business Solution is generated. You can review the solution in an XML Solution Document.

Free, Regular Train, Kia Hotel, Beijing's Modern Business City, Soccer Week, Supper Shuttle, China Travel Agent, Korea Food, Korea World Cup Students Association,
Web Service Outsourcing Manager

Dynamic Web Service Flow Composition Framework

6. XML Solution Document

Your XML Solution Document (in WSFL) is created. You can download the Solution Document. This is the end of the demo of using WSOM for business solutions. Thanks!

You can go back Home to model another business process.
Example: WSOM Eclipse Edition

- Capture Customer’s Requirements
- Capture Business Rules
- Generate Requirement Annotation Document
- Configure Optimal Business Process
- Generate WSFL (or BPEL4WS) Skeleton
WSOM (Eclipse Screens)

1. Customer Requirements
2. Business Rules
3. Service Selection Algorithm (GA)
4. BE4WS
5. Dynamic Business Process Configuration
6. WSFL or BPEL4WS
Result: Trip Planning Business Process

The final business process is constructed by the following service set.

\[ S^* = \{0010, 001000, 01, 010, 001, 001, 10, 100, 010\} \]
Agenda

- Introduction

- Adaptive Web Services Composition Framework
  - Major Components

- Research prototype
  - Web based composition portal
  - Eclipse composition toolkit

- Summary
Integrating Model Driven Tool with BPEL4WS

1. BPEL4WS -- (Abstract Level)
   Requirement Annotation

2. BPEL4WS Web Services Flow

3. Dynamic Finding and Binding of Web Services
   WSOM

Workflow template (Business level)

BPEL4WS
Executable XML
With Real WS Bindings

OMG WS Workshop
Summary

- A framework that enables dynamic composition of Web services flow
  - Using requirement annotation document to capture customers’ requirements for Web services outsourcing (Skeleton based)
  - Automatic search script generation
  - Optimal business process configuration to match customer’s requirements

- WSOM can be extended to support Grid Services Composition
Resources

- Business Process Execution Language (BPEL4WS)

- Web Services Outsourcing Manager

- Web Services Toolkit
  - BE4WS (http://www.alphaworks.ibm.com/tech/be4ws)
  - WSIL Explorer

- Conference Invitation
  - 2003 International Conference on Web Services (ICWS’03)
    - http://tab.computer.org/tfec/icws03
    - June 23 - 26, 2003, Las Vegas, Nevada, USA