

FUNDAMENTALS OF WEBSERVICES

Course Description

This two-day course is designed to provide a firm understanding of the process of designing and planning the implementation of a web service. It discusses reasons for implementing web services, interface considerations, system reusability, interoperability, and security. It explains and contrasts RPC-oriented services (SOAP) with XML-centric alternatives (REST).

Course Length: 2 Days

Course Tuition: \$500 (US)

Prerequisites

Familiarity with web and data processing concepts. Fundamentals of XML.

Course Outline

- Getting Started with Web Services
 - Web Services Defined
 - Why Build a Web Service?
 - Design Considerations
 - Physical Constraints
 - HTTP Interface Design
 - Resources
 - Parameter Passing
 - Client Issues
 - XML Considerations
- Web Services Overview
 - Interfaces vs. Platforms
 - Web Services Applications
 - RPC-style Web Services
 - RESTful Web Services
 - Implementing XML Standards
 - Internationalization
 - Security and Authentication
 - Transactions
 - Interface Versioning
- Interface Design
 - Web Service Development
 - Selecting A Platform
 - Planning for Reuse
 - Security Considerations
 - Supporting Transactions
 - Staged Development
 - Deployment & Ongoing Maintenance
 - Building Web Services in Java
 - The Java Web Services Developer Pack
 - Selecting A Development Framework
 - Java Servlets
 - JavaServer Pages
 - Using SOAP with JAX-RPC
 - Creating A WSDL Document
 - Securing the Application
 - Deployment
 - Calling from the Client
 - Building Web Services in .NET
 - .NET Architecture Overview
 - The .NET Framework
 - The Common Language Runtime (CLR)

Physical Considerations
Web Services Protocols
HTTP Review
Defining Resources
URL Design
Query Parameters
GET and POST
SOAP Over HTTP
Stateless and Stateful Operations
Planning for Scalability
• Publishing and Discovery
Knowing Your Audience
Discovery Protocols
UDDI
WSDL
RDF
XML Schema
• Basic SOAP
SOAP Overview
The SOAP Envelope
SOAP Over HTTP
A SOAP Server
The Deployment Descriptor
A SOAP Client
Complex Data Types
Complex Data Types: Client Code

.NET Remoting vs. ASP .NET
A Simple ASP .NET Web Service
Using the wsdl Utility
Securing the Application
Deployment
Calling from the Client
• Using Web Services
Client Types
Software Clients
Browser Clients
HTML Interfaces
Aggregating Web Services
Distributed Transaction Strategies
Alternative Transports (SMTP)