

SIXTH EDITION

# Enterprise JavaBeans 3.1

*Andrew Lee Rubinger and Bill Burke*

O'REILLY\*

Beijing • Cambridge • Farnham • Kbln • Sebastopol • Tokyo

# Table of Contents

<b>Preface</b> .....	<b>xv</b>
----------------------	-----------

## **Part I. Why Enterprise JavaBeans?**

<b>1. Introduction</b> .....	<b>3</b>
The Problem Domain	3
Breaking Up Responsibilities	3
Code Smart, Not Hard	6
The Enterprise JavaBeans™ 3.1 Specification	8
Review	10
<b>2. Component Types</b> .....	<b>11</b>
Server-Side Component Types	12
Session Beans	12
Message-Driven Beans (MDBs)	15
Entity Beans	16
The Java Persistence Model	17
The Model Isn't Everything	- 17
<b>3. Container Services</b> .....	<b>19</b>
Dependency Injection (DI)	20
Concurrency	21
Instance Pooling/Caching	21
Transactions	23
Security	23
Timers	24
Naming and Object Stores	24
Interoperability	25
Lifecycle Callbacks	25
Interceptors	26

Platform Integration	27
Bringing It Together	27
<b>4. Developing Your First EJBs</b> .....	<b>29</b>
Step 1: Preparation	29
Definitions	29
Naming Conventions	32
Conventions for the Examples	32
Step 2: Coding the EJB	33
The Contract	33
The Bean Implementation Class	34
Out-of-Container Testing	35
Integration Testing	36
Summary	39

## Part II. Server-Side Component Models

<b>5. The Stateless Session Bean</b> .....	<b>43</b>
The XML Deployment Descriptor	45
SessionContext	46
EJBContext	47
The Lifecycle of a Stateless Session Bean	49
The Does Not Exist State	50
The Method-Ready Pool	50
Example: The EncryptionEJB	52
The Contract: Business Interfaces	53
Application Exceptions	54
Bean Implementation Class	55
Accessing Environment Properties (Injection and Lookup)	57
Asynchronous Methods	60
<b>6. The Stateful Session Bean</b> .....	<b>63</b>
The Lifecycle of a Stateful Session Bean	64
The Does Not Exist State	65
The Method-Ready State	65
The Passivated State	66
Example: The FileTransferEJB	68
The Contract: Business Interfaces	69
Exceptions	70
Bean Implementation Class	70
POJO Testing Outside the Container	74
Integration Testing	77

<b>7. The Singleton Session Bean.....</b>	<b>.81</b>
Concurrency	82
Shared Mutable Access	84
Container-Managed Concurrency	86
Bean-Managed Concurrency	87
Lifecycle	87
Explicit Startup	87
Example: The RSSCacheEJB	88
Value Objects	89
The Contract: Business Interfaces	92
Bean Implementation Class	92
<b>8. Message-Driven Beans.....</b>	<b>.97</b>
JMS and Message-Driven Beans	98
JMS as a Resource	98
JMS Is Asynchronous	99
JMS Messaging Models	100
Learning More About JMS	103
JMS-Based Message-Driven Beans	103
@MessageDriven	103
The Lifecycle of a Message-Driven Bean	108
The Does Not Exist State	109
The Method-Ready Pool	109
Connector-Based Message-Driven Beans	111
Message Linking	114
Session Beans Should Not Receive Messages	114
The JMS APIs	115
Example: The StatusUpdateEJBs	118

## Part III. EJB and Persistence

<b>9. Persistence: EntityManager.....</b>	<b>.127</b>
Entities Are POJOs	128
Managed Versus Unmanaged Entities	130
Persistence Context	130
Packaging a Persistence Unit	133
The Persistence Unit Class Set	135
Obtaining an EntityManager	136
EntityManagerFactory	137
Obtaining a Persistence Context	138
Interacting with an EntityManager	140
Example: A Persistent Employee Registry	141

A Transactional Abstraction	141
Persisting Entities	142
Finding and Updating Entities	144
Removing Entities	147
refresh	148
containsO and clear()	148
flushO and FlushModeType	148
Locking	149
unwrap() and getDelegate()	149
<b>10. Mapping Persistent Objects.....</b>	<b>151</b>
The Programming Model	152
The Employee Entity	152
The Bean Class	152
XML Mapping File	154
Basic Relational Mapping	155
Elementary Schema Mappings	155
Primary Keys	157
@Id	157
Table Generators	158
Sequence Generators	159
Primary-Key Classes and Composite Keys	160
Property Mappings	164
©Transient	164
@Basic and FetchType	164
@Lob	166
©Temporal	166
©Enumerated	167
©Embedded Objects	167
<b>11. Entity Relationships.....</b>	<b>171</b>
The Seven Relationship Types	171
One-to-One Unidirectional Relationship	173
One-to-One Bidirectional Relationship	176
One-to-Many Unidirectional Relationship	178
Many-to-One Unidirectional Relationship	181
One-to-Many Bidirectional Relationship	182
Many-to-Many Bidirectional Relationship	184
Many-to-Many Unidirectional Relationship	187
Mapping Collection-Based Relationships	188
Ordered List-Based Relationship	188
Map-Based Relationship	189
Detached Entities and FetchType	190

Cascading	191
PERSIST	192
MERGE	192
REMOVE	193
REFRESH	193
ALL	193
When to Use Cascading	194
<b>12. Entity Inheritance</b>	<b>195</b>
Single Table per Class Hierarchy	196
Advantages	198
Disadvantages	198
Table per Concrete Class	199
Advantages	200
Disadvantages	200
Table per Subclass	200
Advantages	203
Disadvantages	203
Mixing Strategies	203
Nonentity Base Classes	203
<b>13. Queries the Criteria API and JPAQL</b>	<b>205</b>
Query API	206
Parameters	208
Date Parameters	209
Paging Results	209
Hints	210
FlushMode	210
JPAQL	211
Abstract Schema Names	211
Simple Queries	212
Selecting Entity and Relationship Properties	213
Constructor Expressions	215
The IN Operator and INNER JOIN	215
LEFT JOIN	216
Fetch Joins	217
Using DISTINCT	217
The WHERE Clause and Literals	218
The WHERE Clause and Operator Precedence	218
The WHERE Clause and Arithmetic Operators	219
The WHERE Clause and Logical Operators	219
The WHERE Clause and Comparison Symbols	219
The WHERE Clause and Equality Semantics	220

The WHERE Clause and BETWEEN	221
The WHERE Clause and IN	221
The WHERE Clause and IS NULL	222
The WHERE Clause and IS EMPTY	223
The WHERE Clause and MEMBER OF	223
The WHERE Clause and LIKE	224
Functional Expressions	224
The ORDER BY Clause	228
Bulk UPDATE and DELETE	229
Native Queries	230
Scalar Native Queries	230
Simple Entity Native Queries	230
Complex Native Queries	231
Named Queries	232
Named Native Queries	233
<b>14. Entity Callbacks and Listeners.....</b>	<b>235</b>
Callback Events	235
Callbacks on Entity Classes	236
Entity Listeners	237
Default Entity Listeners	238
Inheritance and Listeners	238
Part IV. Container Services	
<b>15. Security.....</b>	<b>243</b>
Authentication and Identity	244
Authorization	245
Example: A Secured School	246
The Business Interface	246
Assigning Method Permissions	247
Programmatic Security	249
The RunAs Security Identity	251
<b>16. JNDI, the ENC, and Injection.....</b>	<b>255</b>
Global JNDI	255
The JNDI ENC	256
What Can Be Registered in the JNDI ENC?	257
How Is the JNDI ENC Populated?	257
How Are Things Referenced from the ENC?	258
Reference and Injection Types	264
• EJB References	264

EntityManagerFactory References	266
EntityManager References	269
Resource References	271
Resource Environment and Administered Objects	275
Environment Entries	275
Message Destination References	277
<b>17. Transactions</b> .....	<b>279</b>
ACID Transactions	279
Example: The BlackjackEJB	281
Helper EJBs for Testing Transactions	283
Is the BlackjackEJB Atomic?	284
Is the BlackjackEJB Consistent?	285
Is the BlackjackEJB Isolated?	285
Is the BlackjackEJB Durable?	285
Declarative Transaction Management	286
Transaction Scope	286
Transaction Attributes	287
Transaction Propagation	293
Isolation and Database Locking	298
Dirty, Repeatable, and Phantom Reads	298
Database Locks	299
Transaction Isolation Levels	300
Balancing Performance Against Consistency	301
Optimistic Locking	302
Programmatic Locking	303
Nontransactional EJBs	303
Explicit Transaction Management	304
Transaction Propagation in Bean-Managed Transactions	307
Heuristic Decisions	308
UserTransaction	309
Status	310
EJBContext Rollback Methods	312
Exceptions and Transactions	313
Application Exceptions Versus System Exceptions	313
Transactional Stateful Session Beans	318
The Transactional Method-Ready State	320
Conversational Persistence Contexts	321
<b>18. Interceptors</b> .....	<b>323</b>
Intercepting Methods	323
Interceptor Class	325
Applying Interceptors	327

Interceptors and Injection	331
Intercepting Lifecycle Events	333
Custom Injection Annotations	333
Exception Handling	335
Aborting a Method Invocation	336
Catch and Rethrow Exceptions	336
Interceptor Lifecycle	338
Bean Class ©AroundInvoke Methods	338
<b>19. Timer Service.....</b>	<b>341</b>
Example: A Batch Credit Card Processing System	342
The Business Interface	343
javax.ejb.ScheduleExpression and @javax.ejb.Schedule	344
The Bean Implementation Class	345
The TimerService	347
The Timer	348
Transactions	350
Stateless Session Bean Timers	351
Message-Driven Bean Timers	352
<b>20. EJB 3.1: Web Services Standards.....</b>	<b>355</b>
Web Services Overview	355
XML Schema and XML Namespaces	356
XML Schema	356
XML Namespaces	361
SOAP 1.1	368
Web Services Styles	369
Exchanging SOAP Messages with HTTP	370
Now You See It, Now You Don't	370
WSDL 1.1	371
The <definitions> Element	372
The <portType> and <message> Elements	374
The <types> Element	375
The <binding> and <service> Elements	377
UDDI 2.0	379
From Standards to Implementation	380
<b>21. EJB 3.1 and Web Services.....</b>	<b>381</b>
Accessing Web Services with JAX-RPC	38.1
Generating JAX-RPC Artifacts from WSDL	382
Calling a Service from an EJB	387
The <service-ref> Deployment Element	388
The JAX-RPC Mapping File	389

Defining a Web Service with JAX-RPC	391
The WSDL Document	392
The Service Endpoint Interface	393
The Stateless Bean Class	393
The Deployment Files	394
Using JAX-WS	396
The @WebService Annotation	396
The @WebMethod Annotation	397
The @SOAPBinding Annotation	398
The @WebParam Annotation	399
The @WebResult Annotation	400
The @OneWay Annotation	401
Separating the Web Services Contract	401
The Service Class	402
The Service Endpoint Interface	403
The @WebServiceRef Annotation	403
Other Annotations and APIs	405
JAXB	405
Taking JAXB Further	407
Conclusion	407

## PartV. Examples

A. FirstEJB Example.....	415
B. Stateless Session EJB: Encryption Example.....	429
C. Stateful Session EJB: FTP Client Example.....	453
D. Singleton Session EJB: RSS Cache Example.....	505
E. Message-Driven EJB: Status Update Listeners Example.....	527
F. Java Persistence APIs: Employee Registry Example.....	557
G. Security: Secured School Example.....	643
H. Transactions: Blackjack Game Example.....	663
I. Interceptors: TV Channel Service Example.....	691

J. TimerService: CreditCard Processor Example .....719

Index .....727