INTERNATIONAL
Systems Group, Inc.

Implementing Service Oriented Applications – An Introduction to Service Oriented Analysis, Modeling & Design

About this seminar

A SOA-based approach to application development promises that applications can be developed with less effort, once an inventory of business services has been implemented that provides much of the functionality required by new applications. Furthermore, application logic that is typically referred to as business process logic, can be elevated from the level of coding to the level of service orchestration through GUI driven modeling. This can be facilitated through Business Process Management (BPM) tools and higher level process languages.

What should make this all possible is the concept of SOA – a concept that is not completely new, but finally has found its time. SOA prescribes loosely coupled business services, minimizing the dependencies between service provider and service requestor. It calls for standard interfaces, standard access protocols, and a separation of interface from implementation. This may sound very familiar – previous technologies such as distributed objects and distributed components were promoting similar ideas. Today's SOA, however, calls for less rigid and thus less brittle interfaces that greatly minimize the dependency between the service requester and service provider. A SOA can now be defined and built using the most recent Internet standards and technologies.

This seminar introduces service oriented analysis, modeling and design. It provides an overview of the key qualities of Service Oriented Architectures like reusability, statelessness, loose coupling, and composability of services, to name a few. Keeping these qualities in mind is essential when defining a SOA that is based on well thought through service layers and service boundaries. These qualities form a continuous thread into the modeling of orchestration services, application services and infrastructure which can also be called foundation services. The seminar then drills down into high level service design and how Web Services can be used to form the basis for service design and development. The abstract concepts of SOA are contrasted against the characteristics of the core and extended Web Services standards and illustrated through a case study.

Benefits of Attending

- Understand the key elements of a service oriented Software Development Life Cycle.
- Identify the differences between traditional analysis and service oriented analysis.
- Learn a step by step approach to service oriented modeling.
- Understand how the core and extended Web Services standards can be used as foundation technologies and how they relate to service oriented design.
- Discover the principles of service oriented design.
- Understand how service oriented analysis, modeling and design differs from component oriented analysis and design.

INTERNATIONAL
Systems Group, Inc.

Who Should Attend

- Architects who want to adopt the principles of service oriented application design and development.
- IT professionals who need to see how SOA can be applied to development as well as integration projects.
- Architects and Application Developers who want to get an overview of service oriented analysis, modeling, and design, and how it differs from an object oriented approach.
- Consultants who need to recommend and use different implementation strategies for building a SOA.

Prerequisite: This class requires attendees to have knowledge of Service Oriented Architectures, and conceptual understanding of Middleware, component technologies, distributed computing, and application integration.

INTERNATIONAL
Systems Group, Inc.

Seminar Outline

1. How to Define a Service Oriented Architecture

a) SOA Architectures – recap of days 1 through 3
b) Defining loosely coupled architectures
c) Principles of service orientation
   Reusability, statelessness, loose coupling, composability, etc
d) The Service Layer Model
   Orchestration services
   Application services
   Infrastructure services

2. Service Oriented Analysis and Modeling

a) Overview of a service oriented Software Development Lifecycle
   Project phases
   Top-down vs. bottom-up approach
b) Service oriented analysis
   Requirements analysis
   Mining existing systems
c) Service modeling
   Business process decomposition
   Deriving service operations
   Organizing operations into services
   Verifying service definitions against SOA principles
d) Putting it all together: a customer case study – part I

3. Selecting Technologies for Service Design & Development

a) Selecting supporting technologies for SOA
   Why Web Services are today's choice for service design and development
   Mapping Web Services to SOA
b) Core technologies for service design and development
   Services need data: organizing information through XML Schemas
   Establishing connectivity through Simple Object Access Protocol (SOAP)
   Defining service interfaces with Web Services Definition Language (WSDL)
   Discovering services with Universal Description, Discovery and Integration (UDDI)
   Using Business Process Execution Language (BPEL) and Business Process Management Notation (BPMN) for orchestration
c) Extended technologies for service design and development
   Asynchronous service interaction: Notification
   Reliable messaging
   Security
   Transactions

4. Service Design

a) Guiding principles for service design
b) Putting it all together: a customer case study – part II
   Designing conceptual service interfaces
   Encapsulation of existing business logic
   Designing the Schemas
   Designing concrete service interfaces
   Composing business processes

5. Conclusions

a) Summary of best practices
b) Comparing component oriented development to service oriented development
c) Measuring success: the SOA Maturity Model
d) SOA outlook