An integrated approach to SOA tooling for ServiceMix and other platforms

Jerry Preissler, SOPERA GmbH
Agenda

- Introduction
  - What is „SOA Tooling“?
  - Identifying SOA Artefacts
  - Available Tools
  - Demo
Introduction

- The company
  - SOPERA GmbH, Bonn, Germany
  - Offering services for open source SOA software

- The speaker
  - Jerry Preissler, Dipl. Inform.
  - working in the SOA field since 2002

- Our involvement with Eclipse
  - initiated and leading the Eclipse Swordfish Project
  - contributing to Eclipse SOA Tools Platform Project
Agenda

- Introduction
- What is „SOA Tooling“?
- Identifying SOA Artefacts
- Available Tools
- Demo
SOA Tooling is about providing an integrated set of tools for developing for and operating an SOA.

- Create
- Edit
- Visualize
- Assist
- Validate
- Store
- Search
- Link
- Collaborate
- Version

Diagram:

- `<xmi>`
- `.....`
- `</xmi>`
- `</wsdl>`
- `Java`
- `transform`
Agenda

- Introduction
- What is „SOA Tooling“?
- Identifying SOA Artefacts
- Available Tools
- Demo
SOA Artefacts can be identified by analyzing the Service Development Lifecycle

- **Analysis**: Abstract Process Definition, Business Guidelines
- **Design**: Technical Process Definition, Configuration Directives, Business Policies, Rules Definition
- **Implementation**: Executable Process Definition, Participant Configuration, Technical Policies, Rules Implementation
- **Packaging**: Implementation package
- **Deployment**: Deployment package, Deployment descriptor, Infrastructure plan, Operational Policies, Operational monitoring configuration
- **Operation**: BAM configuration, SAM configuration

**Owners**
- Analysis: Business analyst
- Design: Service designer
- Implementation: Developer, Developer
- Packaging: Admin
- Deployment: Operator

**Process**
- Abstract Process Definition
- Technical Process Definition
- Executable Process Definition
- Configuration Directives
- Participant Configuration
- Business Policies
- Technical Policies
- Rules Definition
- Rules Implementation

**Config**
- Business Guidelines
- Deployment package
- BAM configuration
- SAM configuration

**Policies**
- Business Policies
- Technical Policies

**Rules**
- Rules Definition
- Rules Implementation

**Service**
- Service Specification
- Service Definition
- Service Implementation

**Artefact type**
- Process Definition
- Technical Process Definition
- Executable Process Definition
- Business Guidelines
- Configuration Directives
- Participant Configuration
- Business Policies
- Technical Policies
- Rules Definition
- Rules Implementation
- Service Specification
- Service Definition
- Service Implementation
- Deployment package
- BAM configuration
- SAM configuration
- Operational Policies
- Operational monitoring configuration
Applying the big picture to concrete use cases allows the identification of concrete artefacts

WSDL-first service development for ServiceMix
Agenda

- Introduction
- What is „SOA Tooling“?
- Identifying SOA Artefacts
- Available Tools
- Demo
Eclipse STP provides tools that address some of the artefacts.

<table>
<thead>
<tr>
<th>Artefact type</th>
<th>Process</th>
<th>Config</th>
<th>Policies</th>
<th>Rules</th>
<th>Service</th>
</tr>
</thead>
<tbody>
<tr>
<td>Artefact type</td>
<td>Definition</td>
<td>Policies</td>
<td>Rules</td>
<td>Service Specification</td>
<td>Service Creation</td>
</tr>
<tr>
<td>Owner</td>
<td>Business analyst</td>
<td>Service designer</td>
<td>Developer</td>
<td>Developer</td>
<td>Admin</td>
</tr>
<tr>
<td>Analysis</td>
<td>Design</td>
<td>Implementation</td>
<td>Packaging</td>
<td>Deployment</td>
<td>Operation</td>
</tr>
<tr>
<td>BPMN</td>
<td>BPEL 2 JAVA</td>
<td>Intermediate Model</td>
<td>SOA System</td>
<td>SOA Tools Platform</td>
<td>SOA Tools Platform</td>
</tr>
<tr>
<td>BAM configuration</td>
<td>SAM configuration</td>
<td>Operational Policies</td>
<td>Operational monitoring configuration</td>
<td>Deployment package</td>
<td>Deployment descriptor</td>
</tr>
</tbody>
</table>
SOPERA provides tooling for ServiceMix

<table>
<thead>
<tr>
<th>Analysis</th>
<th>Design</th>
<th>Implementation</th>
<th>Packaging</th>
<th>Deployment</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Business analyst</td>
<td>Service designer</td>
<td>Developer</td>
<td>Developer</td>
<td>Admin</td>
<td>TBD</td>
</tr>
</tbody>
</table>

- **Service definition**
  - WSDL
- **Service implementation**
  - business code
  - interface code
  - Eclipse project
- **Test config**
  - WSDL
- **Service assembly**
  - ZIP
- **Application**
  - SM3
- **Packaging**
- **Deployment**
- **Operation**

**Initial version available start Q2/08**

**Test gui**

**Test execution**
Agenda

- Introduction
- What is „SOA Tooling“?
- Identifying SOA Artefacts
- Available Tools
- Demo
The initial ServiceMix tooling available today will be expanded in the future

- will be released as open source
- support all other relevant use-cases:
  - code first, all interaction styles
- provide editors and validation for all artefact types
- improve testing:
  - automated testing, regression testing
The SOPERA platform complements the ServiceMix Core with advanced capabilities.

SOA Platform

SOPERA ToolSuite

Service Editor
Policy Editor
Test Consumer
Development Box

Admin Tool
Test Provider

SOPERA ToolSuite

ServiceMix (deploy & test)

SOPERA ESB

Management Services
Service Registry

Eclipse Swordfish
JBI Kern
ServiceMix

Notification Manager
Transformation Messaging
Transport

Security
Monitoring
SOPERA Advanced Services Framework

THE SOPERA ADVANCED SERVICES FRAMEWORK (ASF)

- **Design**
  - SOPERA ToolSuite
    - Service Editor
    - Process Editor
    - Policy Editor
    - Code Generation
  - DevBox
- **Delivery**
  - Infrastructure Administration
  - Eclipse
  - Other UML Modelling Tools
- **Operations**
  - Business Applications
  - Business Processes
  - Business Data/Legacy Systems
  - Third Party ESB's
  - SOA Security Services
  - Messaging/Transport Services
  - Process Management Services

- **Eclipse**
  - Service Monitoring
  - Ressource Management

- **Runtime Platform (J2EE, J2SE,.Net)**
  - SOA Tooling | © 2008 by SOPERA; made available under the EPL v1.0
SOPERA ToolSuite

Introduction

- **SOPERA graphical tools are based on Eclipse WTP:**
  - Perspectives inside Eclipse

- **Service Studio**
  - Used by service developer
  - Development of service descriptions and policy descriptions (service and policy editor)
  - Code generation

- **DevBox**
  - Complete development environment including basic testing capabilities
  - Used by developer

- **Administration Tool**
  - Interface to infrastructure services
SOPERA ToolSuite

ServiceStudio

- Editors for service-related artefacts
  - Service Description
  - Service Provider Description
  - Policies
  - Processes

- Wizards and validation
  - support creation of valid documents

- Common look & feel
  - GUI based on Eclipse WTP WSDL editor
  - raw XML available also
SOPERA ToolSuite

ServiceStudio - Service Editor

- Description Editor
  - edit service interface
- Provider Editor
  - edit service endpoint
SOPERA ToolSuite
ServiceStudio - Policy Editor

- **Operation Policy**
  - defines behaviour for one operation
  - plain WS-Policy

- **Participant Policy**
  - maps policies to service operations
  - uses WS-PolicyAttachment
SOPERA ToolSuite
Process Editor

- SOPERA builds upon the Eclipse BPEL Editor for process design support:
  - BPEL 2.0-compliant
  - Intuitive user interface
  - Direct deployment into process engine delivered with SOPERA ASF
  - Integration with Service Registry to facilitate re-use of existing services
SOPERA ToolSuite

Devbox - Scope

- Build an SOPERA infrastructure locally on the development system
  - Lightweight environment bundled with code generation, testing tools and demo participants

- Stand - alone
  - All required SOPERA components are integrated

- Provides SOPERA ASF run-time environment on one machine:
  - Authentication, Service lookup (authorization), XML document validation, Transport

- Components exchangeable with production counter parts
SOPERA ToolSuite

Devbox

- Infrastructure
  - LDAP, Tomcat, JMS
- Simple Consumer
  - for manual tests
  - Log file
  - Status table
- Test Consumer
  - batch-driven
  - for regression tests
SOPERA ToolSuite

Code generation

- **Service Description -> Java code**
  - creates consumer proxy, provider skeleton, JAXB-objects for types
  - available for JSE, JEE
  - code re-generation possible
SOPERA ToolSuite
Administration Tool

- provides access to central administration services
  - Service Registry, Configuration, Security Services

- task-based user interfaces available
  - GUI for manual inspection and modification
  - CLI for batch-mode operation
  - Tested with JRuby, BeanShell, JavaScript for scripting support

- Service packages to bundle all artefacts for a service
  - contains sdx, spdx, policies and mappings for service, domain or whole infrastructure
SOPERA ToolSuite
Administration Tool
An integrated approach to SOA tooling
for ServiceMix and other platforms

Jerry Preissler, SOPERA GmbH
Click to add title

Click to add text
Agenda

- Introduction
  - What is „SOA Tooling“?
  - Identifying SOA Artefacts
  - Available Tools
  - Demo
Introduction

- The company
  - SOPERA GmbH, Bonn, Germany
  - Offering services for open source SOA software

- The speaker
  - Jerry Preissler, Dipl. Inform.
  - working in the SOA field since 2002

- Our involvement with Eclipse
  - initiated and leading the Eclipse Swordfish Project
  - contributing to Eclipse SOA Tools Platform Project
Agenda

- Introduction
- What is „SOA Tooling“?
- Identifying SOA Artefacts
- Available Tools
- Demo
SOA Tooling is about providing an integrated set of tools for developing for and operating an SOA.

Diagram:
- create
- edit
- visualize
- assist
- store
- search
- collaborate
- transform
- link
- version
- transform

SOAP messages are transformed into Java code.
Click to add title

Click to add an outline
# Agenda

- Introduction
- What is „SOA Tooling“?
- **Identifying SOA Artefacts**
- Available Tools
- Demo
SOA Artefacts can be identified by analyzing the Service Development Lifecycle

<table>
<thead>
<tr>
<th>Owner</th>
<th>Business analyst</th>
<th>Service designer</th>
<th>Developer</th>
<th>Developer</th>
<th>Admin</th>
<th>Operator</th>
</tr>
</thead>
</table>

**Process**
- Abstract Process Definition
- Technical Process Definition
- Executable Process Definition

**Config**
- Configuration Directives
- Participant Configuration
- Business Policies
- Technical Policies
- Rules Definition
- Rules Implementation

**Rules**
- Business Guidelines
- Service Specification

**Service**
- Service Definition
- Service Implementation

**Artefact type**
- BAM configuration
- SAM configuration
- Operational Policies
- Operational monitoring configuration
- Deployment descriptor
- Infrastructure plan
- Implementation package

**Service Specification**
- Service Definition
- Service Implementation

**Process**
- Abstract Process Definition
- Technical Process Definition
- Executable Process Definition

**Operation**
- BAM configuration
- SAM configuration
- Operational Policies
- Operational monitoring configuration
- Deployment descriptor
- Infrastructure plan
- Implementation package
Applying the big picture to concrete use cases allows the identification of concrete artefacts

WSDL-first service development for ServiceMix
Agenda

- Introduction
- What is „SOA Tooling“?
- Identifying SOA Artefacts
- Available Tools
- Demo
Eclipse STP provides tools that address some of the artefacts.
SOPERA provides tooling for ServiceMix

- Analysis: Business analyst
- Design: Service designer
- Implementation: Developer
- Packaging: Developer
- Deployment: Admin
- Operation: TBD

Test config
- WSDL
- Service implementation
  - Business code
  - Interface code
  - Eclipse project

Service definition
- WSDL

Code generation
- Test GUI

Test execution
- Application
  - SM3
  - ZIP

Packaging
- Deployment

Initial version available start Q2/08
Agenda

- Introduction
- What is „SOA Tooling“?
- Identifying SOA Artefacts
- Available Tools
- Demo
The initial ServiceMix tooling available today will be expanded in the future

- will be released as open source
- support all other relevant use-cases:
  - code first, all interaction styles
- provide editors and validation for all artefact types
- improve testing:
  - automated testing, regression testing
The SOPERA platform complements the ServiceMix Core with advanced capabilities

SOPERA ToolSuite
- Service Editor
- Policy Editor
- Test Consumer
- Development Box

ServiceMix (deploy & test)

SOPERA ESB
- Service Registry
- Security
- Monitoring
- Transformation
- Messaging
- Transport

SOPERA ToolSuite

Eclipse Swordfish

JBI Kern

ServiceMix
- Standard WSDL with extensions for JMS endpoints
- Communication style request callback
- Available within SOPERA ASF 3.0
-JORAM ➔ WebSphere MQ
-Manual changes to code are preserved during re-generation
- GUI based + command line scripting interface