INDEX

Symbols

.NET 307, 309

Numerics

4+1 view model 655, 656

A

abstract class 260, 520
  in UML 250, 260
abstract conceptual class 519
Abstract Factory 597
abstract operation 260
  in UML 250, 260
abstract use case 497
action 477
activation bar 228
active class 239, 269
active object 239, 269, 584
activity diagram 477, 478
actor 63, 80
  in use case 66
  offstage 66
  primary 66
  supporting 66
Adapter 436
adaptive development 21
adaptive vs. predictive planning 674
addition use case 497
aggregation 264
agile methods 17, 27, 28, 326, 673
agile modeling 14, 30
agile UP 31
analysis 6
analysis and design
  definition 6
analysis object models 134
architectural
  analysis 543, 544
  baseline 128
  decisions 544
  design principles 553
  factors 544
  patterns-promotion of 555
  proof-of-concept 557, 669
  prototype 128
  synthesis 557, 669
  view
    data 658
    deployment 658
    implementation 658
    logical 657
    process 658
    security 658
    use case 659
architectural approach documents 550
architectural factors 545
architectural view 656
architecturally significant requirements 545
architecture 200, 559
  cross-cutting concerns 554
  documenting 655
  factor table 546
  issue cards 550
  layered 202
  logical 197
  separation of concerns 554
  technical memos 550
artifact
  UML deployment diagrams 623
artifacts 34
  organizing 680
ASP.NET 307
aspect-oriented programming 555
association 150
  criteria for useful 150
  end name 255
  finding with list 155
  for UML properties 252
  link 240
  multiple between types 155
  multiplicity 153
  naming 152
  qualified 531
  reflexive 532
  role names 528
  UML notation 151
association class 266, 523
asynchronous message 588
  in communication diagrams 246
  in sequence diagrams 238
attribute 158
  and quantities 165
  data type 162
  derived 160, 530
  for UML properties 252
  in UML 250
  no foreign keys 165
  non-primitive types 163
  UML notation 159
  valid types 160

UML and Patterns.book Page 696 Thursday, September 16, 2004 9:58 PM

B

base use case 497
behavioral decomposition 424
benefits of iterative development 22
black-box use cases 81
Booch Method 194
boundary objects 307
brief use case 66
Business Modeling discipline 34, 483
Business Object Model 134, 484
business rules 117
Business Rules artifact 58, 102
business use case 73

C

CASE tools 398
casual use case 66
Change Cases 432
class
  abstract 520
  association 523
  conceptual 14, 136
  conceptual & abstract 519
  definitions 13
  design 14
  hierarchy 404, 510, 522
  implementation 14
  in UML 226, 250
  mapping from DCD 371
  partition 514
  partitioning 514
  software 14
  UML meaning 13
  UML notation 226
class diagram 133, 249, 250
class diagrams 249
class hierarchy 404, 510
class method calls 236, 245
class operation
  in UML 250
classifier 14
  in the UML 251
classifier name
  in UML 250
client-driven development 27
COCOMO II 180
code
  mapping OO designs 369
code smells 390
cohesion 290, 314
collaboration
  in RDD 277
collaboration diagram
  conditional messages 243
  example 224, 225
  instance creation 241
  iteration 244
  links 240
  message sequencing 242
  message to self 241
  messages 240
  mutually exclusive conditionals 244
  sequence number 242
collaborations
  of objects in RDD 276
collection
  in OO programming 374
  in UML 250
  iteration over in UML 233
  n UML 245
Command pattern 610, 645
Command-Query Separation Principle 358
comment
  in UML 256
communication diagram 223
  iteration over a collection 245
  message to class object 245
  strength and weaknesses 223
communication path
  in deployment diagram 623
compartments
  in class box 251
  in class diagrams 268
component 624
component diagram 623
component-based modeling 623
Composite 452
composite aggregation 264, 282, 525
composition 264, 293, 525
concept
  extension 136
  finding with noun identification 141
  intension 136
  mistake while finding 146
  specification or description concepts 147
  symbol 136
  versus role 529
conceptual class 14, 136
  abstract 519
conceptual model 8, 134
conceptual object model 9
Concrete Factory 440
concrete use case 81, 497
conditional messages in sequence diagrams 231
constraint
  in note symbols 256
  in UML 265
  UML 192
construction phase 33
constructors
  in UML 250
container (in Decorator pattern) 555

UML and Patterns.book Page 697 Thursday, September 16, 2004 9:58 PM

context diagram 64, 90
continuous integration 18
contract
  example 183
  guidelines 189
  postcondition 184
  section descriptions 183
contracts 181
control objects 307
Controller 302
  application 328, 331, 351
  bloated 311
  definition 302
Convert Exceptions 587
coupling 285, 299
  relation to dependency 261
create
  stereotype on dependency line 360
Creator 281, 282, 291, 330
  application 330, 340
cross-cutting concerns 554
CruiseControl
  build-test tool 387

D

data dictionary 58, 115
data flow diagrams 479
data holder objects 571
data model 136, 629
data modeling 629
data type 161, 162, 254
data view 658
Database Broker pattern 632
Database Mapper pattern 632
database mapping 625
data-driven design 442
datastore node 480
decision symbol
  activity diagrams 481
delegate 287
delegation 287
Delegation Event Model 463
dependency 201, 260
deployment architecture 199
deployment diagram 621
deployment view 658
derived attributes 160
description class 147
design 7
Design by Contract 194
design class 14
design class diagram 10, 251
Design discipline 34
Development Case 37
device node 622
diagram fragment 230
diagram frames 230
discipline 34
  and phases 35
Do It Myself 297, 600
documenting architecture 655
domain layer 136, 206, 282, 287
Domain Model 134
  domain vocabulary 134
  finding concepts 139
  map-maker strategy 145
  modeling changing states 521
  modeling the unreal 146
  multiple meanings 136
  organizing in packages 533
domain model 131
domain object models 134
domain objects 136, 206, 287
domain rules 58, 117

E

eager initialization 444
EBP 88
Eclipse 387
EJBs
  in Controller pattern 308
elaboration phase 33, 123, 127
elementary business process 73, 88
entity objects 307
Environment discipline 34
error definition 586
essential use case style 80
estimates 678
estimation 180, 678
event 486
Evo 545
evolution point 432, 542, 552
evolutionary development 17, 18, 19
exceptions
  in class diagrams 250, 256
exceptions in UML 588
executable architecture 128
execution environment node 622
execution specification 228
Expert 283, 294
  application 332, 335, 336
extend use case relationship 497
extension 136
extension point 498
Extreme Programming 18, 40, 280

F

Facade 461
factor table 546
Factory 440

UML and Patterns.book Page 698 Thursday, September 16, 2004 9:58 PM

failure definition 586
fault definition 586
feature of system 112
features 58, 108, 112
final class
  in UML 260
final operation
  in UML 250, 260
fixture
  in testing 387
focus of control 228
fork 477
formal specifications 193
forward engineering 398
found message 228
frames 230
  opt, loop, alt, ref operators 230
framework 627
  persistence 627
frameworks 625
fully dressed use case 67
function points 180
functional requirements 57
Fusion method 194

G

Gang of Four patterns 280, 436
generalization 260, 404, 510
  abstract class notation 520
  and conceptual class sets 512
  and conceptual classes 511
  conformance 512
  in UML 260
  overview 509
  partitioning 514
  subclass validity tests 513
  UML notation 510
generalize
  use case relationship 499
generics 267
Glossary artifact 58, 102, 115
GoF design patterns 280
GRASP patterns 271, 277, 321
  Controller 302
  Creator 291
  Expert 294
  High Cohesion 314
  Indirection 426
  Low Coupling 299
  Polymorphism 414
  Protected Variations 427
  Pure Fabrication 421
guarded methods 250, 639

H

Hibernate 626
High Cohesion 314
Hollywood Principle 627

I

IDEs 387
implementation 34
implementation class 14
Implementation Model 369, 614, 658
implementation view 658
inception phase 33, 47, 126
include use case relationship 494
Indirection 426
Information Expert 283, 294
information hiding 433
inheritance 522
  in UML 260
initial domain object 348
instance
  in UML 623
  UML notation 226
intension 136
interaction diagram 221, 222
  class 226
  instance 226
  message syntax 226
interaction fragment 230
interaction frames 177, 230
interaction occurrence 235
interaction overview diagram 222, 235
interaction use 235
interface 192
  in UML 250, 263
  when to use 420
interface realization
  interface implementation 263
issue cards 550
iteration over a collection in UML 245
Iteration Plan 32, 676
iterations 19
iterative development 17, 18, 19
  benefits 22
  planning 673
iterative lifecycle 19

J

join 477
JUnit 387

UML and Patterns.book Page 699 Thursday, September 16, 2004 9:58 PM

K

keyword
in UML 258

L

Law of Demeter 430
layer 199
layered architecture 199, 202
lazy initialization 444
lazy materialization 647
lifeline 227
UML interaction diagrams 226
link 240
Liskov Substitution Principle 429
logical 559
logical architecture 197, 199, 559
logical view 657
lollipop notation 264
looping
sequence diagrams 177, 231
Low Coupling 284, 299
low representational gap 138, 281, 330
lower representational gap 11
LRG
low representational gap 281, 330
LSP 429

M

MDA 11, 12
merge symbol
activity diagrams 481
message
asynchronous 588
UML notation 228, 240
metaclass 236
metadata 116, 633
method 192, 257
from collaboration diagram 372
in note symbols 256
in UML 250
Model Driven Architecture 11, 12
Model-View Separation 209, 331
Model-View-Controller 209, 307
modular designs 317
Moment-Interval 528
multiplicity 153, 255
MVC 209, 307

N

N+1 view model 655, 656
namespace
in UML packages 201
navigability arrow
in UML class diagrams 253
navigation model 490
node 622
non-functional requirements 57
Supplementary Specification 104
note
in UML 256
NUnit 387

O

object
active 584
in UML 226
persistent 626
Object Constraint Language 265
object databases 626
object design 7, 271, 321, 363
introduction 213
object node 477
object-oriented analysis 7
object-oriented analysis and design
definition 7
dice game example 8
object-relational mapping 625
Observer 463
OCL 192, 265
offstage actor 66
Open-Closed Principle 434
operation
in UML 250, 257
operation contracts 181
operations 192
ordered
keyword in UML 255
organizing artifacts 680

P

package
dependencies 533
design 613
organization guidelines 614
ownership 533
reference 533
package attributes
in UML 250
package diagram 197, 199, 201
package name
in UML classifier name 250
parameterized types 267
parameters
in class diagrams 256
participants
in interaction diagrams 226

UML and Patterns.book Page 700 Thursday, September 16, 2004 9:58 PM

partition 477
path name 564
pattern 4, 278, 279
  Abstract Factory 597
  Adapter 436
  Command 610
  Composite 452
  Controller 302
  Convert Exceptions 587
  Creator 291
  Do It Myself 297, 414, 600
  Expert 294
  Facade 461
  Factory 440
  High Cohesion 314
  Indirection 426
  Layers 202
  Low Coupling 299
  Model-View Separation 331
  names 279
  Observer 463
  Polymorphism 414
  Protected Variations 427
  Proxy 591
  Publish-Subscribe 463
  Pure Fabrication 421
  Redirection Proxy 592
  Remote Proxy 591
  Singleton 442
  State 191
  Strategy 447
  Template Method 634
  Virtual Proxy 648
patterns
  architectural 555
  history 280
persistence 625
persistence framework 627
  key ideas 628
  materialization 634
  pattern-Cache Management 641
  pattern-Object Identifier 630
  pattern-Representing Objects as Tables 629
  representing relationships in tables 651
  requirements 628
persistent objects 626
Petri nets 484
Phase Plan 32, 676
phases in UP 33
physical architecture 621
physical design 614
PLanguage 545
planning
  adaptive 674
  iterative 673
polymorphism 414
  in communication diagrams 246
  in sequence diagrams 237
Polymorphism pattern 414
  for payments 600
postcondition 184
  a metaphor 186
  in use case 74
precondition
  in use case 74
primary actor 66
Principle of Least Surprise 359
private attributes
  in UML 250
process
  iterative 19
process view 658
profile
  in UML 11, 259
project management 673
property
  in UML 250, 260
  multiple meanings in UML 252
  structural, in UML 252
property string
  for association ends 255
  in UML 260
property strings
  in class diagrams 256
protected attributes
  in UML 250
Protected Variations 427
Proxy 591
  Virtual Proxy 648
Proxy pattern 647
public attributes
  in UML 250
Publish-Subscribe 463
Pure Fabrication 421

Q

qualified association 265, 531
qualifier 265, 531
quality attributes 57, 107
quality scenario 545

R

rake symbol
  activity diagrams 481
Rational Unified Process 18
RDD 271, 276, 321
read-only attributes
  in UML 250
Redirection Proxy 592
ref
  tag in frames 235
ref frames 357

UML and Patterns.book Page 701 Thursday, September 16, 2004 9:58 PM

refactoring 389
reference
in interaction diagrams 235
reference attribute 450
reflexive association 532
relational cohesion 615
relational databases 626
Remote Proxy 591
replicates 571
representational decomposition 424
required interface
in UML 263
requirements 54
functional in Use-Case Model 64
in the UP 58
non-functional in Supplementary Specification 104
overview 53
various types 101
Requirements discipline 34
responsibilities 271, 276, 321
and interaction diagrams 277
and methods 276
doing 276
importance of 6
knowing 276
patterns 278
responsibility-driven design 276
return in sequence diagram 229
return type
in class diagrams 256
in UML operations 256
reverse engineering 398
risk-driven development 27
role 153
of objects in RDD 276
versus concept 529
rolename 255
in DCDs 253
round-trip engineering 398
rules 58
business or domain 58
domain or business 117
RUP 18

S

SAD 656
scenario 63
schema mapping 628
Scrum 18, 40
sd frames 357
frame tag 235
security view 658
selector
in lifeline box 234
separation of concerns 204, 441, 554
sequence diagram 176, 222
activation box 228
conditional message 232
conditional messages 231
instance creation 229
iteration over collection 233
lifelines 230
looping 231
loops 177
message to class 236
message to self 229
messages 228
mutually exclusive conditional 232
object destruction 230
return 229
strength and weaknesses 223
Servlet 308
Session objects
in EJB and Controller pattern 308
Singleton 442
UML shorthand notation 444
socket notation 264
software architecture 200
Software Architecture Document 557, 655, 656
software class 14
software development process 18
specialization 404, 510
state 486
modeling 521
state machine diagram 486
example 491
guard conditions 489
nested states 489
overview 485
transition actions 489
State pattern 191, 642
statechart diagram
nested states 489
state-independent 487
static method calls 236, 245
static operation
in UML 250
stereotype 91, 241, 259
Strategy 447
structural properties
in UML 252
Struts
Java Struts example 310
subclass
conformance 512
creating 514
in UML 250, 260
partitioning 514
validity tests 513
superclass
creating 516
Supplementary Specification artifact 58, 102, 104, 107, 557
supporting actor 66

UML and Patterns.book Page 702 Thursday, September 16, 2004 9:58 PM

SWEBOK 59
Swing
  Java Swing example 309
symbol 136
synchronized method 250, 639
system behavior 176
system events 176, 184
  naming 178
system feature 112
system interface 184
system operation 176, 183, 302
system sequence diagram 173, 176
  showing use case text 178
system use case 72

T

tagged value 241
tags
  in stereotypes 259
technical memos 550
Template Method 634
templates 267
templatized types 267
test fixture 387
test-driven development 18, 29, 376, 386
test-first development 386
  see test-driven development 376
testing 386
threads 238
threads in the UML 584
three-tier architecture 574
tier 207
time intervals 527
timeboxing 23
transition 487
transition phase 33

U

UI modeling
  navigation modeling 490
UML 11
  class diagrams 250
  constraint 192
  Data Modeling Profile 629
  meta-model 11
  overview 11
  profiles 11, 629, 630
  stereotype 241
  tagged value 241
  UML 2 14
Unified Process 18
unique
  keyword in UML 255
unit testing 386
UP 18
  agile 31
  phases 33
use case 61, 63
  abstract 497
  actor 66
  addition 497
  and development process 95
  base 497
  black-box 81
  brief 66
  business 73
  casual 66
  concrete 497
  essential style 80
  extend 497
  fully dressed 67
  include 494
  instance 63
  postcondition 74
  precondition 74
  subfunction level 73
  system 72
  user goal level 73
  when create abstract use cases 497
use case realization 273
use-case driven development 95
Use-Case Model 58, 64
use-case realizations 95, 321, 322
use-case view 657, 659
user task 88

V

value objects 163, 571
variation point 432, 542
VDM 194
version thrashing 613
Virtual Proxy 648
virtual proxy 647
visibility 333, 356, 363, 364
  attribute 365
  global 367
  in class diagrams 253
  in UML 250, 251
  local 366
  parameter 366
Vision artifact 58, 102, 109, 111
visual modeling 14

W

waterfall 18, 23
WebForms 307, 309
whitebox frameworks 634
WinForms 309