

contents

foreword to the first edition xix
preface xxi
acknowledgments xxiii
about this book xxv
about the cover illustration xxxii

PART 1 GETTING STARTED WITH ORM.....1

1 Understanding object/relational persistence 3

1.1 What is persistence? 4

Relational databases 5 ▪ *Understanding SQL* 6
Using SQL in Java 6

1.2 The paradigm mismatch 8

The problem of granularity 10 ▪ *The problem of subtypes* 11
The problem of identity 12 ▪ *Problems relating to associations* 14
The problem of data navigation 15

1.3 ORM and JPA 16

1.4 Summary 18

2 Starting a project 19

2.1 Introducing Hibernate 19

- 2.2 “Hello World” with JPA 20
 - Configuring a persistence unit* 21 ▪ *Writing a persistent class* 23 ▪ *Storing and loading messages* 24
- 2.3 Native Hibernate configuration 26
- 2.4 Summary 29

3 **Domain models and metadata** 30

- 3.1 The example CaveatEmptor application 31
 - A layered architecture* 31 ▪ *Analyzing the business domain* 33 ▪ *The CaveatEmptor domain model* 34
- 3.2 Implementing the domain model 36
 - Addressing leakage of concerns* 36 ▪ *Transparent and automated persistence* 37 ▪ *Writing persistence-capable classes* 38 ▪ *Implementing POJO associations* 41
- 3.3 Domain model metadata 46
 - Annotation-based metadata* 46 ▪ *Applying Bean Validation rules* 49 ▪ *Externalizing metadata with XML files* 51
 - Accessing metadata at runtime* 55
- 3.4 Summary 58

PART 2 MAPPING STRATEGIES.....59

4 **Mapping persistent classes** 61

- 4.1 Understanding entities and value types 62
 - Fine-grained domain models* 62 ▪ *Defining application concepts* 62 ▪ *Distinguishing entities and value types* 64
- 4.2 Mapping entities with identity 65
 - Understanding Java identity and equality* 65 ▪ *A first entity class and mapping* 66 ▪ *Selecting a primary key* 67 ▪ *Configuring key generators* 69 ▪ *Identifier generator strategies* 71
- 4.3 Entity-mapping options 74
 - Controlling names* 74 ▪ *Dynamic SQL generation* 77
 - Making an entity immutable* 78 ▪ *Mapping an entity to a subselect* 79
- 4.4 Summary 80

- 5 Mapping value types 81**
- 5.1 Mapping basic properties 82
 - Overriding basic property defaults* 82
 - *Customizing property access* 84
 - *Using derived properties* 86
 - Transforming column values* 87
 - *Generated and default property values* 88
 - *Temporal properties* 89
 - Mapping enumerations* 89
 - 5.2 Mapping embeddable components 90
 - The database schema* 90
 - *Making classes embeddable* 91
 - Overriding embedded attributes* 94
 - *Mapping nested embedded components* 95
 - 5.3 Mapping Java and SQL types with converters 97
 - Built-in types* 97
 - *Creating custom JPA converters* 103
 - Extending Hibernate with UserTypes* 109
 - 5.4 Summary 116
- 6 Mapping inheritance 117**
- 6.1 Table per concrete class with implicit polymorphism 118
 - 6.2 Table per concrete class with unions 120
 - 6.3 Table per class hierarchy 122
 - 6.4 Table per subclass with joins 125
 - 6.5 Mixing inheritance strategies 128
 - 6.6 Inheritance of embeddable classes 131
 - 6.7 Choosing a strategy 133
 - 6.8 Polymorphic associations 135
 - Polymorphic many-to-one associations* 135
 - *Polymorphic collections* 137
 - 6.9 Summary 139
- 7 Mapping collections and entity associations 140**
- 7.1 Sets, bags, lists, and maps of value types 141
 - The database schema* 141
 - *Creating and mapping a collection property* 141
 - *Selecting a collection interface* 142
 - Mapping a set* 145
 - *Mapping an identifier bag* 146
 - Mapping a list* 147
 - *Mapping a map* 148
 - *Sorted and ordered collections* 149

- 7.2 Collections of components 152
 - Equality of component instances* 152
 - *Set of components* 154
 - Bag of components* 156
 - *Map of component values* 157
 - Components as map keys* 158
 - *Collection in an embeddable component* 160
- 7.3 Mapping entity associations 161
 - The simplest possible association* 162
 - *Making it bidirectional* 163
 - *Cascading state* 164
- 7.4 Summary 171

8 **Advanced entity association mappings** 172

- 8.1 One-to-one associations 173
 - Sharing a primary key* 173
 - *The foreign primary key generator* 176
 - *Using a foreign key join column* 179
 - Using a join table* 180
- 8.2 One-to-many associations 182
 - Considering one-to-many bags* 183
 - *Unidirectional and bidirectional list mappings* 184
 - *Optional one-to-many with a join table* 186
 - *One-to-many association in an embeddable class* 188
- 8.3 Many-to-many and ternary associations 190
 - Unidirectional and bidirectional many-to-many associations* 190
 - Many-to-many with an intermediate entity* 192
 - *Ternary associations with components* 197
- 8.4 Entity associations with Maps 200
 - One-to-many with a property key* 200
 - *Key/Value ternary relationship* 201
- 8.5 Summary 202

9 **Complex and legacy schemas** 203

- 9.1 Improving the database schema 204
 - Adding auxiliary database objects* 205
 - *SQL constraints* 208
 - Creating indexes* 214
- 9.2 Handling legacy keys 215
 - Mapping a natural primary key* 215
 - *Mapping a composite primary key* 216
 - *Foreign keys in composite primary keys* 218
 - *Foreign keys to composite primary keys* 221
 - *Foreign key referencing non-primary keys* 222

- 9.3 Mapping properties to secondary tables 224
- 9.4 Summary 225

PART 3 TRANSACTIONAL DATA PROCESSING227

10 *Managing data* 229

- 10.1 The persistence life cycle 230
 - Entity instance states* 231 ▪ *The persistence context* 232
- 10.2 The EntityManager interface 234
 - The canonical unit of work* 234 ▪ *Making data persistent* 236
 - Retrieving and modifying persistent data* 237 ▪ *Getting a reference* 239 ▪ *Making data transient* 240 ▪ *Refreshing data* 241 ▪ *Replicating data* 242 ▪ *Caching in the persistence context* 243 ▪ *Flushing the persistence context* 244
- 10.3 Working with detached state 245
 - The identity of detached instances* 245 ▪ *Implementing equality methods* 247 ▪ *Detaching entity instances* 250
 - Merging entity instances* 251
- 10.4 Summary 253

11 *Transactions and concurrency* 254

- 11.1 Transaction essentials 255
 - ACID attributes* 255 ▪ *Database and system transactions* 256
 - Programmatic transactions with JTA* 256 ▪ *Handling exceptions* 258 ▪ *Declarative transaction demarcation* 261
- 11.2 Controlling concurrent access 261
 - Understanding database-level concurrency* 262
 - Optimistic concurrency control* 266 ▪ *Explicit pessimistic locking* 273 ▪ *Avoiding deadlocks* 277
- 11.3 Nontransactional data access 278
 - Reading data in auto-commit mode* 279
 - Queueing modifications* 281
- 11.4 Summary 282

12 *Fetch plans, strategies, and profiles* 283

- 12.1 Lazy and eager loading 284
 - Understanding entity proxies* 285 ▪ *Lazy persistent collections* 289 ▪ *Lazy loading with interception* 291
 - Eager loading of associations and collections* 294

- 12.2 Selecting a fetch strategy 296
 - The n+1 selects problem* 296
 - The Cartesian product problem* 297
 - Prefetching data in batches* 299
 - Prefetching collections with subselects* 302
 - Eager fetching with multiple SELECTs* 303
 - Dynamic eager fetching* 304
- 12.3 Using fetch profiles 306
 - Declaring Hibernate fetch profiles* 306
 - Working with entity graphs* 307
- 12.4 Summary 311

13 *Filtering data* 312

- 13.1 Cascading state transitions 313
 - Available cascading options* 314
 - Transitive detachment and merging* 315
 - Cascading refresh* 317
 - Cascading replication* 319
 - Enabling global transitive persistence* 320
- 13.2 Listening to and intercepting events 321
 - JPA event listeners and callbacks* 321
 - Implementing Hibernate interceptors* 325
 - The core event system* 329
- 13.3 Auditing and versioning with Hibernate Envers 330
 - Enabling audit logging* 331
 - Creating an audit trail* 332
 - Finding revisions* 333
 - Accessing historical data* 334
- 13.4 Dynamic data filters 337
 - Defining dynamic filters* 337
 - Applying the filter* 338
 - Enabling the filter* 338
 - Filtering collection access* 339
- 13.5 Summary 341

PART 4 WRITING QUERIES 343

14 *Creating and executing queries* 345

- 14.1 Creating queries 346
 - The JPA query interfaces* 347
 - Typed query results* 349
 - Hibernate's query interfaces* 349
- 14.2 Preparing queries 351
 - Protecting against SQL injection attacks* 351
 - Binding named parameters* 352
 - Using positional parameters* 353
 - Paging through large result sets* 354
- 14.3 Executing queries 355
 - Listing all results* 355
 - Getting a single result* 356
 - Scrolling with database cursors* 357
 - Iterating through a result* 358

- 14.4 Naming and externalizing queries 359
 - Calling a named query* 360 ▪ *Defining queries in XML metadata* 360 ▪ *Defining queries with annotations* 361
 - Defining named queries programmatically* 362
- 14.5 Query hints 363
 - Setting a timeout* 364 ▪ *Setting the flush mode* 365
 - Setting read-only mode* 365 ▪ *Setting a fetch size* 366
 - Setting an SQL comment* 366 ▪ *Named query hints* 366
- 14.6 Summary 368

15 *The query languages* 369

- 15.1 Selection 370
 - Assigning aliases and query roots* 371 ▪ *Polymorphic queries* 372
- 15.2 Restriction 373
 - Comparison expressions* 374 ▪ *Expressions with collections* 378 ▪ *Calling functions* 380
 - Ordering query results* 382
- 15.3 Projection 383
 - Projection of entities and scalar values* 383 ▪ *Using dynamic instantiation* 385 ▪ *Getting distinct results* 387 ▪ *Calling functions in projections* 387
 - Aggregation functions* 389 ▪ *Grouping* 391
- 15.4 Joins 392
 - Joins with SQL* 393 ▪ *Join options in JPA* 395
 - Implicit association joins* 395 ▪ *Explicit joins* 396
 - Dynamic fetching with joins* 398 ▪ *Theta-style joins* 401 ▪ *Comparing identifiers* 403
- 15.5 Subselects 404
 - Correlated and uncorrelated nesting* 405
 - Quantification* 406
- 15.6 Summary 407

16 *Advanced query options* 408

- 16.1 Transforming query results 409
 - Returning a list of lists* 410 ▪ *Returning a list of maps* 410
 - Mapping aliases to bean properties* 411 ▪ *Writing a ResultTransformer* 412
- 16.2 Filtering collections 414

16.3 The Hibernate criteria query API 416

Selection and ordering 416 ▪ *Restriction* 417
Projection and aggregation 419 ▪ *Joins* 420
Subselects 422 ▪ *Example queries* 423

16.4 Summary 425

17 Customizing SQL 426

17.1 Falling back to JDBC 427

17.2 Mapping SQL query results 429

Projection with SQL queries 430 ▪ *Mapping to an entity class* 431 ▪ *Customizing result mappings* 432
Externalizing native queries 444

17.3 Customizing CRUD operations 448

Enabling custom loaders 448 ▪ *Customizing creation, updates, and deletion* 449 ▪ *Customizing collection operations* 451 ▪ *Eager fetching in custom loaders* 453

17.4 Calling stored procedures 455

Returning a result set 457 ▪ *Returning multiple results and update counts* 458 ▪ *Setting input and output parameters* 460 ▪ *Returning a cursor* 463

17.5 Using stored procedures for CRUD 464

Custom loader with a procedure 465
Procedures for CUD 466

17.6 Summary 468

PART 5 BUILDING APPLICATIONS 469**18 Designing client/server applications 471**

18.1 Creating a persistence layer 472

A generic data access object pattern 474 ▪ *Implementing the generic interface* 475 ▪ *Implementing entity DAOs* 477
Testing the persistence layer 479

18.2 Building a stateless server 481

Editing an auction item 481 ▪ *Placing a bid* 484
Analyzing the stateless application 488

18.3 Building a stateful server 489

Editing an auction item 490 ▪ *Analyzing the stateful application* 495

18.4 Summary 497

19 *Building web applications* 498

- 19.1 Integrating JPA with CDI 499
 - Producing an EntityManager* 499
 - *Joining the EntityManager with transactions* 501
 - *Injecting an EntityManager* 501
- 19.2 Paging and sorting data 503
 - Offset paging vs. seeking* 503
 - *Paging in the persistence layer* 505
 - *Querying page-by-page* 511
- 19.3 Building JSF applications 512
 - Request-scoped services* 513
 - *Conversation-scoped services* 516
- 19.4 Serializing domain model data 524
 - Writing a JAX-RS service* 525
 - *Applying JAXB mappings* 526
 - *Serializing Hibernate proxies* 528
- 19.5 Summary 531

20 *Scaling Hibernate* 532

- 20.1 Bulk and batch processing 533
 - Bulk statements in JPQL and criteria* 533
 - *Bulk statements in SQL* 538
 - *Processing in batches* 539
 - *The Hibernate StatelessSession interface* 543
- 20.2 Caching data 544
 - The Hibernate shared cache architecture* 545
 - Configuring the shared cache* 549
 - *Enabling entity and collection caching* 551
 - *Testing the shared cache* 554
 - *Setting cache modes* 557
 - *Controlling the shared cache* 558
 - *The query result cache* 558
- 20.3 Summary 561
 - references* 562
 - index* 563