Swordfish
SOA Runtime Framework

Aleksander Pohl

Katedra Informatyki, Akademia Górniczo-Hutnicza

4 maja 2009
Agenda

Introduction

Technologies

Extensions

Development plan
<table>
<thead>
<tr>
<th>Introduction</th>
<th>Technologies</th>
<th>Extensions</th>
<th>Development plan</th>
</tr>
</thead>
</table>

Aleksander Pohl

Swordfish SOA Runtime Framework
“The goal of the Swordfish project is to provide an **extensible SOA runtime framework** based on the proven Eclipse Equinox runtime technology. The framework is designed to be complemented by additional open source components such as service registry, a messaging system, a process engine, etc. to form a comprehensive open source SOA runtime environment based on both established and emerging open standards.”
Swordfish

▫ Allows application developers and system integrators to build their own ESB
▫ Built upon
  ▪ Apache ServiceMix
  ▪ Apache CXF
  ▪ Eclipse Equinox
Features

- Integrates dynamic Service Registry
- Based on Distributed ESB pattern
- Remote configuration mechanism
- Extensive monitoring capabilities
- OSGi as a component model
- JBI 1.0/2.0 for messaging abstraction
Service registry

- Consumer refers to provider by logical identifier
- Logical identifier signifies the service’s interface
- Consumers Policy covers its non-functional capabilities and requirements
- Service Registry selects matching provider and calculates an effective Policy
- Service Registry will be complemented by a Service Repository that allows all SOA-related artifacts to be stored in reliable and traceable manner
Service registry

Logical service name + policy

Registry/Repository

Service Resolver

JBI SE Java

JBI BC HTTP

physical endpoint address

Provider 1

Provider 2

Provider 1

Provider 2
Distributed ESB

- No central components are needed after the initial communication set up
- The whole system is much more scalable
- Increased administration and management complexity are compensated for by
  - Remote configuration mechanism
  - Extensive monitoring capabilities
Distributed ESB

Service consumer
- Swordfish framework
  - Apache ServiceMix 4
  - Swordfish core
  - Configuration Plugin
  - Monitoring Plugin
  - Registry Plugin
- Process Engine (Apache ODE)
- Business Code (BPEL)
- SOAP Binding (Apache CXF)

Service provider
- Swordfish framework
  - Apache ServiceMix 4
  - Swordfish core
  - Configuration Plugin
  - Monitoring Plugin
  - Registry Plugin
- SOAP Binding (Apache CXF)
- Business Code (Java)
- JAX-WS Client (Apache CXF)

SOAP
- lookup
- register
- retrieve configuration
- retrieve configuration

Aleksander Pohl
Swordfish SOA Runtime Framework
OSGi and JBI

- OSGi provides
  - component model
  - deployment mechanism for modules
  - class loading system

- Subset of the JBI
  - messaging abstraction
  - message routing between components
  - JBI 1.0 component model dropped in favor of OSGi
Extending Swordfish

- General interceptor API
- Service resolver API
- Event API
- Configuration API
Interceptor API

- Swordfish hooks into the Normalized Message Router in Servicemix, intercepts all messages flowing back and forth and feeds them into an interceptor chain
- Interceptors in that chain can do (almost) anything with a message: read and modify payload, read and modify properties, re-route etc.
- The interceptor chain is dynamically built from all registered instances of Interceptor
- A component called Planner is responsible for building the chain
- A default implementation of Planner (DefaultPlanner) is part of the Swordfish core
Service resolver API

- The Service Resolver API builds upon the general interceptor API
- A special kind of interceptor (ServiceResolverInterceptor) is provided as part of the Swordfish core
- The ServiceResolverInterceptor is responsible for translating the (logical) service interface name carried inside a message exchange into a physical endpoint address where the message exchange will be ultimately sent to
- The ServiceResolverInterceptor collaborates with a number of other (replaceable) components in order to perform its task
Event API

- The Event system is the basis for Swordfish’s monitoring capabilities
- The Event API builds upon the OSGi EventAdmin Service
- Swordfish core generates Events of type TrackingEvent for each MessageExchange flowing through the NMR
- Other pre-defined Event types include OperationEvent and ConfigurationEvent
Configuration API

- The Configuration API builds upon the OSGi ConfigurationAdmin service.
- A ConfigurationAgent receives configurations from a ConfigurationSource (possibly a remote one) and provides them to other components through the OSGi Configuration Admin service.
- A component that implements ConfigurationConsumer receives updated configurations as they become available.
# Releases

<table>
<thead>
<tr>
<th></th>
<th>Galileo June 2009 Release 0.9</th>
<th>Eclipse Summit Europe November 2009 Release 1.0</th>
<th>EclipseCon March 2010 Release 1.1</th>
<th>Eclipse Release Summer 2010 Release 2.0</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ESB</strong></td>
<td>Swordfish framework basic plugins basic tool support</td>
<td>Integration with open source security framework</td>
<td>Test support</td>
<td>Full enterprise ESB</td>
</tr>
<tr>
<td><strong>Process Orchestration</strong></td>
<td>BPEL process engine</td>
<td>Integration with open source BPM suite (Spagic)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Registry &amp; Repository</strong></td>
<td>Basic runtime registry</td>
<td>Advanced runtime registry/service locator basic repository</td>
<td>Service repository w/service lifecycle management</td>
<td>Full enterprise service repository</td>
</tr>
<tr>
<td><strong>Management</strong></td>
<td>JMX-based management</td>
<td>Plugin for integration into Hyperic HQ</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Data Integration</strong></td>
<td></td>
<td>Plugins for open source ETL tools (e.g. Talend)</td>
<td>Plugins for open source EDI tools (e.g. Smooks)</td>
<td></td>
</tr>
<tr>
<td><strong>Service/Business Activity Monitoring</strong></td>
<td></td>
<td>Service activity reporting</td>
<td>Complex event processing</td>
<td>Full business activity monitoring</td>
</tr>
</tbody>
</table>
Resources

- http://www.eclipse.org/swordfish
- http://wiki.eclipse.org/Swordfish