



# SOA Using Java Web Services

By Mark D. Hansen

Download now

Read Online ➔

SOA Using Java Web Services By Mark D. Hansen

## Expert Solutions and State-of-the-Art Code Examples

*SOA Using Java™ Web Services* is a hands-on guide to implementing Web services and Service Oriented Architecture (SOA) with today's Java EE 5 and Java SE 6 platforms. Author Mark Hansen presents in explicit detail the information that enterprise developers and architects need to succeed, from best-practice design techniques to state-of-the-art code samples.

Hansen covers creating, deploying, and invoking Web services that can be composed into loosely coupled SOA applications. He begins by reviewing the “big picture,” including the challenges of Java-based SOA development and the limitations of traditional approaches. Next, he systematically introduces the latest Java Web Services (JWS) APIs and walks through creating Web services that integrate into a comprehensive SOA solution. Finally, he shows how application frameworks based on JWS can streamline the entire SOA development process and introduces one such framework: SOA-J.

The book

- Introduces practical techniques for managing the complexity of Web services and SOA, including best-practice design examples
- Offers hard-won insights into building effective SOA applications with Java Web Services
- Illuminates recent major JWS improvements—including two full chapters on JAX-WS 2.0
- Thoroughly explains SOA integration using WSDL, SOAP, Java/XML mapping, and JAXB 2.0 data binding
- Walks step by step through packaging and deploying Web services components on Java EE 5 with JSR-181 (WS-Metadata 2.0) and JSR-109
- Includes specific code solutions for many development issues, from publishing REST endpoints to consuming SOAP services with WSDL
- Presents a complete case study using the JWS APIs, together with an Ajax front end, to build a SOA application integrating Amazon, Yahoo Shopping, and eBay
- Contains hundreds of code samples—all tested with the GlassFish Java EE 5 reference implementation—that are downloadable from the companion Web site, <http://soabook.com>.

*Foreword*

*Preface*

*Acknowledgments*

*About the Author*

Chapter 1: Service-Oriented Architecture with Java Web Services

Chapter 2: An Overview of Java Web Services

Chapter 3: Basic SOA Using REST

Chapter 4: The Role of WSDL, SOAP, and Java/XML Mapping in SOA

Chapter 5: The JAXB 2.0 Data Binding

Chapter 6: JAX-WS–Client-Side Development

Chapter 7: JAX-WS 2.0–Server-Side Development

Chapter 8: Packaging and Deployment of SOA Components (JSR-181 and JSR-109)

Chapter 9: SOAShopper: Integrating eBay, Amazon, and Yahoo! Shopping

Chapter 10: Ajax and Java Web Services

Chapter 11: WSDL–Centric Java Web Services with SOA-J

Appendix A: Java, XML, and Web Services Standards Used in This Book

Appendix B: Software Configuration Guide

Appendix C: Namespace

Prefixes

*Glossary*

*References*

*Index*

 [Download SOA Using Java Web Services ...pdf](#)

 [Read Online SOA Using Java Web Services ...pdf](#)

# SOA Using Java Web Services

*By Mark D. Hansen*

SOA Using Java Web Services By Mark D. Hansen

## Expert Solutions and State-of-the-Art Code Examples

*SOA Using Java™ Web Services* is a hands-on guide to implementing Web services and Service Oriented Architecture (SOA) with today's Java EE 5 and Java SE 6 platforms. Author Mark Hansen presents in explicit detail the information that enterprise developers and architects need to succeed, from best-practice design techniques to state-of-the-art code samples.

Hansen covers creating, deploying, and invoking Web services that can be composed into loosely coupled SOA applications. He begins by reviewing the “big picture,” including the challenges of Java-based SOA development and the limitations of traditional approaches. Next, he systematically introduces the latest Java Web Services (JWS) APIs and walks through creating Web services that integrate into a comprehensive SOA solution. Finally, he shows how application frameworks based on JWS can streamline the entire SOA development process and introduces one such framework: SOA-J.

The book

- Introduces practical techniques for managing the complexity of Web services and SOA, including best-practice design examples
- Offers hard-won insights into building effective SOA applications with Java Web Services
- Illuminates recent major JWS improvements—including two full chapters on JAX-WS 2.0
- Thoroughly explains SOA integration using WSDL, SOAP, Java/XML mapping, and JAXB 2.0 data binding
- Walks step by step through packaging and deploying Web services components on Java EE 5 with JSR-181 (WS-Metadata 2.0) and JSR-109
- Includes specific code solutions for many development issues, from publishing REST endpoints to consuming SOAP services with WSDL
- Presents a complete case study using the JWS APIs, together with an Ajax front end, to build a SOA application integrating Amazon, Yahoo Shopping, and eBay
- Contains hundreds of code samples—all tested with the GlassFish Java EE 5 reference implementation—that are downloadable from the companion Web site, <http://soabook.com>.

*Foreword*

*Preface*

*Acknowledgments*

*About the Author*

Chapter 1: Service-Oriented Architecture with Java Web Services

Chapter 2: An Overview of Java Web Services

Chapter 3: Basic SOA Using REST

Chapter 4: The Role of WSDL, SOAP, and Java/XML Mapping in SOA

Chapter 5: The JAXB 2.0 Data Binding

Chapter 6: JAX-WS–Client-Side Development

Chapter 7: JAX-WS 2.0–Server-Side Development

Chapter 8: Packaging and Deployment of SOA Components (JSR-181 and JSR-109)

Chapter 9: SOAShopper: Integrating eBay, Amazon, and Yahoo! Shopping

Chapter 10: Ajax and Java Web Services

Chapter 11: WSDL-Centric Java Web Services with SOA-J

Appendix A: Java, XML, and Web Services Standards Used in This Book

Appendix B: Software Configuration Guide

Appendix C: Namespace

Prefixes

*Glossary*

*References*

*Index*

## **SOA Using Java Web Services By Mark D. Hansen Bibliography**

- Sales Rank: #1736665 in Books
- Published on: 2007-05-19
- Released on: 2007-05-09
- Original language: English
- Number of items: 1
- Dimensions: 9.22" h x 1.25" w x 7.00" l, 2.07 pounds
- Binding: Paperback
- 608 pages

 [Download SOA Using Java Web Services ...pdf](#)

 [Read Online SOA Using Java Web Services ...pdf](#)

## **Editorial Review**

From the Back Cover

Expert Solutions and State-of-the-Art Code Examples "SOA Using Java(TM) Web Services" is a hands-on guide to implementing Web services and Service Oriented Architecture (SOA) with today's Java EE 5 and Java SE 6 platforms. Author Mark Hansen presents in explicit detail the information that enterprise developers and architects need to succeed, from best-practice design techniques to state-of-the-art code samples. Hansen covers creating, deploying, and invoking Web services that can be composed into loosely coupled SOA applications. He begins by reviewing the "big picture," including the challenges of Java-based SOA development and the limitations of traditional approaches. Next, he systematically introduces the latest Java Web Services (JWS) APIs and walks through creating Web services that integrate into a comprehensive SOA solution. Finally, he shows how application frameworks based on JWS can streamline the entire SOA development process and introduces one such framework: SOA-J. The book introduces practical techniques for managing the complexity of Web services and SOA, including best-practice design examples. Offers hard-won insights into building effective SOA applications with Java Web Services. Illuminates recent major JWS improvements-including two full chapters on JAX-WS 2.0. Thoroughly explains SOA integration using WSDL, SOAP, Java/XML mapping, and JAXB 2.0 data binding. Walks step by step through packaging and deploying Web services components on Java EE 5 with JSR-181 (WS-Metadata 2.0) and JSR-109. Includes specific code solutions for many development issues, from publishing REST endpoints to consuming SOAP services with WSDL. Presents a complete case study using the JWS APIs, together with an Ajax front end, to build a SOA application integrating Amazon, Yahoo Shopping, and eBay. Contains hundreds of code samples-all tested with the GlassFish Java EE 5 reference implementation-that are downloadable from the companion Web site, <http://soabook.com>.

"Foreword ""

Preface "

"Acknowledgments ""

About the Author "

Chapter 1: Service-Oriented Architecture with Java Web Services

Chapter 2: An Overview of Java Web Services

Chapter 3: Basic SOA Using REST

Chapter 4: The Role of WSDL, SOAP, and Java/XML Mapping in SOA

Chapter 5: The JAXB 2.0 Data Binding

Chapter 6: JAX-WS-Client-Side Development

Chapter 7: JAX-WS 2.0-Server-Side Development

Chapter 8: Packaging and Deployment of SOA Components (JSR-181 and JSR-109)

Chapter 9: SOA Shopper: Integrating eBay, Amazon, and Yahoo! Shopping

Chapter 10: Ajax and Java Web Services

Chapter 11: WSDL-Centric Java Web Services with SOA-J

Appendix A: Java, XML, and Web Services Standards Used in This Book

Appendix B: Software Configuration Guide

Appendix C: Namespace

Prefixes "

Glossary ""

References ""

Index "

About the Author

**Mark Hansen, Ph.D.**, is a software developer, consultant, and entrepreneur. His company, Javector Software, provides consulting and software application development focused on Web services. Mark is also a content developer for Project GlassFish and has developed the open source SOA-J application framework for WSDL-centric Web services development.

Previously, Mark was a visiting scholar at MIT, researching applications for process and data integration using Web services technology. Prior to that, Mark was an executive vice president for Xpedior, Inc., a leading provider of e-business consulting services. He joined Xpedior when they acquired his consulting firm, Kinderhook Systems.

Mark founded Kinderhook in 1993 to develop custom Internet solutions for Fortune 1000 firms in the New York metropolitan area. Prior to founding Kinderhook Systems, Hansen was a founder and vice president of technology for QDB Solutions, Inc., a software firm providing tools for data integrity management in corporate data warehouses.

Mark's work has been featured in publications such as the *Wall Street Journal*, *Information Week*, *Computer World*, *Database Management*, *Database Programming and Design*, *Business Communications Review*, *EAI Journal*, and *IntelligentEnterprise*.

Mark earned a Ph.D. from the MIT Laboratory for Computer Science, a master's degree from the MIT Sloan School of Management and a bachelor's degree in mathematics from Cornell University.

Mark and his wife, Lorraine, live in Scarsdale, New York, with their three children, Elizabeth, Eric, and Emily.

Excerpt. © Reprinted by permission. All rights reserved.

Java became a powerful development platform for Service-Oriented Architecture (SOA) in 2006. Java EE 5, released in May 2006, significantly enhanced the power and usability of the Web Services capabilities on the application server. Then Java SE 6, released in December 2006, incorporated the majority of those capabilities into the standard edition of the Java programming language.

Because robust Web Services technology is the foundation for implementing SOA, Java now provides the tools modern enterprises require to integrate their Java applications into SOA infrastructures.

Of course, Java has had basic Web Services capabilities for some time. JAX-RPC 1.0 was released in June 2002. J2EE 1.4, finalized in November 2003, included JAX-RPC 1.1. So what is significant about the latest versions of the Java Web Services (JWS) APIs?

The answers are power and ease of use. Programmers will find it much easier to build enterprise-class applications with Web Services in Java EE 5 than in J2EE 1.4. Evidence of that is contained in Chapters 9 and 10, which describe an application I developed to integrate online shopping across eBay, Yahoo! Shopping, and Amazon. It's a pure Java EE 5 application, called SOAShopper, that consumes REST and SOAP services from those shopping sites. SOAShopper also provides its own SOAP and REST endpoints for cross-platform search, and supports an Ajax front-end. SOAShopper would have been a struggle to develop using J2EE 1.4 and JAX-RPC. With the new Java Web Services standards, it was a pleasure to write.

This book focuses on the following standards comprising the new Java Web Services:

- JAX-WS 2.0 JSR 224—The Java API for XML-Based Web Services. The successor to JAX-RPC, it enables you to build and consume Web services with Java.
- JAXB 2.0 JSR 222—The Java Architecture for XML Binding. Tightly integrated with JAX-WS, the JAXB standard controls how Java objects are represented as XML.
- WS-Metadata JSR 181—Web Services Metadata for the Java Platform. WS-Metadata provides annotations that facilitate the flexible definition and deployment of Java Web Services.
- WSEE 1.2 JSR 109—Web Services for Java EE. WSEE defines the programming model and run-time behavior of Web Services in the Java EE container.

These standards contain a few big improvements and many little enhancements that add up to a significantly more powerful Web Services programming platform. New annotations, for example, make it easier to write Web Services applications. And the delegation, in JAX-WS 2.0 JSR 224, of the Java/XML binding to JAXB 2.0 JSR 222 greatly improves the usability of JAX-WS as compared with JAX-RPC. The deployment model has been greatly simplified by WS-Metadata 1.0 JSR 181 and an improved 1.2 release of WSEE JSR-109.

Chapters 1 and 2 review these JWS standards in detail and describe how they improve on the previous set of JWS standards. Chapters 3 through 10 focus on writing code. To really understand the power and ease of use of the new Java Web Services, you need to start writing code. And that is primarily what this book is about. Chapters 3 through 10 are packed with code examples showing you how to best take advantage of the powerful features, avoid some of the pitfalls, and work around some of the limitations.

Chapter 11 looks to the future and offers some ideas, along with a prototype implementation, for a WSDL-centric approach to creating Web Services that might further improve JWS as a platform for Service-Oriented Architecture.

I started writing this book in 2002, when JAX-RPC first appeared on the scene. I soon ran into trouble, though, because I wanted it to be a book for programmers and I had a hard time writing good sample code with JAX-RPC. Four years later, when I started playing around with beta versions of the GlassFish Java EE 5 application server, I noticed that things had significantly improved. It was now fun to program Web Services in Java and I recommitted myself to finishing this book.

The result is a book with lots of code showing you how to deal with SOAP, WSDL, and REST from inside the Java programming language. Hopefully this code, and the writing that goes with it, will help you master Java Web Services and enable you to start using Java as a powerful platform for SOA.

## About This Book

### An Unbiased Guide to Java Web Services for SOA

My primary goal in this book is to offer an unbiased guide to using the Java Web Services (JWS) standards for SOA. Of course, any author has a bias, and I admit to believing that the JWS standards are quite good. Otherwise, I would not have written this book.

Having admitted my bias, I also freely admit that JWS has weaknesses, particularly when it comes to the development approach known as *Start from WSDL and Java*. As you will see described in many different ways in this book, the JWS standards present a Java-centric approach to Web Services. That approach can be troublesome when you need to work with established SOA standards and map your Java application to existing XML Schema documents and WSDLs.

In such situations, it's helpful to be able to take a WSDL-centric approach to Web Services development. In

this area, JWS is less strong. Throughout the book, I point out those shortcomings, and offer strategies you can use to overcome them. Chapter 11 even offers a prototype framework, called SOA-J, that illustrates an alternative, WSDL-centric approach to Java Web Services.

## **Written for Java Developers and Architects**

This is a book for people who are interested in code—primarily the developers who write systems and the architects who design them. There are a lot of coding examples you can download, install, and run.

Being a book for Java programmers working with Web Services, the discussion and examples provided within assume you have a working knowledge of Java and a basic understanding of XML and XML Schema. You don't need to know a lot about SOAP or WSDL to dive in and start learning. However, as you go along in the book, you might want to browse through an introductory tutorial on WSDL and/or XML Schema if you need to firm up your grasp on some of the Web Services basics. Throughout the book, I offer references to Web sites and other books where you can brush up on background material.

## **Knowledge of J2SE 5.0 Is Assumed**

This book assumes you have a basic understanding of J2SE 5.0—particularly the Java language extensions generics and annotations. If you are not familiar with generics or annotations, you can learn all you need to know from the free documentation and tutorials available at <http://java.sun.com>.

Don't be intimidated by these topics. Generics and annotations are not hard to master—and you need to understand them if you are going to do Web Services with Java EE 5 and Java SE 6. The reason I have not written an introduction to generics and annotations as part of this book is that there is so much good, free information available on the Web. My focus in this book is to go beyond what is freely available in the online tutorials and documentation.

## **Why GlassFish?**

All the code examples presented in this book have been developed and tested using the GlassFish GLASSFISH open source Java EE 5 reference implementation. At the time I wrote this, it was the only implementation available. Now that the book is going to press, there are more, and the code should run on all these platforms without change. The only changes that will need to be made have to do with the build process where GlassFish specific tools (e.g., the `wsimport` WSDL to Java compiler, the `asadmin` deployment utility) are used.

I plan to test the example code on other platforms as they become available and to post instructions for running them on JBoss, BEA, IBM, and so on, as these vendors support the JWS standards. Check the book's forthcoming Web site (<http://soabook.com>) for updates on progress with other platforms.

If you haven't tried GlassFish, I suggest you check it out at <https://glassfish.dev.java.net>. It supports the cutting edge in Java EE and the community is terrific. In particular, I've had good experiences getting technical support on the mailing lists. It's not uncommon to post a question there and have one of the JSR specification leads respond with an answer within minutes!

## **Why Some Topics Aren't Covered**

Both SOA and Web Services are vast topics. Even when restricting the discussion to Java technology, it is impossible to cover everything in one book. Faced with that reality, I decided to focus on what I consider to be the core issues that are important to Java developers and architects. The core issues involve creating,



deploying, and invoking Web Services in a manner that enables them to be composed into loosely coupled SOA applications.

In narrowing the book's focus, it is inevitable that I will have disappointed some readers because a particular topic of interest to them isn't covered. Some of these topics, pointed out by my reviewers, are listed here, along with the reasons why I didn't include them.

## **SOA Design Principles**

This is not a book that covers the concepts and design philosophy behind SOA. It is a how-to book that teaches Java developers to code SOA components using Java Web Services. For a thorough introduction to SOA concepts and design, I recommend Thomas Erl's *Service-Oriented Architecture* Erl.

## **UDDI**

UDDI is very important. And Java EE 5 includes the JAX-R standard interface to UDDI repositories. But JAX-R hasn't changed since J2EE 1.4. And it is covered well in many other books and online tutorials. So, in an effort to keep this book to a manageable size, I have left it out.

## **Enterprise Messaging**

I wish I could have included a chapter on Enterprise Messaging. After all, it is a cornerstone of SOA. However, this book restricts itself to the capabilities provided by JWS. JWS does not support WS-Reliable Messaging WS-RM or any other SOAP/WSDL-oriented reliable messaging mechanism. Of course, Java EE 5 includes support for the Java Message Service API (JMS). And JMS is a useful tool for implementing SOA applications. But by itself, JMS isn't a Web Services tool. So, in the interest of focus, I have left it out.

## **WS-Addressing, WS-Security, and the Many Other WS-\* Standards**

Explaining the myriad standards for the Web Services stack would require many thousands of pages. Since these WS-\* standards are not yet part of JWS, I have not covered them. In addition, my sense is that most Java developers are still mastering SOAP over HTTP. The need for a programmer's guide to WS-\* is probably several years away.

## **Fonts and Special Characters**

Courier font is used for Java types, XML Schema components, and all code examples included in the text. Courier font is also used to signify software-environment-specific items (e.g., paths, directories, environment variables) and text interactions with the computer.

Italics indicate that a term is defined in the glossary. I don't always use italics—only when a term may not have been used before and I think the reader might want to know it is defined in the Glossary. Italics are also used for emphasis.

< > indicates an environment-specific directory location.

## **Code Fragments in Text**

The text contains lots of code fragments to illustrate the discussion. At the bottom of each code fragment is the file path showing where it came from. Moreover, there are line numbers to show you the line number in the file where the code comes from.

## **Users Review**

### **From reader reviews:**

#### **Kyle Guthrie:**

In this 21st millennium, people become competitive in each and every way. By being competitive today, people have to do something to make these people survive, being in the middle of the actual crowded place and notice by surrounding. One thing that at times many people have underestimated the idea for a while is reading. Yeah, by reading a publication your ability to survive enhance then having chance to remain than other is high. For you personally who want to start reading a book, we give you this specific SOA Using Java Web Services book as basic and daily reading reserve. Why, because this book is usually more than just a book.

#### **Martin Williams:**

This book untitled SOA Using Java Web Services to be one of several books which best seller in this year, here is because when you read this reserve you can get a lot of benefit in it. You will easily to buy this kind of book in the book store or you can order it through online. The publisher with this book sells the e-book too. It makes you quickly to read this book, as you can read this book in your Mobile phone. So there is no reason for your requirements to past this guide from your list.

#### **Peter Christensen:**

Reading a book can be one of a lot of pastime that everyone in the world loves. Do you like reading book consequently. There are a lot of reasons why people fantastic. First reading a book will give you a lot of new facts. When you read a publication you will get new information since book is one of various ways to share the information or even their idea. Second, reading through a book will make an individual more imaginative. When you reading through a book especially fictional book the author will bring you to imagine the story how the figures do it anything. Third, you are able to share your knowledge to other individuals. When you read this SOA Using Java Web Services, you are able to tells your family, friends and also soon about yours publication. Your knowledge can inspire different ones, make them reading a guide.

#### **Michael Clements:**

The publication untitled SOA Using Java Web Services is the reserve that recommended to you to learn. You can see the quality of the book content that will be shown to anyone. The language that publisher use to explained their ideas are easily to understand. The writer was did a lot of investigation when write the book, and so the information that they share to you personally is absolutely accurate. You also will get the e-book of SOA Using Java Web Services from the publisher to make you considerably more enjoy free time.

**Download and Read Online SOA Using Java Web Services By Mark  
D. Hansen #QANRV5G1WT3**

## **Read SOA Using Java Web Services By Mark D. Hansen for online ebook**

SOA Using Java Web Services By Mark D. Hansen Free PDF d0wnl0ad, audio books, books to read, good books to read, cheap books, good books, online books, books online, book reviews epub, read books online, books to read online, online library, greatbooks to read, PDF best books to read, top books to read SOA Using Java Web Services By Mark D. Hansen books to read online.

### **Online SOA Using Java Web Services By Mark D. Hansen ebook PDF download**

**SOA Using Java Web Services By Mark D. Hansen Doc**

**SOA Using Java Web Services By Mark D. Hansen Mobipocket**

**SOA Using Java Web Services By Mark D. Hansen EPub**