Fast Track to Core Web Services for JEE Developers

Course Summary

Description

This course focuses on what separates an ad hoc set of web services (NOT SOA) from a managed, vibrant, reusable catalog of enterprise services (definitely SOA). This course provides an overview of the entire spectrum from the promise of cloud computing to the grit of XML content. Students will gain practical, hands-on experience with implementations of the Java XML and Web Service APIs including JAXP, JAXB, JAX-WS 2.0, WSEE, and XWSS. In terms of Java, the focus is on using the JAX-WS 2.0, and JAXB APIs and implementations as the basic for generating and deploying web services and service clients.

Objectives

At the end of this course, students will be able to:

- Understand and apply the basic concepts of SOA to the identification and design of web services
- Understand and intelligently discuss Web Services and the core technologies involved
- Understand and apply the basic concepts of SOA to the identification and design of web services
- Appreciate the concept of layered services including orchestration
- Understand and intelligently discuss Web Services and the core technologies involved
- Design, develop, and deploy real-world JEE Web Services
- Expose existing Java components as XML Web Services

- Write Java components that access remote Web Services hosted by a third party
- Read and understand a WSDL document
- Parse, process, and respond to a SOAP message
- Understand the concepts behind REST and implement a REST-based web service
- Implement handlers to inject cross-cutting solutions for security, logging, auditing, and other needs
- Work with WS-Security to protect content, resources, and other assets

Topics

- Services Via the Web
- Web Services Overview
- Web Services, Java, and JEE
- Web Services Quickstart
- XML, Namespaces & Schemas
- XML in Java - JAXP and JAXB
- SOAP Overview
- SOAP in Detail
- REST

- WSDL
- JAX-WS Overview
- Working with JAX-WS
- Web Services for JEE (WSEE)
- Transactions in Web Services: WS-TX
- Web Services Discovery
- XML Signature and Encryption
- WS-Security
- Securing Untrusted Input

Audience

This is an intermediate level training course, designed for experienced Java developers and architects who need to identify, design, and implement web services.

Prerequisites

Students should have 1-2 years of working knowledge with Servlets and JSPs, and should be familiar with XML, Namespaces, and XML Schema.

Duration

Three days

Fast Track to Core Web Services for JEE Developers

Course Summary

I.   Services Via the Web
   A. What is a Service?
   B. Let's Focus on One Definition of SOA
   C. Architectural Style: Common Framework
   D. Loose Coupling: Spectrum of Options
   E. Software Agents: Services
   F. Interacting: Orchestrated
   G. SOA is Not Revolutionary
   H. What is the Difference Between Services and SOA?
   I. Logical Components of a SOA
   J. Characteristics of a Good Service
   K. SOA Reference Architecture
   L. Business versus Application
   M. Service Layers
   N. Application Services
   O. Functional Services
   P. Business Process Services
   Q. Messaging
   R. Common Framework: Addressing Cross-Cutting Concerns
   S. Role and Uses of an Enterprise Service Bus
   T. High-level View of an Enterprise Service Bus
   U. Common Framework: Addressing Cross-Cutting Concerns
   V. Governance and Compliance
   W. Governance and Expertise
   X. Governance and Service Lifecycle
   Y. Governance and Consumer/Provider Management
   Z. SOA Anti-Patterns
   AA.   Incremental Adoption of SOA

II.   Web Services Overview
   A. What are Web Services?
   B. Web Services Characteristics
   C. Web Services Architecturally
   D. Web Services Enable Decoupling
   E. Web Services Challenges
   F. Spec and Standard Evolution
   G. Web Services Interoperability Organization
   H. Basic Profile 1.1 Consists of:
   I. Additional WS-I Profiles
   J. .NET Platform
   K. .NET Web Services
   L. Java and Web Services
   M. Web Services in Action

III.   Web Services, Java, and JEE
   A. XML and Java APIs at a Glance
   B. XML Signature

   C. XML Encryption
   D. JAXP
   E. JAXB
   F. JAXP, JAXB and Web Services
   G. Web Services APIs at a Glance
   H. JAX-WS
   I. SAAJ
   J. JAX-WSA and XWSS
   K. Web Services APIs
   L. Web Services for JEE (JSR109)
   M. JEE and Web Services
   N. Web Services Metadata
   O. Web Services Stacks at a Glance
   P. Apache Axis 2.x
   Q. JBossWS
   R. WSIT/Metro
   S. WebSphere WS
   T. WebSphere/RAD Web Service Stacks
   U. Spring-WS

IV.   Web Services Quickstart
   A. What is a "Typical" Web Services Stack?
   B. Typical Tool Support
   C. How Stack is Used on the Service-Side
   D. How Stack is Used on the Client-Side
   E. Debugging Web Services
   F. TCP/IP Monitors Provide View of Wire
   G. Implementing a Web Service
   H. Debugging Web Services

V.   XML, Namespaces & Schemas
   A. What is XML?
   B. XML Can Provide Application-Specific Information
   C. Content: XML Document Syntax Rules
   D. Structure: A Document Type Definition
   E. XML Transformation to HTML
   F. XML Separates Structure, Content and Format
   G. TriveraTunes Purchase Order
   H. Content as MarkUp
   I. Tell Parser That Text is Data
   J. Use Predefined Entities
   K. Well-Formed and Valid XML Documents
   L. Why Are These Definitions Important?
   M. XML Namespaces
   N. Name Collision - Example
   O. Inter-Organization Name Collisions
   P. W3C's Solution: Namespaces
   Q. Uniform Resource Indicator
   R. Declaring a Namespace
   S. Namespace Scope

T. Default Namespace
U. Attributes and Namespaces
V. Example of Namespaces
W. Namespaces Best Practices
X. Benefits From Valid XML
Y. W3C XML Schemas
Z. Impacts of Schemas
AA. General Form of an XML Schema
BB. Elements, Attributes and Types
CC. Simple Schema and XML Document
DD. Element Definitions
EE. Corresponding XML Schema
FF. Simple Types - Primitive Datatypes
GG. Restricting Simple Types: Facets
HH. Complex Types Bring More to Validation
II. Repetition Control
JJ. Restricting Simple Types
KK. Complex Types Can be Derived
LL. Derivation by Extension
MM. Extension of Phone Number
NN. Associating Schemas with XML Instances
OO. Using XML Schema with Namespaces
PP. Namespaces Provide Thread of Connection
QQ. Schema Defines a Target Namespace
RR. XML Doc Uses Schema-Defined Namespace
SS. schemaLocation Links Namespace to Location
TT. Relating Schemas to XML
UU. Namespaces and Schemas

VI. XML in Java - JAXP and JAXB
A. XML Parsers Are Complex and Powerful
B. Parsers and API's
C. Parser Generates DOM, Then Hands to App
D. Parsing With a DTD or Schema
E. Many Options to Consider
F. XML and Java
G. JAXP: Java API for XML Processing
H. Challenges to Mapping XML
I. Generating XML is Nondeterministic
J. JAXB: Binding XML to Java
K. JAXB 2.0 Incorporated Changes
L. JAXB 2.0 and Java Versions
M. Defining the Rules in JAXB
N. Turning Rules into Java Classes
O. Using the Generated Classes
P. Creating Content
Q. Some JAXB Type Bindings
R. XML Schema for List of Items

S. Corresponding Class Interface
T. A Word About Validation
U. Working With JAXB

VII. SOAP Overview
A. SOAP in a Nutshell
B. SOAP Specification Provides:
C. Anatomy of a SOAP Message
D. SOAP and HTTP
E. Simple Scenario
F. More Complex Scenario
G. Uses of SOAP
H. A Simple Example
I. Remote Procedure Calls
J. Example of SOAP RPC Call
K. Example of Response
L. SOAP in Action

VIII. SOAP in Detail
A. Protocols Used With Web Services
B. Request and Response Example
C. The SOAP Envelope
D. SOAP Header
E. SOAP Body
F. SOAP Request/Response Example
G. SOAP Defines a Fault Element
H. SOAP Data Model
I. Example Using XSD Data Type
J. SOAP Styles and Modes
K. SOAP With Attachments
L. SOAP Messaging
M. Endpoint Behavior
N. SOAP Encoding Styles
O. SOAP 1.1 versus SOAP 1.2
P. What is SAAJ?
Q. Understanding SAAJ
R. SOAPElement
S. SOAPElement Methods
T. SOAPMessage
U. SOAPPart
V. SOAPEnvelope
W. SOAPBody
X. SOAPBodyElement
Y. SOAPHeaderElement
Z. SOAPHeaderElement Methods
AA. SOAPFault
BB. Connections
CC. Accessing a Service with a Servlet

Fast Track to Core Web Services for JEE Developers

Course Summary (cont'd)

IX. REST
- A. REpresentational State Transfer
- B. REST Characteristics
- C. REST Elements
- D. REST in Web Service Terms
- E. REST: Another Option for Binding
- F. Characterizing REST
- G. REST Example
- H. REST Design Principles
- I. REST/SOAP Comparison
- J. SOAP versus REST
- K. RESTing in Java
- L. Working With REST

X. WSDL
- A. Describing Web Services
- B. WSDL in Practice
- C. WSDL Extensibility
- D. WSDL/SOAP Namespaces
- E. WSDL Elements
- F. WSDL Anatomy
- G. <definitions> element
- H. <documentation> element
- I. <types> element
- J. <message> element
- K. <portType> and <operation>
- L. <binding> element
- M. <port> and <service> elements
- N. Looking Ahead – WSDL 2.0
- O. WSDL in Action

XI. JAX-WS Overview
- A. JAX-WS and JAX-RPC
- B. JAX-RPC – Where It Stands
- C. JAX-WS Overview
- D. JAX-WS Architecture
- E. JAX-WS Under the Hood
- F. JAX-WS Features
- G. JAX-WS Programming Model
- H. JAX-WS Handlers
- I. JAXB
- J. WSIT
- K. JAX-WS vs RMI

XII. Working with JAX-WS
- A. JAX-WS Service
- B. Bottom-up Building of a Web Service
- C. Top-Down Building of a Web Service
- D. JAX-WS Development Process
- E. Web Service Annotations
- F. @WebService
- G. @WebMethod

- H. Additional Web Service Annotations
- I. Generating the Portable Artifacts
- J. Example of Artifacts Generated From WSDL
- K. Deploying the Web Service
- L. Lab: Cost Service
- M. JAX-WS Client
- N. Types of JAX-WS Clients
- O. Creating the JAX-WS Artifacts
- P. Example of Artifacts Generated From WSDL
- Q. JAX-WS Java Client
- R. Cost Service Clients
- S. Membership Registration Service
- T. Membership Registration Client
- U. Modifying the Registration Service
- V. Handlers
- W. Handlers: The Basics
- X. JAX-WS and Handlers
- Y. Handler Life Cycle
- Z. Applying Handlers
- AA. SOAPMessageContext
- BB. Example of Handler Class
- CC. Configuring Handlers
- DD. Handler Configuration using webservices.xml
- EE. Handler Configuration using Annotations
- FF. Working with Handlers
- GG. Checking Compliance with Handlers (optional)

XIII. Web Services for JEE (WSEE)
- A. WSEE
- B. WSEE Server Programming Model
- C. Server Programming Model
- D. Servlets as Web Services
- E. EJBs as Web Services
- F. Routing SOAP Requests To An EJB
- G. WSDD
- H. WSDD Example
- I. WSEE Client Programming Model
- J. Client Programming Model
- K. Types of WSEE Clients
- L. WSEE Basic Client Operations
- M. Static WSEE Client
- N. Dynamic WSEE Client
- O. DII WSEE Client
- P. WSEE Client Packaging

Fast Track to Core Web Services for JEE Developers

Course Summary (cont'd)

XIV. Transactions in Web Services: WS-TX
- A. Transactions
- B. The ACID Transaction Properties
- C. Transaction Lifecycle
- D. Overview of a Transactional System
- E. JEE Transaction Support
- F. SOA's Challenge of Handling Transactions
- G. Solution: WS-Transaction (WS-TX)
- H. WS-Transaction (WS-TX)
- I. WS-Coordination
- J. WS-AtomicTransaction
- K. WS-BusinessActivity
- L. Creating Transacted Web Service
- M. Option 1
- N. Creating Transacted Web Service
- O. Option 2
- P. Transaction Mapping
- Q. Client Coordinated Transaction
- R. WS-AT and JEE
- S. WS-BA and JEE

XV. Web Services Discovery
- A. Issues With Broadly Scoped Discovery
- B. What is UDDI?
- C. UDDI Registries
- D. What is WSIL?
- E. What is WS-Discovery?
- F. Examples of Tools That Support Discovery
- G. Discovery in Action

XVI. XML Signature and Encryption
- A. Cryptography
- B. Common Solutions to Big Three
- C. XML Challenges
- D. XML Signature
- E. XML Digital Signatures
- F. XML Signature Usage
- G. Standard For Digital Signature
- H. XML Encryption
- I. XML Encryption Usage
- J. XML Encryption Protects Data

XVII. WS-Security
- A. Transport-Level Security
- B. SSL In Action
- C. When to Use Transport-Level Security
- D. Message-Level Security
- E. Web Services Security Roadmap
- F. WS-Security Enables Interoperability
- G. What is XWSS?
- H. XWSS Provides Many Functions

XVIII. Securing Untrusted Input
- A. Input Data Attacks
- B. Protecting a Web Service
- C. Tenacious D
- D. Defending a Web Service
- E. Responding to Error State
- F. Additional Types of Attacks
- G. Defenses to Consider
- H. Networking Devices Targeting XML
- I. Support Message Level Security
- J. Address Cross-cutting Concerns
- K. Advantages to These Devices
- L. Enterprise Integration Needs
- M. Device Usage Scenario: Authentication
- N. Variety of Products Out There
- O. Insecure XML Processing