

SOA -Service-Oriented Architecture Technologie

3 dagen hands-on training (Nederlandstalig)

► Doel & voor wie geschikt?

In this course, we will go beyond the hype of Web services and explore implementation architectures, as well as best practices for implementing SOAs and realizing the benefits of composite applications. Composite applications combine new functionality with existing data and applications. Because new development is usually involved, the tools as well as the implementers tend to be different from those used in traditional EAI projects. This training will focus on the evolving architecture alternatives and component technologies necessary for implementing composite applications. It will also include best practices for maximizing reuse and ROI.

- Understand the role of SOA in composite applications
- Experience the exact role of Web services in composite applications
- What Webservices will and will not do
- Architecture and technologies required for implementing composite applications
- Best practices in composite application implementations
- SOMA Service Oriented Modeling Architecture
- Administer a SOA solution and monitor business processes
- Increase process reliability through fault handling and exception management
- Invoke a SOA service through adapters for writing to files and databases
- Administer SOA processes using process monitoring and sensor technology
- Deploy and run a SOA process in development and production environments

► Onderwerpen

Module 1

A Service Oriented Methodology

- Introduction to a SOA adoption roadmap
- Service lifecycle
- Three analysis approaches
- Service oriented analysis
- Service oriented design
- Introduction to service oriented patterns

Advantages of SOA

- Traditional EAI Approach
- Problems With Traditional EAI Approach
- Enter Service Oriented Architecture (SOA)
- Build the Services
- We Can Easily Change the Process
- Change Flow Using Legacy Approach
- Replacing an Application
- Other Advantages
- Business Advantages
- Adoption Stages

SOA Past and Present

- From XML to Web Service to SOA
- How SOA was done before
- Emerging standards for SOA
- Compare SOA with other architectures

What is service oriented architecture?

- Creating a common understanding of SOA
- The evolution of SOA
- Introduce the concepts of services and SOA
- Design principles of SOA
- The relationship between SOA and web services
- The advantages and risks of SOA

Introduction

- Why use models with SOA.
- Difference between model and methodology.
- Why use the Unified Modeling Language?
- Introducing UML, the notation.
- Identifying business processes.
- Notation, Patterns and Methodology.
- Which Methodology to choose?

Fundamental Concepts

- Building from components.
- Modeling concepts.
- What is an object?
- Containment.
- Messages and methods.

- Object interaction.
- Exercise: testing some basic concepts.

Introduction to Business Process

- How a collection of services perform a task.
- Simple request response interaction
- Complex interaction involving many services.
- Need for a coordinator service emerges.
- Birth of orchestration or business process.
- Composing processes using processes.
- Business Process Execution Language (BPEL)
- Example business processes

Module 2

Service Enablement

- Basic web services elements
- Core web services standards stack
- The Importance of WSDL
- The design of SOAP
- The use of registries via UDDI
- The basic concepts of service orientation

Service Oriented Analysis & Design

- Objectives
- Stages of SOAD
- Identifying services
- Producing service specifications
- Functional areas of the business.
- Services belonging to these functional areas
- Functionalities belonging to these services
- Documenting service hierarchy
- Best practices
- Summary

Business Process Implementation

- Business Process Diagram
- Challenges in Process Implementation
- BPEL4WS
- Partnership
- Example: a Buy-Sell Partnership
- Modeling Partnership in BPEL
- Variables
- Simple Activities
- Invoke Activity
- Structured Activities
- Lifecycle of Process Development
- Follow Integration Patterns
- Example: A Simple Process

Distributing Services Across a Network

- Aligning functional and nonfunctional requirements
- The role of Intermediaries In Service Networks
- Introductions to WS-* Extensions
- SOA Tenets

Modeling SOA building blocks

- Using UML to analyze and design interfaces
- Generating a domain model
- Implementing and realizing Use Cases
- Showing web service collaboration
- Usage of communication diagrams

Enterprise Service Bus (ESB)

- Objectives
- Service Invocation
- Legacy System Integration
- Web Services to the Rescue
- The role of ESB in SOA
- Security and ESB

Process Driven Services

- Service layer abstraction
- Introduction to business process layer
- Process patterns
- Orchestration and choreography
- The use WS-BPEL for process automation

Module 3

A Service Oriented Reference Model

- Reference models and reference architectures
- The SOA reference model and architecture
- SOA vendors and their relationship with SOA
- SOA support in .NET and J2EE platforms

Layered Architecture

- The layers pattern.
- Classic three-tier architecture.
- Connecting to the domain layer.
- Linking to the User interface.
- Using packages to decompose a system.
- Avoiding mutual dependencies.

- What is layering and why we need them?

- Application service layer
- Business service layer
- Orchestration service layer

XML Schema Basics

- Defining a Simple Element
- Defining a Complex Element
- Defining Element Attributes
- Referring to an Element From Another Element
- Defining Abstract Data Types
- Adding Restrictions

Defining a Service in WSDL

- Sample WSDL Document Structure
- One-way
- Request-Response
- Solicit-Response
- Syntax
- SOAP Binding Example
- WSDL SOAP Binding Extensions

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

Security Architecture

- Special challenges in SOA
- Pillars of security
- Authentication
- Authorization
- Confidentiality
- Integrity
- Web Services Security
- Transport Layer Security
- Audit logging
- Common threat profiles and mitigation techniques

Module 4

The Nature of a Service

- What does a service do?
- Components that make up a service
- Common implementation techniques
- What is the communication protocol?
- How does a service execute?
- How does a client work?
- What is the message format?

Messaging Architecture

- What is messaging and why do we need them?
- How to use messaging in SOA?
- SOAP over JMS details.
- Modeling services well suited for messaging.
- Correlation and why do we need them?
- How to use correlation in SOA.
- How to implement publish subscribe in SOA?
- Sample scenarios

Transaction Management

- The ACID properties.
- Local vs. distributed transaction.
- New challenges with transaction in SOA.
- Transaction from a specific service call.
- Transaction in a long running business process.
- What is compensation and why do we need them?
- How to implement compensation?

Software Platform for SOA

- Software Tools for SOA
- The Need for a Tool
- SOA Development Life Cycle
- Oracle BP Manager
- Microsoft BizTalk Server 2006
- Rational Application Developer (RAD)
- Key Features
- Web Services Support
- Runtime Products for SOA

Deployment Architecture

- Deploying a web service
- Deploying a JCA based service
- Deploying a business process service

- Monitoring processes
- Web Service caching
- Clustering Web Services
- Load balancing service calls

Conclusions

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

► Instapniveau

XML, XSD and XSL knowledge

► Opzet

Het lesmateriaal is in het Engels maar de open-rooster training wordt in het Nederlands gegeven. Op verzoek kunnen de bedrijfstrainingen ook in het Engels of Frans gegeven worden.

Tools usage

XMLSpy
Enterprise Architect
Oracle JDeveloper 10g BPEL Designer
Oracle BPEL Process Manager
BizTalk Server
Visual Studio

Deze cursus duurt 3 dagen, van 9.30 - ca. 16.30 uur. Tussen 12.00 - 12.45 uur is er een lunch. Koffie, thee en frisdrank zijn de gehele dag beschikbaar.

► Kosten & Iestijden

De cursus kost EUR 1395,00 per persoon, excl. BTW. Inclusief cursusmateriaal, deelnamecertificaat, lunch en consumpties.