Service Oriented Architecture with JBI and ServiceMix

Bruce Snyder
LogicBlaze
Who Is This Guy?

- 10+ years of software development
  - Service Oriented Architecture
  - Event Driven Architecture
  - J2EE
  - Middleware
  - Relational Databases
Book Author

Professional Apache Geronimo

Bruce Snyder — Service Oriented Architecture with JBI and ServiceMix
Agenda

- Integration is Pervasive
- Service Oriented Architecture
- Custom Business Integration Solution Using Open Source
- Java Business Integration
- Apache ServiceMix
Business Integration Problem
Service Oriented Analogy

- Gated community in a Home owners assoc.
  - Provides services (cutting grass, plowing snow, providing cable, security, clubhouse w/ pool, etc.)
  - Used by many people (consumers)
  - HOA community is comprised of many owners
  - Adherence to basic conventions
    - HOA requirements
    - Address on house
    - Mailbox at end of driveway
    - Pay dues!
Service Oriented Architecture

- Language and platform neutral
- Integration services via encapsulation
- Loosely coupled
- Interoperability
- Open standards
- Reusable components
- Standards based
- Vendor diversity
Characteristics of Service Orientation

- Loose coupling
- Service contracts
- Autonomy
- Abstraction
- Reusability
- Composability
- Statelessness
- Discoverability
A Custom Business Integration Solution using Open Source

- Energy trading
- Pulling data from many disparate sources
- Custom ESB solution
  - All open source
  - JBoss, jBPM, JMS, EJB, Spring, Quartz
  - Ajax for web UI
Disadvantages

- Lack of standards
  - Custom adapters
  - Custom expansion
  - Custom maintenance

- Dependencies
  - Budgets
  - Knowledge base of developers
Java Business Integration

- Java Business Integration (JSR-208)
  - A standards-based architecture for enterprise integration
  - A standard for a container of containers with an API for integration amongst them
  - WSDL-based integration model
Plugin Component Architecture
JBI Characteristics

- Integration architecture via plug-in components
- WSDL message exchange model
  - Service providers
  - Service consumers
- WSDL service description model
  - Services decoupled from business logic
  - Orchestration houses business logic
- Message transformation
  - Independent of business logic
JBI Abstraction

Component A

send()

JBI determines service provider, Component B

accept()

Component B accepts message from JBI

Component A sends message to JBI
JBI Components

- Plug-in components provide and/or consume services
- Services described using WSDL
  - Abstract
  - Technology neutral
JBI Components (cont’d)

- **Service Engines**
  - Provide business logic

- **Binding Components**
  - May implement one or more communications protocol
  - Allow service engines to expose and consume services outside of the JBI bus
JBI Components (cont’d)

- Managed through JMX
  - Installing Components
  - Managing a component’s life cycle
  - Deploying service artifacts to components
Architecture of the JBI Environment

- WSDL-based Messaging Model
  - Abstract Service Model
  - Concrete Service Model
- Service Provider - performs a given service
- Service Consumer - invokes a given service
- WSDL Naming
  - Qualified
  - Simple
Abstract Service Model

- Provide an abstract definition of a service without reference to a protocol or wire encoding

- Abstract Message Types
  - Normal
  - Fault

- Abstract Operations
  - Operation Name
  - Message Exchange Pattern
  - Message Type

- Abstract Service Types
  - Interface name
  - Extended Interfaces
Concrete Service Model

- Map the abstract definition to a particular communications protocol and communications endpoint.
- Binding Types
- Endpoints
  - Endpoint name
  - Binding type
- Service
  - Service name
  - Service type name
  - Endpoints
WSDL Component Model
The JBI Environment
Normalized Message Exchange

- External Service Consumer
Normalized Message Exchange

- External Service Provider

```
1. SE sends ME
Consumer BC/SE
Delivery Channel

2. BC accepts ME
Normalized Message Router
Delivery Channel

3. BC sends request
External Service Provider

WS - I BC
```
Binding Components

- Communicates with providers or consumers via a protocol other than a normalized message on the NMR
  - Processes external to the JBI Container
- Act as sort of a proxy for external providers and consumers to other services within the JBI container
- Describe an endpoint in the NMR that represents a remote service
Binding Components (cont’d)

- Handle message normalization
- Can connect to a remote service using that remote service’s protocol
- May also perform translation or other services
Binding Components

Diagram showing the interaction between a SOAP Endpoint, HTTP(SOAP) Binding Component, and a JBI Container.
Service Engines

- Communicates with providers or consumers locally via normalized messages on the NMR
  - Process is internal to the JBI Container
  - The Service Engine is that process
- Either consumes or provides a service
Service Engines

![Diagram showing JBI Container and Service Engines A, B, and C]
JBI Packaging

- **Service Unit (SU)**
  - Component-specific artifact

- **Service Assembly (SA)**
  - Aggregate deployment file for service units
Apache ServiceMix

- Open source Enterprise Service Bus
- Combines SOA and Event Driven Architecture
  - http://enterpriseintegrationpatterns.com/docs/EDA.pdf
- Based on JBI
  - http://servicemix.org/site/what-is-jbi.html
- Connectivity centric
- Smart Routing capabilities
- Orchestration
- Management
Apache ServiceMix Architecture

ServiceMix Enterprise Service Bus

JBI Transformation, Routing, and Correlation Services

JBI Binding Components

BPEL  XSLT  Rules  Scripting

SOAP  Files  JCA Resources  Legacy Apps
ServiceMix JBI Components

- servicemix-bean
- servicemix-bpe
- servicemix-eip
- servicemix-file
- servicemix-ftp
- servicemix-http
- servicemix-jms
- servicemix-jsr181

- servicemix-lwcontainer*
- servicemix-sca
- servicemix-wsn2005
- servicemix-xmpp
External JBI Components

- BPEL support via PXE
- BpmScript
- JAFS
- Jbi4Corba
- Jbi4Cics
ServiceMix Lightweight Components

- Service Engines
  - Drools
  - Cache
  - Groovy
  - JCA
  - Quartz
  - Scripting
  - XSLT
  - Schema Validation
  - XSQL

- Transport Bindings
  - Email
  - File
  - FTP
  - HTTP
  - Jabber
  - JMS
  - RSS
  - VFS
Apache ActiveMQ

- ActiveMQ
  - Enterprise class messaging system
  - JMS 1.1, clustering, peer networks, discovery, TCP, SSL, multicast, persistence, XA
  - C, C++, C#/.Net, Python, PHP, Perl, Pike, Ruby and Java
  - Integration with Geronimo, Spring, Tomcat and any J2EE 1.4 container
  - Wicked fast, highly scalable, easy to use
  - Designed for embeddability
SOA with ServiceMix
Summary

- Integration is Pervasive
- Service Oriented Architecture
- Custom Business Integration Solution Using Open Source
- Java Business Integration
- Apache ServiceMix
Wrap Up

- Open Source ESB (JBI)
  - http://servicemix.org/
- Open Source JMS
  - http://activemq.org/
- Lingo
  - http://lingo.codehaus.org/
- Support and Services
  - http://logicblaze.com/
Thank You for Attending

- Questions and answers
- Please fill out your surveys