



RESTful Web Services

Duration 2 day(s) (WS-REST-02)

Development of RESTful Web Services with Apache CXF and JAX-RS

Description

By their distributed nature, information systems face two major issues: reuse and interoperability. Several technologies have tried to meet these objectives, the Web Services are part of the latest arrivals. We will present the concepts and principles that made Web Services a must-have technology. We will rely on the Java language and dedicated frameworks in order to quickly implement a sustainable distributed architecture.

Goals

- Understand the REST architecture and the approach to go into RESTful service
- Using JAVA frameworks for creating RESTful web services
- Know how to deploy Web Services REST
- Call Remote REST Web Services

Public

- Architect - Developer - Project Manager

Prerequisites

- Notions XML/Java

Structure

50% Theory, 50% Practice

Program

Introduction to Web Services

- What is a (Web) Service?
- Distributed architectures
- Granularity of services: reuse and interoperability
- Principles of design: contract, conventions, loose coupling ...
- Web Services and ROA (Resources Oriented Architecture)
- The HATEOAS principle
- Glory of REST
- Caching
- Management of cache headers
- Competition access to resources
- Secure a REST service
- CORS principle and implementation
- Auto API documentation

The fundamental concepts of REST architecture

- What is a resource, a URI?
- Objectives of this architecture
- REST Features
- REST and / or RESTful?

REST, the return to the sources of the Web

- Reminder on the HTTP protocol
- The operations
- The settings
- The answer codes
- Data (XML, JSON, other MIME types)

Java and REST

- Overview of the Java / Web Services offer
- JAX-RS, the JSR 311
- Jersey, the reference implementation
- Implementation of JAX-RS
- Headers, cookies
- The JAX-RS context

XML / XSD: Reminders

- XML / XSD: pivot of Web Services
- XML: universal language
- XSD: describe a document
- Namespaces and types
- Java and XML

RESTful advanced

REST vs. SOAP WS- *

- Use case of REST and SOAP WS- *
- What solution (s) to choose?
- Advantages disadvantages