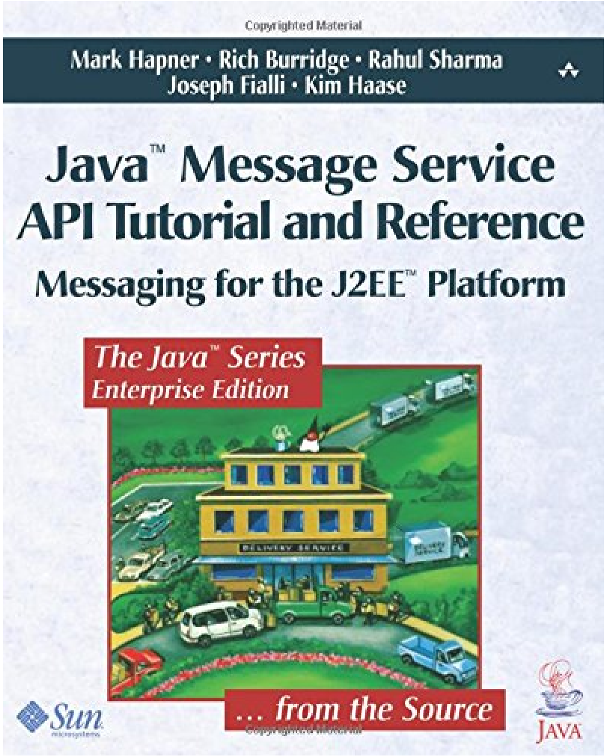
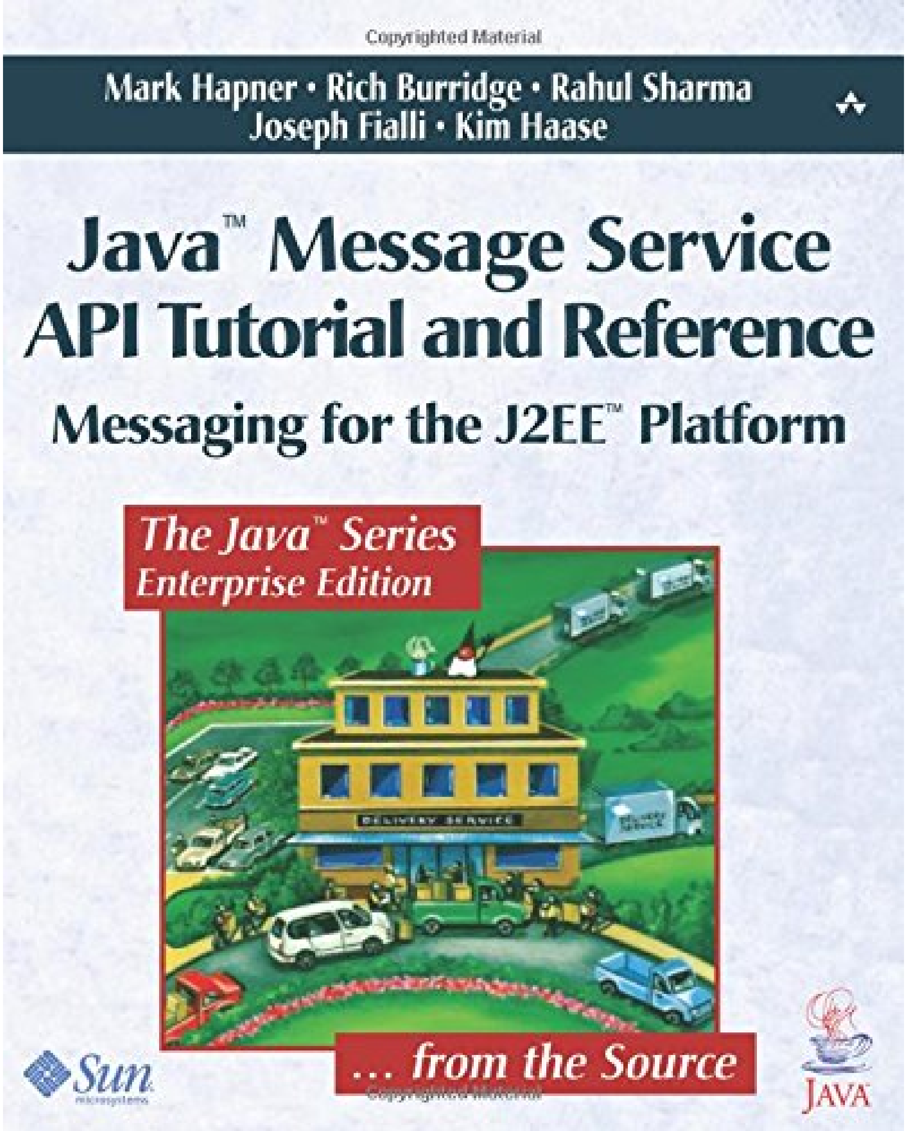
JAVA¿ MESSAGE SERVICE API TUTORIAL AND REFERENCE: MESSAGING FOR THE J2EE¿ PLATFORM BY MARK HAPNER, RICH BURRIDGE, RAHUL SHA



DOWNLOAD EBOOK : JAVA¿ MESSAGE SERVICE API TUTORIAL AND REFERENCE: MESSAGING FOR THE J2EE¿ PLATFORM BY MARK HAPNER, RICH BURRIDGE, RAHUL SHA PDF



Click link bellow and free register to download ebook:

JAVA¿ MESSAGE SERVICE API TUTORIAL AND REFERENCE: MESSAGING FOR THE J2EE¿ PLATFORM BY MARK HAPNER, RICH BURRIDGE, RAHUL SHA

DOWNLOAD FROM OUR ONLINE LIBRARY

JAVA¿ MESSAGE SERVICE API TUTORIAL AND REFERENCE: MESSAGING FOR THE J2EE¿ PLATFORM BY MARK HAPNER, RICH BURRIDGE, RAHUL SHA PDF

When going to take the experience or thoughts kinds others, book *Java¿ Message Service API Tutorial And Reference: Messaging For The J2EE¿ Platform By Mark Hapner, Rich Burridge, Rahul Sha* can be a great source. It holds true. You could read this Java¿ Message Service API Tutorial And Reference: Messaging For The J2EE¿ Platform By Mark Hapner, Rich Burridge, Rahul Sha as the source that can be downloaded and install right here. The method to download and install is also simple. You could see the web link web page that we provide and afterwards buy guide to make a bargain. Download Java¿ Message Service API Tutorial And Reference: Messaging For The J2EE¿ Platform By Mark Hapner, Rich Burridge, Rahul Sha as well as you could deposit in your personal device.

Amazon.com Review
Aimed at the more experienced Java developer who needs to work with enterprise messaging, Java Message Service API Tutorial and Reference delivers starter code and a complete reference to all JMS classes that you will need to know to work effectively with this powerful feature of the J2EE platform.

The no-nonsense, just-the-facts approach of this dual tutorial/reference is perhaps its salient feature. It explains the basics of asynchronous messaging and its advantages for robust enterprise-level applications before digging right in to JMS. The authors do a good job at explaining the difference between point-to-point and publish/subscribe models of message delivery. They also give a laundry list of areas to look at to ensure reliability and robustness in JMS systems, including looking at dos and don'ts for acknowledgement, message priority, and durability.

The real focus early in the book is on the simple, but effective, sample code used to illustrate the basic APIs with short, complete examples. Though somewhat demanding, this code will be for the more experienced reader all you need to get going with JMS. There are examples of both point-to-point and publish-subscribe APIs in action. Importantly, because working with JMS can be tricky, the authors don't skimp on the practical details of compiling, deploying, and running each application. Other examples look at JMS used with Enterprise JavaBeans, including the new EJB 2.0 message bean, plus how to use JMS correctly with session and entity beans. (This can also be tricky, and the authors go through the steps of packaging up and deploying bean JAR files, as well.)

More than half of this book is an alphabetical listing of the 46 classes available in JMS. Each class is explained, along with options and tips for using the class where appropriate. There is full coverage of basic message and exception classes and the classes you need to use for basic point-to-point and publish/subscribe processing. Obviously, this material is a must for those who prefer a printed reference instead of online help. With a quick-start tutorial and concise (yet complete) reference to all JMS classes, this title will serve a useful function for the working enterprise Java developer. --Richard Dragan

Topics covered: Introduction to messaging, the Java Message Service (JMS) API architecture, point-to-point and publish/subscribe messaging domains described, essential JMS programming APIs (connections, sessions, message producers and consumers), the structure of messages, simple point-to-point and publish/subscribe examples (including guide to deployment and execution), JMS used with multiple systems (including J2EE and non-J2EE messaging interoperability), JMS reliability mechanisms (acknowledgement, persistence, priority levels, durable subscriptions, local transactions), building J2EE clients used with message, session, and entity beans (including deployment and execution tips), a sample using multiple J2EE servers, reference to all JMS APIs: including topic and message classes, exception classes, queue classes, sessions and connections, and appendix with JMS client examples.

From the Back Cover

Java™ Message Service API Tutorial and Reference provides a clear and complete introduction to the Java™ Message Service (JMS) API. This book illustrates how to use the JMS API to build applications that create, send, receive, and read messages. Beginning with comprehensive descriptions of fundamental JMS concepts and building blocks, the coverage extends to all aspects of running and creating JMS applications. Each topic area is supported by relevant and well-crafted JMS program examples that demonstrate how to put the technology to work.

The JMS API is an integral part of the Java 2 Platform, Enterprise Edition (J2EE™ platform). Developed by Sun Microsystems in close cooperation with enterprise messaging partners, JMS works together with other technologies to provide reliable, asynchronous communication between components in a distributed computing environment. It delivers a new, powerful tool for enterprise messaging--the cornerstone of today's enterprise applications.

You will find in-depth coverage on how to:

- Create and run JMS client programs
- Use the JMS API within the J2EE platform
- Consume messages asynchronously with a message-driven bean
- Produce messages from an application client and from a session bean
- Access an entity bean from a message-driven bean
- Create applications for the J2EE platform that use the JMS API to Consume messages Produce messages Access an entity bean

From BytesMessage to TransactionRolledBackException, a useful alphabetical reference provides complete information on all facets of the JMS API.

Additionally, the tutorial example programs are downloadable from the Sun Web site, so that you can adapt them to implementations of the JMS API and the J2EE platform.

Written by an expert team, the book offers an unparalleled technical understanding of JMS and its integration into the J2EE platform. Its thorough and practical coverage of JMS makes it easy for developers working in a distributed Java technology environment, and those familiar with the J2EE platform, to efficiently integrate the JMS API.

0201784726B02212002

About the Author

Mark Hapner is Lead Architect for the Java 2 Platform, Enterprise Edition. He participated in the development of the JDBC API, wrote the Java Message Service specification, and co-authored the Enterprise JavaBeans specification.

Rich Burridge is a Staff Engineer at Sun Microsystems, where he has worked for over fifteen years. Currently with the Accessibility group, he is the creator of the Java Shared Data Toolkit (JSDT) and a coauthor of the Java Message Service API specification. Previously he worked on the "Netscape 6 for Solaris," OpenStep, and OpenWindows products.

Rahul Sharma is the lead architect of the J2EE Connector architecture and a Senior Staff Engineer at the Java Software division of Sun Microsystems, Inc. Presently, he is the lead architect of the JAX-RPC (Java APIs for XML based RPC) 1.0. Rahul has been with Sun for the last five years. Rahul holds a computer engineering degree from the Delhi University, India, and an MBA from the Haas School of Business, UC Berkeley.

Joseph Fialli is a Senior Staff Engineer at Sun Microsystems, where he has worked for the past four years. Currently, he is the lead architect for the Java Architecture for XML Binding (JAXB). Previously, he was the technical lead for the Java Message Service API within the J2EE Software Development Kit, version1.3, and maintained Java serialization within the Java 2 Platform, Standard Edition, version 1.2.

Kim Haase is a staff writer with Sun Microsystems, where she documents the J2EE platform. In previous positions she has documented compilers, debuggers, and floating-point programming. She currently writes about the Java Message Service and J2EE SDK tools.

JAVA¿ MESSAGE SERVICE API TUTORIAL AND REFERENCE: MESSAGING FOR THE J2EE¿ PLATFORM BY MARK HAPNER, RICH BURRIDGE, RAHUL SHA PDF

Java¿ Message Service API Tutorial And Reference: Messaging For The J2EE¿ Platform By Mark Hapner, Rich Burridge, Rahul Sha. In undergoing this life, numerous people consistently attempt to do and also obtain the very best. New knowledge, experience, driving lesson, and also every little thing that can enhance the life will be done. Nonetheless, many individuals in some cases feel puzzled to obtain those things. Really feeling the minimal of encounter and also sources to be far better is among the does not have to possess. Nevertheless, there is a quite straightforward thing that could be done. This is exactly what your teacher consistently manoeuvres you to do this one. Yeah, reading is the response. Checking out an e-book as this Java¿ Message Service API Tutorial And Reference: Messaging For The J2EE¿ Platform By Mark Hapner, Rich Burridge, Rahul Sha as well as various other references can enrich your life top quality. Just how can it be?

Reviewing, as soon as more, will offer you something new. Something that you do not know then exposed to be populared with the e-book *Java¿ Message Service API Tutorial And Reference: Messaging For The J2EE¿ Platform By Mark Hapner, Rich Burridge, Rahul Sha* message. Some knowledge or session that re got from reading e-books is vast. A lot more e-books Java¿ Message Service API Tutorial And Reference: Messaging For The J2EE¿ Platform By Mark Hapner, Rich Burridge, Rahul Sha you read, more expertise you get, as well as more chances to consistently enjoy reviewing e-books. Due to this factor, reading book ought to be begun from earlier. It is as exactly what you could obtain from the book Java¿ Message Service API Tutorial And Reference: Messaging For The J2EE¿ Platform By Mark Hapner, Rich Burridge, Rahul Sha

Obtain the perks of reading behavior for your lifestyle. Reserve Java¿ Message Service API Tutorial And Reference: Messaging For The J2EE¿ Platform By Mark Hapner, Rich Burridge, Rahul Sha notification will constantly relate to the life. The actual life, expertise, scientific research, health, religious beliefs, entertainment, as well as much more could be found in created publications. Numerous writers supply their experience, science, research, as well as all points to show you. Among them is through this Java¿ Message Service API Tutorial And Reference: Messaging For The J2EE¿ Platform By Mark Hapner, Rich Burridge, Rahul Sha This e-book Java¿ Message Service API Tutorial And Reference: Messaging For The J2EE¿ Platform By Mark Hapner, Rich Burridge, Rahul Sha will offer the needed of notification and declaration of the life. Life will be completed if you recognize a lot more points with reading e-books.

JAVA¿ MESSAGE SERVICE API TUTORIAL AND REFERENCE: MESSAGING FOR THE J2EE¿ PLATFORM BY MARK HAPNER, RICH BURRIDGE, RAHUL SHA PDF

Java Message Service (JMS) represents a powerful solution for communicating between Java enterprise applications, software components, and legacy systems. In this authoritative tutorial and comprehensive reference, Sun's Java Message Service architects offer start-to-finish coverage of peer-to-peer JMS development with Java 2 Platform, Enterprise Edition, Release 1.3. JMS is now fully integrated into the J2EE platform -- and this is the first book to show how to make the most of JMS in the context of sophisticated J2EE application development. The authors begin by introducing the JMS API to developers who are new to it. Then, with the help of extensive programming examples, they demonstrate key JMS techniques for enabling applications to create, send, receive, and read messages, and for integrating with existing back office and enterprise systems. Coverage includes: consuming messages asynchronously with message-driven beans; producing messages from application clients; accessing entity beans from message-driven bean; producing messages from session beans; and much more. For all Java developers building applications that must communicate and share information.

- Sales Rank: #2317806 in Books
- Published on: 2002-03-08
- Released on: 2002-02-26
- Original language: English
- Number of items: 1
- Dimensions: 9.20" h x 1.20" w x 7.40" l, 2.10 pounds
- Binding: Paperback
- 544 pages

Amazon.com Review
Aimed at the more experienced Java developer who needs to work with enterprise messaging, Java Message Service API Tutorial and Reference delivers starter code and a complete reference to all JMS classes that you will need to know to work effectively with this powerful feature of the J2EE platform.

The no-nonsense, just-the-facts approach of this dual tutorial/reference is perhaps its salient feature. It explains the basics of asynchronous messaging and its advantages for robust enterprise-level applications before digging right in to JMS. The authors do a good job at explaining the difference between point-to-point and publish/subscribe models of message delivery. They also give a laundry list of areas to look at to ensure reliability and robustness in JMS systems, including looking at dos and don'ts for acknowledgement, message priority, and durability.

The real focus early in the book is on the simple, but effective, sample code used to illustrate the basic APIs with short, complete examples. Though somewhat demanding, this code will be for the more experienced reader all you need to get going with JMS. There are examples of both point-to-point and publish-subscribe APIs in action. Importantly, because working with JMS can be tricky, the authors don't skimp on the practical details of compiling, deploying, and running each application. Other examples look at JMS used with Enterprise JavaBeans, including the new EJB 2.0 message bean, plus how to use JMS correctly with session and entity beans. (This can also be tricky, and the authors go through the steps of packaging up and deploying bean JAR files, as well.)

More than half of this book is an alphabetical listing of the 46 classes available in JMS. Each class is explained, along with options and tips for using the class where appropriate. There is full coverage of basic message and exception classes and the classes you need to use for basic point-to-point and publish/subscribe processing. Obviously, this material is a must for those who prefer a printed reference instead of online help. With a quick-start tutorial and concise (yet complete) reference to all JMS classes, this title will serve a useful function for the working enterprise Java developer. --Richard Dragan

Topics covered: Introduction to messaging, the Java Message Service (JMS) API architecture, point-to-point and publish/subscribe messaging domains described, essential JMS programming APIs (connections, sessions, message producers and consumers), the structure of messages, simple point-to-point and publish/subscribe examples (including guide to deployment and execution), JMS used with multiple systems (including J2EE and non-J2EE messaging interoperability), JMS reliability mechanisms (acknowledgement, persistence, priority levels, durable subscriptions, local transactions), building J2EE clients used with message, session, and entity beans (including deployment and execution tips), a sample using multiple J2EE servers, reference to all JMS APIs: including topic and message classes, exception classes, queue classes, sessions and connections, and appendix with JMS client examples.

From the Back Cover

Java™ Message Service API Tutorial and Reference provides a clear and complete introduction to the Java™ Message Service (JMS) API. This book illustrates how to use the JMS API to build applications that create, send, receive, and read messages. Beginning with comprehensive descriptions of fundamental JMS concepts and building blocks, the coverage extends to all aspects of running and creating JMS applications. Each topic area is supported by relevant and well-crafted JMS program examples that demonstrate how to put the technology to work.

The JMS API is an integral part of the Java 2 Platform, Enterprise Edition (J2EE™ platform). Developed by Sun Microsystems in close cooperation with enterprise messaging partners, JMS works together with other technologies to provide reliable, asynchronous communication between components in a distributed computing environment. It delivers a new, powerful tool for enterprise messaging--the cornerstone of today's enterprise applications.

You will find in-depth coverage on how to:

- Create and run JMS client programs
- Use the JMS API within the J2EE platform
- Consume messages asynchronously with a message-driven bean
- Produce messages from an application client and from a session bean
- Access an entity bean from a message-driven bean
- Create applications for the J2EE platform that use the JMS API to Consume messages Produce messages Access an entity bean

From BytesMessage to TransactionRolledBackException, a useful alphabetical reference provides complete information on all facets of the JMS API.

Additionally, the tutorial example programs are downloadable from the Sun Web site, so that you can adapt them to implementations of the JMS API and the J2EE platform.

Written by an expert team, the book offers an unparalleled technical understanding of JMS and its integration into the J2EE platform. Its thorough and practical coverage of JMS makes it easy for developers working in a distributed Java technology environment, and those familiar with the J2EE platform, to efficiently integrate the JMS API.

0201784726B02212002

About the Author

Mark Hapner is Lead Architect for the Java 2 Platform, Enterprise Edition. He participated in the development of the JDBC API, wrote the Java Message Service specification, and co-authored the Enterprise JavaBeans specification.

Rich Burridge is a Staff Engineer at Sun Microsystems, where he has worked for over fifteen years. Currently with the Accessibility group, he is the creator of the Java Shared Data Toolkit (JSDT) and a coauthor of the Java Message Service API specification. Previously he worked on the "Netscape 6 for Solaris," OpenStep, and OpenWindows products.

Rahul Sharma is the lead architect of the J2EE Connector architecture and a Senior Staff Engineer at the Java Software division of Sun Microsystems, Inc. Presently, he is the lead architect of the JAX-RPC (Java APIs for XML based RPC) 1.0. Rahul has been with Sun for the last five years. Rahul holds a computer engineering degree from the Delhi University, India, and an MBA from the Haas School of Business, UC Berkeley.

Joseph Fialli is a Senior Staff Engineer at Sun Microsystems, where he has worked for the past four years. Currently, he is the lead architect for the Java Architecture for XML Binding (JAXB). Previously, he was the technical lead for the Java Message Service API within the J2EE Software Development Kit, version1.3, and maintained Java serialization within the Java 2 Platform, Standard Edition, version 1.2.

Kim Haase is a staff writer with Sun Microsystems, where she documents the J2EE platform. In previous positions she has documented compilers, debuggers, and floating-point programming. She currently writes about the Java Message Service and J2EE SDK tools.

Most helpful customer reviews

12 of 12 people found the following review helpful.
Network Computing Using JMS
By W Boudville
A lucid and authoritative description of Java Message Service, from Sun Microsystems, which developed and owns it. JMS is designed for an environment of distributed computers, where applications need to communicate with each other and databases across the network. You can think of JMS as one of the enablers

of Sun's longtime slogan "The Network IS The Computer".

JMS is loosely coupled distributed networking, where the sender and receiver do not have to be running at the same time. Plus they do not need to know each other's methods, quite unlike RMI [a tightly coupled technology]. This makes for potentially much greater flexibility in network computing.

The book emphasises this, with detailed examples of source code showing how to use JMS with Enterprise Java Beans, another technology invented by Sun. You can see how to hook JMS to a session bean or an entity bean, and how to combine JMS with several Message Driven Beans. The text is clearly written, with attention paid to how you can run the examples under Microsoft Windows or Unix.

The book also suggests two sequels. It describes using JMS with J2EE, the Java Enterprise Edition, which is the full Java environment. But in a world of PDAs, cell phones and other mobile gadgets, what would be interesting is a description of JMS running under a slimmed down Java environment, like kvm, and how this would scale with the number of devices. A second sequel might be a comparison of JMS with JXTA, another Sun technology for mobile computing. Who know? Perhaps Sun is already working on this!

If you are programming in a distributed computing environment, consider using JMS as an enabling technology, and this book as its indispensible guide.

3 of 4 people found the following review helpful.

Excellent Jump Starter

By Ray Ye

With the increase of the popularity of loosely-coupled systems in enterprise integration and various serviced-oriented application architectures, messaging-based standards and strategies is geting more and more widely utilized. JMS is a JAVA based message framework (and standard), it allows application components based on the Java 2 Platform, Enterprise Edition (J2EE) to create, send, receive, and read messages. It enables distributed communication that is loosely coupled, reliable, and asynchronous.

The book provides all the bits to get you started quickly as well as provides fair details about the architecture of JMS and its API programming model. Various sample applications and code snippets were supplied for better understanding the technology.

It is a really good and concise tutorial on the topic

0 of 11 people found the following review helpful.

One of the good books I dearsay !!!

By A Customer

Covers reasonable amount of JMS. As an SCJA I recommand this book, since you don't have much choices this should do !!

See all 3 customer reviews...

JAVA¿ MESSAGE SERVICE API TUTORIAL AND REFERENCE: MESSAGING FOR THE J2EE¿ PLATFORM BY MARK HAPNER, RICH BURRIDGE, RAHUL SHA PDF

From the explanation over, it is clear that you have to read this book Java¿ Message Service API Tutorial And Reference: Messaging For The J2EE¿ Platform By Mark Hapner, Rich Burridge, Rahul Sha We supply the online publication qualified Java¿ Message Service API Tutorial And Reference: Messaging For The J2EE¿ Platform By Mark Hapner, Rich Burridge, Rahul Sha right here by clicking the link download. From shared book by on-line, you could offer a lot more perks for lots of people. Besides, the readers will certainly be additionally effortlessly to obtain the favourite e-book Java¿ Message Service API Tutorial And Reference: Messaging For The J2EE¿ Platform By Mark Hapner, Rich Burridge, Rahul Sha to review. Discover one of the most favourite as well as needed e-book **Java¿ Message Service API Tutorial And Reference: Messaging For The J2EE¿ Platform By Mark Hapner, Rich Burridge, Rahul Sha** to check out now and also below.

Amazon.com Review
Aimed at the more experienced Java developer who needs to work with enterprise messaging, Java Message Service API Tutorial and Reference delivers starter code and a complete reference to all JMS classes that you will need to know to work effectively with this powerful feature of the J2EE platform.

The no-nonsense, just-the-facts approach of this dual tutorial/reference is perhaps its salient feature. It explains the basics of asynchronous messaging and its advantages for robust enterprise-level applications before digging right in to JMS. The authors do a good job at explaining the difference between point-to-point and publish/subscribe models of message delivery. They also give a laundry list of areas to look at to ensure reliability and robustness in JMS systems, including looking at dos and don'ts for acknowledgement, message priority, and durability.

The real focus early in the book is on the simple, but effective, sample code used to illustrate the basic APIs with short, complete examples. Though somewhat demanding, this code will be for the more experienced reader all you need to get going with JMS. There are examples of both point-to-point and publish-subscribe APIs in action. Importantly, because working with JMS can be tricky, the authors don't skimp on the practical details of compiling, deploying, and running each application. Other examples look at JMS used with Enterprise JavaBeans, including the new EJB 2.0 message bean, plus how to use JMS correctly with session and entity beans. (This can also be tricky, and the authors go through the steps of packaging up and deploying bean JAR files, as well.)

More than half of this book is an alphabetical listing of the 46 classes available in JMS. Each class is explained, along with options and tips for using the class where appropriate. There is full coverage of basic message and exception classes and the classes you need to use for basic point-to-point and publish/subscribe processing. Obviously, this material is a must for those who prefer a printed reference instead of online help. With a quick-start tutorial and concise (yet complete) reference to all JMS classes, this title will serve a useful function for the working enterprise Java developer. --Richard Dragan

Topics covered: Introduction to messaging, the Java Message Service (JMS) API architecture, point-to-point and publish/subscribe messaging domains described, essential JMS programming APIs (connections, sessions, message producers and consumers), the structure of messages, simple point-to-point and publish/subscribe examples (including guide to deployment and execution), JMS used with multiple systems (including J2EE and non-J2EE messaging interoperability), JMS reliability mechanisms (acknowledgement, persistence, priority levels, durable subscriptions, local transactions), building J2EE clients used with message, session, and entity beans (including deployment and execution tips), a sample using multiple J2EE servers, reference to all JMS APIs: including topic and message classes, exception classes, queue classes, sessions and connections, and appendix with JMS client examples.

From the Back Cover

Java™ Message Service API Tutorial and Reference provides a clear and complete introduction to the Java™ Message Service (JMS) API. This book illustrates how to use the JMS API to build applications that create, send, receive, and read messages. Beginning with comprehensive descriptions of fundamental JMS concepts and building blocks, the coverage extends to all aspects of running and creating JMS applications. Each topic area is supported by relevant and well-crafted JMS program examples that demonstrate how to put the technology to work.

The JMS API is an integral part of the Java 2 Platform, Enterprise Edition (J2EE™ platform). Developed by Sun Microsystems in close cooperation with enterprise messaging partners, JMS works together with other technologies to provide reliable, asynchronous communication between components in a distributed computing environment. It delivers a new, powerful tool for enterprise messaging--the cornerstone of today's enterprise applications.

You will find in-depth coverage on how to:

- Create and run JMS client programs
- Use the JMS API within the J2EE platform
- Consume messages asynchronously with a message-driven bean
- Produce messages from an application client and from a session bean
- Access an entity bean from a message-driven bean
- Create applications for the J2EE platform that use the JMS API to Consume messages Produce messages Access an entity bean

From BytesMessage to TransactionRolledBackException, a useful alphabetical reference provides complete information on all facets of the JMS API.

Additionally, the tutorial example programs are downloadable from the Sun Web site, so that you can adapt them to implementations of the JMS API and the J2EE platform.

Written by an expert team, the book offers an unparalleled technical understanding of JMS and its integration into the J2EE platform. Its thorough and practical coverage of JMS makes it easy for developers working in a distributed Java technology environment, and those familiar with the J2EE platform, to efficiently integrate the JMS API.

0201784726B02212002

About the Author

Mark Hapner is Lead Architect for the Java 2 Platform, Enterprise Edition. He participated in the development of the JDBC API, wrote the Java Message Service specification, and co-authored the Enterprise JavaBeans specification.

Rich Burridge is a Staff Engineer at Sun Microsystems, where he has worked for over fifteen years. Currently with the Accessibility group, he is the creator of the Java Shared Data Toolkit (JSDT) and a coauthor of the Java Message Service API specification. Previously he worked on the "Netscape 6 for Solaris," OpenStep, and OpenWindows products.

Rahul Sharma is the lead architect of the J2EE Connector architecture and a Senior Staff Engineer at the Java Software division of Sun Microsystems, Inc. Presently, he is the lead architect of the JAX-RPC (Java APIs for XML based RPC) 1.0. Rahul has been with Sun for the last five years. Rahul holds a computer engineering degree from the Delhi University, India, and an MBA from the Haas School of Business, UC Berkeley.

Joseph Fialli is a Senior Staff Engineer at Sun Microsystems, where he has worked for the past four years. Currently, he is the lead architect for the Java Architecture for XML Binding (JAXB). Previously, he was the technical lead for the Java Message Service API within the J2EE Software Development Kit, version1.3, and maintained Java serialization within the Java 2 Platform, Standard Edition, version 1.2.

Kim Haase is a staff writer with Sun Microsystems, where she documents the J2EE platform. In previous positions she has documented compilers, debuggers, and floating-point programming. She currently writes about the Java Message Service and J2EE SDK tools.

When going to take the experience or thoughts kinds others, book *Java¿ Message Service API Tutorial And Reference: Messaging For The J2EE¿ Platform By Mark Hapner, Rich Burridge, Rahul Sha* can be a great source. It holds true. You could read this Java¿ Message Service API Tutorial And Reference: Messaging For The J2EE¿ Platform By Mark Hapner, Rich Burridge, Rahul Sha as the source that can be downloaded and install right here. The method to download and install is also simple. You could see the web link web page that we provide and afterwards buy guide to make a bargain. Download Java¿ Message Service API Tutorial And Reference: Messaging For The J2EE¿ Platform By Mark Hapner, Rich Burridge, Rahul Sha as well as you could deposit in your personal device.