



442: Core Web Services for J2EE Developers (3 days)

Course Description...

Web Services represent a groundbreaking evolution in distributed computing. The concepts are not altogether new, but the application of them, and the unanimous acceptance of core standards like HTTP, XML, SOAP, WSDL, and UDDI, has paved the way for XML Web Services and service-oriented architecture.

This three-day course introduces J2EE developers to the core standards that enable Web Services, and provides them hands-on experience with the Java Web Services Developer's Pack (JWSDP) released by Sun Microsystems. The JWSDP is a comprehensive toolkit for J2EE Web Services development, complete with the Apache Tomcat web server, Ant Build Tool, a UDDI registry server, and a variety of other tools to enable the development and deployment of robust J2EE Web Services.

Environment...

This course can be offered using Sun Microsystems JWSDP or IBM WSAD

Who Should Attend...

Java Developers and Architects.

Prerequisites...

Students should have 1-2 years experience with Servlets and JSPs, and should be familiar with XML. This course can be combined with 2 days of XML, 2 days of Serv/JSP or 2 days of struts, depending on your needs.

Learning Objectives...

After successfully completing this course, students will be able to:

- Understand and intelligently discuss Web Services and the core technologies involved
- Design, develop, and deploy real-world J2EE 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
- Effectively use the tools and APIs provided by the JWSDP



We will look at the current state of the art of Web services, what works and what doesn't work, and also at newer standards, and how they fit into the Web services picture. Web services are still evolving rapidly, and this course will give you a thorough understanding of the current Web Services architecture, and the technologies that support Web services including:

- **SOAP** – Simple Object Access Protocol - A remote invocation (RPC) and messaging mechanism.
- **WSDL** – Web services Description Language – An XML language that describes the interface and semantics of a Web service.
- **UDDI** – Universal Description, Discovery, and Integration – A standard for describing, publishing and finding Web services.
- **JSR101/JAX-RPC** – Accessing Web services using Java
- **ebXML** – Simple Object Access Protocol - A remote invocation (RPC) and messaging mechanism.
- **WS-I** – Web Services Interoperability
- **WS Security** – Web Services Security

Hands-on Approach:

This course is hands on, and students will actually build and deploy a Web service during the course. This course consists of

- 14 individual exercises, all tied together
- Instructor demonstrations
- Class discussion
- Daily reviews

Course Outline...

Session 1 – Introduction to Web Services

- ❑ Introduction
- ❑ Web Services Overview
- ❑ Programming Web Services
- ❑ Apache Axis

Session 2 – Simple Object Access Protocol (SOAP)

- ❑ Introduction to SOAP
- ❑ Communication Protocols (HTTP & SMTP)
- ❑ SOAP and HTTP
- ❑ SOAP in a Nutshell
- ❑ SAAJ Basics

Session 3 – Web Services Description Language (WSDL)

- ❑ Introduction
- ❑ WSDL Elements

Session 4 – Universal Description, Discovery and Integration (UDDI)

- ❑ UDDI Introduction



Session 5 – Java API for XML-based RPC (JAX-RPC) / JSR101

- ❑ JAX-RPC Overview
- ❑ Creating a JAX-RPC Client
- ❑ Creating a JAX-RPC Service

Session 6 – J2EE Web Services / JSR109

- ❑ J2EE Web Services (Server-side)
- ❑ J2EE Web Services (Client-side)

Session 7 – Java Web Services Security

- ❑ Introduction
- ❑ Securing JAX-RPC
- ❑ WS-Security

Session 8 – Emerging Web Services Standards

- ❑ The Future of Web Services