1.

Record Nr.	UNINA9910813187003321
Titolo	Self-service applications using IBM WebSphere v5.0 and WebSphere MQSeries Integrator / / [Peter Kovari ... et al.]
Pubbl/distr/stampa	Raleigh, NC, : IBM, International Technical Support Organization, c2003
Edizione	[1st ed.]
Descrizione fisica	xvi, 298 p. : ill
Collana	IBM redbooks
Altri autori (Persone)	KovariPeter
Soggetti	Software patterns Electronic commerce - Computer programs Application software - Development
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	"This edition applies to WebSphere Application Server v5, WebSphere MQ v5.3, WebSphere MQ Integrator v2.1 for use with the Windows 2000 server, AIX 5L, and Red Hat Linux 7.2"--Verso t.p. Rev. ed. of: Self-Service Applications using IBM WebSphere V4.0 and IBM MQSeries Integrator / [Carla Sadtler, Barry Procopio, Robert Rehms]. 2002. "July 2003." "SG24-6875-00."
Nota di bibliografia	Includes bibliographical references (p. 289-291) and index.
Nota di contenuto	Front cover -- Contents -- Notices -- Trademarks -- Preface -- The team that wrote this redbook -- Become a published author -- Comments welcome -- Part 1 Patterns for e-business -- Chapter 1. Patterns for e-business -- 1.1 The Patterns for e-business layered asset model -- 1.2 How to use the Patterns for e-business -- 1.2.1 Selecting a Business, Integration, or Composite pattern, or a Custom design -- 1.2.2 Selecting Application patterns -- 1.2.3 Review Runtime patterns -- 1.2.4 Review Product mappings -- 1.2.5 Review guidelines and related links -- 1.3 Summary -- Chapter 2. The Self-Service business pattern -- 2.1 Self-Service applications -- 2.2 Self-Service application patterns -- 2.3 Application patterns used in this book -- 2.3.1 Router pattern -- 2.3.2 Decomposition pattern -- Chapter 3. Runtime patterns -- 3.1 Nodes -- 3.2 Basic Runtime pattern for the Router pattern -- 3.2.1 Variation 1 -- 3.3 Basic Runtime pattern for Decomposition -- 3.3.1 Variation 1 -- 3.4 For more information --

Chapter 4. Product mapping -- 4.1 Runtime product mappings -- 4.2 Product summary -- Part 2 Guidelines -- Chapter 5. Technology options -- 5.1 Web client -- 5.1.1 Web browser -- 5.1.2 HTML -- 5.1.3 Dynamic HTML -- 5.1.4 CSS -- 5.1.5 JavaScript -- 5.1.6 Java applets -- 5.1.7 XML (client side) -- 5.1.8 XHTML 1.1 (HTML 4.01) -- 5.1.9 VoiceXML -- 5.1.10 XForms -- 5.1.11 XSLT -- 5.1.12 Mobile clients -- 5.2 Web application server -- 5.2.1 Java servlets -- 5.2.2 JavaServer Pages (JSPs) -- 5.2.3 JavaBeans -- 5.2.4 XML -- 5.2.5 Enterprise JavaBeans -- 5.2.6 Additional enterprise Java APIs -- 5.3 Integration technologies -- 5.3.1 Web services -- 5.3.2 J2EE Connector Architecture -- 5.3.3 Java Message Service -- 5.3.4 Message Oriented Middleware -- 5.3.5 Others -- 5.4 Where to find more information -- Chapter 6. Application design -- 6.1 Application structure.

6.1.1 Model-View-Controller design pattern -- 6.1.2 Struts -- 6.1.3 Sample application -- 6.2 EJB design guidelines -- 6.2.1 Local and remote home interfaces -- 6.2.2 Using the Singleton pattern -- 6.2.3 The Facade pattern -- 6.3 JMS design guidelines -- 6.3.1 Message models -- 6.3.2 JMS point-to-point model -- 6.3.3 JMS publish/subscribe model -- 6.3.4 JMS messages -- 6.3.5 Synchronous versus asynchronous design considerations -- 6.3.6 Where to implement message producers and consumers -- 6.3.7 Message-driven beans -- 6.3.8 Managing JMS objects -- 6.3.9 JMS and JNDI -- 6.3.10 Embedded JMS Provider versus WebSphere MQ -- 6.3.11 WebSphere to MQ connection options -- 6.3.12 Best practices for JMS and IBM WebSphere MQ -- Chapter 7. Application development -- 7.1 MVC development using the Struts framework -- 7.1.1 Creating a Web diagram -- 7.1.2 Coding Struts elements -- 7.2 Developing a message-driven bean with WebSphere Studio -- 7.2.1 Message-driven bean implementation -- 7.2.2 Life cycle of a message-driven bean -- 7.2.3 Creating an MDB using WebSphere Studio -- 7.2.4 Coding the message-driven bean -- 7.3 XML and XSLT development -- 7.3.1 XML as data transfer technology -- 7.3.2 Guidelines for creating an XML message -- 7.3.3 Performing XML transformations -- 7.3.4 Working with XSLTC -- 7.3.5 WebSphere Studio XML support -- 7.3.6 Using XML JavaBeans -- Chapter 8. Developing WebSphere MQ Integrator message flows -- 8.1 What is a broker domain? -- 8.2 Developing message flows -- 8.2.1 Preparations: creating queue managers and defining queues -- 8.2.2 Using the Control Center -- 8.2.3 Creating message flows -- Chapter 9. Security -- 9.1 End-to-end security -- 9.2 Applying security to our Runtime patterns -- 9.3 Security guidelines -- 9.4 Application security -- 9.5 Messaging security -- 9.5.1 Securing WebSphere MQ resources.

9.5.2 Securing WebSphere MQ Integrator resources -- 9.6 Security design principles summary -- Chapter 10. Performance and availability -- 10.1 Introduction -- 10.2 Performance analysis -- 10.3 Performance considerations in messaging -- 10.3.1 Connection pooling -- 10.3.2 Multithreaded programs -- 10.3.3 Persistent versus non-persistent messages -- 10.3.4 One-phase commit optimization -- 10.3.5 Caching WebSphere MQ JMS objects -- 10.3.6 Message-driven beans performance considerations -- 10.4 High availability with WebSphere MQ -- 10.4.1 Overview of WebSphere MQ cluster components -- 10.4.2 WebSphere MQ simplified management -- Part 3 Implementation -- Chapter 11. Technical scenarios -- 11.1 Application flow -- 11.2 System setup -- 11.2.1 Products used to prove the scenarios -- 11.2.2 Development environment -- 11.2.3 Runtime environment -- Chapter 12. Configuring WebSphere -- 12.1 Defining JMS resources to WebSphere -- 12.1.1 Determining the correct scope -- 12.2 Using the embedded JMS server -- 12.2.1 Defining a queue connection factory -- 12.2.2 Defining a queue destination -- 12.2.3 Define the queue for the JMS server -- 12.3 Using WebSphere MQ V5.3 -- 12.3.1 Defining a queue connection factory -- 12.3.2 Define a queue destination -- 12.3.3 Define the queue for WebSphere MQ -- 12.4 Deploying message-driven beans in WebSphere V5.0 -- 12.5 Testing, logging, debugging -- Chapter 13. Configuring WebSphere MQ and MQ Integrator -- 13.1 WebSphere MQ objects -- 13.2 WebSphere MQ system management -- 13.2.1 Remote administration -- 13.3 Creating the WebSphere MQ Integrator databases -- 13.4 Creating the WebSphere MQ Integrator Configuration Manager -- 13.4.1 Creating the brokers -- 13.4.2 Transaction behavior -- 13.5 Testing, logging, debugging -- Appendix A. Additional material -- Locating the Web material -- Using the Web material.

System requirements for downloading the Web material -- How to use the Web material -- 13.5.1 Supplier application configuration -- 13.5.2 Running -- Abbreviations and acronyms -- Related publications -- IBM Redbooks -- Other resources -- Referenced Web sites -- How to get IBM Redbooks -- IBM Redbooks collections -- Index -- Back cover.

Sommario/riassunto

This IBM Redbooks publication introduces the Router and Decomposition application patterns for Self-Service e-business applications. The book discusses the messaging and transactional capabilities of an application. This book is a valuable source for IT architects, IT specialists, application designers, application developers, system administrators, and consultants. Part 1, Patterns for e-business, introduces the Patterns for e-business concept, focusing particularly on the Self-Service business pattern and the Router and Decomposition application patterns. Part 2, Guidelines, provides guidelines for messaging and transactional applications, including application design and development and some of the non-functional requirements for such applications, including security and system management and performance. In the Appendix, you will find details on how to set up and configure both the development and runtime environments for the sample application discussed in this book. Please note that the additional material referenced in the text is not available from IBM.