

SERVICE TECHNOLOGY SERIES FROM THOMAS ERL



**The Top-Selling, De Facto Guide to SOA**

Now Updated with New Content and Coverage of Microservices!

SECOND EDITION

# Service-Oriented Architecture

Analysis and Design for Services and Microservices

ServiceTech  
PRESS

**Thomas Erl**

With contributions by  
Paulo Merson and Roger Stoffers

# Service-Oriented Architecture

# Service-Oriented Architecture: Analysis and Design for Services and Microservices

## Table of Contents

Cover

Half Title

Title Page

Copyright Page

Contents

Acknowledgments

Reader Services

CHAPTER 1: Introduction

1.1 How Patterns Are Used in this Book

1.2 Series Books That Cover Topics from the First Edition

1.3 How this Book Is Organized

Part I: Fundamentals

Chapter 3, Understanding Service-Oriented

Chapter 4, Understanding SOA

Chapter 5, Understanding Layers with Services and Microservices

Part II: Service-Oriented Analysis and Design

Chapter 6, Analysis and Modeling with Web Services and Microservices

Chapter 7, Analysis and Modeling with REST Services and Microservices

Chapter 8, Service API and Contract Design with Web Services

Chapter 9, Service API and Contract Design with REST Services and Microservices

Chapter 10, Service API and Contract Versioning with Web Services and REST Services

Part III: Appendices

# **Table of Contents**

Appendix A, Service-Orientation Principles Reference

Appendix B, REST Constraints Reference

Appendix C, SOA Design Patterns Reference

Appendix D, The Annotated SOA Manifesto

## **1.4 Page References and Capitalization for Principles, Constraints, and Patterns**

### **Additional Information**

Symbol Legend

Updates, Errata, and Resources ([www.servicetechbooks.com](http://www.servicetechbooks.com))

Service-Orientation ([www.serviceorientation.com](http://www.serviceorientation.com))

What Is REST? ([www.whatisrest.com](http://www.whatisrest.com))

Referenced Specifications ([www.servicetechspecs.com](http://www.servicetechspecs.com))

SOASchool.com® SOA Certified Professional (SOACP)

CloudSchool.com Cloud Certified Professional (CCP)

BigDataScienceSchool.com Big Data Science Certified Professional (BDSCP)

Notification Service

## **CHAPTER 2: Case Study Backgrounds**

### **2.1 How Case Studies Are Used**

### **2.2 Case Study Background #1: Transit Line Systems, Inc**

### **2.3 Case Study Background #2: Midwest University Association**

## **PART I: FUNDAMENTALS**

### **CHAPTER 3: Understanding Service-Orientation**

#### **3.1 Introduction to Service-Orientation**

Services in Business Automation

Services Are Collections of Capabilities

Service-Orientation as a Design Paradigm

Service-Orientation Design Principles

#### **3.2 Problems Solved by Service-Orientation**

Silo-based Application Architecture

It Can Be Highly Wasteful

Its Not as Efficient as It Appears

# **Table of Contents**

It Bloats an Enterprise

It Can Result in Complex Infrastructures and Convoluted Enterprise Architectures

Integration Becomes a Constant Challenge

The Need for Service-Oriented

Increased Amounts of Reusable Solution Logic

Reduced Amounts of Application-Specific Logic

Reduced Volume of Logic Overall

Inherent Interoperability

## **3.3 Effects of Service-Oriented on the Enterprise**

Service-Oriented and the Concept of Application

Service-Oriented and the Concept of Integration

The Service Composition

## **3.4 Goals and Benefits of Service-Oriented Computing**

Increased Intrinsic Interoperability

Increased Federation

Increased Vendor Diversification Options

Increased Business and Technology Domain Alignment

Increased ROI

Increased Organizational Agility

Reduced IT Burden

## **3.5 Four Pillars of Service-Oriented**

Teamwork

Education

Discipline

Balanced Scope

## **CHAPTER 4: Understanding SOA**

Introduction to SOA

### **4.1 The Four Characteristics of SOA**

Business-Driven

Vendor-Neutral

Enterprise-Centric

Composition-Centric

Design Priorities

# Table of Contents

## 4.2 The Four Common Types of SOA

- Service Architecture
- Service Composition Architecture
- Service Inventory Architecture
- Service-Oriented Enterprise Architecture

## 4.3 The End Result of Service-Oriented Architecture and SOA

## 4.4 SOA Project and Lifecycle Stages

- Methodology and Project Delivery Strategies
- SOA Project Stages
- SOA Adoption Planning
- Service Inventory Analysis
- Service-Oriented Analysis (Service Modeling)
  - Step 1: Define Business Automation Requirements
  - Step 2: Identify Existing Automation Systems
  - Step 3: Model Candidate Services
- Service-Oriented Design (Service Contract)
- Service Logic Design
- Service Development
- Service Testing
- Service Deployment and Maintenance
- Service Usage and Monitoring
- Service Discovery
- Service Versioning and Retirement
- Project Stages and Organizational Roles

## CHAPTER 5: Understanding Layers with Services and Microservices

### 5.1 Introduction to Service Layers

- Service Models and Service Layers
- Service and Service Capability Candidates

### 5.2 Breaking Down the Business Problem

- Functional Decomposition
- Service Encapsulation
- Agnostic Context
- Agnostic Capability

# Table of Contents

Utility Abstraction

Entity Abstraction

Non-Agnostic Context

Micro Task Abstraction and Microservices

Process Abstraction and Task Services

## 5.3 Building Up the Service-Oriented Solution

Service-Orientation and Service Composition

Capability Composition and Capability Recomposition

Capability Composition

Capability Composition and Microservices

Capability Recomposition

Logic Centralization and Service Normalization

## PART II: SERVICE-ORIENTED ANALYSIS AND DESIGN

### CHAPTER 6: Analysis and Modeling with Web Services and Microservices

#### 6.1 Web Service Modeling Process

Case Study Example

Step 1: Decompose the Business Process (into Granular Actions)

Case Study Example

Step 2: Filter Out Unsuitable Actions

Case Study Example

Step 3: Define Entity Service Candidates

Case Study Example

Step 4: Identify Process-Specific Logic

Case Study Example

Step 5: Apply Service-Orientation

Step 6: Identify Service Composition Candidates

Case Study Example

Step 7: Analyze Processing Requirements

Case Study Example

Step 8: Define Utility Service Candidates

Case Study Example

Step 9: Define Microservice Candidates

# Table of Contents

Case Study Example

Step 10: Apply Service-Oriented

Step 11: Revise Service Composition Candidates

Case Study Example

Step 12: Revise Capability Candidate Grouping

## CHAPTER 7: Analysis and Modeling with REST Services and Microservices

### 7.1 REST Service Modeling Process

Case Study Example

Step 1: Decompose Business Process (into Granular Actions)

Case Study Example

Step 2: Filter Out Unsuitable Actions

Case Study Example

Step 3: Define Entity Service Candidates

Case Study Example

Step 4: Identify Process-Specific Logic

Case Study Example

Step 5: Identify Resources

Case Study Example

Step 6: Associate Service Capabilities with Resources and Methods

Case Study Example

Step 7: Apply Service-Oriented

Case Study Example

Step 8: Identify Service Composition Candidates

Case Study Example

Step 9: Analyze Processing Requirements

Case Study Example

Step 10: Define Utility Service Candidates (and Associate Resources and Methods)

Case Study Example

Step 11: Define Microservice Candidates (and Associate Resources and Methods)

Case Study Example

Step 12: Apply Service-Oriented

Step 13: Revise Candidate Service Compositions



# **Table of Contents**

Case Study Example

Step 14: Revise Resource Definitions and Capability Candidate Grouping

## **7.2 Additional Considerations**

Uniform Contract Modeling and REST Service Inventory Modeling

REST Constraints and Uniform Contract Modeling

REST Service Capability Granularity

Resources vs. Entities

## **CHAPTER 8: Service API and Contract Design with Web Services**

### **8.1 Service Model Design Considerations**

Entity Service Design

Utility Service Design

Microservice Design

Task Service Design

Case Study Example

### **8.2 Web Service Design Guidelines**

Apply Naming Standards

Apply a Suitable Level of Contract API Granularity

Case Study Example

Design Web Service Operations to Be Inherently Extensible

Case Study Example

Consider Using Modular WSDL Documents

Case Study Example

Use Namespaces Carefully

Case Study Example

Use the SOAP Document and Literal Attribute Values

Case Study Example

## **CHAPTER 9: Service API and Contract Design with REST Services and Microservices**

### **9.1 Service Model Design Considerations**

Entity Service Design

Utility Service Design

Microservice Design

# Table of Contents

Task Service Design

Case Study Example

## 9.2 REST Service Design Guidelines

Uniform Contract Design Considerations

Designing and Standardizing Methods

Designing and Standardizing HTTP Headers

Designing and Standardizing HTTP Response Codes

Customizing Response Codes

Designing Media Types

Designing Schemas for Media Types

Complex Method Design

Stateless Complex Methods

Fetch Method

Store Method

Delta Method

Async Method

Stateful Complex Methods

Trans Method

PubSub Method

Case Study Example

## CHAPTER 10: Service API and Contract Versioning with Web Services and REST Services

### 10.1 Versioning Basics

Versioning Web Services

Versioning REST Services

Fine and Coarse-Grained Constraints

### 10.2 Versioning and Compatibility

Backwards Compatibility

Backwards Compatibility in Web Services

Backwards Compatibility in REST Services

Forwards Compatibility

Compatible Changes

Incompatible Changes

### 10.3 REST Service Compatibility Considerations

# **Table of Contents**

## 10.4 Version Identifiers

## 10.5 Versioning Strategies

The Strict Strategy (New Change, New Contract)

Pros and Cons

The Flexible Strategy (Backwards Compatibility)

Pros and Cons

The Loose Strategy (Backwards and Forwards Compatibility)

Pros and Cons

Strategy Summary

## 10.6 REST Service Versioning Considerations

## **PART III: APPENDICES**

APPENDIX A: Service-Orientation Principles Reference

APPENDIX B: REST Constraints Reference

APPENDIX C: SOA Design Patterns Reference

Whats a Design Pattern?

Whats a Design Pattern Language?

Pattern Profiles

APPENDIX D: The Annotated SOA Manifesto

The SOA Manifesto

The SOA Manifesto Explored

Preamble

Priorities

Guiding Principles

About the Author

Index