A
abstract classes 137
abstract operation 137
abstract syntax 129, 133
account 52
activity-dimension 238
activity diagrams 12, 294, 315
activity graph 298
actor stereotypes 361
aggregation 131, 225
algebraic specification 51, 200
alternative tasks 326
alternatives 242
analysis and design 2
analysis models 241
applicability conditions 247
application programming interfaces (APIs) 232
approval concern 344
architectural design 254
architectural planning 255
architecture-based approach 377
artifact-dimension 238
artifacts 240
aspect-oriented programming 376

asset concern 343
association 131
association class 152
association end–adornments 226
association ends 133
atomic activity 301
attribute 135, 222

B
basic model 82
behavior modules 120
behavioral responses 82
Bill of Material Processor (BOMP) 255, 258
Bill of Systems Processor (BOSP) 255, 258
black box components 48
bottom-up verbs 100
business actors 88
business components 52
business entities 88
business level metamodel and profile 28
business objects model 81, 88
business process 294
business process diagrams (BPDs) 315
business processes 315
business rule model 82
business rules construction process 87
business systems 262
business tasks 52
business use cases 88
business use cases model (BUC) 81, 88
business-organizational point 52

C

C++ 265
calculate balance 50
catalysis 23
cause 87
CCTA Risk Analysis and Management Methodology (CRAMM) 355
Central Computer and Telecommunications Agency (CCTA) 355
changeability 135, 226
class 221
class diagrams 131
class invariant 210
class stereotypes 362
class-scoped_attribute_checking 179
classifier roles 243
class name 204
client-oriented requirements baseline 81
client-oriented requirements models 80
closure principle 84
code and design scavenging 49
collaboration diagrams 12
collaboration model 248
common object request broker architecture (CORBA) 52
communication protocols modeling 281
component reuse 66
component stereotypes 363
composition 131, 225
compositional reuse techniques 49
comprehensibility appropriateness 6, 13
computer-supported software engineering (CASE) 219

conceptual models 131
conceptual systems 69
concrete system 69
conditional branch states 266
configuration view 82
conformance declaration definer 250
connectors 316
constraint 84
construction block 276
container managed persistence (CMP) 233
content-composition-verb 100
contextual part 70
contractual agreements 49
conversation definer 250
CORAS™ methodology 332
CORAS ontology 340
corporate information systems (CIS) 259

D
data flow diagrams 254
data mining system (DMS) 255
dealers information system (DIS) 259
decorative stereotypes 282
definition stage 40
dependency 133
dependency relations checker (DRC) 249
dependent system 361
derived attributes 120
descriptive stereotypes 282
“design by contract” principle 200
design class diagrams 131
design patterns 376
design stage 40
determine demand 50
development process 67, 358
direct components 257
documentation 275
domain 3, 10
domain and business object analysis 365, 367
domain appropriateness 5
domain models 131
domain-specific models 23
dummy activity 301
dynamic model 265

E

eyearly reuse decision 74
Eiffel 207
Eiffel code 207
Eiffel programming language 50
elaboration phase 72
empirical model 69
empirical quality 3, 8
enterprise information portal (EIP) 255
Enterprise JavaBeans (EJB) 232
enterprise performance management (EPM) 255
enterprise viewpoint 342
entity-relationship (ER) 217
environment designer 75
evaluation framework 2
event-driven process chains (eEPC) 52, 315
events 316
evolutionary stereotypes 273, 282
exception 84
explicit knowledge 3, 6
extension mechanisms 273
externalization 8
externalized model 3

F

failure mode and effect criticality analysis (FMECA) 337
fault tree analysis (FTA) 337
Fleksit 28
Fleksit basis service 30
Fleksit business service 30
Fleksit service 30
fork/join model elements 266
formal semantics 130
formal specification 129, 200
formal syntax 7, 130
frameworks modeling 281
functional domain 364
functional domain expertise 359
functional iterations 380
functional mental model 70
functions 316

G
generalization 131, 227
generalized distributed programming language (GDPL) 281
generalized model 255
generative techniques 49
generic quality framework 1
graphic modeling 255
group information system (GIS) 259
group level 255
GSBl0o algebraic language 199
guarded branch 322
guidelines 358
guiding model 49

H

HAZard and OPerability study (HazOp) 336
headquarters level 255
hierarchical information system 255
hierarchical management information system (MIS) 255
hierarchical systems architecture 254
horizontal dimension 238
horizontal-dimension relations 246
“how-it-works” model 70
“how-to-use-it” model 70
hypertext view 82

I

illustration 44
impacts 82
implementation methods 376
info-mathics 254
information systems (IS) 254
inheritance 131
inline block 301
input models 71
instance-level 33
instantiations 139
integration scenarios 85
inter scenarios 85
inter-component coordination 51
interface definition languages (IDL) 51
interface stereotypes 363
internal dimension 238
internalizability 8
intra scenarios 85
intra-component coordination 51
intrusion detection exchange format (IDMEF) 336
inventory level 52
iteration-dimension 238

J
Java™ data objects (JDO) 233
JECKO 365

K
knowledge externalizability appropriateness 6
KOBRA 23

L
language 10
language extended lexicon (LEL) 82
language extension 3
language quality 2
late reuse decisions 74
leaf 135
LEL verification 85
level of SI 257
lexicon model 82
location 275
loop activity 301
lower-level activity diagrams 268

M
management information system (MIS) 255, 259
mapping 69
mapping associations 211, 222
mapping attributes 222
mapping binary associations 222
mapping classes 207, 221
mapping many-to-many associations 223
mapping one-to-many associations 223
mapping one-to-one associations 223
Markov analysis 337
mathematical modeling 254, 255
mathematical notations 254
mathematical semantics 7
meaning triangle 68
mental models 70
meta-model 273, 294
meta-modeling 10
mindset 74
model creation 69
model interpretation 69
model object data management system (MDMS) 219
model-based development process 71
model-based risk assessment 332
modeling 2, 68
modeling process 68
modeling task 3
modeling triangle 69
multiple relationships 26
multiplicity 131
multiplicity_checking 179

N
n-ary relationships 218
navigability 226
navigation 133
new system 361
NoFun (acronym for “NOn-FUNctional”) 117, 119
non-functional code 379
non-functional requirements 117
non-verb phrase 86
notions 82

O
object 131
object/relational mapping tools 233
object constraint language (OCL) 49, 51, 126, 219, 240, 276, 298
object diagrams 160
object identifier 229
object management group (OMG) 1, 2, 51, 276
object modeling technique (OMT) 218
object-oriented analysis 200
object-oriented code 199
object-oriented computer systems 262
object-oriented modeling 2
object-oriented software development process 237
on-line business model 46
open systems 70
operation owner 268
operational semantics 7
operations 131
ordering 226
order processing 53
organizational goal completeness 4
organizational goal validity 4
organizational model 300, 303
organizational quality 10
organizational quality 4
out-of-scope 360

P
package stereotypes 363
parallel tasks 325
parameterized classes 165
parameters 135
participant knowledge 6
participant language knowledge appropriateness 6
perceived semantic quality 4
perceived system 69
Petal File parser 250
petri nets 315
physical quality 3, 8
plant level 255
platform-specific profile 32
players in a business model 41
post-condition 210
pragmatic quality 4, 9
pre-condition 209
precise UML group (pUML) 201
primary key 229
prime modeling rules 365
process algebras 52
process calculi 52
product-oriented approach 118

production planning and control (PPC) 53
production control 53
production planning 53
programming by contract 50
progressive implementation method (Pim) 375, 376, 378
property 86
pseudo code 268

Q
qualifier 226
quality entities 120
quality model 119
quality requirements 123

R
RAISE specification language (RSL) 129
Rational Rose 2, 201
Rational Rose data modeler 221
rational unified process (RUP) 23, 80, 375, 376
RBM 89
RDD 89
re-use 360
reactive system design support (RSDS) 355
real time systems modeling 281
redefined stereotypes 282
reference model for open distributed processing (RMODP) 335
refinement of events 324
relation 86
relational algebra 254
relational databases 217
relationship between players 42
relationships 131
requirement analysis 365
requirements baseline 84
requirements engineering (RE) 116
restrictions 273, 274
restrictive stereotypes 282
retail information system (RIS) 259
retail pharmacy (drug dispensing) systems 220
reuse decision 73
risk analysis 334
risk assessment 333
risk management concern 343
role name 226
root 135
RSL 161
RSL templates 130
rules on meta-model 282
running software 359
RUP business model 80
RUP’s dynamic aspects 380
RUP’s static aspects 380

S
safeguarding 358
scenarios model 82
scope 133
security policy concern 343
security-critical systems 332, 333
select perspective 23
semantic quality 4
semantics 129, 275, 298
semitic triangle 68
separation of concerns 376
sequence diagram 12, 33
sequential events 325
service level agreement (SLA) 28
service provider 30
service user 30
SINTEF 24
social actor interpretation 3
social quality 4, 10
software attributes 119
software contracts 50
software design 262
software development process 238
software engineering 116
software process 376
software quality metrics 119
software quality standard 117
software views 316
special iterations 380
specific language code 268
specific modeling rules 365
specification-level collaboration 33
SpReIm 211
state machines 241, 298
stereotypes 273, 274, 278
stock 52
stock keeping 51
structural mental model 70
structural model 265
structure diagram 44
structured design 254
structured programming 254
sub-models 240
subclass 137
subflow activity 301
subsystems 257
superclasses 141
swimlanes 12, 267, 320
SWOT (strengths, weaknesses, opportunities, threats) 342
SWOT analysis 342
SWOT-diagram 342
symbol discrimination 7
symbol name 82
symbolic simplicity 7
symbolic systems 69
syntactic and semantic reference 129
syntactic disjointness 7
syntactic quality 3, 8
system & environmental data 300, 303
system and software architecture 365
system answers 242
system deepness 254
system depth 257
system designer 75
system end 254
system level 254, 257
system list 254
system product 254, 257
system reliability and quality 254
system schema 257
system width 254, 257

T
tag values 273, 274
tailoring framework 25	
target of evaluation concern 343
technical actor interpretation 3
technical actor interpretation appropriateness 7
technical terms 52
template classes 135
top-down verbs 100
transactional part 70
transfer values 42
translation 69
triggers 266

U
UC1-completeness 247
UC1-correctness 248
UML activity diagram 265
UML case tools 200
UML class diagram 265
UML constraints 133
UML extension mechanisms 274
UML meta-model 221
UML model translator 250
UML profile 26, 359
UML static models 199
UML-based methodology 23
unified modeling language (UML) 1, 2, 39, 49, 51, 67, 116, 117, 131, 200, 217, 254, 255, 262, 273, 294, 332, 360
unified process (UP) 23, 80, 238, 332, 333
unified system development process (USDP) 67
universe of discourse (UofD) 81
use case design 383
use case model 243
use case stereotypes 360
use cases 241
use-case driven 377
use-case model 246
user actions 242
user-defined stereotypes 263

V
value diagram 43
value of business 41
verbal phrase 86
verification 85

W
Web applications modeling 280
well-formed class diagram 134
workflow activity 299, 301
workflow application 300, 302
workflow enactment services (WfES) 295
workflow management coalition (WfMC) 294
workflow meta-model 299
workflow participant 300, 303
workflow process 294
workflow process definition language (WPDL) 297
workflow reference model 296
workflow relevant data 300, 302
workflow transition 300, 301
workflows modeling 281

X
XML metadata interchange (XMI) 336