

# SOAP Design and Implementation

## detailed contents

### Introduction

- High level overview of SOAP, XML, WSDL, and UDDI
- Demystifying the underlying core technologies
- The classic service provider
- The service interface
- The service consumer

### Web Services Demo

- Web service in action
- A web services client and server demo
- Quote service demonstration
- Interaction between SOAP, XML and WSDL
- Connecting to an Internet web service on the web
- Demo using XMLSpy, BizTalk, Eclipse, Java...

### Web Services Standards and Organizations

- The web services standards space
- The standards body and their associated recommendations
- Overview of OASIS, W3C, WS-I, OMG and Java Community

### Simple Object Access Protocol (SOAP)

- Objectives
- SOAP Overview
- Why do you need SOAP?
- SOAP In Protocol Stack
- Header Attributes
- SOAP Body
- SOAP Fault
- Document/Literal Style
- Document/Literal Wrapped Style
- Details of the Wrapped Style

### Fundamental Service Contract Design

- SOAP Fundamentals
- Web Service Contracts
- Anatomy of a Web Service Contract
- A Plain English Guide to Namespaces
- Types and Message Structure Basics
- Abstract Description Design
- Concrete Description Design
- Design Options
- WS-Policy: Expression
- Assertion and Attachment
- SOAP Envelope Structure
- Header Block Processing

### Advanced Service Contract Design

- Message Flexibility
- Type Inheritance and Composition
- Reusability
- Derived Types
- Relational Design
- Modularization, Extensibility
- Asynchrony
- Message Dispatch
- Service Instance Identification
- Non-SOAP HTTP Binding

### Service Policies

- Policy Centralization
- Nested, Parameterized Assertions
- Ignorable Assertions
- Custom Policy Assertion Design
- Runtime Representation
- Compatibility
- WS-Addressing Vocabularies
- WS-Addressing Rules
- Design Techniques

### Service Contract Versioning

- Versioning Fundamentals
- Versioning WSDL Definitions
- Versioning Message Schemas
- Advanced Versioning

### The Wrapped-Doc-Literal Style for WSDL

- Discussion of different message encodings and styles
- Historical evolution and the on-the-wire format
- Why doc-literal is preferred over RPC-encoded
- Why using the wrapped form?
- Design guidelines for wrapped-doc-literal approach.

### Best Practice in WSDL Design

- Advice on how to design interoperable WSDL
- Making WSDL easy to understand and re-use
- Operation Granularity
- Naming: portTypes, services, ports and bindings
- Client access
- WSDL-first design

### Orchestrating Services with BPEL

- SOAP in BPEL context
- Importing XML Schemas and WSDL Locators
- Support for Validation, Deployment, and Lifecycle Management
- Orchestrate services through BPEL for a Synchronous Service
- Orchestrate services through BPEL for an Asynchronous Service
- Partner Link Concepts and Usage

### Software Platform for SOA (Service Orientation)

- Software Tools for SOA
- SOA Development Life Cycle
- Oracle BP Manager
- Microsoft BizTalk Server 2006
- Key Features
- Web Services Support
- Runtime Products for SOA

### Conclusions

- New implementation paradigms
- The benefits of employing SOA
- Review of common business goals
- The risks associated with the SOA approach
- Evaluating tradeoff strategies