

JVA-561. Developing SOAP Web Services in Java

Version 2.2

A comprehensive look at the state of the art in developing interoperable web services on the Java EE 6 platform. Students learn the key standards -- SOAP, WSDL, and the WS-I Basic Profile -- and the Java architecture that has evolved to build interoperable services and clients. We begin with an introductory module that covers both SOAP-based and RESTful services, and therefore both JAX-WS and JAX-RS.

But JAX-WS is central to the course, and we cover both WSDL-driven and Java-driven development paths, as well as message handlers and attachment support. With the new Provider and Dispatch APIs, it's now much easier to integrate SAAJ, JAXB, and JAXP code into services and clients, and we explore these strategies in depth as well.

Course Length: 5 days

Prerequisites

- Strong Java programming skills are essential – **Course 103** is excellent preparation.
- Students must be able to read XML documents and to write well-formed XML by hand
- Knowledge of XML Schema will be helpful, too, but is not a strict prerequisite
- Experience with other Java EE standards, especially servlets and JSP, will be very helpful in class, but is not strictly required.

Learning Objectives

- Be able to describe the interoperable web services architecture, including the roles of SOAP and WSDL in component-based services and XML and HTTP in the REST architecture.
- Understand the importance of the WS-I Basic Profile for interoperable web services.
- Build JAX-WS services and clients that take full advantage of the automated data binding of JAXB.
- Build WSDL-to-Java and Java-to-WSDL services, with equal facility.
- Apply advanced techniques and best practices including proper exception handling, care around possible polymorphism, and use of context and lifecycle services.
- Use lower-level SOAP and XML APIs for services and/or clients.
- Customize data binding by specifying specific type mappings or altering method or parameter names.
- Incorporate binary data, such as images, into service and client code.

Server Support: GlassFish

- This version of the course works with the GlassFish server. Our Java EE courses are available in variants that support various server products, including Tomcat, GlassFish, JBoss, and WebLogic..

IDE Support: Eclipse Helios

- In addition to the primary lab files, an optional overlay is available that adds support for Eclipse Helios. Students can build and test all exercises from within Eclipse, deploying to a managed application server.

Course Outline

Chapter 1. Overview of Web Services

- Why Web Services?
- Service-Oriented Architecture
- HTTP and XML
- SOAP
- WSDL
- The SOAP Vision
- The REST Vision
- UDDI
- The WS-I Basic Profile
- Security

Chapter 2. Web Services for Java EE

- Hosting Web Services: Scenarios
- Web Services for Java EE
- JAX-WS and JAXB
- Web-Services Metadata
- WSDL-to-Java and Java-to-WSDL Paths
- Provider and Dispatch APIs
- SAAJ and JAXP
- JAX-RS for RESTful Services
- JAXR

Chapter 3. The Java API for XML Binding

- The Need for Data Binding
- XML Schema
- Two Paths
- JAXB Compilation
- Mapping Schema Types to Java
- Java-to-XML Mapping Using Annotations
- Marshaling and Unmarshaling
- Working with JAXB Object Models

Chapter 4. The Simple Object Access Protocol

- Messaging Model
- Namespaces
- SOAP over HTTP
- The SOAP Envelope
- The Message Header
- The Message Body
- SOAP Faults
- Attachments

Chapter 5. Web Services Description Language

- Web Services as Component-Based Software
- The Need for an IDL
- Web Services Description Language
- WSDL Information Model
- The Abstract Model -- Service Semantics
- Message Description
- Messaging Styles
- The Concrete Model -- Ports, Services, Locations
- Extending WSDL -- Bindings
- Service Description

Chapter 6. The Java API for XML-Based Web Services

- Two Paths
- How It Works: Build Time and Runtime
- The Service Endpoint Interface
- Working from WSDL
- Working from Java
- RPC and Document Styles
- One-Way Messaging
- Binary Protocols

Chapter 7. WSDL-to-Java Development

- The @WebService Annotation
- Generated Code
- Scope of Code Generation

- Parameter Order
- More JAXB: Mapping Collections
- More JAXB: Mapping Enumerations
- Applying JAXB Customizations

Chapter 8. Client-Side Development

- Stubs and Proxies
- Generated Code
- Locating a Service
- Invoking a Service
- The `@WebServiceRef` Annotation

Chapter 9. Java-to-WSDL Development

- Generating the WSDL and Schema
- The `@WebMethod`, `@XmlParam`, and Related Annotations
- More JAXB: Mapping Inheritance
- Controlling the XML Model
- Controlling the WSDL Description
- JAXB Customizations with `@XmlJavaTypeAdapter`

Chapter 10. Exception Handling

- SOAP Faults vs. Java Exceptions
- Mapping Faults from WSDL
- Mapping Exceptions from Java
- JAX-WS Exception API and Handling
- Client Exception Handling

Chapter 11. JAX-WS Best Practices

- Which Way to Go?
- Interoperability Impact
- Portability Impact
- Polymorphism in WebServices
- Web Services as Java EE Components
- Lifecycle Annotations
- Context Interfaces

Chapter 12. Metadata

- Abstract and Concrete Model Metadata

- Defaults
- Annotations
- webservices.xml and web.xml
- Best Practices and Pitfalls

Chapter 13. Provider and Dispatch APIs

- Stepping Down
- The @WebServiceProvider Annotation
- The Provider<T> Interface
- Implementing a Provider
- The @ServiceMode Annotation
- JAXB Without WSDL
- Integrating JAXP
- The Dispatch<T> Interface
- Building Clients

Chapter 14. The SOAP with Attachments API for Java

- The SAAJ Object Model
- Parsing a SOAP Message
- Reading Message Content
- Working with Namespaces
- Creating a Message
- Setting Message Content
- Relationship to the DOM

Chapter 15. Message Handlers

- Handling SOAP Headers
- Handlers and Handler Chains
- Configuration by XML or Annotation
- MessageContext and SOAPMessageContext
- Processing Model and Patterns
- Logical and Protocol Handlers
- Client-Side Handlers

Chapter 16. Handling Binary Content

- The WS-I Attachments Profile
- Using base64Binary

- MTOM and XOP
- JAX-WS Support
- Configuration by XML or Annotation
- Client-Side Support
- SAAJ Support

Appendix A. Course Tools and Utilities

Appendix B. Handy Guide to Web-Services Acronyms