

Mastering SOA and Web Services – Developer's Workshop

Course Summary

Description

This course introduces developers to the fundamentals of Service Oriented Architectures (SOA) and the core standards that enable Web Services. 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). It provides an overview of the entire spectrum from the promise of cloud computing to the grit of XML content.

This lab-intensive course provides students with hands-on experience with implementations of the Java XML and Web Service APIs including JAXP, JAXB, JAX-WS, WSEE, and XWSS. In terms of Java, the focus is on using the JAX-WS and JAXB APIs and implementations as the basic for generating and deploying web services and service clients. SOA and 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 the various WS-* policies, has paved the way for XML Web Services and service-oriented architectures. An overreaching concern related to services is security.

Various aspects of security are woven into this course, allowing students to see the entire spectrum of issues as well as solutions. These include encryption, digital signatures, authentication and authorization assertions, and recognized application security issues such as Cross-Site Scripting and Injection attacks.

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 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
- 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
- Understand and utilize web service-related design patterns

Topics

- SOA Overview
- SOA: the Business Proposition
- Service-oriented Architecture
- Modeling Business Processes
- Service-oriented Analysis and Design
- Common Framework: Governance
- Common Framework: Infrastructure
- Mapping Frameworks to SOA

Due to the nature of this material, this document refers to numerous hardware and software products by their trade names. References to other companies and their products are for informational purposes only, and all trademarks are the properties of their respective companies. It is not the intent of ProTech Professional Technical Services, Inc. to use any of these names generically

Mastering SOA and Web Services – Developer's Workshop

Course Summary (cont'd)

- Case Study Workshop
- Web Services Overview
- Web Services, Java, and J2EE
- Web Services Quickstart
- XML, Namespaces, & Schema
- XML in Java - JAXP and JAXB
- Binding – SOAP/REST
- SOAP Overview
- SOAP in Detail
- REST
- WSDL
- JAX-WS Overview
- Working with JAX-WS
- Web Services for J2EE (WSEE)
- Handlers
- Discovery Overview
- XML Signature and Encryption
- WS-Security
- Securing Untrusted Input
- SOA Best Practices
- SOA Patterns
- SOA Anti-Patterns

Audience

This is an intermediate and beyond level web services training course, designed for J2EE developers and architects who need to identify, design, and implement web services. We will explore and apply the terminology, the specification, the processes and technologies specific to 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

Five days

Mastering SOA and Web Services – Developer's Workshop

Course Outline

I. SOA Overview

- A. Service Defined
- B. SOA Defined
- C. Organizational Framework
- D. Technical Framework
- E. Orchestration
- F. Reusability
- G. Services vs. SOA
- H. SOA in the past
- I. What is new in SOA
- J. Business impact/ROI
- K. Myths/Reality
- L. Adoption issues

II. SOA: the Business Proposition

- A. Drivers for business-orientation
- B. Accessible Services and Data
- C. Leveraging business processes
- D. Leveraging legacy applications
- E. Challenges to adoption
- F. Role of governance
- G. Role of an ESB
- H. SOA Maturity Model
- I. SOA Adoption

III. Service-oriented Architecture

- A. Principles
- B. Business Process-driven development
- C. SOA team
- D. Messaging
- E. Orchestration
- F. Business modeling
- G. Integrating legacy applications
- H. Extending the lifetime of legacy apps
- I. Governance
- J. What needs to be governed and what is already governed?
- K. Governing IT vs. SOA
- L. Continuous improvement
- M. Strategies

IV. Modeling Business Processes

- A. Top-down Process Design
- B. Bottom-up Process Design
- C. WSDL as Service Description
- D. Identifying services

- E. Identifying messages
- F. Identifying collaborations
- G. BPEL Overview

V. Service-oriented Analysis and Design

- A. The SOAD Process
- B. Analysis
- C. Design
- D. Implementation
- E. Process

VI. Common Framework: Governance

- A. Governance Overview
- B. Importance
- C. Responsibilities
- D. Implementation

VII. Common Framework: Infrastructure

- A. Overview
- B. Role in SOA
- C. Security Issues
- D. Scenarios and Analysis
- E. ESB Issues

VIII. Mapping Frameworks to SOA

- A. SOA Concepts and .Net
- B. SOA Concepts and JEE

IX. Case Study Workshop

- A. Discovery and Identification
- B. Recognizing Potential for Reusability
- C. Motivation for Service-Based Approach
- D. Problems with Ad Hoc Approach
- E. Advantages of Managed Approach
- F. Harvesting Reusable Services
- G. Refactoring and Agility in Face of Change
- H. Issues Yet to Be Adequately Resolved

X. Web Services Overview

- A. Crossing Boundaries
- B. What are Web Services?
- C. Six Key Components
- D. Web Services Characteristics
- E. Web Services Architecturally
- F. Technology Comparison

Due to the nature of this material, this document refers to numerous hardware and software products by their trade names. References to other companies and their products are for informational purposes only, and all trademarks are the properties of their respective companies. It is not the intent of ProTech Professional Technical Services, Inc. to use any of these names generically

Mastering SOA and Web Services – Developer's Workshop

Course Outline (cont'd)

- G. Architectural Perspective
- H. Web Services Enable Decoupling
- I. Many Web Services Challenges
- J. Secure Services
- K. Spec and Standard Evolution
- L. Web Services Interoperability Organization
- M. WS-I Has Many Deliverables
- N. Basic Profile 1.0 Consists of:
- O. Has > 100 Requirements and Suggestions
- P. .NET Platform & .NET Web Services
- Q. Java and Web Services
- R. Exercise: Web Services in Action

XI. Web Services, Java, and J2EE

- A. XML and Java APIs at a Glance
- B. XML Signature
- C. XML Digital Signatures
- D. XML Encryption
- E. JAXP
- F. JAXB
- G. JAX-WS
- H. SAAJ
- I. JAX-WSA and XWSS
- J. Web Services APIs
- K. Web Services for J2EE (JSR109)
- L. J2EE and Web Services
- M. Web Services Metadata
- N. Web Services Stacks at a Glance
- O. WSIT
- P. Apache Axis2
- Q. JBossWS
- R. JWSDP
- S. WebSphere WS
- T. Spring-WS
- U. Key Features

XII. Web Services Quickstart

- A. What is WSIT?
- B. WSIT Tool Support
- C. How is WSIT Used?
- D. Web Service Development with WSIT
- E. Debugging Web Services
- F. TCP/IP Monitors Provide View of Wire
- G. Exercise: Implementing a Web Service

- H. Exercise: Debugging Web Services

XIII. XML, Namespaces, & Schema

- 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. Parsers Use URI, Not the Alias
- U. Default Namespace
- V. Attributes and Namespaces
- W. Example of Namespaces
- X. Namespaces Best Practices
- Y. Benefits From Valid XML
- Z. W3C XML Schemas
- AA. Impacts of Schemas
- BB. General Form of an XML Schema
- CC. Elements, Attributes, and Types
- DD. Simple Schema and XML Document
- EE. Element Definitions
- FF. Corresponding XML Schema
- GG. Simple Types - Primitive Datatypes
- HH. Restricting Simple Types: Facets
- II. Complex Types Bring More to Validation
- JJ. Repetition Control
- KK. Restricting Simple Types

Due to the nature of this material, this document refers to numerous hardware and software products by their trade names. References to other companies and their products are for informational purposes only, and all trademarks are the properties of their respective companies. It is not the intent of ProTech Professional Technical Services, Inc. to use any of these names generically

Mastering SOA and Web Services – Developer's Workshop

Course Outline (cont'd)

- LL. Complex Types Can be Derived
- MM. Derivation by Extension
- NN. Extension of Phone Number
- OO. Associating Schemas with XML Instances
- PP. Using XML Schema with Namespaces
- QQ. Namespaces Provide Thread of Connection
- RR. Schema Defines a Target Namespace
- SS. XML Doc Uses Schema-Defined Namespace
- TT. schemaLocation Links Namespace to Location
- UU. Relating Schemas to XML
- VV. Exercise: Namespaces and Schemas

XIV. XML in Java - JAXP and JAXB

- A. XML Parsers Are Complex and Powerful
- B. Parsers Are Integral to XML Processing
- C. Parsers and API's
- D. Parser Generates DOM, Then Hands to App
- E. Parsing With a DTD or Schema
- F. Many Options to Consider
- G. XML and Java
- H. Security Concerns Relative to Parsing
- I. Bridging Application Data and XML
- J. JAXP: Java API for XML Processing
- K. JAXP and Transformations
- L. Challenges to Mapping XML
- M. Generating XML is Nondeterministic
- N. JAXB: Binding XML to Java
- O. JAXB 2.0 Incorporated Changes
- P. JAXB 2.0 and Java Versions
- Q. Defining the Rules in JAXB
- R. Turning Rules into Java Classes
- S. Using the Generated Classes
- T. Creating Content
- U. Some JAXB Type Bindings
- V. XML Schema for List of Items
- W. Corresponding Class Interface
- X. A Word About Validation...
- Y. Exercise: Working With JAXB

XV. SOAP Overview

- A. SOAP in a Nutshell
- B. Anatomy of a SOAP Message
- C. SOAP and HTTP
- D. A Typical Scenario
- E. What is SOAP?
- F. SOAP Specification Provides:
- G. Why do I need SOAP?
- H. Simple Scenario
- I. Less Simple Scenario
- J. Uses of SOAP
- K. A Simple Example...
- L. Remote Procedure Calls
- M. Example of SOAP RPC Call
- N. Example of Response
- O. Exercise: SOAP in Action

XVI. 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 vs. 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. Security Concerns Relative to SOAP

Due to the nature of this material, this document refers to numerous hardware and software products by their trade names. References to other companies and their products are for informational purposes only, and all trademarks are the properties of their respective companies. It is not the intent of ProTech Professional Technical Services, Inc. to use any of these names generically

Mastering SOA and Web Services – Developer's Workshop

Course Outline (cont'd)

- DD. Exercise: Creating a SOAP Message
- EE. Exercise: Accessing a Service with a Servlet

XVII. 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 vs. REST
- K. RESTing in Java
- L. Security Concerns Relative to REST
- M. Exercise: Working With REST (Optional)

XVIII. 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. <portType> and <operation>
- K. <binding> element
- L. <port> and <service> elements
- M. Looking Ahead – WSDL 2.0
- N. Exercise: WSDL in Action

XIX. JAX-WS Overview

- A. JAX-WS
- B. JAX-WS Introduction
- C. JAX-WS Architecture
- D. JAX-WS vs RMI
- E. What about JAX-RPC?
- F. JAX-WS Under the Hood
- G. JAX-WS Basics
- H. JAX-WS Features
- I. Web Service Annotation Examples
- J. JAX-WS Programming Model

- K. JAX-WS Handlers
- L. Working with JAX-WS
- M. JAX-WS Service
- N. JAX-WS Development Process
- O. Bottom-up Building of a Web Service
- P. Top-Down Building of a Web Service
- Q. Exercise: WSDL-Generated Service
- R. JAX-WS Client
- S. Types of JAX-WS Clients
- T. JAX-WS Basic Client Operations
- U. Static Web Service Client
- V. Dynamic JAX-WS Client
- W. DII JAX-WS Client
- X. SOAP Handlers
- Y. Handlers Applied Via Declaration
- Z. Exercise: WSDL-Generated Client

XX. Web Services for J2EE (WSEE)

- A. WSEE & WSEE Server Programming Model
- B. Server Programming Model
- C. Servlets as Web Services
- D. EJBs as Web Services
- E. Routing SOAP requests to an EJB
- F. WSDD
- G. WSDD Example
- H. WSEE Client Programming Model
- I. Client Programming Model
- J. Types of WSEE Clients
- K. WSEE Basic Client Operations
- L. Static WSEE Client
- M. Dynamic WSEE Client
- N. DII WSEE Client
- O. WSEE Client Packaging

XXI. Handlers

- A. Handlers: The Basics
- B. WSEE and Handlers
- C. Handler Life Cycle
- D. Applying Handlers
- E. Handler Interface and Implementation
- F. SOAPMessageContext
- G. Example of Handler Class
- H. Configuring Handlers
- I. Web Service Handler Configuration

Due to the nature of this material, this document refers to numerous hardware and software products by their trade names. References to other companies and their products are for informational purposes only, and all trademarks are the properties of their respective companies. It is not the intent of ProTech Professional Technical Services, Inc. to use any of these names generically

Mastering SOA and Web Services – Developer's Workshop

Course Outline (cont'd)

- J. Using Handlers for Security
- K. Exercise: Handlers in Action
- XXII. Discovery Overview**
 - A. Discovery Options
 - B. What is UDDI?
 - C. UDDI Background
 - D. Interacting With UDDI
 - E. What is WSIL?
 - F. What is WS-Discovery?
 - G. Exercise: Discovery in Action
- XXIII. XML Signature and Encryption**
 - A. Cryptography Addresses Many Aspects of Security
 - 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
- XXIV. WS-Security**
 - A. Securing a Web service
 - B. Web Service Security Exposures
 - C. Transport-Level Security
 - D. Secure Sockets Layer (SSL)
 - E. SSL In Action
 - F. When to Use Transport-Level Security
 - G. Message-Level Security
 - H. Web Services Security Roadmap
 - I. WS-Security Enables Interoperability
 - J. Security Tokens; Example of Security Token
 - K. Message Authentication
 - L. XML Signature and Encryption
 - M. Picture is Evolving
 - N. What is XWSS?
 - O. XWSS Provides Many Functions
 - P. Exercise: XWSS in Action
- XXV. 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. Best Practices for Untrusted Data
 - G. Defenses to Consider
 - H. Additional Types of Attacks
 - I. Exercise: Insecure Web Services
- XXVI. SOA Best Practices**
 - A. Planning
 - B. Standardizing
 - C. Designing
 - D. Managing
 - E. Implementing
- XXVII. SOA Patterns**
 - A. Direct Connections
 - B. Broker Interactions
 - C. Serial Process Flows
 - D. Serial and Parallel Processes
- XXVIII. SOA Anti-Patterns**
 - A. SOA Adoption antipatterns
 - B. Service identification & design antipatterns
 - C. Service realization antipatterns