

# Index

## Symbols

- <abstract-schema-name> 384, 394, 406
- <acknowledge-mode> 460
- <assembly-descriptor> 503
- <cascade-delete/> 393
- <cmp-field> 384
- <cmr-field-name> 388
- <cmr-field-type> 388
- <cmr-field> 388
- <container-transaction> 502
- <destination-type> 449
- <ejb-link> 615
- <ejb-local-ref> 614
- <ejb-ref-name> 613
- <ejb-ref-type> 613
- <ejb-ref> 613
- <ejb-relation> 388
- <ejb-relationship-role-name> 388
- <ejb-relationship-role> 388
- <env-entry-name> 607
- <env-entry-type> 607
- <env-entry-value> 607
- <env-entry> 607
- <exclude-list> 574
- <message-driven-destination> 449
- <message-driven> 449
- <method-intf> 582
- <method-param> 505
- <method-params> 505, 579
- <method-permission> 574, 578
- <method> 504, 579
- <multiplicity> 388, 422
- <persistence-type> 385
- <relationships> 395
- <res-auth> 610, 611
- <res-ref-name> 610
- <res-sharing-scope> 610
- <res-type> 610
- <resource-env-ref> 616
- <resource-ref> 610
- <role-link> 589
- <role-name> 574
- <run-as> 591, 592
- <security-identity> 591
- <security-role-ref> 589

<security-role> 574  
<trans-attribute> 476, 502, 504  
<transaction-type> 476, 502  
<unchecked/> 574, 581

## A

abstract 313  
abstract getters and setters 315, 383  
abstract getter and setter 313  
abstract persistence schema 384, 387  
abstract schema type 408  
ACID test 472  
acknowledgement. *See* message acknowledgement  
actor playing an entity 350  
administered object 616  
Advice Guy bean 15  
afterBegin() 512  
afterCompletion() 512  
Application 33  
    making a new J2EE app in the deploytool 33  
Application Assembler 26  
Application Assembler's responsibility  
    <ejb-link> 615  
    deployment descriptor 617  
    security 574, 588  
Application exceptions 532  
application exceptions 500, 548

architectural overview 63  
arguments 67  
Atomic 472  
attributes. *See* transaction attributes  
authentication 572  
authorization 572  
Auto-acknowledge 460

## B

Bean-managed transactions. *See* BMT  
beanness 188, 191, 196  
beans 11. *See also* entity beans; *See also* message-driven beans; *See also* session beans  
    bean inheritance 186  
    five things you do to build a bean 16  
    overview 98  
bean creation 188, 192, 194, 196. *See also* entity beans  
bean getting shot 283  
Bean Provider 26  
    exception responsibilities 541  
    passivation responsibility for stateful session beans 203  
    responsibility for the Deployment Descriptor 617  
    security responsibilities 589  
    writing a CMP entity bean 313  
    writing a message-driven bean 447  
beat box. *See also* appendix A  
beforeCompletion() 512

begin() 476, 478  
 BeingJohn Entity Bean 350  
 BETWEEN 416. *See* EJB-QL  
 BMP 303  
 BMT 476, 478, 482, 485, 487, 490, 502, 508  
     getStatus() 489  
     restrictions 481  
     summary of bean-managed demarcation 508  
 bodyguard 94  
 bullet points 166  
     architectural overview 97  
     client view 131  
     EJB-QL 420  
     entity client view 287  
     entity relationships 420  
     exceptions 555  
     local interfaces 166  
     overview 14  
     Remote interfaces 144  
     RMI 76  
     session bean lifecycle 221  
 business interface 79  
 business methods 199, 226

## C

cascade-delete 387, 388  
 cast 120  
     narrow before cast 120  
 client 55  
     client Code 57, 219  
     client view 111  
     component interface 136  
     exceptions 159  
     local 145  
     code example 119  
     creating a test client 55  
     InitialContext 117  
     local client code 160  
     lookup 116  
 Client before and after message-driven beans 441  
 client Jar 51  
 client view 111, 131  
     EJBHome methods 129  
     EJBObject methods 137  
     entity bean 287  
     exceptions 159  
     local 145  
     local interface code 156  
 CMP 302  
     vs. BMP 303  
 CMR 383, 396. *See also* cascade-delete  
 cmr-field 387

- CMR field 383, 386
  - CMT 476, 482, 485, 487, 499, 502, 509
    - summary of container-managed transactions 509
  - CMT. 491
  - code examples
    - AdviceStatefulBean code 218
    - client 119
  - AdviceStatefulBean client 219
  - exceptions in client interfaces 538
  - CMP entity bean complete 318
  - CustomerBeanCMP 318
  - Customer BMT entity bean 268
  - Customer CMT entity bean 316
  - entity bean Remote home interface 275
  - exceptions in client interfaces 538
  - local client code 160
  - local client interfaces 156
  - message-driven beans 446
  - MovieBean CMP code 396
  - MovieHomeRemote interface 398
  - Remote home interface 126
  - WelcomeBean 446
- Coffee Cram
- appendix A: final mock exam 637
  - chapter 1: overview 59
  - chapter 10: exceptions 563
  - chapter 11: security 593
  - chapter 12: environment and programming restrictions 627
  - chapter 3: client view 168
  - chapter 4: session beans 244
  - chapter 5: entity bean client view 288
  - chapter 6: entity bean synchronization 362
  - chapter 7: entity relationships and EJB-QL 425
  - chapter 8: message-driven beans 463
  - chapter 9: transactions 516
- commit 471, 473
- commit() 478
- commit options 356
- compare
  - equals() vs. isIdentical() 150
  - Remote vs. local interfaces 154
- compiling a bean 30
- component-based development 3
- component interface 87, 133, 234, 271, 272
  - implementing 87
- confusingly-named exceptions 554
- Consistent 472
- constructor 195, 196
- container 6
- container-managed persistence. *See* CMP
- container-managed relationship. *See* CMR
- Container-managed transactions. *See* CMT
- container callbacks 181, 310
  - ejbActivate() 183
  - ejbCreate() 181
  - ejbPassivate() 183
  - ejbRemove() 183
  - implementing in a session bean 183
  - setSessionContext() 183

Container responsibilities

exceptions 546

create() 123, 232, 237

entity vs. session bean 281

CreateException 125, 548

create methods 159, 181

creation

entity beans 329

entity vs. session bean 281

session beans 188, 192

## D

database 301

database to entity mapping 400

DataSource 602, 610

dead bean 284

entity vs. instance death 285

declarative security 572

declarative transactions. *See* CMT

DELETE 265

deploy-time customization 602, 618

Deployer 27

deployment descriptor responsibilities 618

responsibilities for mapping abstract roles to people 583

deployment descriptor

<ejb-link> 615

abstract persistence schema 384

abstract schema 394

intro 23

message-driven bean 449

resource manager connection factories 610

transactions 502, 503, 504

method parameters 505

wildcard 579

deploytool 32

deserializing 71

Director 378

Director talking to entity bean 284

distributed transactions 473

two-phase commit 473

DuplicateKeyException 548

Dups-ok-acknowledge 460

Durable 472

durable subscription 450

## E

EJB

and RMI 81

architectural overview 63

EJBObject 82

ejb-jar 24, 620, 624

requirements 24

ejb-jar.xml 624

- EJB-QL 401, 402, 411
  - AND 405
  - BETWEEN 416
  - Collection types 413
  - FROM clause 403
  - identifiers 410
  - IN operator for the FROM clause 415
  - IN with the WHERE clause 417
  - IS EMPTY 418
  - IS NOT EMPTY 418
  - LIKE expression 419
  - parameters 405
  - path navigation 404
  - reserved words 410
  - SELECT clause 403, 406
  - String literals 417
  - topics and queues 450
  - using collections in a path 414
  - WHERE 412
  - WHERE clause 403, 411
- ejb-relation 387
- ejb-relationship-role 387
- ejbActivate() 183, 200, 204, 207, 308, 314, 354
- EJBContext 477
- ejbCreate() 181, 195, 196, 237, 331, 333
- EJBException 537, 539, 552, 558
- EJBHome 128
  - methods 129
- ejbLoad() 314, 354
- EJBLocalHome 145
- EJBLocalObject 145, 156
- EJBMetaData 149
- EJBObject 82, 94, 133, 137
- ejbPassivate() 183, 200, 204, 207, 308, 314, 355
- ejbPostCreate() 334, 396
- ejbRemove() 183, 210, 213, 214, 216, 314
  - missed ejbRemove() in session beans 213
- ejbStore() 314, 355
- EJB 2.0 guarantees
  - services and behavior 623
  - supported APIs 621
- EJB references 613
  - <ejb-local-ref> 614
- EJB Roles and Responsibilities 26
  - Application Assembler 26
  - Bean Provider 26
  - Deployer 27
- EJB security mapping 590
- EntityBean 312, 314
- entity beans 99
  - activation 352, 354
  - actor playing an entity 350
  - business method 348
  - client view 287
  - CMP bean construction 326, 327
  - CMP bean creation 328, 338
  - ejbCreate() 331, 333
  - ejbPostCreate() 334
  - OID for bean creation 329
    - CMP class example 316

- CMP complete class code 318
- container callbacks 310, 312
- `ejbActivate()` 308
- `ejbPassivate()` 308
- `ejbPostCreate()` 396
- entity realization (how a bean becomes an entity) 306
- entity vs. instance death 285
- finder methods 276, 339, 341
- home business method 345
- home interface 273, 275
- lifecycle 320
- mapping from abstract schema to a real database 400
- overview 268
- passivation 308, 355
- relationships 378
- unidirectional 391
  - relationship assignments 421
- `remove()` 282, 283, 284
- session remove vs. entity remove 282
- synchronization 298, 301
- environment
  - creating a subcontext 609
  - resource mapping 612
  - subcontexts 604
- environment entries 607
  - scope 608
  - types 608
- `equals()` 150
- exam objectives
  - bean environment 600
  - client view 112
  - EJB overview 2
  - entity bean client view 260
  - entity bean lifecycle 296
  - entity relationships: CMP and CMR 374
  - exceptions 526
    - message-driven beans 438
    - security 570
    - session beans 174
    - transactions 470
  - exceptions 555
    - application exceptions 533, 548
    - client interfaces 159
    - common system exceptions 551
    - `CreateException` 548
    - `DuplicateKeyException` 548
    - `EJBException` 537, 539
    - `FinderException` 538, 548
    - logging system exceptions 535
    - `ObjectNotFoundException` 548
    - `RemoteException` 78, 133, 537
    - `RemoveException` 548
    - scenarios 556
    - system exceptions 534
- exception pathways 530
- exception scenarios 556
- Exercise. *See* Sharpen your pencil
- exotic casting 122. *See* narrow

# F

findByPrimaryKey() 274  
FinderException 538, 548, 559  
finder methods 274, 276, 339, 341  
    Container writes all the code 342  
FROM 411

# G

getCallerPrincipal() 241, 587  
getEJBHome() 137, 241  
getEJBLocalHome() 241  
getEJBLocalObject() 241  
getEJBMetaData() 129  
getEJBObject() 241  
getHandle() 137  
getHomeHandle() 129  
getPrimaryKey() 137  
getRollbackOnly() 241, 488, 500  
getStatus() 489  
getUserTransaction() 241, 476, 477, 478

# H

Handle 139, 149  
handle 149  
home 91  
    business methods 278  
    create() 123  
    EJBHome 128  
    home interface 114  
    methods 129  
home business method 278, 345  
home interface 273  
    EJBHome 128  
    entity bean Remote home interface 273, 279  
    methods 129  
    rules 125  
home object 91

# I

IIOP 90  
IllegalStateException 551, 552, 558, 560  
IN. *See* EJB-QL  
inheritance 186  
InitialContext 117  
input parameters 412. *See* EJB-QL  
INSERT 265

instance-level security 586

interface

- business interface 79
- comparing Remote vs. local 154, 163
- component interface 82, 133, 271
- EJBHome 129
- EJBObject 137
- extending 85
- home interface 91, 114, 125, 273
- javax.ejb.EntityBean 182
- javax.ejb.EntityContext 182
- javax.ejb.MessageDrivenBean 182
- javax.ejb.MessageDrivenContext 182
- javax.ejb.SessionBean 182
- javax.ejb.SessionContext 182
- local client interfaces 145, 166

exceptions 159

- Remote 78
- squeamish about not implementing 86

isCallerInRole 587

isCallerInRole() 587

isCallerInRole(String s) 241

isIdentical 141

isIdentical() 142

isIdentical(Object o) 137

Isolated 472

IS EMPTY. *See* EJB-QL

IS NOT EMPTY. *See* EJB-QL

# J

J2EE Reference Implementation (RI)

- ClientJar 51
- deploytool 32
- JNDI name 52
- New Enterprise Bean Wizard 37
- server 31
- verifier 47

java.net.Socket 625

java:comp/env 602, 603

- subcontexts 604

JavaMail 621

javax.ejb.EJBContext 477

javax.ejb.EntityBean 312. *See also* EntityBean

javax.ejb.SessionBean 182. *See also* SessionBean

javax.ejb.SessionSynchronization 511

javax.jms.QueueConnectionFactory 611

javax.jms.Topic 449

javax.jms.TopicConnectionFactory 611

javax.mail.Session 611

javax.transaction.UserTransaction 471, 477

JAXP 621

JDBC 621

JMS 452, 621

JNDI 92, 602

- bean's special context 189, 602
- InitialContext 117, 602
- lookup 116

narrow the result 120

JNDI API 1.2 621

JNDI name 52

JTA 621

JTS 471

## L

LIKE expression

EJB-QL. *See* EJB-QL

local client view 145

local interface 148, 156

exceptions 159

vs. Remote interface 154

locking (database) 301

logging 535

lookup 116

## M

Mandatory 492

Many-to-Many 380

message-driven beans 106, 437, 442

BMT 460

CMT 460, 500

concurrent processing 444

JMS 452

lifecycle 445

MessageDrivenContext 455, 456

MessageListener 452

message acknowledgement 458, 460

queues 454

rules for writing the class 448

topics 453

transaction attribute 500

MessageDrivenContext 455, 456

MessageListener 452

message acknowledgement 458

methods

afterBegin() 512

afterCompletion() 512

beforeCompletion() 512

begin() 476

create() 128, 274

create methods 159

ejbActivate() 183, 200, 204, 207, 309, 354

ejbCreate() 183, 195, 331, 333

ejbLoad() 354

ejbPassivate() 183, 200, 204, 207, 309

ejbPostCreate() 334

ejbRemove() 213, 214, 216

findByPrimaryKey() 274

getCallerPrincipal() 587

getEJBHome() 137

getEJBMetaData() 129

getHandle() 137

- getHomeHandle() 129
- getPrimaryKey() 137
- getRollbackOnly() 488
- getStatus() 489
- getUserTransaction() 476
- isIdentical(Object o) 137, 141
- remove() 137, 282
- remove(Handle h) 129
- remove(Object key) 129
- remove methods 159
- setRollbackOnly() 485
- setSessionContext() 183
- setSessionContext(SessionContext sc) 195
- method permissions 578
  - <unchecked/> 581
  - union 580
- missed `ejbRemove()` in session beans 213, 215
- mock exam. *See* Coffee cram
- Movie 378
- MovieBean code 396
- MovieHomeRemote interface 398
- MovieSchema 408
- multiplicity 380, 387, 422
  - affects return type 382
  - Many-to-Many 380
  - One-to-Many 380
  - One-to-One 380

## N

- narrow 120
  - PortableRemoteObject.narrow() 121
- network transparency 65, 66
- Never 492, 495
- NoSuchEntityException 551, 552
- NoSuchObjectException 551, 553
- NoSuchObjectLocalException 551, 552, 553
- NotSupported 492, 495
- NOT IN. *See* EJB-QL

## O

- objectives. *See* exam objectives
- ObjectNotFoundException 548, 553, 559
- ome business method 346
- One-to-Many 380
- One-to-One 380

## P

- passing objects remotely 68
- passivation
  - stateful session bean 200
  - transactions 206

- pass by copy 69
- pass by reference 68
- path navigation 404
- persistent field 313
  - virtual 315
- point-to-point 451. *See also* queue; *See also* topic
- PortableRemoteObject.narrow() 121
- primary key 332, 333
- Principal 584
- programmatically security 586, 589
- programmatically transactions. *See* BMT
- programming restrictions 625
- propagation
  - transactions 474, 482
- proxy. *See* stub
- publish/subscribe 451

## Q

- queries 406. *See* EJB-QL
- queues 450, 454

## R

- relationships 378, 394
  - unidirectional 391
- relationship assignments 421
- Remote 65
  - arguments 67, 72
- Serializable 72
  - business interface 79
  - component interface 82
  - home interface 125
  - interface 78
- PortableRemoteObject.narrow() 121
- RemoteException 78
  - return values 67, 72
- Serializable 72
  - vs. local interface 154
- RemoteException 78, 133, 537, 559
- Remote home interface 125, 279
  - EJBHome 128
- Remote interface 78, 144
  - component interface 82, 137, 271
  - home interface 125
- EJBHome 128
  - vs. local interface 154
- Remote Method Invocation 63, 64
- Remote object 65, 67
  - EJBObject 82
- remove 129, 151, 159, 282

- entity beans 284
- entity vs. session bean 282, 283
- remove() 137, 151, 208, 282, 286
  - entity vs. session bean 283
- remove(Handle h) 129
- remove(Object key) 129
- RemoveException 548, 559
- remove methods 159
- Required 492
- RequiresNew 492
- resource manager connection factory 610
  - resource authorization 611
- return values 67
  - Serializable 72
- RMI 63
  - arguments 67
  - how its used by EJB 81
  - IIOP 90
  - PortableRemoteObject.narrow() 121
  - RemoteException 78
  - Remote object 65
  - return values 67
  - RMI-IIOP 90
  - rmic 90
  - skeleton 65, 67
  - stub 64
- rmic 90
- Role 584
- role reference 589
- rollback 473

- rollback() 478
- RPC 67
- run-as 591
- RuntimeException 545

## S

- scalable
  - stateless session beans 103
- security
  - App Assembler 574, 576
  - authentication 572
  - authorization 572
  - context propagation 592
  - declarative security 572
  - Deployer 583
  - identity 591
  - instance-level security 586
  - isCallerInRole() 587
  - Principal 584
  - programmatic security 586
  - roles 574
  - role definition 576
- security identity 592
- security mapping 590
- security roles 574, 576
  - <method-permission> 578
- SELECT 265, 402, 406

- Serializable 67, 72
- server 31
- Server dude at the pool 441
- SessionBean 182, 183
- SessionContext 189, 199, 240, 241
- SessionSynchronization 511, 512
  - callbacks 512
- session beans 182
  - bean lifecycle 180, 184
  - container callbacks 182
  - implementing 183
  - missed `ejbRemove()` calls 215
  - overview 98, 100
  - passivation 200, 207
  - removal 208, 210
  - scalability 103
  - SessionBean interface 182
  - stateful session beans 178, 184
  - creation 179, 192, 196
    - OID for bean creation 194
  - passivation/activation 207
  - removal 210
    - stateless session bean 102, 222
  - creation
    - OID for bean creation 227
- session context 240
- `setEntityContext(EntityContext ec)` 314
- `setRollbackOnly()` 485, 487
- `setRollbackOnly(boolean b)` 241
- `setSessionContext(SessionContext ctx)` 183
- `setSessionContext(SessionContext sc)` 195
- Sharpen your pencil
  - APIs 621
  - architectural overview 107
  - bean table 108
  - bean types 13
  - client view exercise 165
  - CMP entity creation 359
  - compiler checked? 447
  - EJB-QL 407
  - EJB architecture 10, 80
  - EJB Lifecycle 96
  - EntityBean interface 323
  - entity bean 280
  - entity bean callbacks 353
  - entity bean class methods 361
  - entity bean relationships (Director-to-Movie) 390
  - entity business method 360
  - entity CMP code annotation 319
  - entity lifecycle activity 324
  - environment 626
  - exceptions 561
  - exception exercise 562
  - exception scenarios 557
  - home interface 132
  - local and Remote client interfaces 155
  - local clients 158
  - local client view 147
  - message-driven bean 461
  - message-driven bean context methods 455

- Multiplicity and Assignments 423
- package name 119
- security roles 575
- SessionBean interface 186
- session bean overview 104
- stateful session bean state diagram 187
- stateless to stateful 220
- transactions 480
- transactions puzzle 514
- transaction attributes 496, 497, 507
- UserTransaction 484
  - write an entity bean 317
- skeleton 65, 67
- squeamish 86
- stateful session bean 178
  - creation 179, 194, 196
  - Object Interaction Diagram (OID) for bean creation 194
  - removal 210
  - SessionSynchronization 512
- stateless session
  - transactions in 481
- stateless session bean 102, 222
  - business methods 226
  - creation 224
  - Object Interaction Diagram (OID) for bean creation 227
- String literals. *See* EJB-QL
- stub 64, 67
- subcontexts 604

- supported APIs 621
- Supports 492, 495
- suspended transactions 483
- synchronization 298
- system exceptions 500, 551
  - logging 535

## T

- The Advice Guy 15
- Tikibean Lounge
  - BMT/CMT 501
  - EJBObject/bean 89
  - session bean/bartender 14
- topics 450, 453
  - durable subscription 450
- topics and queues 450
- TransactionRequiredException 551, 552
- TransactionRequiredLocalException 551, 552
- TransactionRolledbackException 551, 552
- TransactionRolledbackLocalException 551, 552
- transactions 471, 473. *See also* commit; *See also* roll-back; *See also* setRollbackOnly()
  - Bean-managed transactions 476
  - begin() 476
  - BMT 476, 481
  - call stack example 479
  - database locking 301

- deployment descriptor 503
- getRollbackOnly() 488, 500
- message-driven beans 460, 500
- propagation 474, 482
- SessionSynchronization 512
- stateful session bean passivation 206
- suspension 475, 483
- transaction attributes 492
- two-phase commit 473
- unspecified transaction context 499

transaction attributes 492, 493, 498

- attributes that require a transaction 494
- Mandatory 492
- Never 492, 495
- NotSupported 492, 495
- Required 492
- RequiresNew 492
- Supports 492, 495

Tutorial 28

two-phase commit 473

## U

- unidirectional 391
- unsetEntityContext() 314
- unspecified transaction context 499

UPDATE 265

Users 584

UserTransaction 476, 477

- getStatus() 489

## V

- vendor lock-in 5
- verifier 47
- virtual persistent fields 315

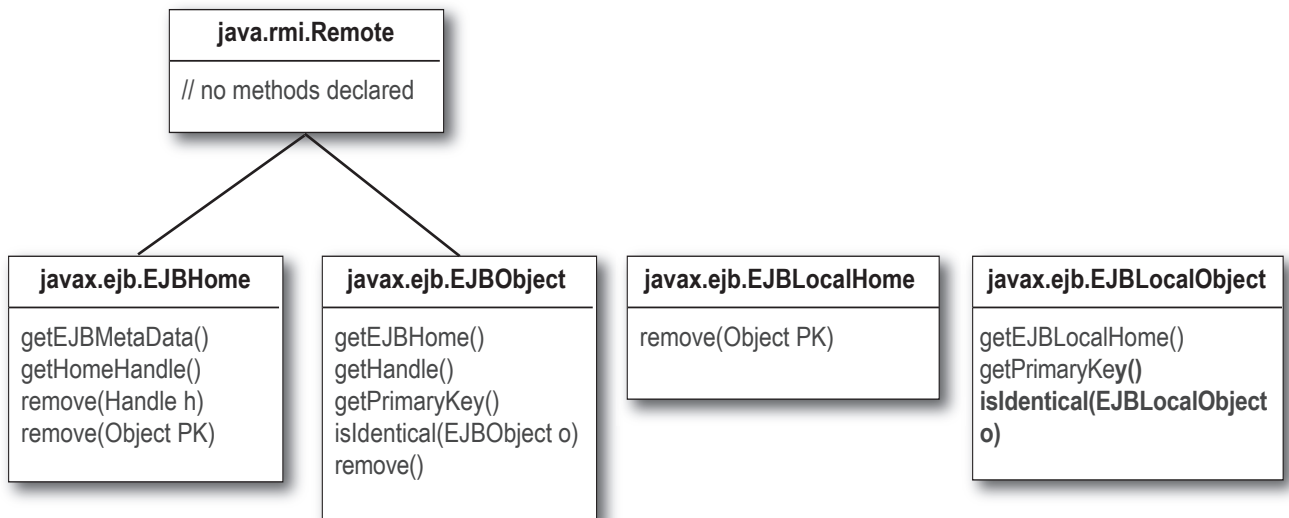
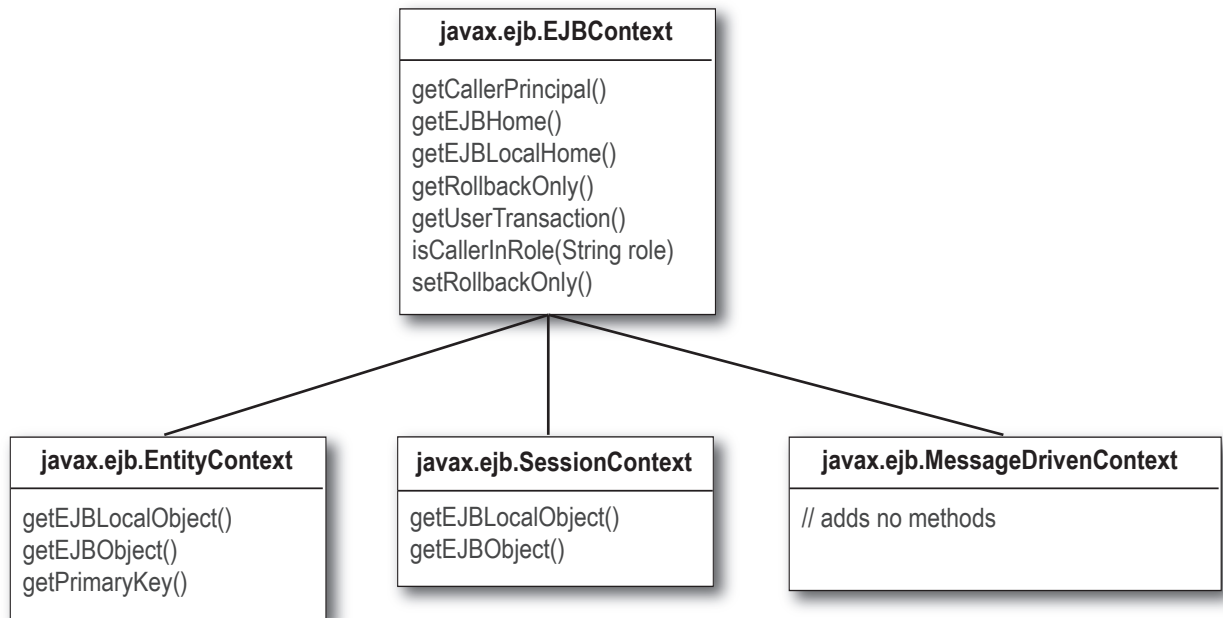
## W

WHERE 403, 411, 412

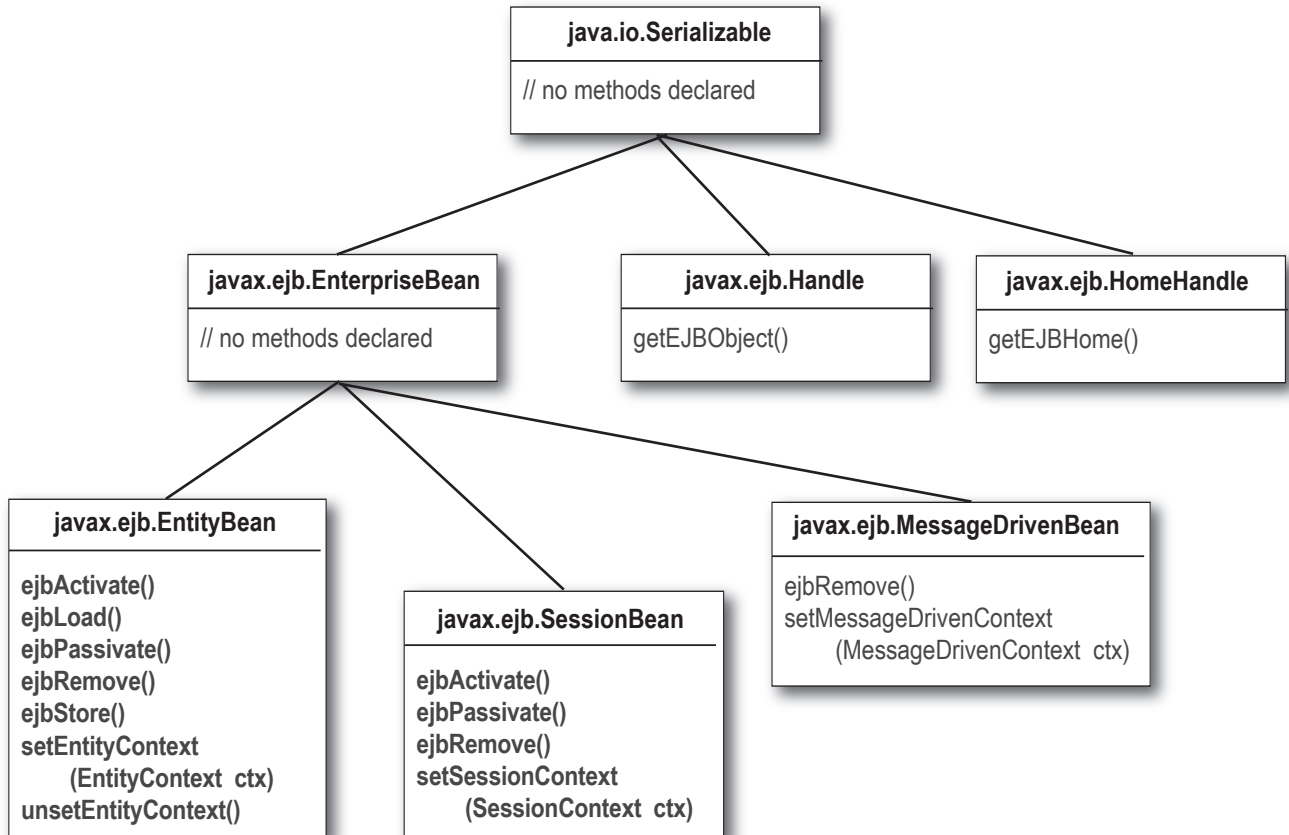
wildcard

- method elements 504
- transaction attributes 504

# Interface summary



# Interface summary



# Interface summary

<b>javax.transaction.UserTransaction</b>
begin() commit() getStatus() rollback() setRollbackOnly() setTransactionTimeout(int seconds)



*setTransactionTimeout() is not on the exam...  
it lets you tune the elapsed time before the  
Container decided the transaction has timed  
out, and automatically rolls it back.*

<b>javax.ejb.SessionSynchronization</b>
afterBegin() afterCompletion( boolean committed) beforeCompletion()

<b>javax.jms.MessageListener</b>
onMessage(Message message)