

Model Driven Service-Oriented Architecture

<p>Objectives</p> <ul style="list-style-type: none"> • Discuss SOA background / benefits • Discuss SOA development method <ul style="list-style-type: none"> • Integrated with Unified Process • UML 2.0 modeling • Apply Model Driven Architecture® (MDA®) • Review UML-based business modeling • Discuss service identification activities <ul style="list-style-type: none"> • Business models • Existing assets • Discuss business goals and associated services • Understand how to model services in UML • Identify service partitions and gateways • Model service specifications, service contracts, security patterns with UML • Discuss message design considerations • Discuss service orchestration and choreography • Introduce SOA-related standards 		<p>Description</p> <p><i>Model Driven Service-Oriented Architecture</i> teaches the experienced software architect how to represent the architecture and design of a software application that uses web services. Beginning with a discussion of the business benefits and technical background of service-oriented architecture (SOA), the course continues with a review of typical business modeling activities. Then the course describes several techniques for identifying candidate services including domain decomposition, business model analysis, goal-service modeling, and asset analysis. Once a candidate portfolio of services is identified, the course discusses how to partition services, capture non-functional requirements and service polices, and service repositories. Topics of message design and security policies and requirements are discussed. The course finishes with describing how to design services, specify their internal components, and service orchestration and choreography. Students use the UML Profile for Software Services to describe SOA using composite structures, collaborations, components, and interfaces.</p>	
		<p>Course Outline (Modules and Topics)</p>	
		<ul style="list-style-type: none"> • SOA Overview <ul style="list-style-type: none"> • Business situation and benefits • UML Profile for Software Services • Business Modeling Overview <ul style="list-style-type: none"> • Workflow, activities, tasks, work products • UML for business modeling • Domain Decomposition <ul style="list-style-type: none"> • Service identification activities and approaches • Business component map • Business Model Analysis <ul style="list-style-type: none"> • Candidate services from process models • Goal Service Modeling <ul style="list-style-type: none"> • Identification of business goals • Business Motivation Metamodel (BMM) • Asset analysis <ul style="list-style-type: none"> • Candidate services from existing assets 	<ul style="list-style-type: none"> • Partition Service <ul style="list-style-type: none"> • Services as UML composite structures • Service repository, service gateway • Describe Service <ul style="list-style-type: none"> • Non-functional requirements and service policies • Design Message <ul style="list-style-type: none"> • Message exchange patterns • Identify Security Patterns <ul style="list-style-type: none"> • Security policies and requirements • Design Service <ul style="list-style-type: none"> • Service components and patterns • Specify Service Components <ul style="list-style-type: none"> • Service component interfaces • Realize Service <ul style="list-style-type: none"> • Acquiring service components • Service orchestration and choreography
<p>Duration 3 days</p>	<p>Course # 01-0204</p>	<p>Prerequisites</p> <ul style="list-style-type: none"> • Object-Oriented Design with UML course <p>Continuing education</p> <ul style="list-style-type: none"> • Requirements Management Fundamentals • User Interaction Design with UML • Test Case Design with UML <p>Classroom requirements</p> <ul style="list-style-type: none"> • No computers required 	
<p>Audience</p> <ul style="list-style-type: none"> • Enterprise architect • Business architect • Software architect • Software designer • Senior developer • System engineer 		<p>In partnership with</p>	

Proven ▶▶▶▶▶ Practical ▶▶▶▶▶ Process™