

Custom

TRAINING INSTITUTE

# Developing Web Services with WebSphere 6.0 and Rational Application Developer (5 Days)

**Custom Training Institute**  
9085 Coyote Springs Road  
Prescott Valley, AZ 86314  
(928) 772-3811  
FAX (928) 441-6444

## Developing Web Services with WebSphere 6.0 and Rational Application Developer (CTI 239)

*Students will learn how to develop Web Services with WebSphere 6.0 and Rational Application Developer (RAD) using standards such as SOAP, WSDL and JAX-RPC. This course covers the basics of XML and includes all the techniques needed to program Web Services.*

*Prerequisites: Familiarity with the Java language, JavaBeans, and web application architecture and concepts. Experience with RAD helpful. Successful completion of course CTI 238, "Enterprise Java Programming Using Rational Application Developer," satisfies these requirements.*

*Minimum software requirements: Microsoft Windows 2000 Professional (SP 3 or later ) or Windows XP (SP 1 or later). WebSphere 6.0 and Rational Application Developer. Java-compatible browser, J2SE 5.0 SDK.*

*Minimum hardware requirements: Pentium III at 1 GHz; minimum of 1 Gb RAM; 3.5 Gb free disk space, plus additional space for development; Internet connection.*

*Microsoft PowerPoint on instructor's workstation for presentation purposes.*

### Module 1: The Web Services Architecture

- Evolution of Web Services
- Motivation for Web Services
- HTTP and XML
- Interoperability Stacks
- The Wire Stack
- Simple Object Access Protocol (SOAP)
- The Description Stack
- Web Service Description Language (WSDL)
- The Discovery Stack
- Universal Description, Discovery and Integration (UDDI)
- Hosting Web Services: Scenarios
- Observing SOAP Traffic

### Module 2: RAD and Web Services

- The WebSphere Platform
- Web Services Features and Support
- RAD -Speak for Web Services
- The Studio
- The WebSphere Server
- Creating a Domain
- Ant Tasks for Web Services
- Development Process

### Module 3: Java and Web Services

- Java and Web Services
- Web Services and the J2EE
- RAD Support for Standard APIs
- The Java API for XML Processing (JAXP)
- The Java API for XML Binding (JAXB)
- The SOAP With Attachments API for Java (SAAJ)
- The Java API for XML Messaging (JAXM)
- Low-Level Web Services in WebSphere (SAAJ)
- The Java API for XML-Based RPC (JAX-RPC)
- High-Level Web Services in RAD (JAX-RPC)
- WSDL-to-Java vs. Java-to-WSDL
- The Java API for XML Registries (JAXR)
- RAD UDDI

### Module 4: The Simple Object Access Protocol (SOAP)

- SOAP Messaging Model
- SOAP Namespaces
- SOAP over HTTP
- The SOAP Envelope
- The Message Header
- The Message Body
- SOAP Faults
- Attachments
- XML Schema
- Validating Message Content
- The SOAP "Section 5" Encoding
- Arrays
- Avoiding Redundant Serialization

## Developing Web Services with WebSphere 6.0 and Rational Application Developer (CTI 239) - continued

---

### Module 5: The Java APIs for SOAP Messaging (SAAJ)

- The SAAJ Object Model
- Parsing a SOAP Message
- Reading Message Content
- Bridges to JAXP
- Working with Namespaces
- Creating a Message
- Setting Message Content
- RAD SAAJ: Bugs and Limitations

### Module 6: SAAJ Web Services

- JAXM vs. RAD JMS
- Messaging Scenarios
- Point-to-Point Messaging
- SAAJ Services Using JAX-RPC
- Creating a JAXM Connection
- Sending a Message

### Module 7: Web Services Description Language (WSDL)

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

### Module 8: The Java API for XML-Based RPC (JAX-RPC)

- The Java Web Services Architecture
- Two Paths
- How It Works - Build Time and Runtime
- Mapping Between WSDL/XML and Java
- Generating Java from WSDL
- What Gets Generated
- What the Application Sees
- Generating WSDL from Java
- Which Way to Go?
- Passing Objects
- Another CORBA?

### Module 9: Generating Web Services from Java Code

- The Java-to-XML Mapping
- Primitive Types and Standard Classes
- Value Types and JavaBeans
- The Java-to-WSDL Mapping
- Service Endpoint Interface
- Scope of Code Generation
- Inheritance Support
- RAD JAX-RPC: Bugs and Limitations
- Multi-Tier Application Design
- Analyzing the Domain
- High-Level Ant Tasks
- web-services.xml
- When Things Don't Fit
- Polymorphism
- Extensible Type Mapping

### Module 10: Generating Java Web Services from WSDL

- The XML-to-Java Mapping
- Simple and Complex Types
- Enumerations
- Arrays
- The WSDL-to-Java Mapping
- Mapping Operation Inputs and Outputs
- Building a Service Client
- Locating a Service
- Client-Side Validation
- Interoperability Under Java-to-WSDL
- Creating a Web Service
- Mid-Level Ant Tasks
- XML and WSDL Design Guidelines
- Deploying the Service
- Interoperability Under WSDL-to-Java
- Controlling Names and URIs

### Module 11: Web Services and EJB

- Enterprise JavaBeans
- Three Tiers for J2EE
- EJB 2.1 and JAX-RPC
- Session Beans as Web Service Endpoints
- How It Works – Build Time and Runtime
- The Bean's Service Endpoint Interface
- SOAP as an RMI Transport
- Adding a SOAP Interface to a Session Bean

### Module 12: Message Context and Message Handlers

- Handling SOAP Headers
- Servlet Endpoint Context
- EJB Endpoint Context
- Using SAAJ
- JAX-RPC Message Handlers
- Handler Chains
- Processing Model and Patterns
- The <handlerChain> Ant Task

## Developing Web Services with WebSphere 6.0 and Rational Application Developer (CTI 239) - continued

---

### **Module 13: SOAP Attachments**

- SOAP Support for Attachments
- SAAJ Object Model, Revisited
- The SOAPMessage Class
- MIME
- The Java Activation Framework
- The MimeHeaders Class
- The AttachmentPart Class
- Adding SOAP Attachments
- Identifying Attachments
- Reading Attachments

### **Module 14: Web Services and JMS**

- Asynchronous Messaging
- The Java Message Service
- Queues and Topics
- Message Types
- Message-Driven Beans
- Asynchronous Web Services
- Message Queues as Web Services
- Ant Tasks and JMS Services

### **Module 15: Security**

- Web Services and Security
- Threats
- Technology and Techniques
- Public Key Encryption
- Digital Signature
- J2EE Techniques
- Securing Web-Service URIs
- HTTPS
- XML and SOAP Solutions
- XML Encryption and Signature
- WS-Security
- SAML
- XACML
- RAD Support for WS-Security
- Securing a Service's Messages
- Key Pairs and Keystores
- Enhancing the Client