Java Database Best Practices, ?? Edition
by George Reese
Published by O'Reilly, , , .

Revision History for the :

To my beautiful wife, Monique, and the child she carries.
# Table of Contents

Preface ......................................................... xi

Part I. Data Architecture ............................... 3

1. Elements of Database Applications ......................... 4
   Database Application Architectures 
   The Network Architecture  
   Various System Architectures  
   Software Architecture  
   Component Models  
   JavaBeans  
   Enterprise JavaBeans  
   Persistence Models  
   EJB Persistence  
   Other Persistence Models ................................... 24

2. Relational Data Architecture .............................. 25
   Relational Concepts  
   The Relational Model  
   Entities  
   Constraints  
   Relationships  
   NULL  
   Modeling  
   Normalization  
   Before Normalization  
   Basic Normalization  
   Specialized Normalization  
   Denormalization ........................................... 52
Object-Relational Mapping 55
Inheritance Mapping 57
Multivalued Attributes 59

3. Transaction Management ........................................ 61
Transactions 62
ACID Requirements 62
Transaction Design 63
Concurrency 67
Isolation Levels 67
Locking 69
JDBC Transaction Management 71
Basic Transaction Management 71
Optimistic Concurrency 72
Batch Transactions 76
Savepoints 78
Transaction Management Paradigms 80

Part II. Persistence Models

4. Persistence Fundamentals ........................................... 85
Patterns of Persistence 85
Division of Labor 85
Sequence Generation 87
Mementos 88
Object Caching 88
A Guest Book Application 90
The View 91
The Controller 94
The Business Object (Model) 96
The Data Access Objects 100

5. EJB CMP .......................................................... 109
Which CMP Model to Use? 110
The EJB 1.0 CMP Model 111
Field Mapping 113
Persistence Methods 114
Searches 116
Transactions 116
The EJB 2.0 CMP Model 118
Container-Managed Relationship 118
### Table of Contents

#### 6. EJB BMP
- EJBs Revisited
  - The Components of a Bean
  - Kinds of Beans
- BMP Patterns
  - Data Access Objects
  - Value Objects
  - Sessions as Transactions
- State Management
  - Lazy-Loading
  - To Store or Not to Store
- Exception Handling

#### 7. JDO Persistence
- JDO or EJB?
- Basic JDO Persistence
  - Transaction Management
  - Query Control
- EJB BMP with JDO
  - Transaction Management
  - Persistence Strategies

#### 8. Alternative Persistence Frameworks
- Why Alternative Frameworks?
- Persistence Approach
  - Castor Field Mapping
  - Hibernate Field Mapping
- Persistence Operations
  - Castor Persistence
  - Hibernate Persistence
- Searches
  - Castor Searches
  - Hibernate Searches
- Beyond the Basics

### Part III. Tutorials

#### 9. J2EE Basics
10. SQL .................................................................................................................. 205
    Background ........................................................................................................ 206
        The SQL Story ............................................................................................... 206
        Database Interaction .................................................................................. 207
        Basic Syntax ............................................................................................... 208
    Database Creation ............................................................................................ 210
    Table Management .......................................................................................... 211
        The Basics of Table Creation .................................................................. 211
        Data Types .................................................................................................. 212
        Indexing ........................................................................................................ 215
    Data Management .............................................................................................. 216
        Inserts ........................................................................................................... 216
        Primary Keys .............................................................................................. 217
        Updates ......................................................................................................... 217
        The WHERE Clause .................................................................................... 218
        Deletes .......................................................................................................... 218
        Queries .......................................................................................................... 219
        Operators ...................................................................................................... 222
        Functions ...................................................................................................... 225
        Joins ................................................................................................................ 226

11. JDBC .................................................................................................................. 231
    Architecture ...................................................................................................... 231
        The Core Interfaces .................................................................................... 232
        Databases and Drivers ............................................................................... 235
Simple Database Access
  The Connection
  Query Execution
  Transactions
  Error Handling and Cleanup
  Prepared SQL
Advanced JDBC
  Batch Processing
  Metadata
  Hidden Features

12. JDO
  Architecture
  Business Objects
  Applications
  Implementations
  Data Stores
  Enhancement
  Class Metadata
  Running the Enhancer
  The Database
  Queries
  The JDO Extent
  The JDO Query
  Complex Queries
  The Filter Language
  Changes
  Transactions
  The Transaction Class
  Managed Versus Nonmanaged Environments
  Inheritance

Index
Java database programming has grown much more complex than it was in 1996 when I wrote the first edition of my book *Database Programming with JDBC and Java* (O’Reilly & Associates). The J2EE platform did not exist. Distributed programming was RMI, JDBC was simple, and transaction management and persistence did not exist in the Java vocabulary. Database programming in 1996 was quite simply JDBC programming.

To place database programming in a real-world context, I spent much of that book introducing ways to build robust persistence models and manage transactions using only the JDBC API. As you can imagine, you had to do a lot of things for yourself that developers now take for granted in the Java platform.

The Java world has certainly changed since then. Not only does Java provide you with a persistence model, it provides you with three different persistence models built right into the core J2EE platform. Outside the J2EE platform is the popular JDO persistence model. In addition, many tools exist to enable you to effectively use third-party and custom persistence models. All of these choices present a problem for database programmers that simply did not exist in 1996: what are the best approaches to database programming with the Java language?

This book seeks to aid the Java developer in appreciating the different approaches Java provides for database programming. It helps you assess what approaches fit which problems, and what the best practices are under each model.
Audience

This book is not an introductory text. It is also not a tutorial on any particular API. It is, instead, a description of the best practices for using a database to drive a variety of Java application architectures. It assumes you have at least a passing familiarity with one or more of the Java enterprise APIs, as well as SQL. You do not, however, need to be an expert in all of them. To help you with any holes in your knowledge of these tools, I provide a few tutorial chapters at the end of the book.

Organization of This Book

This book is divided into three distinct sections. The first two sections are the meat of this book: best practices for Java database architecture and development. The first section focuses on the architecture aspect and the second section on the development aspect.

Part I

Chapter 1 is an overview of the art of database programming. It examines the various tools and skills needed for database programming and covers common database application architectures. The chapter is mostly review material for experienced database programmers.

Chapter 2 tackles one of the more difficult aspects of database programming, especially for the object-oriented programmer: data architecture. This chapter begins with relational theory and covers critical topics such as normalization and object-relational modeling. It is a very important chapter for database programmers of all levels of experience.

Though relational architecture is one of the more difficult aspects of database programming, transaction management is where database programmers make most of their mistakes. Chapter 3 covers transactions and transaction management.

Part II

The second section begins with an overview of persistence concepts. In short, persistence is the practice of saving application state to a data store. Chapter 4 introduces this practice with an eye on using relational databases as your data store for Java applications.

Chapter 5 through Chapter 8 go into the best practices for different Java persistence models. Chapter 5 begins with container-managed persistence under the Enterprise JavaBeans component model—for Versions 1 and 2. Chapter 6 tackles the other EJB persistence model, bean-managed persistence. Chapter 7 dives into an evolving, popular persistence model, Java Data Objects. Finally, Chapter 8 looks at alternatives to the standard Java persistence models.
Part III

The third section of the book contains tutorials on the core technologies covered in this book. No reader should need to read all of the tutorial chapters. Instead, I expect that most readers will be familiar with the subject in several, but not all, of the tutorial chapters. The tutorial chapters provide the basic knowledge necessary to understand key concepts used in the first two sections. Don't look to any of the tutorial chapters to make you an expert in its subject matter. I have provided tutorials on the J2EE platform (Chapter 9), SQL (Chapter 10), JDBC (Chapter 11), and JDO (Chapter 12).

I recommend reading the first two sections in order, breaking that order only to refer to a tutorial chapter for a subject on which you lack familiarity.

Conventions Used in This Book

The following typographical conventions are used in this book:

*Italic*

Used for filenames and directory names, programs, compilers, tools, utilities, URLs, emphasis, and first use of a technical term.

*Constant width*

Used in code examples and to show the contents of files. Also used for tags, attributes, and environment variable names appearing in the text.

*Constant width italic*

Used as a placeholder to indicate an item that should be replaced with an actual value in your program.

*Constant width bold*

Used to highlight a particular section or change in code, such as a custom tag or a change in a transaction.

Comments and Questions

Please address comments and questions concerning this book to the publisher:

O'Reilly & Associates, Inc.
1005 Gravenstein Highway North
Sebastopol, CA 95472
(800) 998-9938 (in the United States or Canada)
(707) 829-0515 (international/local)
(707) 829-0104 (fax)

There is a web page for this book which lists errata, examples, or any additional information. You can access this page at:
About the Philosophers

Daniel Dennett (Chapter 1)
Dennett, who teaches at Tufts University, is probably my favorite philosopher. His books are actually well written, which is a rare quality among philosophy texts. His works run the spectrum of philosophy, but his greatest influence lies in the philosophies of mind and science. If you want a fun philosophy book to read that does not require you to be a philosopher, pick up his book *Elbow Room*. If you are looking for something more weighty, but equally accessible, read *Darwin’s Dangerous Idea*.

René Descartes (Chapter 2)
Though he lived from 1596 until 1650, Descartes’s writings mark the beginning of modern philosophy. He was a French philosopher who emphasized a solipsistic approach to epistemology. He is the author of the famous quote “Cogito, ergo sum,” or “I think, therefore I am.”

Donald Davidson (Chapter 3)
Donald Davidson is among the most important philosophers of the late 20th century. He is particularly influential in the philosophy of language and action theory. He is currently a professor at the University of California, Berkeley. My senior thesis at Bates College was based on his writings.

Ludwig Wittgenstein (Chapter 4)
Ludwig Wittgenstein was a German philosopher who lived from 1889 until 1951. His primary contributions to philosophy were in the philosophy of language. He once wrote that “philosophy is a battle against the bewitchment of our intelligence by means of language.”

Friedrich Nietzsche (Chapter 5)
Nietzsche, who lived in Germany from 1844 until 1900, is likely the most controversial “serious” philosopher. His writings have influenced nearly every kind of philosophy but have had their greatest impact—both positive and negative—in the area of ethics.
transitive dependencies, 46
transparency of persistence models, 145
Truth and Existence (Sartre), 261
tuples (see rows)
two-phase commits, 258
two-tier architecture (see client/server architecture)
typographical conventions used in this book, xiii

U
UI (user interface)
   patterns, 16–17
   view logic for, 86
UI (user interface) layer, distributed architectures, 12
unary operators, in SQL, 222
UnicastRemoteObject class (RMI), 193
unique index, 29, 29–32
unknown value (see NULL)
unmarshaling, 194
update anomalies, 45, 46
UPDATE statement (SQL), 209, 217
URL for JDBC database connection, 239, 241
use cases, corresponding to transactions, 64
user actions, control logic for, 86
User class (JNDI), 178
user interface (see UI)
UserFactory class (JNDI), 178
UUID algorithm (Castor), 160
uuid.hex algorithm (Hibernate), 162
uuid.string algorithm (Hibernate), 162

V
value objects, 132, 152
VARCHAR datatype (SQL), 214
VARCHAR2 datatype (SQL), 215
view logic
   as part of application logic, 86
   for Guest Book application, 91–93
Vlissides, John (Design Patterns), 85
vm.hex algorithm (Hibernate), 162
vm.long algorithm (Hibernate), 162

W
wasNull( ) method (JDBC), 243
web applications
   EJB for, 144
   JDBC for, 144
   JDO for, 145
   MVC pattern used for, 16
web pages, generating dynamically (see JSP)
web services layer, distributed architectures, 12
web site architecture, 10
web sites
   Castor, 154
   for this book, xiii
   Guest Book application, 90
   Hibernate, 154
   JDBC drivers, 236
   JDO reference implementation, 262, 265
   JNDI service providers, 174
   O’Reilly & Associates, Inc., xiv
WHERE clause (SQL), 218, 225
wildcard (*), JNDI search filter, 181
Wittgenstein, Ludwig (philosopher), xiv, 85
WORA (Write Once, Run Anywhere), 236
Write Once, Run Anywhere (see WORA)

X
X/OPEN SQL, CLI (Call Level Interface), 232
XHTML, generating dynamically (see JSP)
XML
   Castor field mapping, 158–160
   EJB CMP field mapping, 113
   generating dynamically (see JSP)
   Hibernate field mapping, 160–162
   JDO metadata descriptions, 266
   JNDI data source configuration, 238
   JSP class mappings in, 186
Colophon

Our look is the result of reader comments, our own experimentation, and feedback from distribution channels. Distinctive covers complement our distinctive approach to technical topics, breathing personality and life into potentially dry subjects.

The animal on the cover of Java Database Best Practices is a taguan. The taguan (Petaurista petaurista) is giant flying squirrel. It lives in dense, tropical rainforests, ranging from the eastern regions of Afghanistan to Java, and from Kashmir, Taiwan, and southern China to Sri Lanka. It is most often found in the Pakistan's temperate forests.

The squirrel conceals its nest in the cavity of a tree, raising 2-3 young at a time. It has a lifespan of approximately 16 years. It is a nocturnal animal, recognizable by its big eyes and reddish color. The taguan consumes a diet of pine cones, fruit, leaves, and nuts.

The taguan is an excellent climber. Additionally, it is referred to as a “flying” squirrel because of the muscular membrane that extends from its wrists to its hind legs, enabling it to glide long distances. It leaps from high tree branches and the tops of trees, controlling the direction of its flight by flexing and relaxing the muscles of the membrane.

Colleen Gorman was the production editor, and Norma Emory was the copyeditor for Java Database Best Practices. Linley Dolby and Jane Ellin provided quality control. Angela Howard wrote the index.

Emma Colby designed the cover of this book, based on a series design by Edie Freedman. The cover image is a 19th-century engraving from Animate Creation, Volume II. Emma Colby produced the cover layout with QuarkXPress 4.1 using Adobe's ITC Garamond font.

David Futato designed the interior layout. This book was converted by Joe Wizda to FrameMaker 5.5.6 with a format conversion tool created by Erik Ray, Jason McIntosh, Neil Walls, and Mike Sierra that uses Perl and XML technologies. The text font is Linotype Birka; the heading font is Adobe Myriad Condensed; and the code font is LucasFont’s TheSans Mono Condensed. The illustrations that appear in the book were produced by Robert Romano and Jessamyn Read using Macromedia FreeHand 9 and Adobe Photoshop 6. The tip and warning icons were drawn by Christopher Bing. This colophon was written by Colleen Gorman.

The online edition of this book was created by the Safari production group (John Chodacki, Becki Maisch, and Madeleine Newell) using a set of Frame-to-XML conversion and cleanup tools written and maintained by Erik Ray, Benn Salter, John Chodacki, and Jeff Liggett.