Index

A
abstract class 71, 82, 234, 309, 332
abstract colleague decorator class 72
abstract mediator class 72
abstract syntax tree (AST) 215
abstraction 289
abstraction by default 328
account maintenance charge 330
ACM Digital Library 295
action definitions 228
action transformation model 43
adapting the rules 136
affected properties 83
aggregation 65, 38, 164
analysis 16, 17
analysis of pattern descriptions 147
analysis or interpretation 25
antipatterns 205
antisymmetric 67, 68, 81
application of the template method pattern 82
applying a quality model 111
applying the rules 136
assessing the quality of motifs 118
assessment 113, 137, 293, 295
associated object creation 79, 80
associated object creation implementation 86
associated object destruction 81
association 62, 68, 74, 164
association ends 65, 71, 73, 74
association relationships 61, 64
association/aggregation 65
automatic deduction of interactions 48
automatic verification 143
automation 133, 241, 256, 261

B
bad smells 5, 193, 204, 295, 318, 341, 343
Bansiya and Davis’s QMOOD 299
basics 228
BBDD scheme 336
behavior 24, 31, 36, 41, 58, 70, 72
behavior analysis 40
behavior specification 25, 35
behavioral view 74, 79
best practices 5, 9, 11, 294
black box 315
bloaters 205
bounds 228
building a quality model 109, 115
building numerical signatures 122
built-in primitives 62
business elements 338
business process profile 177
C

calls to methods 334
cascade 68, 69, 81, 89, 92, 97
case study 183, 237
catalog of design rules 307
change in an interface 324
changed process 180
chaos 2
characterizing pattern models 150
choosing internal attributes 117
classes and methods 83
classification 26, 65, 160, 202
client relations 228
CMMI 283, 286
cohesion 265, 291
collaborations 311
collapse hierarchy 317
collecting parameters 234
COM 312
command 124, 127, 234
communication 10, 36, 172
complex programming 340
complexity 291
component collaboration architecture (CCA) 177
component initialisation 179
composite 134, 135, 163, 167, 317
composition 65, 291
computational linguistics 31
computing metrics 118
conceptual framework 59, 63, 65, 77, 101
conceptual models 60, 61, 64, 74, 75, 101
conceptual perspective 36
conclusion validity 18
concrete class 329
concrete elements 308
concrete superclasses 332
conflict of roles 331
consistency 25

constraint resolution 115, 131
coordinator component 170, 172
CORBA 312
correction 219
coupling 265, 291
creating instances 312

D
data class 238
data extraction 297
data mining techniques 218
declarative knowledge 11
decorator 59, 71, 82, 317
decorator class 71
decorator colleague class 72
decorator design pattern 158, 159
definition level 43
definition of the sequence of calls 82
delete propagation 81
deletion of instantiated objects 179
dependencies 308, 310
dependency analysis 273
dependency inversion principle (DIP) 308
DePMoVe (Design and Pattern Modeling and Verification) 144
depth inheritance tree (DIT) 236
description 44, 198
design 2, 9, 10, 13, 24, 56, 63, 72, 101, 106, 120, 125, 144, 233, 284, 288, 307
design classes 74
design motif identification 114
design patterns 70, 144
design quality assessment models 294
design size 289
design structure 155
design structure component 177
design structure load 179
design structure verification 179
destination 45
destruction 79, 80
detection strategies 302
different behavior 312
directionality of associations 61
discussion 112, 136, 201
documentation of components 177
domain classes 71
domain entities 337
dynamic 77
dynamic behavior 31, 34, 54
E
ease of use 25
effectiveness hypotheses 16
efficiency 286
EJB 312
empirical reality 258
empirical studies 19, 254, 255
empirical validation 14
encapsulate field (EF) 253, 254, 260, 273
encapsulation 291
engine framework extension 232
entities 11
entity profile 177
equality 49, 176
equivalent 49
evolution studies 219
evolution-sensitive 212
examples 77, 198, 201
experimental evidence 271
external validity 19
F
family of languages 217
form template method 13
formal basis of OOD 154
formal languages 151, 153
formal models of patterns 153
formal specifications 148
formalization of patterns 158
Fowler’s catalog 198
frameworks 69, 71, 216, 236
functionality 286
functions 176
fuzzy reasoning 115
G
generic parameters 228
genericity 62, 70
goal 82, 283, 301
goal, question, metrics (GQM) 206
graphical user interface (GUI) 338
H
Henderson-Sellers metrics 291
heuristic-oriented techniques 27
heuristics 5, 250, 251, 275, 295, 301
hierarchies 163, 289
hierarchy of classes 315
high level indicators 295
human factor 108
hypotheses formulation 16
I
identification projection 78
identifying micro-architectures 135
identifying programs with motifs 118
identifying roles 121
idiom 4
IEEE Digital Library 295
IF a superclass is a concrete class 330
impact 324
implementability 64
implementation strategy 82
implementation techniques 57, 62
independent variables 116, 209
inheritance 291
inheritance relationships 228, 315
initiator 45
instantiation component 171
interaction comparison 48
interaction fragment 37, 41
interaction linguistic patterns 42
interaction modelling techniques 26,
Index 359

30, 32, 34
interaction unit 41
interactions 83
internal state 312, 313
internal validity 18
interpretation 17
introduction of abstractions 311

J
Java Parser 171

K
knowledge 2

L
Lack of Cohesion of the Methods of a class LCOM 265
language dependent 230
language independent bad smell detection 219
language independent refactoring 198, 226
large class 335
lazy class 238, 308, 318
lessons learned 295
lines of code (LOC) 208
linguistic patterns 23
linguistics-oriented techniques 29
link 81
link creation 79, 80, 81, 83
link destruction 84
lists of parameters 334
little used items 317
location 45
logic metaprogramming 216
loose coupling 70

M
manual deduction of interactions 48
mapping association elements 74
MDSD 63
measuring internal attributes 135
mechanics 198, 253
mediator 59, 71, 82
messaging 291
meta level interface (MLI) 217
metaclasses 63
metamodel traversal 233
metamorphic interactions 34
metamorphosis interaction 41
method 295
metric calculation 236
metric collection 216
metric customization 234
metric extraction 123
metrics 206, 250, 251, 256, 276, 293, 295
metrics analysis 218
metrics in software refactoring 210
metrics smoothing 193
micro-architecture 307
middle man 318
minimal object-oriented notation (MOON) 221
minimal set of methods 165
model interactions 23
model of actions 43
model-driven architecture (MDA) 57
model-driven software development (MDSD) 57
modelling techniques 26
monitoring growth 270
MOOSE 216, 226, 227
motivation 198, 309
move field 260
moving member variable to subclasses 230
multiplicity 80

N
name 44
navigability 61, 66, 78
new alternatives 32
no verifiable 230
normalization function 45
not navigable 79
not reflexive 69, 80
number of attributes (NOA) 215, 238, 256
number of children (NOC) 125, 236, 237, 292
number of methods (NOM) 125, 238, 272

O
object declaration 176
Object Management Group (OMG) 57
object-orientated design knowledge 1, 8
object-oriented design quality improvement 282
object state 45
object-oriented (OO) 3, 4, 10, 205, 253, 287, 289, 295
object-oriented analysis/design (OOAD) 4
object-oriented programming languages (OOP) 149
observations 44
ontology 10
OO abusers 205
OO design 3, 4, 295, 307
OO design maturity model 284
OO knowledge
OO metrics 209, 291
OOD metamodel 155
OOD metamodel component 171
OO micro-architectural design 1, 10
OO micro-architectural design knowledge 10
OO micro-architectures 307
OO refactoring 250
OO software development 196
OOD pattern applications 143
OODSLA 2001 Workshop 5
operative knowledge 11
OptimalJ 60
owner 45

P
parallel inheritance hierarchy 239
participating classes 74
particular semantic interpretation 68
pattern application level 44
Pattern and Abstract-level Description Language (PADL) 124
pattern instantiation approach 149
pattern metamodel component 173
pattern specification level 43
pattern-based design 143, 146, 180
patterns 4, 13, 149, 154, 295, 308, 326, 335
performer 45
platform independent models (PIM) 57
polymorphism 71, 274, 291
portability 287
precision 64
preconditions 198
preconditions analysis 201
primitives, operators, metrics (POM) 118, 124, 135
principal verb 45
principles 5, 295
process of verification 173
program architecture 108
program slicing 197
program transformation 195
pull up field 261, 264
push down field 261, 329

Q
quality 295
quality characteristics 116
quality factors 70
quality model 107, 113, 116, 135, 206
quality model for object-oriented design (QMOOD) 299
quality theory 108
quantified expressions 176
quantitative evaluation 115

R

recursion 177
refactor 251
refactoring 4, 193, 195, 206, 251, 267, 295, 308, 326, 335
refactoring engine 230
refactoring framework 223
refactoring repository 231
refactoring support 203
refactoring theory 253
references 62
reflexivity 80
refused bequest 343
relation profile 177
relationship 13, 74
relationship objects 62
reliability 286
rename field 261, 274, 275
renaming 157
reorganizing 195
repository creation 122
representation richness 25
restrictive 81
reusability 70
reused based on object-oriented techniques (REBOOT) 207
review methods 295
rule learning 125
rules 13, 295

S

scalability 332
schema declaration 176
selected design patterns 70
semantic context 40, 44
semantic fragment 44
semantic perspective 38
separate domain from presentation 340
separated by a bridge 144
separation of concerns 70
shotgun surgery 308
simplicity 64
single responsibility principle 335
software elegance 107
software product quality 285
software quality assurance 299
special sentences 37
specialization 65
specific modules 340
speculative generality 318
SPICE 283
static 77
strategies 301
structural view 74
structured expressions 176
subclasses 160, 230, 254, 264, 320
subjects 19
superclass 320, 329
surveys 202
SWEBOK 2, 10
swing class library 259
switch statements 239
symmetry 69, 81
syntactic normalization 38
syntactic perspective 37
system classes 337
system sequence diagrams 28

T

tabular analysis method (TAM) 29
taxonomies 202
template method 59, 71, 234
temporal behavior 77
threats to validity 18
Tichelaar, S. 198
traceability 25, 137
transformation patterns 48
transformation rules 44
transformation strategy 47
translation of RSL structures into Java 174
type declaration 176
type equivalence 175
types and generic instantiations 228

U
UML association concepts 63
UML based metamodel 224
unidirectional 37
unidirectional action 37
unification 5, 114
usability 286
user interface 183, 187, 337, 339
using an observer 144

V
validating the quality model 119
validation 125, 208
validation experiment 47
variations 311
verification layer 170

W
“what” of refactoring 257
white box 315
WorkFor relationship 73

Z
zero attribute classes 259