



Applied SOA

Service-Oriented Architecture and Design Strategies

Mike Rosen
Boris Lublinsky
Kevin T. Smith
Marc J. Balcer



WILEY

Wiley Publishing, Inc.



About the Authors

Mike Rosen is chief scientist at Wilton Consulting Group, which provides expert consulting on software architecture, SOA, and enterprise architecture. He is also director of enterprise architecture for the Cutter Consortium and editorial director of the SOA Institute. He frequently speaks at industry symposia and contributes to industry journals.

Boris Lublinsky is lead architect at Navteq, where he is responsible for SOA and BPM implementations. He is a frequent contributor to technology magazines and a speaker at industry conferences. Boris is also an SOA news editor for InfoQ.

Kevin T. Smith is a technical director at ManTech MBI (formally McDonald Bradley, Inc.), where he builds highly secure and data-driven SOA solutions for the U.S. government. He is the author of many SOA technology articles in industry magazines, such as the SOA/Web Services Journal, and has coauthored several technology books, including *The Semantic Web* (Wiley, 2003), *Professional Portal Development with Open Source Tools* (Wrox, 2004), *More Java Pitfalls* (Wiley, 2003), and *Essential XUL Programming* (Wiley, 2001), in addition to the books where he has written chapters as a contributing author. Kevin has led SOA workshops and has presented at numerous industry conferences, such as the RSA Security Conference, JavaOne, the Semantic Technology Conference, the Apache Open Source Conference, Net-Centric Warfare, the Object Management Group, and the Association for Enterprise Integration.

Marc J. Balcer is the founder of ModelCompilers.com, a provider of tools and services for realizing the power of model-based development, and the coauthor of *Executable UML: A Foundation for Model-Driven Architecture* (Addison-Wesley, 2002). He has over 15 years of experience in

developing, deploying, and managing projects based upon executable models and model-driven development techniques.

As a party to many enterprise development projects, Marc has witnessed firsthand how the precision of application and architecture models can make the difference between spectacular success and miserable failure. He has applied Executable UML to projects in such diverse areas as medical instrumentation, transportation logistics, telecommunications, and financial services.



Acknowledgments

Well, who to thank for all the help? First, thanks to all the people who supported me throughout this process. There were many, but a few stand out for special mention: all my friends and clients who cut me a little slack when I might have been slightly unresponsive during the final push to finish everything; my friends in the travel industry who inspired the case study; everyone at Cutter Consortium for constant encouragement; SOAInstitute for providing a forum to teach and discuss all things SOA; Robert Elliott at Wiley, who had the uncanny timing to call me during a lull in my consulting practice and ask if I wanted to be involved in an SOA book; and Sydney Jones, our project editor, for putting up with our changes and delays. I hope she wasn't just being nice when she said we weren't the worst group of authors ever. Thanks to Jim Amsden, a friend and colleague, who also turned out to be the best technical editor you could imagine; Jeroen van Tyn and Laura O'Brian for the great Business Use Cases in Chapters 6 and 7 and Appendix A; my good friend Ken Orr for teaching me about business architecture, processes, and semantics over the years; my coauthors, for contributing to a collaborative project where we all learned from each other and everyone's chapters, and the book, benefited; and most importantly, to my awesome wife, Tamar Krichevsky, who not only put up with it all, but who also read every single chapter of the book and compiled and wrote the fantastic Evaluating SOA Services appendix. Thanks.

— Mike Rosen

I would like to thank Mike for calling me out of the blue and asking whether I would like to participate in this exciting project. I really enjoyed collaborating with Mike, Kevin, and Marc. It allowed me to learn more about SOA and significantly improved the quality of my chapters. Many thanks to the people whom I used to work with over the years, especially Didier Le Tien, Dmitry

Tyomkin, and Deborah Shaddon, for always challenging me with tough architecture questions and pointing at deficiencies in my solutions; Jay Davidson and Edward Kuffert for explaining to me the importance of business architecture and the way the insurance industry works; and Jerry Daus, Matt O'Neal, and Maria Mernandez for helping me to understand how IBM software works and the best ways to use it. I am also thankful for all of the failed and successful projects that I worked on, which taught me what is important and what is not, and why things fail or succeed. Most importantly, to my wonderful wife, Lilia, for patiently putting up with me spending more time with my computer than with her. Thanks.

— Boris Lublinsky

I would like to thank my three talented coauthors, Mike, Boris, and Marc — it has been a pleasure working with you on this exciting and challenging project. Mike, you did a great job of guiding us in this process, and I would especially like to thank Boris for his additions to the chapters on Composing Services (Chapter 8) and SOA Governance (Chapter 12). I would like to thank Vaughn Bullard for his suggestions on Chapter 12 and Layer7's Toufic Boubez for his support of my discussion on dynamic policy adaptation ("Policy Application Points") in Chapters 11 and 12. Special thanks to Ken and Myrtle Ruth Stockman for allowing me to use their nicknames in one of my examples, and thanks to my "readability editors," Helen G. Smith and Lois G. Schermerhorn.

I would like to thank my company, ManTech MBI (formerly McDonald Bradley, Inc.) in Herndon, VA, with special and sincere thanks to those who encouraged my writing of this book on my own time — specifically, thanks to Danny Proko, Bill Pulsipher, Waymond Edwards, John Sutton, Gail Rissler, Mark Day, and Ken Bartee. I would like to give my thanks (and apologies) to my wonderful wife, Gwen, and my sweet daughters, Isabella and Emma! Thank you for putting up with me as I went into isolation for countless nights and weekends while writing this book. I would like to thank Ashland Coffee and Tea, who once again didn't kick me out when I camped out there for days at a time for writing, research, and of course, caffeine.

Thanks to the Washington Redskins, who thoughtfully did not have a good enough football season that it would distract me from writing on Sundays. Thanks to other people, places, and things that most likely affected my writing in a positive way (in no particular order): Gavin Sutcliffe; Eric Monk; Nick Duan; Sue Lee; Joanie Barr; John Medlock; Kyle Hendrickson; Tom Diepenbrock; Scooby-Doo; Jeff Phelps; Ruben Wise; Kim Gumabay; Mike Hoops, the AMC Pacer, Ralph Perko, Kathleen Ferris, Brad Giaccio, Kevin Moran; Mike Daconta; Leo Obrst; Fox; my community group (Russ and Debi Garber, Ed and Lori Buchanan, Steve and Ani Tetrault, Ed Hoppe); Kyle Rice; Thai Gourmet in Kings Charter; the Apostle Paul; Sean, Jen, Garrett, and Parker Cullinan; Daniel Buckley; Ken Pratt; Adam Dean; Mike Rohan; Carl and

Sharon Smith, Emma when she sleeps past 4:00 A.M., Bill, Farron, Casey, and Will Smith, New Hanover Church, Grace Community Presbyterian Church, Mungo, and T3. Finally, all glory, laud, and honor to the one who was, who is, and who is to come.

— Kevin T. Smith

Many ideas emerge from the everyday work of developing real solutions. In addition to my coauthors, I would like to acknowledge the contributions, criticism, and insights from current and former colleagues, including Steve Dowse of International Asset Systems, and Brian Itow, Gary Marcos, Julio Roque, and Matt Samsonoff of AZORA Technologies. Most importantly, I would like to thank my partner, Canares (“Chicho”) Aban, for his dedication and support during this project.

— Marc J. Balcer



Contents at a Glance

Part One	Understanding SOA	
Chapter 1	Realizing the Promise of SOA	3
Chapter 2	SOA – Architecture Fundamentals	27
Chapter 3	Getting Started with SOA	77
Part Two	Designing SOA	
Chapter 4	Starting with the Business	119
Chapter 5	Service Context and Common Semantics	159
Chapter 6	Designing Service Interfaces	203
Chapter 7	Designing Service Implementations	253
Chapter 8	Composing Services	273
Chapter 9	Using Services to Build Enterprise Solutions	311
Chapter 10	Designing and Using Integration in SOA Solutions	353
Chapter 11	SOA Security	391
Chapter 12	SOA Governance	449
Part Three	Case Studies	
Chapter 13	Case Study – Travel Insurance	495
Chapter 14	Case Study – Service-Based Integration in Insurance	541

Appendix A Business Use Cases	579
Appendix B Evaluating SOA Services	589
Appendix C Additional Reading	621
Index	631



Contents

Acknowledgments	ix
Introduction	xxvii
Part One Understanding SOA	
Chapter 1 Realizing the Promise of SOA	3
Once Upon a Time . . .	4
Learning from History	7
What Went Wrong?	8
What Went Right?	9
What Can You Learn?	10
The Promise of SOA	10
The Challenges of SOA	11
Reuse	11
Efficiency in Development	14
Integration of Applications and Data	15
Agility, Flexibility, and Alignment	16
Meeting the Challenge	18
Reference Architecture	19
Common Semantics	19
Governance	20
Business Process Modeling	22
Design-Time Service Discovery	22
Model-Based Development	23
Best Practices in SOA Analysis and Design	24
Summary	25
Chapter 2 SOA – Architecture Fundamentals	27
What Is Architecture?	28

Architectural Styles	29
Architectural Principles and Practices	30
What Is Service-Oriented Architecture?	33
1. Defining a Service	37
2. Defining How Services Are Built and Used	38
3. Integrating Packaged and Legacy Systems into the Service Environment	39
4. Combining Services into Enterprise Processes	39
5. Specifying the Technology Infrastructure	39
Specifying the Technology Infrastructure	39
Specifying the Application Infrastructure Required to Support Services	40
6. Defining Common Semantics and Data	40
7. Aligning Services with the Business	40
8. Determining How to Use the Architecture	41
Determining the Development Environment, Frameworks, Infrastructure, and Tools	41
Defining Metrics for Measuring Success	41
Business-Driven SOA	41
SOA and Other Architectures	44
Enterprise Architecture	44
Software Architecture	46
EA, 4+1, and Services	49
What Is a Service?	50
A Word about Information Architecture	52
Service Characteristics	53
Service Granularity	56
Service Dimensions	60
Loose Coupling Is King	64
Location Transparency	65
Interface and Implementation	66
Data	66
Versioning	67
Interoperability and Platform Independence	67
Usage, Assumptions, and Knowledge	68
Common Service Patterns	68
Service Types and Purpose	70
SOA Reference Architecture	73
Summary	75
Chapter 3 Getting Started with SOA	77
Overview of SOA Implementation Methodology	78
SOA Reference Architecture	82
Minimum Architecture	83
9-Month Checkpoint	84
18-Month Checkpoint	84
Long Term	85

Business Architecture	85
Business Processes	86
Information Design	88
Service Identification	90
Service Specification	94
Service Expectations	96
Interaction Model	97
Service Constraints	97
Service Location	98
Services Realization	98
Buying Services	99
Outsourcing Services	99
Building Services	102
Summary of Service Identification and Realization Concerns	102
Service Life Cycle	104
The Service Design Process	106
Top-Down Approaches	106
Enterprise System Analysis	107
Business Process Model	107
Bottom-Up Approaches	108
Utility Services	108
Service Enabling	108
Middle-Out: The Best of Both	109
Process Summary	109
Activities	110
Artifacts	111
Repositories	111
Governance	111
Process Phases	111
Architectural Context	111
Business	112
Design	112
Implementation	112
Test	112
Practical steps	113
Summary	115

Part Two Designing SOA

Chapter 4 Starting with the Business	119
Business Architecture	121
Enterprise Business Architecture	124
Project Business Architecture	124
Value Chain	125
Business Context	126
Understanding the Business Motivation Model	132
Ends	134

	Vision	134
	Desired Results	134
	Means	134
	Mission	134
	Course of Action	135
	Directives	135
	Influencers	136
	Alignment and Traceability	136
	Business Process Management and Modeling	137
	Basic Business Process Model Components	139
	Executable Models	140
	Business Process Models in an SOA World	142
	How to Create Business Process Models	143
	Use Cases	143
	Use Cases and Business Process Models	144
	One Use Case, Multiple Scenarios	144
	Step Reuse	146
	Documents	146
	Conditional Business Process Models	148
	Conditional Flows	148
	Conditional Operation Outputs	148
	Recap: Processes and Services	149
	Organizing Services	151
	Domains	152
	Types of Domains	154
	The Service Inventory	155
	Summary	156
Chapter 5	Service Context and Common Semantics	159
	The Importance of Semantics in SOA	160
	Core Information Modeling	163
	Objects and Attributes	163
	Classes, Attributes, and Instances	164
	Attributes and Instances	165
	Associations	166
	Association Multiplicities	166
	Finding Classes	167
	Defining Types	167
	Simple Types	168
	Numeric Types	168
	Symbolic Types	169
	Enumeration Types	169
	Composite Types	170
	Implementing Types	170
	Beyond the Basics	170
	Identifiers and Uniqueness Constraints	170
	Identifier and Identity	171

Contrived Identifiers	171
Multiple Population Identifiers	171
Subpopulation Identifiers	172
Specializations	172
Derived Attributes	174
Value Constraints	176
Structuring Information Models	176
Documents	177
Defining Documents	178
Adapting the Information Model	179
Multiple Documents	180
Documents and XML	181
XML Schema	184
Types in Schemas	185
Document Variations in Schemas	187
Designing for Change	188
XML Patterns	190
Derivation Using Abstract Classes	192
Derivation by Extension	193
Derivation by Restriction	194
Disallowing Derivations	195
Russian Doll	195
Salami Slice	196
Venetian Blind	197
Best Practices for the SOA Architect	198
Using Abstraction to Avoid “SOA Stovepipes”	199
Reuse Standards to Avoid Reinventing the Wheel	200
Develop Information Models Based on Use Cases	201
With Change, Crawl, Walk, Then Run	201
Summary	202
Chapter 6 Designing Service Interfaces	203
Services Revisited	204
Service Characteristics	204
Granularity	205
Scope	205
Visibility	206
Interaction Styles	207
Parameter Passing	208
Document Passing	208
Data Passing	209
Request/Reply	210
Events	211
Mixed Style	212
Design Guidelines	213
Isolating Responsibilities	213
Understanding Overall Context	215

Identifying Granularity	217
Stateless Interfaces	218
Exceptions	220
Designing Documents	221
Interface Design Illustrated	222
Overview of Models and Diagrams	223
ACME Insurance Example	224
Conceptual Architecture	225
Problem Space Model	227
Use Case Diagrams	227
Actors	228
Initial Scenario Diagrams	229
Purchase Insurance Scenario	230
Enterprise Service Context and Inventory	232
Detailed Scenario Diagrams	234
Information Model	239
Service Specification	239
Solution Model	241
Service Model	241
Service Definition Diagrams	243
Operations Procedures	246
More Information Model	246
Document Model	248
Summary	249
Chapter 7 Designing Service Implementations	253
Basic Service Architecture	254
Layer Responsibilities	256
Using Activity Diagrams for Modeling Operational Logic	257
Implementation Components	259
Implementing the Interface Layer	260
Document Receipt	261
Syntactic Validation	261
Transformations	262
Implementing the Business Layer	263
Semantic Input Validation	263
Performing the Business Logic of the Operation	264
Computing and Returning Results	267
Implementing the Resource Layer	267
Implementation Design Illustrated	268
Business Layer	268
Create Quote Operation (Quoting Service — Request Quote Scenario)	269
Price for Quote Operation (Automobile LOB Pricing)	269
Summary	272

Chapter 8	Composing Services	273
	Understanding Service Composition	274
	Separation into Service Layers	275
	Orchestration and Choreography	276
	The Relationship between BPM and Composition	278
	Architectural Models in Service Composition	279
	Hierarchical and Conversational Composition	279
	Conductor-Based and Peer-to-Peer Composition	280
	Service Composition Implementation	281
	Programmatic Composition	281
	Service Component Architecture Composition	282
	Event-Based Composition	285
	Orchestration Engine–Based Composition	286
	Centralized and Decentralized Orchestration Approaches	290
	Service Composition and Business Rules	292
	Service Composition and Transactions	295
	Incorporating Human Activities into Service Composition	297
	Orchestration with BPEL	299
	Composition Example — Case Study	301
	The Problem	301
	High-Level Design Decisions	302
	Process Modeling	303
	Dos and Don'ts in Service Composition	307
	Avoid Static, Programmatic Orchestration	307
	Use a Layered Service Approach	308
	When Using BPEL, Use Abstract Processes	308
	Summary	309
Chapter 9	Using Services to Build Enterprise Solutions	311
	Enterprise Solutions versus Applications	312
	Service-Based Enterprise Solutions	313
	Layered SOA Architecture and Multitiered Application	
	Architecture	317
	Locating Services	321
	Example: Implementing Service Access for Policy Issuance	325
	Versioning: Dealing with Service Changes	325
	Version Deployment and Access Approaches	327
	Example: Coping with Changes in Policy Issuance Solutions	329
	Architecting Security for Service-Based Solutions	330
	Using a Security Gateway	330
	Using an Interceptor in Security Implementations	331
	Example: Architecting Security for Policy Issuance Solutions	333
	Exception Handling and Logging in Enterprise Solutions	333
	Monitoring and Managing Enterprise Solutions	337

Business Activity Monitoring	338
Technical Monitoring and Management of SOA Solutions	340
Enterprise Service Bus-Unified Infrastructure for Enterprise Solutions	343
Defining ESB	344
ESB Architecture	346
Stand-alone ESB	348
ESB as a Service Container	348
ESB as a Framework	349
Choosing an ESB	350
Summary	351
Chapter 10 Designing and Using Integration in SOA Solutions	353
Challenges of Integration in SOA	354
Characteristics of Islands of Data	354
Characteristics of Islands of Automation	355
Characteristics of Islands of Security	355
Integration in SOA Defined	358
Integration Services	360
Integration Access Implementations	364
Using Messaging Infrastructure to Implement Integration Access	365
Using a Message Broker to Implement Integration	367
Using Existing Web Services to Implement Integration	369
Using JCA/J2C Adapters to Implement Integration	372
Using Web Service Wrappers to Implement Integration	374
Using Direct Database Access to Implement Integration	375
Using an Enterprise Service Bus to Implement Integration	376
Special Considerations for Implementing of Integration	377
Data Mapping in Integration	378
Security Support for Integration	380
Transactional Support in Integration	381
Versioning Integration	383
Dealing with Large Messages	384
Data Virtualization and Enterprise Data Bus	386
Summary	389
Chapter 11 SOA Security	391
SOA Security Goals and Fundamentals	392
Authentication	392
Authorization and Access Control	395
Two Types of Access Control — DAC and MAC	396
Federated Identity and Cross-Enterprise Access	397
Confidentiality	400
Integrity	401
Non-Repudiation	404

Web Service Security Standards and Specifications	405
WS-Security SOAP Messaging	405
WS-Trust	406
WS-Federation	409
WS-SecureConversation	410
WS-SecurityPolicy and the WS-Policy Framework	410
SAML	411
XACML	413
XML Signature	415
XML Encryption	415
SOA Security Blueprints	416
Separation of Security into Components and Services	416
Authentication and Identity Blueprints	419
Identity Propagation for SSO Solutions	420
Point-to-Point Authentication	425
Access Control Blueprints	427
Controlling Access to Data, Not Just Services	427
Access Control Policy Enforcement Approaches	428
Auditing and Troubleshooting	435
Flexibility with Dynamic WS-SecurityPolicy Adaptation	436
Complete Architecture Analysis	437
Applying Concepts from This Chapter — A Simple Case Study	437
Establishing Enterprise Security Services	440
Defining Identity Propagation and Access Control	441
The Security Game Plan for the SOA Architect	443
Plan from the Beginning, Focusing on Requirements	443
Crawl and Walk before Running	444
Use Accepted Standards (in a Standard Way)	444
Understand the Details of the Standards	445
Understand the Impact of Security on Performance	446
Try to Keep It Simple	446
Summary	447
Chapter 12 SOA Governance	449
SOA Management and Governance Defined	450
The Case for SOA Governance	453
The Reality of Change in Real-World Deployments	453
The Need for an Enterprise Big Picture	455
The Need for Explicit Run-Time Service Policies	456
The Need to Separate Policy Logic from Business Logic	457
SOA Governance and the Service Life Cycle	459
Design-Time Governance	462
The Service Identification Process	464
The Service Design and Specification Process	465
The Service Implementation Process	467

Deploy-Time Governance	469
Run-Time Governance	471
Practical SOA Governance	475
Structuring Your Organization for Governance	475
Developing Enterprise Policy	477
Using the Service Repository	481
Cataloging and Discovery	483
Validation	483
Dependency Management	484
Service Evolution and Versioning	484
Artifacts Publishing Governance	484
Support for Multiple Artifact Types	484
Developing and Registering Run-Time Policies	486
Run-Time Policy Enforcement and Adaptation	488
Summary	490
Part Three Case Studies	
Chapter 13 Case Study – Travel Insurance	495
Travel Insurance	496
The Scenario	496
Conceptual Architecture	497
Business Concerns	498
Business Value Chain	499
Business Motivation	500
Brief Review	502
Business Analysis	506
Business Process Model	509
Service Conceptual Architecture	510
Use Cases	512
Enterprise Context	517
Solutions Architecture	518
Authentication	519
Authorization	521
Confidentiality	522
Integrity and Non-Repudiation	522
The Big Security Picture	523
Service Inventory	524
Entity Diagram	525
Information Model	527
Document Model	529
Service Interface Design	530
Service Interactions	532
Document Design	533
Service Implementation Design	534

Service Specification	534
Implementation Layers	535
Operation Procedure	536
Summary	538
Chapter 14 Case Study – Service-Based Integration in Insurance	541
ACME Insurance	542
High-Level Integration Design	547
Establishing Policy Submission	548
Rate Insurance Policy	551
Assess Insured Location	552
Get Driver’s Information	553
Generate Policy Notice Documents	554
Collect Policy Financials	555
Work with Documents	556
Integration Requirements for the ACME Implementation	556
Integration with Existing CICS Transactions	557
Integration Approaches	558
ACME’s Implementation of a CICS-Based Integration	562
Integration with the Existing COM Components	565
ACME’s Implementation of Integration with COM Components	567
Integration Based on the Existing Java APIs	568
ACME’s Implementation of Integration Based on Java APIs	569
Integration with the Existing J2EE Applications	570
ACME’s Implementation of Integration for J2EE-Based Applications	572
Integration with Existing Databases	573
ACME’s Implementation of Database Integrations	575
Integration Based on the Vendor’s Web Services	576
ACME’s Implementation of Integration Using the Vendor’s Web Services	577
Summary	578
Appendix A Business Use Cases	579
Business Use Case BU01 — Quote Insurance	579
Basic Workflow	580
Alternative Workflow: Unacceptable Risk	582
Performance Goals	582
Business Use Case BU02 — Process Application	582
Basic Workflow	583
Alternative Workflow: Unacceptable Risk	585
Performance Goals	585

Business Use Case BU03 — Change Policy	585
Basic Workflow	585
Alternative Workflow: Unacceptable Risk	587
Extension Points	588
Appendix B Evaluating SOA Services	589
How Do I Assess Services?	590
Alignment Characteristics	597
Business Alignment	598
Specification	599
Fit for Purpose	601
Security	602
Semantic Alignment	603
Design Characteristics	605
Isolation of Responsibilities	605
Abstraction	606
Coupling	607
Granularity	609
Stateless	610
Composable	611
Governance	612
Technical Characteristics	613
Specification	613
Service Level Agreement	615
Extensibility	616
Variability and Configurability	617
Autonomy	618
Housekeeping Characteristics	619
Appendix C Additional Reading	621
Index	631