Integrating Legacy Assets Using J2EE Web Services

Jonathan Maron
Oracle Corporation
Agenda

- SOA-based Enterprise Integration
- J2EE Integration Scenarios
  - J2CA and Web Services
- Service Enabling Legacy Resources
  - Service-Relational Mapping
  - Service-MOM Mapping
- WSIF and JBI
- BPEL Integration
Agenda

- SOA-based Enterprise Integration
- J2EE Integration Scenarios
  - J2CA and Web Services
- Service Enabling Legacy Resources
  - Service-Relational Mapping
  - Service-MOM Mapping
- WSIF and JBI
- BPEL Integration
What Is SOA?

- IT architecture for request - reply applications
- Application functions are modularized and presented as services
- Services are loosely coupled
  - Service interface is independent of the implementation
Characteristics of SOA

- Services have platform independent, self describing interfaces (XML)
- Messages are formally defined
- Services can be discovered
- Services have quality of service characteristics defined in policies
- Services can be provided on any platform

SOA

Composable

Interoperable

Re-Usable

Loosely Coupled
Why SOA?

Source: Gartner
SOA-based Integration

- Respond to business changes
- Address new needs with existing applications
- Unlock existing application investments
- Support new channels & complex interactions
- Support organic business

Oracle Financials
- Standard Interface

Custom Order Application
- Standard Interface

New Web Application
- Standard Interface

New Supply Chain Management Application (Business Process or Composite Application)
- Standard Interface

Siebel CRM

Mainframe HR Application

.NET Inventory Application
Agenda

- SOA-based Enterprise Integration
- J2EE Integration Scenarios
  - J2CA and Web Services
- Service Enabling Legacy Resources
  - Service-Relational Mapping
  - Service-MOM Mapping
- WSIF and JBI
- BPEL Integration
Java Connector Architecture

- J2EE Application Server
- Container-Component Contract
- J2EE Application Component
- Client API
- System Contracts
- Resource Adapter
- EIS specific contracts
- Enterprise Information System
- Lifecycle Management
- Work Management
- Transaction Management
- Security Management
- Connection Management
**J2CA Advantages**

Simplify EIS integration by providing:

- Comprehensive integration framework
- Ease of Integration (reduce integration points/cost)
- Industry Standard
- Flexibility
- Not limited to connectivity, can be used for other services.
J2CA and Web Services

Order Entry

Credit Rating

United Loan

Fulfillment

Loan Application

Application Server

Star Loan
Agenda

- SOA-based Enterprise Integration
- J2EE Integration Scenarios
  - J2CA and Web Services
- Service Enabling Legacy Resources
  - Service-Relational Mapping
  - Service-MOM Mapping
- WSIF and JBI
- BPEL Integration
Integration Scenario

Web Service Distributed Management
- Web Service Choreography
- Web Service Orchestration
- Web Service Coordination
- Web Service Transactions
- Web Service Security
- Web Service Policy
- Web Service Reliable Messaging

J2EE 1.4

WS-I Basic Profile

SOAP, WSDL, UDDI

J2EE Application Server

TPM, Mainframe, Legacy Sys

Applications

Databases

B2B Partners

SOA Tools
App Dev Framework
J2EE Web Service Integration

Web Service Client
(J2EE, .NET, Portal, Mobile, PL/SQL …)

SOAP 1.1/1.2

Request Handler (HTTP JMS)

WS-Security

WS-Reliability

Message Auditing

Content Based Logging

Service Management

Stateless Session EJB
EJB Container

Java Class

PL/SQL/AQ/ DML/SQL

JMS Topic Queue

Corba

Custom Provider
Servlet Container

Transact’ns
Messaging
Security
Data Access
Pooling

J2EE Application Server

Find

Publish

WSDL

UDDI

Service Management
Service Relational Motivation

Web Service Client ➔ SOAP ➔ Application Server ➔ JDBC

- PL/SQL
- Java
- SQL/DML
- XML
- SQL/Query
- AQ/Streams
Service-Relational Mapping

- Enterprises have significant investment in their databases
- Many years of effort
- Inability to leverage existing assets may prevent uptake of new technologies
- Technologies exist to “service-enable” database assets
Database Web Services

- Existing assets accessible as Web services
  - Stored Procedures
  - Embedded Java
  - SQL Queries and DML
  - Database XML facilities
  - Other features
    - Messaging

- Native and middle tier implementations
- Service callouts available as well
Database Web Services

Application Server

- Endpoint implicitly specifies the type of service provided by the server
- SOAP

Web Services Runtime
- Decoding
- Encoding

SOAP Libraries
- XML Parser

O/R Java Classes
- JDBC

Libraries
- Java Classes
Database Web Service

```java
package company as
/* add a new employee */
procedure addemp (emp IN employee);
/* get employee */
function getemp (id IN int) return employee;
/* get employee address */
...
end company;

public interface DatabaseWebService extends java.rmi.Remote{

    public void addemp(EmployeeUser emp) throws java.rmi.RemoteException;
    public EmployeeUser getemp(Integer id) throws java.rmi.RemoteException;
    ...
}
```
Demonstration

Database Web Services
Service-MOM Mapping

- Leveraging existing Message-Oriented Middleware assets
- Messaging integral to processes of enterprise
- Natural mapping between messaging technologies
- Leverage messaging infrastructure transport and/or destinations
JMS Web Services

- WSDL
- UDDI Registry
- Application Server
- JMS Topic
- JMS Queue
- SOAP Servlet
- SOAP Stub
- Java/Servlet
- Send Client
- SOAP Servlet
- SOAP Stub
- Java/Servlet
- Receive Client
- Client(s)

Java/Servlet

Send Client

SOAP Stub

SOAP

HTTP

Message Driven Bean
JMS Transport

- Leverage existing messaging infrastructure
  - Guaranteed message delivery
- Interact with existing messaging components
JMS Web Service

```xml
<queue name="The MDB Queue" location="jms/ws/mdb/theQueue"/>
<queue name="Log MDB Queue" location="jms/ws/mdb/logQueue"/>
<queue-connection-factory
location="jms/ws/mdb/theQueueConnectionFactory"/>
<queue-connection-factory
location="jms/ws/mdb/logQueueConnectionFactory"/>
```

```java
public interface JmsDestination extends java.rmi.Remote {
    public javax.xml.soap.SOAPElement receive(…) throws java.rmi.RemoteException;
    public void send(…) throws java.rmi.RemoteException;
}
```
Demonstration

JMS Web Services
Agenda

- SOA-based Enterprise Integration
- J2EE Integration Scenarios
  - J2CA and Web Services
- Service Enabling Legacy Resources
  - Service-Relational Mapping
  - Service-MOM Mapping
- WSIF and JBI
- BPEL Integration
WSIF and JBI

Application

Data Binding

WSIF & JBI

In Memory  REST  JCA  COM+  SOAP

Java  Yahoo  SAP  .net  Oracle
Web Service Invocation Framework (WSIF)

- Use native protocols rather than SOAP
- Best of both Web services and native protocols
WSIF Bindings

- Supported bindings include:
  - J2CA
  - Java, EJB
  - JMS
  - HTTP GET/POST
  - TCP pure socket binding
  - Interconnect, Database, Flat File

- New bindings can be written as needed
WSIF Java Binding WSDL

- <types>
  + <schema attributeFormDefault="qualified" elementFormDefault="qualified"
    targetNamespace="http://services.otn.com"
    xmlns="http://www.w3.org/2001/XMLSchema">
  </types>
+ <message name="AddCommentRequestMessage">
+ <message name="AddCommentResponseMessage">
  - <portType name="HelperService">
    - <operation name="addComment">
      <input message="tns:AddCommentRequestMessage" />
      <output message="tns:AddCommentResponseMessage" />
    </operation>
  </portType>
  - <binding name="JavaBinding" type="tns:HelperService">
    <java:binding />
    - <format:typeMapping encoding="Java" style="Java">
      <format:typeMap typeName="tns:commentType"
        formatType="com.onv.services.CommentType" />
      <format:typeMap typeName="tns:commentsType"
        formatType="com.onv.services.CommentsType" />
    </format:typeMapping>
    - <operation name="addComment">
      <java:operation methodName="addComment" />
      <input />
      <output />
    </operation>
  </binding>
- <service name="HelperService">
  <documentation>HelperService</documentation>
  - <port name="JavaPort" binding="tns:JavaBinding">
    <java:address className="com.onv.services.HelperService" />
  </port>
</service>
WSIF Java Binding Code

public void invokeAddComment () throws Exception {
    WSIFServiceFactory factory = WSIFServiceFactory.newInstance();
    WSIFService service =
        factory.getService("wsdl/testWSIFJAVA_doclit.wsdl",
        null,
        null,
        "http://services.otn.com",
        “HelperService”);
    HelperService stub = (HelperService) service.getStub
        “JavaPort”, HelperService.class);
    stub.addComment ( . . .);
}
Demonstration

WSIF
What Is JBI?

- Unified Metadata (WSDL)
- Unified Addressing (Logical Naming)
- Plug-and-play Binding Components
- Optimized Transport and Data Format
- Both Sync. and Async. Interactions

Service Engines:
- BPEL
- XQuery
- Rules

Binding Components:
- SOAP
- HTTP
- JCA
- In Memory
- COM

Services:
- Amazon
- SAP
- C#
JBI Advantages

- Core design is made for service orientated service provision and consumption
- Encapsulation of technical binding and technology
- Introduce meanings for standard installation and artifact deployment
- Introduce a standard way for managing components
- Support for mediated XML messaging (unlike JCA that is designed to provide EIS connectivity)
- Avoid Vendor Lock-in
Agenda

- SOA-based Enterprise Integration
- J2EE Integration Scenarios
  - J2CA and Web Services
- Service Enabling Legacy Resources
  - Service-Relational Mapping
  - Service-MOM Mapping
- WSIF and JBI
- BPEL Integration
Business Process Framework

How do I …

- Coordinate asynchronous communication between services
- Correlate message exchanges between parties
- Implement parallel processing of activities
- Manipulate/transform data between partner interactions
- Support for long running business transactions and activities
- Provide consistent exception handling
Jon Maron — Integrating Legacy Assets Using J2EE Web Services

BPEL

CREATE ORDER PROCESS FLOW

Create

Validate

Approve

Part 1

Part 2

BAM

Workflows

JBI

JSF

Order

Submit

...Customizable

...Services and Events

Oracle, PeopleSoft
SAP, Mainframe
.NET

Data Hubs

...User Activity

...Java Binding

...Built-in reporting

...Standard
Demonstration

BPEL
Agenda

- SOA-based Enterprise Integration
- J2EE Integration Scenarios
  - J2CA and Web Services
- Service Enabling Legacy Resources
  - Service-Relational Mapping
  - Service-MOM Mapping
- WSIF and JBI
- BPEL Integration
Absolutely Shameless Self Promotion


- From amazon.com reviews:
  - “Required Enterprise Transactions Reading”
  - “All J2EE developers should read this book”