

1. Record Nr.	UNINA9910809734103321
Autore	Dikmans Lonneke
Titolo	SOA made simple : discover the true meaning behind the buzzword that is "service oriented architecture" // Lonneke Dikmans, Ronald van Luttikhuizen
Pubbl/distr/stampa	Birmingham, UK : , : Packt Pub., , 2012
ISBN	1-62198-913-5 1-283-93782-4 1-84968-417-0
Edizione	[First edition]
Descrizione fisica	1 online resource (292 pages)
Altri autori (Persone)	DikmansLonneke Van LuttikhuizenRonald
Disciplina	004.654 658.4038011
Soggetti	Service-oriented architecture (Computer science) Computer network architectures
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Includes index.
Nota di contenuto	Intro -- SOA Made Simple -- Table of Contents -- SOA Made Simple -- Credits -- About the Authors -- About the Reviewers -- www.PacktPub.com -- Support files, eBooks, discount offers and more -- Why Subscribe? -- Free Access for Packt account holders -- Instant Updates on New Packt Books -- Preface -- What this book covers -- What you need for this book -- Who this book is for -- Conventions -- Reader feedback -- Customer support -- Downloading the example code -- Errata -- Piracy -- Questions -- 1. Understanding the Problem -- The importance of information -- Example - insurance company -- Mismatch between business and IT -- Duplication of functionality and data -- Example - insurance company -- Process silos -- Example - utility companies -- Example - international software company -- Example - insurance company -- Strategies to stay ahead -- Example - a software company -- Architecture as a tool -- Layering of architecture -- Models -- Requirements -- Architecture ontology -- Enterprise architecture -- Reference architecture -- Solution architecture -- Project architecture -- Software architecture -- Service

Oriented Architecture -- Summary -- 2. The Solution -- What is a service? -- Elements of a service - contract, interface, and implementation -- Example - let's have breakfast -- Contract -- Interface -- Implementation -- Example - ordering a passport -- Consumer and provider -- Dealing with lots of service providers - service registry -- How can you make sure that people use a service? -- From sunny-side-up eggs to IT -- Example - international software company revisited -- Contract -- Interface -- Implementation -- Consumer and provider -- Reuse -- Drivers for services -- Common myths -- Every service has to be automated by software -- Every service is a web service -- Consumers of services are always IT systems.

Putting it together - what is SOA? -- Solutions -- Example - utility company -- International software company - changing existing processes -- Functional duplication - rationalizing application landscapes -- Standardization - enabling change -- Summary -- 3. Service Identification and Design -- Service identification -- Top-down -- Example of top-down service identification -- Business service identification -- Information service identification -- Technical service identification -- Bottom-up -- Meet in the middle -- I have identified my services, now what? -- Service design -- Provide value -- Meaningful -- Implementation hiding -- Trust -- Idempotent -- Isolated -- Interoperable -- Isolation -- Example: print service -- Trust -- Security -- Fault-prevention and handling -- Business faults -- Faulty user input -- Technical and software faults -- Idempotency -- Idempotency and statefulness -- Granularity -- How big should my lasagna be? -- Classification -- Reusability -- Example - reusability -- Example - good or bad service? -- Service definition revisited -- Summary -- 4. Classification of Services -- Service classification revisited -- Example - insurance company -- Other classifications -- Actor type -- Channel -- Organizational boundaries -- Security level -- Architectural layer -- Combining classifications -- Why classify your services? -- Composability -- Aggregation versus orchestration -- Example - DocumentService as a composite service -- Elementary services -- Realization -- Composite services -- Where to put the composition logic? -- Implementation -- Example 1 - database link -- Example 2 - service invocation -- Process services -- Implementation -- Isolation and composition - a contradiction? -- Passing information from smaller to larger services -- Summary -- 5. The SOA Platform -- Overview -- Services -- Implementation.

Using existing software -- Build the implementation -- Interfaces -- Proprietary interfaces -- Web services -- SOAP-based services -- RESTful services -- Contracts and Policies -- Events -- Interfaces for events -- Service composition -- Enterprise Service Bus -- Business Process Management -- Case Management -- Business rules -- User interface -- Integrated user interfaces -- Information mismatch -- Security -- Applying security in your SOA -- Service registry and service repository -- Canonical Data Model -- Design tooling -- Development tooling -- Example - Order-to-cash revisited -- Designing the solution -- Developing the solution -- Running the solution -- Summary -- 6. Solution Architectures -- Comprehensive suite or best of breed -- Comparison -- Oracle -- Services -- Events -- Oracle Event Processing (OEP) -- Business Activity Monitoring (BAM) -- Service composition -- Oracle Service Bus -- Oracle SOA Suite -- Oracle BPM Suite -- Business rules -- User interface -- Security -- Registry and repository -- Design tooling -- Design tooling for developers -- Design tooling for business analysts -- Development tooling -- Test tooling -- Testing transformations -- SCA testing

framework -- Testing from the console -- Deployment tooling --
 Deployment from the IDE -- Deployment from the console --
 Deployment using scripting -- Monitoring -- Error handling -- IBM --
 Services -- Events -- WebSphere Operational Decision Management --
 IBM Business Monitor -- Service composition -- IBM WebSphere
 Enterprise Service Bus -- IBM Business Process Manager -- Business
 rules -- User interface -- Security -- Registry and repository -- Design
 tooling -- Services -- Composite services -- Development tooling --
 Test tooling -- Deployment tooling -- Deployment from the IDE --
 Deployment from the web interface of the server -- Deployment scripts
 -- Monitoring.
 Error handling -- Microsoft -- Services -- Events -- Message-oriented
 middleware -- Complex Event Processing (CEP) -- Business Activity
 Monitoring -- Service composition -- BizTalk Server -- Windows Server
 AppFabric -- Business rules -- User interface -- Security -- Registry
 and repository -- Design tooling -- Development tooling -- Test
 tooling -- Deployment tooling -- BizTalk Server -- Monitoring -- Error
 handling -- Summary -- 7. Creating a Roadmap, How to Spend Your
 Money and When? -- Organize the SOA effort -- Business case -
 benefits for different stakeholders -- Business case explained --
 Company as a whole -- Example 1 - insurance company WATB needs
 shorter time to market -- Summary of scenarios -- Analysis of the
 scenarios -- Example 2 - insurance company TPIR needs to decrease
 operational cost -- Summary of scenarios -- Analysis of the scenarios
 -- IT -- Example - insurance company TMS needs to consolidate
 systems -- Departmental benefits -- Example - insurance company X
 wants to cut cost -- Analysis of the scenarios -- Approaches --
 Example - Document Management Service -- Top-down identification
 -- Bottom-up identification -- Meet in the middle -- Roadmap -- Work
 packages -- Service by service -- Process by process -- Feature by
 feature -- System by system -- Comparison -- Maturity and stages --
 Stage 0: Starting with SOA -- Stage 1: Newlyweds -- Stage 2: Live --
 Stage 3: Growing up -- Stage 4: Experience -- Stage 5: Maintenance --
 Summary -- 8. Life Cycle Management -- Service stages -- Versioning
 of services -- Type of change - contract, interface, and implementation
 -- Changing the contract -- Changing the interface -- Changing the
 implementation -- Versioning schemes -- Versioning and life cycle
 stages -- Making the version explicit for service consumers --
 Communicating change -- Tooling -- Standards -- Information
 needed.
 Find services -- Troubleshooting -- Change process -- Registries and
 repositories in your IT landscape -- Enterprise architecture tools --
 Business Process Management tool -- Configuration Management
 Database -- Bug and issue tracker system -- ESB -- Business Activity
 Monitoring -- Infrastructure monitoring -- Summary -- 9. Pick your
 Battles -- Governance -- Architecture process -- Ad hoc business need
 -- Define the solution -- Deviations -- Integration in the solution
 architecture -- Planned feature -- Pick your battles -- Development
 process -- Pick your battles -- Operations -- Pick your battles --
 Change management -- Pick your battles -- Summary -- 10.
 Methodologies and SOA -- Demand management -- Methodology --
 Impact of SOA -- Project management -- Methodology -- Impact of
 SOA -- Software development -- Methodology -- Impact of SOA --
 Application management -- Methodology -- Impact of SOA -- IT
 service and operations management -- Methodology -- Impact of SOA
 -- Summary -- Index.

three clear sections, in this book you'll learn from both theory as well as step-by-step implementation examples to aid in your understanding of this often poorly- articulated industry term. If you are an architect who wants to be completely clear in your understanding of what SOA is, then this book is essential. In fact, anyone (designer, developer, administrator or team lead) who is implementing or about to implement an architecture in an IT environmen
