

Index

Note: Pattern names are in **bold**; use case names are in *italic*.

A

Abstraction

- and goals of steps, 153
- use cases written as different levels of, 103

Abstract use cases, 198, 199

Access ATM description, with technology-specific steps, 161

Access ATM use case, improved Technology-Neutral, 161

Access e-mail use case, 131–132

Accuracy

- of requirements, 139
- verifying, 65

Active verbs, 159

ActorIntentAccomplished, xiv, 5, 23, 103, 127, 147, 154, 158–161, 163, 210

- examples, 159–161
- improving *Withdraw Cash* description with, 160

and purpose, 171

Withdraw Cash description without, 159–160

Actor-less ATM example, 159–161

Actor lists

- and briefs, 56
- for Pharmacy System, 93
- for revised Pharmacy System, 94
- for Wings Over the World (sample), 55

Actors, 1, 23, 24, 51, 54, 78, 79, 89, 114, 117

- and accomplishments of value, 118–119
- and context diagrams, 88
- duration of goals for, 119

identifying, 90

naming, 92

naming use cases and goals of, 123

in pharmacy example, 92–94

stakeholder's competing visions about, 81

Adolph, Steve, xix

Adornments, 4, 22, 42, 117, 121, 127, 133–137,

- 142, 146, 147, 155, 163, 169, 178, 206,
- 207, 214, 215, 216

examples, 135–137

and use cases, 22

Wings Over the World with, 135–137

Agile Software Development, xix

Agile Software Development Ecologies, xix

Agile Software Development Series, xviii

Alexander, Christopher, xvi, 6, 7, 8, 9, 10

Alexandrian form, 10, 16

Alternative courses of actions

coping with, 183

handling, 191

Ambiguity, 68–69, 70, 141, 174

and imprecision, 141

reducing, 159

removing, 81

in scenario, 171

Analysis, 202

Anchor points, 12, 17, 96

Apollo Saturn V spacecraft, 38

ARJIS. *See* Automated Regional Justice Information System

Armstrong, Neil, 80

ATM JAD system example, 150–152

- Audience. *See also* Customers; Users
 amount of information needed by, 61
 knowing, 175
 and names of use cases, 122
 and readability of use cases, 138
- Auto insurance claim handling, 128
- Automated railway signaling, 84
- Automated Railway Signaling System, vision statement for, 84
- Automated Regional Justice Information System, 33
- B**
- BalancedTeam, 20, 30, 32, 39–42, 43, 92
- Balanced use cases, 69
- Batch job distribution, example, 60
- Best practices, and pattern form, xiv
- Blob, The* (film), 86
- Book Flight for Frequent Flier*, 187
- Book Flight for Frequent Flier* extension use case, 186
 referencing extension points, 189
- Book Flight* use case
 after *Upgrade Seat* alternative promoted via extends, 194
 after *Upgrade Seat* alternative promoted via includes, 192
 brief for, 56
 with business rules, 135
 confusing frequent flier alternatives for, 184
 with extension points, 188
 with incorrectly applied includes relationship, 178
 more detailed version of, 107
 revised, 179–180
 shortening with *EverUnfoldingStory*, 106
 thirty-page use case horror, 105
- Boundaries, different points of view about, 86–87
- Boundary conditions, 191
- Brainstorming, 33
- Bramble, Paul, xix
- Branch-and-join process, 33
- BreadthBeforeDepth, 20, 47, 48–51, 52, 53, 55, 68, 71, 98, 140, 200
 and alternative conditions, 131
 examples, 51
 and order for use case writing process, 51
 QuittingTime facilitated by, 72
 SpiralDevelopment interacts with, 54, 76
 and stopping points in use case, 127
 and UML, 51
 and use case diagrams, 89
- Bridges of Madison County, The* (film), 90
- Briefs, 55
 and actor lists, 56
 for *Reserve Flight* and *Book Flight*, 56
- “Brittle” requirements set, 4
- Brooks, Frederick, 1, 51, 84
- Builders, and expansion of system scope, 81
- Business purchasing, 61–63
- Business rules, 127, 134, 136, 147, 149
Book Flight with, 135
Change Seat adorned with, 135–136
 specifiers, 2
- Business-value check, 66
- Buy something* (casual version) use case, 61
- C**
- Call processing examples, 206–208, 211–212
- Canadian Pacific Ltd, 204
- Candidate use cases, identifying, 50
- CapturedAbstraction, 51, 175, 198–200
 and UML, 198
- Change Seat* use case
 adorned with business rules, 135–136
 business rules in, 152
 informal version of, 140
 review of, 121
- “Chaos” reports, xiii
- ChiliPLOP conferences, xix
- “Choose your own adventure” story
 differences between use case and, 126
 and writing use cases, 125
- Claim Insurance* use case
 not making sufficient forward progress, 165
 revised to make forward progress, 166
- CleanHouse, 21, 50, 54, 202, 203, 213–215
 examples, 215
- ClearCastOfCharacters, 11, 13, 15, 48, 79, 90–94, 118
 and *EverUnfoldingStory*, 113

- examples, 92–94
 - and services identification, 97
 - and use case sets, 22
 - and `VisibleBoundary`, 87
 - and vision, 82
- Clear vision, lack of, 80
- Cockburn, Alistair, xiv, xv, xix, 7, 27, 104
- Code bloat, 149
- Common steps, rewriting, 176
- `CommonSubBehavior`, 23, 51, 121, 174, 176–181, 196, 206
 - examples, 178–181
 - and includes relationship, 174, 180
 - and UML, 180–181
- Communication
 - common forms and facilitation of, 59
 - strengthening channels of, 81
- Complete group, defining, 66
- `CompleteSingleGoal`, xiv, 8, 15, 97, 117, 118–121, 122, 125, 127, 138, 139, 142, 143, 154, 159, 184, 191, 199, 203, 205, 206, 207, 209, 210
 - and addressing more than one goal, 120
 - examples, 121
 - factors to consider with, 142
 - and goals associated with use cases, 5, 14, 103, 117
 - helper patterns for, 215
 - and individual use cases, 22
 - for *Process Normal Call* use case, 207–208
 - and scope of use case, 178
 - and structuring of use cases, 4
 - use cases as representative of, 209
- Complexity
 - controlling, 119
 - incremental, 126
 - reducing, 196
- Complex use cases, 178, 190
- Computer-Aided Software Engineering (CASE)
 - tools, for generating use case diagrams, xviii
- Concrete details, 167
- Conditions, 148
- Connery, Sean, 90
- Consensus, 57
- Constantine, Larry, 43
- Content, reviews of, 64
- Context
 - and patterns, 15
 - in `UserValuedTransactions` pattern sample, 11
- Context diagrams, for *Wings Over the World*, 88–89
- Control break processing, 191
- Core competencies, 69, 71
- Corporate vision, and technology vision, 83
- Create, Read, Update, and Delete, 12, 78, 80, 96
 - Create Customer* use case, and mortgage origination system, 100
- CRUD. *See* Create, Read, Update, and Delete
- Cruddy mortgage origination system, 100
- Crystal Clear*, xix
- Customers. *See also* Audience; Users
 - final authority on product and, 36
 - movie-making and consultation with, 38
 - soliciting input from, 37
 - and use case development process, 36, 37, 43
- D
- Data
 - formats, 136
 - specifiers, 2
 - validation rules, 134
- Davis, Wade, 73
- Day at the Races, A* (film), 38
- Deadlines, 53
- Delays, expense of, 53
- Delete Customer* use case, and mortgage origination system, 100
- Dependency, 183
- Deployment diagram, 87
- Design Patterns: Elements of Reusable Object Oriented Software* (Gamma), 6–7, 8, 9
- Details
 - addition of, by developers, 138
 - handling, 49
- DetectableConditions, 23, 127, 136, 148–152, 171
 - examples, 150–152
 - and scenarios, 147
 - and variations, 131

- Developers
 - and detectable conditions, 149
 - errors handled by, 130
 - and use case readability, 138
- Development organizations, and end user representation, 35
- Development patterns, 19
- Development team, 66
- Diagrams, and *SpiralDevelopment*, 57
- Dictionaries, iterative writing and, 57
- Diminishing returns, avoiding law of, 54
- Director-style *Book Flight* use case, with incorrectly applied includes relationship, 178
- Diversions
 - creating, 162
 - with insufficient forward progress, 165
 - multiple, 182
 - repeated, 183
 - in scenarios, 163
- Diversity
 - and *BalancedTeams*, 32
 - lack of, 41–42
- Documentation group, and use cases, 37
- Domain experts, 37, 40
- Dr. Strangelove* (film), 90
- Droplets, repairing, 180
- Droplet use cases, merging into new use cases, 212

- E
- Eastwood, Clint, 90
- Editing, 19, 21
- Editing existing use cases, 201–216
 - CleanHouse*, 213–215
 - MergeDroplets*, 209–212
 - RedistributeTheWealth*, 204–208
 - trade-offs and collaborations, 215–216
- EDSAC, 176
- e-mail access example, 131–132
- Encyclopedias, iterative writing and, 57
- Engage Art* use case, 109–110
- Engage Diversity use case, 110–111
- ENIAC, 176
- Errors
 - developers and handling of, 130
 - in requirements models, 177
 - and reviews, 64, 65

- Estimates, 2
- EverUnfoldingStory*, xvi, 14, 15, 79, 97, 100, 102–113, 118, 154, 159, 179, 191, 199, 200, 206, 207, 210
 - and *Book Flight* use case, 112
 - and Centre A: The Museum for Contemporary Asian Art, 109–111
 - examples, 104–110
 - and goals, 120
 - good use cases in context of, 113
 - and includes relationship, 180
 - and *LeveledSteps*, 171
 - shortening *Book Flight* with, 106
 - systems described as, 113
 - and UML models, 111
 - and use case sets, 22
- Examples
 - actor-less ATM, 159–161
 - ATM JAD system, 152–154
 - auto insurance claim handling, 128
 - automated railway signaling, 84
 - batch job distribution, 60
 - call processing, 206–208, 211–212
 - Cruddy mortgage origination system, 100
 - e-mail access, 131–132
 - File Accident Claim*: tied to technology, 169–171
 - hospital claim, 215
 - insurance claims, 123–124, 165–166
 - mobile dispatching, 85
 - Museum for Contemporary Asian Art, 92, 109–110
 - for patterns, 18
 - pharmacy receptionist, 92–94
 - purchasing for a business, 61–63
 - vision statement for *Wings Over the World*, 82–83
 - Wings Over the World*, 54–57, 104–109, 199
 - Wings Over the World*: diversionary scenario, 163–164
 - Wings Over the World*: precise but unreadable, 140–141
 - Wings Over the World*: readable but imprecise, 139–140
 - Wings Over the World* and avoiding form-focus, 98–99

- Wings Over the World and extensions, 184–186
- Wings Over the World and includes, 178–180
- Wings Over the World and User-Valued Transactions, 98–99
- Wings Over the World (continued), 66, 71–72, 75
- Wings Over the World goals, 121
- Wings Over the World seat upgrade alternative, 191–193
- Wings Over the World with Adornments, 135–136
- Exception processing, 130
- ExhaustiveAlternatives, 5, 8, 50, 59, 117, 127, 129–132, 142, 148, 150
 - examples, 131–132
 - and use cases, 22
- Expandability, and high-quality system, 83
- Expert users, 40
- Extends relationship, 23, 108, 173, 175, 180, 186, 196
 - and alternatives, 194
 - formal definitions in UML for, 178
 - includes relationship *versus*, 174
 - interpretations for, 109
 - misunderstanding, 183
 - PromotedAlternative and UML, 195
- Extends use cases, 183, 184, 185, 195, 199
 - Book Flight* after *Upgrade Seat* alternative promoted via, 194
 - Book Flight for Frequent Flier*, 186
 - and UML extension points, 187–189
 - Upgrade Seat* as, 195
- Extension points, *Book Flight* use case with, 188
- Extensions, 24
 - Wings Over the World and, 182–184
- External interface protocols, 134
- Exxon, 202
- F
- Feature creep, 65
- File Accident Claim* use case
 - technology-dependent, 169–170
 - technology-neutral, 170
- Find Flight*, lower level use case for, 105–106
- Forces affecting problem, and patterns, 16–17
- Formality, 59, 175, 177, 183
- Formal specifications, 141
- Formats, choosing for use cases, 59
- Form-focus, Wings Over the World and avoidance of, 99–100
- Forms-based use case set, 99
- ForwardProgress, 23, 127, 147, 154, 159, 161, 162–166
 - examples, 163–164
 - and steps in scenario, 171
- Fowler, Martin, 202
- Fragmented use cases, 121
- Fragments, 24, 127
 - combining of, into cohesive units, 215
 - merging into existing use cases, 210
 - relocating to other use cases, 205
- Frequent flier alternatives, confusing for *Book Flight*, 184–185
- Functional requirements, nonbehavioral information and, 134
- G
- Gabriel, Richard, 9
- Generalization relationships, 174, 200
 - formal definitions in UML for, 178
- Generalizations, 173, 181, 196
- Generalizes relationship, 23
- Get Paid for Car Accident* use case, 128
- Goal fragments, use cases written around, 121
- Goal level, and actor list, 55
- Goals
 - actor, 119
 - good characteristics of, 120
 - improper, 118
 - project, 80–81
 - and SharedClearVision, 96
 - and subgoals, 25
 - of use cases, 117
 - use cases written around, 97
- Gold plating, preventing, 84
- Gold Rush* (Cockburn), 7
- Graham, Ian, 13, 95
- Grammatical errors, 65
- Granularity, 142

Graphical user interface, 169
 Group dynamics, 20
 GUI. *See* Graphical user interface

H

“Happy day” scenario, 24
 High-level use cases, 55, 108. *See also* Briefs
 Hospital claim example, 215

I

If statements, 126
 Implementation details, 91, 167
Improving Software Organizations, xix
 Included use case, 195
 Includes relationship, 23, 108, 111, 112, 173,
 175, 181, 195, 206
 and alternatives, 192
 Book Flight after *Upgrade Seat* alternative
 promoted via, 192
 extends relationship *versus*, 174
 formal definitions in UML for, 178
 interpretations for, 109
 misunderstanding, 177
 restriction on, 180
 Wings Over the World and, 178–180
 Incremental complexity, 126
 Inheritance relationships, 183
 Internal stakeholders, and use case develop-
 ment process, 36, 37
 InterruptsAsExtensions, 23, 51, 175, 182–189,
 196, 199, 206
 examples, 184–186
 and extension use cases, 175
 and UML, 186–187
 Invalid actors, identifying, 88
 Iterative approach, 76
 Iterative development, and *SpiralDevelopment*,
 54
 Iterative life cycle, xiii

J

Jacobson, Ivar, xiii, 1, 2, 23, 110
 JAD. *See* Joint Application Development
 Jargon, 39
 Java, 167
 Job flow engine, 37–38

Joint Application Development, 150
Joy of Cooking, The, 177

K

Kennedy, John F., 80
 “Know your audience” rule, 175
 Kraus, Andy, 33
 Kubrick, Stanley, 90

L

Large groups
 facilitating: no design without representa-
 tion, 33
 and frequent checkpoints, 33
 Large steps, 153
 Large teams, problems with, 32
 Large use cases, 119, 153
 breaking up, 216
 Larman, Craig, 2
20,000 Leagues Under the Sea (Disney film),
 158
 LeveledSteps, 4, 23, 127, 153–157, 162, 163,
 165, 167, 171, 172, 178, 184, 191, 199,
 206, 210
 examples, 154–157
 revised *Purchase Goods* use case with, 155
 scenarios with, 147
 steps of scenario written as, 158
Log Billing Information sub-use case,
 RedistributeTheWealth by creating, 207
 Lower level included use cases, shared courses
 of action expressed with, 178
 Lower level use cases
 creating, 206
 referencing from higher level use cases,
 110

M

Main success scenario, 203
Make a Phone Call use case, 212
Manifesto for Agile Software Development, xix
 Marx Brothers, 38
 McQueen, Steve, 86
 MergeDroplets, 21, 50, 54, 175, 179, 198, 202,
 203, 205, 209–212, 216
 examples, 211–212

Metaphoric story
 and patterns, 16
 in *UserValuedTransactions* pattern sample,
 11–13

MGM, 38

Miscommunication, cost of, 159

Mission of organization, 81

Mistakes, 70

Mobile Dispatching System, vision statement
 for, 85

Model formality, 69

Movie-making, and customer consultation,
 38

MultipleForms, 20, 47, 50, 58–63, 75
 batch job distribution, 60
 examples, 60–63
 purchasing for a business, 61–63

Museum for Contemporary Asian Art
 (Centre A)
 and *EverUnfoldingStory*, 109–111
 use case model for, 111

Museum for Contemporary Asian Art
 (Vancouver, Canada), 92

N

Names and naming
 goal-based, 123
 of patterns, 14–15
 of primary actor goal, 120
 in *UserValuedTransactions* pattern sample,
 11
 value of, 122

NASA, 38, 80

National Software Quality Experiment (2000),
 xiii

Network Administrator
Provision a Cross-Connect (Engineering-
 Centric Version), 41
Provision a Cross-Connect (User-Friendly
 Version), 42

Network Element
Provision a Cross-Connect (Engineering-
 Centric Version), 41
Provision a Cross-Connect (User-Friendly
 Version), 42

Night at the Opera, A (film), 38

Normal Call use case, processing with multiple
 goals, 206

Nouns/noun phrases, for actors, 92

O

Object Constraint Language, 141

Object-oriented software, refactoring, 202

Object-oriented software development, use
 cases as element of, 2

OCL. *See* Object Constraint Language

“One-stop-shopping” rule, 177, 183, 199

On-line store
 long and tedious use case for, 154–155
 use case mixing large and small steps for,
 156
 use case with excessively large steps for, 155

OOPSLA 98 (Vancouver, Canada), xix

“Outer” reviews, purpose of, 66

Outlines, 49–50

Outsourcing, 72

Overlapping responsibilities, discovering, 91

Overspecification, 68, 69

Overstaffing, 32

Overviews, early, 49

P

Parallelism, 49

Partial use cases, 209

ParticipatingAudience, 20, 30, 32, 35–38, 40,
 43, 65, 76, 82, 139
 examples, 37–38

Passive voice, 160
Withdraw Cash Description written in,
 160–161

Pattern Almanac, The (Rising), xix

Pattern Handbook, The (Rising), xix

Pattern Language, A (Alexander), 10

Pattern language heritage, xix

Pattern language organization, 19–23
 development patterns, 19
 editing, 21
 process, 20
 relationships, 23
 scenarios and steps, 22–23
 structural patterns, 21
 team, 20

- Pattern language organization, (*cont.*)
 - use cases, 22
 - use case sets, 21–22
 - Pattern languages
 - Alexander's work on, 6–7
 - use of, 9–10
 - Patterns, xiv
 - description of, 6–8
 - development, 19
 - for evaluating use cases, xiv
 - structural, 19, 21
 - use case set, 78
 - Patterns of Software* (Gabriel), 9
 - Pay Tax* use case, 177
 - Performance information, 127
 - Pharmacy receptionist example, 92–94
 - Pictures, in patterns, 15
 - Placeholders, 50
 - Pols, Andy, xix
 - Porting a system, 167
 - Post-conditions, 56
 - PreciseAndReadable*, 20, 22, 59, 117, 120, 127, 134, 138–141, 142, 158, 159, 167, 175
 - examples, 140–141
 - Precision, 65, 139. *See also* Accuracy
 - Preconditions, 56
 - Preconditions sections, of use cases, 75
 - Problem statement
 - and patterns, 16
 - in *UserValuedTransactions* pattern sample, 11
 - Process, 19, 45–76
 - BreadthBeforeDepth*, 48–51
 - MultipleForms*, 58–63
 - and quality use cases, 20
 - QuittingTime*, 68–72
 - SpiralDevelopment*, 52–57
 - trade-offs and collaborations, 75–76
 - TwoTierReview*, 64–67
 - WritersLicense*, 73–75
 - Process Normal Call* use case, *Redistribute-TheWealth* in, to give use case a *CompleteSingleGoal*, 207
 - Process patterns, xvi
 - Productivity, improving, 49
 - Programmers, and reviews, 66
 - Project priorities, incremental approach based on, 50
 - Projects
 - differing needs of, 58
 - slippage in, 74
 - PromotedAlternative*, 23, 51, 127, 174, 190–196, 199, 206
 - examples, 191–196
 - and extensions, 175
 - and UML, 195–196
 - Engineering-Centric Version, 41
 - User-Friendly Version, 42
 - Proxy (pseudo actor), 106
 - Purchase Goods* use case
 - with excessively large steps, 155
 - revised, 155–157
 - revised, with leveled steps, 155
 - with unlevelled steps, 154
 - Purchase Property* use case, 177
 - Purchasing for a business, 61–63
- Q**
- Quality assurance group, and use cases, 37
 - Queues, 44
 - QuittingTime*, 20, 47, 50, 54, 66, 68–72, 73–74, 76, 140, 202
 - examples, 71–72
 - use cases passing test for, 74
- R**
- Railway signaling, automated, 84
 - RAPIER
 - failure codes, 152
 - reservation system, 145–146
 - Ratliff, Rick, xix, 48, 167
 - Rawsthorne, Dan, xx, 51, 57, 89, 111–113, 175, 186–187, 195–196, 198–200
 - Readability, verifying, 65
 - Reading, and consistent writing style, 73
 - Receive Customer* use case, 93
 - with actor's role defined as Receptionist, 94
 - RedistributeTheWealth*, 21, 50, 175, 191, 202, 203, 204–208, 210, 216
 - by creating *Log Billing Information* sub-use case, 207
 - examples, 206–208

- in *Process Normal Call*, to give a use case a **CompleteSingleGoal**, 207
 - and scope of use case, 178
- Redundancy
 - eliminating, 180, 196
 - and inconsistencies in model, 177
- Redundant behavior, 90
- Refactoring* (Fowler), 202
- Register for Courses* (Use Case with Extensions), 24–25
- Relationships, and use cases, 23. *See also*
 - Extends relationship; Includes relationship
- Release schedules, 2
- Reliability, and high-quality system, 83
- Representative democracy, 64
- Request Upgrade* use case
 - with forward progress, 164
 - with steps diverting forward, 164
- Requirements
 - high cost of mistakes, 53
 - improving definition of, xiii
 - source of errors in models, 177
 - specifications, 68, 202
 - volatility of, 53
- Requirements gathering, 53
 - as process of discovery, 49
 - prolonging, 69
- Reserve Flight*, brief for, 56
- Reserve Flight Segments* use case, 112–113, 145–146, 152
- Reuse, and large use cases, 119
- Reviewers, 33
- Reviewing, 149
- Reviews, 37, 64–66
 - effective, 64
 - two types of, 65
- Rising, Linda, xix, 9
- Risk factors, delaying discovery of, 52
- Robust design, and variations, 130

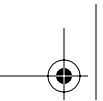
- S
- Scalability, and high-quality system, 83
- ScenarioPlusFragments**, 50, 117, 120, 125–128, 133, 142, 153, 163, 171, 176, 182, 183, 190, 191, 196, 198
 - and goals in use cases, 22
 - examples, 128
 - use cases written as, 129
 - and variations, 131
- Scenarios, 19, 22–23, 117, 147
 - balanced, 171
 - clear steps in, 162
 - mixing levels of detail in, 153
 - number of steps in, 154
 - organizing steps in, 171
- Scope creep, and poorly defined boundaries, 87
- Screen designs, 127
- Sellers, Peter, 90
- Sentence fragments, 5
- Sequential life cycle, xiii
- Serpent and the Rainbow, The* (film), 73
- Services, sets of, 92
- SharedClearVision**, xv, 11, 12, 15, 17, 31, 48, 80–85, 90, 95
 - and **EverUnfoldingStory**, 113
 - examples, 82–85
 - lack of, 82
 - and services identification, 97
 - and system goals, 96
 - and use case sets, 21
 - of use case writers, 78
 - and user identification, 92
 - VisibleBoundary** and limits/support of, 86
- Slippage, 74
- Small steps, 153
- Small use cases, 119, 153, 210
- SmallWritingTeam**, 20, 30, 31–34, 35, 38, 39, 40, 43, 65
 - and audience participation, 38
 - examples, 33–34
 - vision created by, 82
- Software for Use* (Constantine and Lockwood), 37, 43
- Software patterns, benefits of, 123
- Solution
 - of pattern, 18
 - in **UserValuedTransactions** pattern sample, 13–14
- Specialization relationships, 186

- Spelling errors, 65
 - SpiralDevelopment**, 20, 47, 50, 52–57, 64, 96
 - BreadthBeforeDepth** coupled with, 76
 - examples, 54–57
 - and regular assessments of use cases, 66
 - and roles of actors, 92
 - and system boundary, 87
 - and UML models, 57
 - and use case diagrams, 89
 - Stacks, 44
 - Stakeholders, 119
 - competing visions of, 81
 - and good specifications, 141
 - input of, 119
 - and overspecification, 69
 - and system boundaries, 86
 - and use case goals, 204
 - and use case readability, 138
 - use cases understood by, 173
 - and useful systems, 90
 - vested interest in use cases by, 65
 - Standish Group, xiii
 - Star Trek* (television series), 162
 - Statement of purpose, 81
 - Step patterns, 147
 - Steps, 22–23
 - behavior put into, 162
 - clear and succinct, 162
 - conditions met by, 171
 - excessively small and large, 153
 - organizing in scenarios, 171
 - Purchase Goods* use case with excessively large, 155
 - and repeated diversions, 182
 - writing, 158
 - Stockholder meetings, 35
 - Stories
 - good, 145–147
 - quality, 147
 - variations, 126
 - Structural patterns, 2–3, 19, 21
 - Style guides, 74
 - Subfunction goals, 25
 - Subfunction Level, of use case, 104
 - Subgoals, 25
 - Subject matter experts
 - narrow focus of, 91
 - and small use cases, 120
 - Subsets, of briefs, 56, 57
 - Sub-use cases, 207
 - Success scenarios
 - alternatives to, 125
 - identifying, 127
 - SuD. *See* System under discussion
 - Summary goals, 25
 - Summary Level, of use case, 104
 - Supplementary specifications, superfluous fragments relocated to, 206
 - Surface and Dive technique, 56
 - Surviving Object-Oriented Projects*, xix
 - System-focused use cases, 121
 - Systems
 - deficient, 95
 - and detectable conditions, 148
 - useful, 90–91
 - System's scope, documenting interactions outside of, 89
 - System under discussion, 24, 25
- T**
- Tablature (guitar), 138
 - TBDs. *See* To Be Determined issues
 - Teams, 20, 29–44
 - and **BalancedTeam**, 39–42
 - composition of, 30
 - developers and end users on, 40
 - and differing amounts of formality, 58–59
 - organization of, 19, 30
 - and **ParticipatingAudience**, 35–38
 - size of, and use case quality, 30
 - SmallWritingTeams**, 31–34
 - trade-offs and collaborations, 43–44
 - and well-written use cases, 26–27
 - Technical errors, 65
 - Technological details, and increased costs in reading/writing use cases, 168
 - Technology, volatility of, 168
 - Technology details, in use cases, 152
 - TechnologyNeutral**, 4, 23, 133, 134–135, 140, 161, 167–171, 200, 210
 - and *Access ATM* use case, 161

- examples, 169–171
 - and scenarios/steps, 147
 - and steps, 154
- Technology vision, and corporate vision, 83
- Telephone calls, set of small use cases for placing, 211
- Templates, 50, 58, 59, 74, 127
- Terminators, in context diagrams, 88
- Testers, 2
- Time pressures, 80, 91
- Titles, for use cases, 117
- TL1 code, 42
- To Be Determined issues, 134
- Traceability, 198
- Trace relationship, 111–112
- Trainers, 37
- Trust, 72
- TwoTierReview, 8, 20, 32, 37, 47, 54, 64–67, 73, 76
 - examples, 66–67
- U
- UML. *See* Unified Modeling Language
- UML extension points, and extension use cases, 187–189
- UML models
 - and EverUnfoldingStory, 111
 - and SpiralDevelopment, 57
 - and VisibleBoundary, 89
- Unbalanced teams, 30
- Unified Modeling Language, xvii–xviii, 173, 174, 175, 178
 - and BreadthBeforeDepth, 51
 - and CaptureTheAlternative, 197
 - and CommonSubBehavior, 180–181
 - deployment diagram in, 88
 - and InterruptsAsExtensions, 186–187
 - and PromotedAlternative, 195–196
- Unleveled steps, *Purchase Goods* use case with, 154
- Update Customer* use case, and mortgage origination system, 100
- Upgrade Seat*, as extending use case, 195
- Upgrade Seat* alternative
 - Book Flight* after promotion of, via extends, 194
 - Book Flight* use case after promotion of, via includes, 192
 - dominates *Book Flight* use case, 191–192
 - as separate use case, 193
- U.S. Department of Defense, xiii
- Usability, formality favored over, 174
- “Use Case Blue” (Kraus), 33
- Use case development process, 53
 - customers and internal stakeholders involved with, 36
 - customers involved in, 43
- Use case diagrams, styles for, xviii
- Use case forms, standard, 59
- Use case horrors
 - Access ATM* description with technology-specific steps, 161
 - Book Flight* with business rules, 135
 - Claim Insurance* not making sufficient forward progress, 165
 - confusing frequent flier alternatives for *Book Flight*, 184–185
 - Director-style *Book Flight* use case with incorrectly applied includes relationship, 178–179
 - Get Paid for Car Accident*, 128
 - mixing large and small steps in use case, 156
 - naming horrors, 123–124
 - Process Normal Call* use case with multiple goals, 206–207
 - Provision a Cross-Connect* (Engineering-Centric Version), 40
 - Purchase Goods* use case with excessively large steps, 155
 - Purchase Goods* use case with unleveled steps, 154
 - Register for Course*, 3
 - Request Upgrade* with steps diverting **ForwardProgress**, 164
 - set of small use cases for placing telephone call, 211
 - supplementary requirements anchored by, 136–137
 - technology-dependent *File Accident Claim* use case, 169
 - thirty-page *Book Flight* use case, 105

- Use case horrors (*cont.*)
 - Withdraw Cash* description without *ActorIntentAccomplished*, 159
 - Withdraw Cash* description written in passive voice, 160
- Use case models, 142
 - formalism in, 183
 - UML and formalism in, 178
- Use case names, good, 124
- Use case pattern form
 - context, 11, 15
 - description of, 10–18
 - examples, 11, 18
 - forces affecting the problem, 11, 16–17
 - metaphoric story, 11, 16
 - names, 10, 14–15
 - pictures, 10, 15
 - problem statement, 11, 16
 - solution, 11, 18
 - stepping through sample pattern, 11–14
- Use case pattern languages, reasons for, 5–6
- Use case relationships, 19, 173–200
 - CapturedAbstraction*, 198–200
 - CommonSubBehavior*, 176–181
 - InterruptsAsExtensions*, 182–189
 - PromotedAlternative*, 190–196
 - trade-offs and collaborations, 196–197
- Use cases, xiii, xvi, 19, 22, 115–143. *See also*
 - Editing existing use cases; Examples adorning, 117
 - Adornments, 133–137
 - alternatives and cluttering of, 190
 - alternatives for, 129
 - balanced, 68, 171
 - brief tutorial on writing, 23–27
 - Buy something* (Casual Version), 61
 - Buy something* (Fully Dressed Version), 61–63
 - capturing alternatives/failures handled in, 131
 - and changes, 96
 - cleaning up and removing, 213–215
 - clear, succinct steps in, 162
 - complete, 70
 - CompleteSingleGoal*, 118–121
 - complex, 190
 - complex alternatives and cluttering of, 198
 - complicated and imprecise, 138
 - correct, precise and readable, 26–27
 - creating lower level, 206
 - creating new, 205
 - crisscrossing structure revealed with, 126
 - developing in iterative, breadth-first manner, 53
 - different detail levels in, 102
 - effective, 149
 - example of poorly written, 3
 - excessive detail in, 205
 - ExhaustiveAlternatives*, 129–132
 - expense of adding, 205
 - extension-handling behavior for, 54
 - fragmented, 121
 - goal levels, 104
 - groups with vested interest in set of, 35–36
 - high-level, 55
 - high-quality, 75–76
 - large, 119, 153
 - levels of, 102–103
 - localizing closely related behavior into, 216
 - localizing information about features in, 209
 - long, 204
 - main scenario for well-written, 5–6
 - merging, 210
 - minimizing number of, 210
 - necessary information in, 95
 - nonfunctional requirements in, 133, 134
 - organizing, 113
 - outlines for, 50
 - overviews of, 49, 50
 - partial, 209
 - partitioning, 177, 190
 - patterns for evaluation of, xv
 - PreciseAndReadable*, 138–141
 - preconditions sections of, 75
 - purpose of, 74, 133–134
 - quality, 158, 215
 - readability of, 138, 139
 - readable and comprehensible, 117
 - reasons for using, 1–2
 - relocating fragments to, 205
 - reviewing, 64–66
 - ScenarioPlusFragments*, 125–128

- selecting format for, 59
 - signs of quality in, 78
 - small, 119, 153, 210
 - stability of, 96
 - stopping development of, 70
 - story lines in, 117
 - and style issues, 73
 - technology details in, 152
 - and telling of good stories, 2–5
 - trade-offs and collaborations, 142–143
 - unnecessary, 51
 - and value-added services, 96
 - value to business shown by, 103
 - VerbPhraseName**, 122–124
 - for workstation monitor server, 60–61
 - and “writer’s license,” 74
 - writing in technology-neutral manner, 168
 - Use case sets, 19, 21–22, 77–114
 - ClearCastOfCharacters**, 90–94
 - and context diagrams, 88
 - EverUnfoldingStory**, 102–113
 - forms-based, 99
 - organization in, 103
 - SharedClearVision**, 80–85
 - trade-offs and collaborations, 113–114
 - UserValuedTransactions**, 95–101
 - VisibleBoundary**, 86–89
 - Use case structure, levels of, 21
 - Use case template, additional fields in, 134
 - Useful services, identifying, 96
 - User Goal Level, of use case, 104
 - User interface designers, 2
 - User-interface details, 3, 134
 - User-interface navigation, 136
 - User-interface sketches, 134
 - Users. *See also* Audience; Customers
 - services tied to, 91
 - and useful systems, 90
 - UserValuedTransactions**, 8, 15, 16, 17, 25, 50, 79, 95–101, 102, 116, 131, 142, 168, 203, 209, 213, 214
 - and *Book Flight* use case, 108
 - and **EverUnfoldingStory**, 114
 - examples, 98–101
 - stepping through, 11–14
 - and system analysis, 92
 - and use case sets, 22
 - and vision, 82
 - for *Wings Over the World*, 98–99
- V**
- Value-added services, 12, 96
 - Variations
 - capturing, 131
 - having information about, 130–131
 - identifying, 130
 - Vehicle Control Center (VCC), 84
 - VerbPhraseName**, 7, 22, 51, 122–124, 186, 196, 206
 - for each use case, 50
 - examples, 123–124
 - for primary actor goal, 120
 - for use cases, 117, 142
 - Verbs, 159
 - Verne, Jules, 158
 - Vietnam War, 118
 - VisibleBoundary**, 86–89
 - and **EverUnfoldingStory**, 113
 - examples, 88–89
 - and project’s vision and scope, 79, 82
 - and system scope, 92
 - and UML models, 89
 - and use case sets, 21
 - Vision
 - changes to, 82
 - consistency in, 82
 - of stakeholders, 81
 - Vision statements
 - for Automated Railway Signaling System, 84
 - items included in, 81–82
 - for Mobile Dispatching System, 85
 - for *Wings Over the World*, 83, 98
- W**
- Walters, Rusty, 3
 - “Waterfall” life cycle, xiii, 53
 - Web site, use cases discussion at, 27
 - Whiteboard drawings, 51
 - Wilkes, Maurice, 176
 - Wings Over the World*, xvi, xvii, 18, 152
 - actor list (sample) for, 55
 - with **Adornments**, 135–137



- Wings Over the World (cont.)*
- and avoiding form-focus, 99–100
 - briefs for *Reserve Flight* and *Book Flight*, 56
 - context diagrams for, 88–89
 - discussion with chief architect at, 115–116
 - and editing existing use cases, 201–202
 - examples, 54–57, 66–67, 71–72, 75, 191–195, 199–200
 - and extensions, 184–187
 - and goals, 121
 - initial conversation with CIO of, 45–46
 - precise but unreadable example, 141
 - readable but imprecise example, 140
 - red-eye flight to, 77–78
 - and *Reserve Flight Segment* use case, 145–146
 - and *UserValuedTransactions*, 98–99
 - vision statement for, 83, 98
 - winning contract for, 29
- Withdraw Cash*
- description without *ActorIntent-Accomplished*, 159–160
- description written in passive voice, 160
 - improving description with *ActorIntent-Accomplished*, 160
 - use case, 151
- Workstation monitor server, use case for, 60
- WritersLicense*, 20, 47, 70, 73–75, 76
- examples, 75
- Writing, 149
- consistent style of, 73
 - at different levels, 103
 - and “know your audience” rule, 139
 - quality use cases, 158
 - reviews of, 64
 - steps, 159
 - in technology-neutral manner, 168
- Writing Effective Use Cases* (Cockburn), xiv, xv, xix, 3, 25–26, 27, 61, 104
- X
- XML, 44
 - XP, 71

