

A

- ability to's 141
- acceptance testing 242
- acceptance tests 204
- Acme Media
 - adapting
 - during adapt week 265–270
 - during an iteration 261
 - to business changes 271
 - assigning features to iterations 200
 - case study 17–19
 - background 18
 - individuals 19
 - teams 19
 - core team 68
 - current process,
 - documenting 88
 - demonstrating 267
 - deployment
 - considerations 291
 - deployment plan 292
 - feasibility team, feedback from 126
 - feature workflow,
 - modeling 206
 - lessons from pilot project 316–321
 - need for agile 18
 - pilot team 137
 - project backlog 110
 - replanning 272–276
 - selecting pilot project 110–112
 - status, tracking 214
 - team skills 18
 - User Acceptance Testing 270
- Adapt phase 101
- adapt week, schedule 266
- adapting
 - and replanning 265
 - at the end of an iteration 262–265
 - common questions 255
 - common reasons for 256–260
 - during adapt week at Acme Media 265–270
 - during an iteration 260
 - during an iteration at Acme Media 261
 - issues and options 264
 - re-evaluating priorities 263
 - to business changes at Acme Media 271
 - to change in business needs 257
 - to losing a team member 258
 - to low throughput 259
 - to technical constraints 258
- adoption
 - big-bang approach 325
 - evolutionary change 325
 - project-level to enterprise-level 323–325
 - reducing risk of 44–46
- agile 7–9
 - and compliance 74
 - and the bottom line 11–14
 - as a mindset 5–9
 - correct level 36
 - culture 74, 83
 - custom process 37
 - determining usefulness 12
 - enterprise-wide 323
 - goals 28–32
 - good characteristics for adoption 38–39
 - knowledge, lack of 40
 - levels, and SAMI 328
 - maturity levels 325
 - mindset 82–86
 - packaged methods 32–35
 - practices 9, 330
 - reasons to pursue 59
 - reducing risk of adoption 44–46
- agile coach 75–77
 - attributes 75
 - responsibilities 76
 - training core team 76
- Agile Manifesto 6
 - and SAMI 328
- agile principles 329
 - in practice 231
 - supporting during development and testing 231–237
- alignment team 136–150
- alternatives, working on, with the customer 240
- Andersen, Arthur 81
- anonymity, on surveys 300
- application, creating content to test 240
- approach
 - plan-driven vs. value-driven 11
 - value-driven 10

architecture
 assessment, artifacts 224
 vision 224
 archive requirements,
 validating 285
 archiving 286
 Archway Software
 existing process 20
 more agile process 21
 optimal agile process 22
 assessment tables 49
 assignments 238
 changing 239
 automation
 return on 250
 to save money 250

B

background process,
 enabling 288
 backout plan 292
 Beck, Kent 37
 Bezos, Jeff 40
 big-bang approach 325
 blackout window 292
 Boehm, Barry 44
 bonus functionality 234
 bottom line 11–14
 brainstorm, organized 157
 bug stomp 93, 248, 283
 bug tracking 226
 bugs
 tracing 235
 triage meeting 242
 build, automatic 225
 burn down chart 212
 business climate, changes in 271
 business need, changing 257
 business people, working with
 developers 232

C

Capability Maturity Model Inte-
 gration (CMMI) 326
 capacity, calculating 273
 career stage, of team
 members 85
 case study 18
 individuals 19
 introduction 17
 teams 19
 celebrations 294

change
 adapting to 29
 embracing 13
 embracing at Quecreek 5
 embracing during pilot 316
 welcoming 234
 chartering 140
 chasm 325
 CMMI, compared to SAMI 326
 coach 76
 attributes 75
 training core team 76
 coaching, need for 77
 code
 continuous integration
 80, 235
 starting from scratch 265
 writing additional 264
 Code Complete 37
 Cohn, Mike 184, 188, 190, 330
 collaboration, at Quecreek 5
 committing 210
 communication
 face to face 232
 plan 288
 planning, and iteration 0 229
 competitor, catching up
 with 279
 constraints
 and your process 35
 at Quecreek 4
 supporting 278
 verifying 281
 content, creating, to test
 application 240
 continuous integration 235
 contract, fixed-bid 41
 conversation, as effective means
 of communication 8
 core team 66–72
 approach to work 69
 creating at Acme Media 68
 example team 68
 kickoff meeting 69–71
 mentoring 76
 need for 68
 power and influence 66
 related to the sponsor 71
 role 71
 team-member
 characteristics 67
 training 76
 typical questions from 70
 versus outside consulting 67
 cowboy coding 61

Crossing the Chasm 322
 culture, why changing is
 difficult 74
 custom process 37
 customer
 advocate, during pilot 110
 availability 38, 165
 collaboration 13
 contract, supporting 278
 contracts 5
 feedback 235
 interest in working software 8
 involvement, during pilot 317
 involving in triage 264
 proxy 110
 refining requirements 257
 retention 12
 satisfying 7, 232
 working on alternatives
 with 240
 customer value, unrelenting
 pursuit of 80
 customization, avoiding 41

D

data testing 226
 data-recovery requirements,
 validating 285
 death march 279
 delivering software, percent
 complete 230
 and agile principles 231–237
 delivery
 accurate 12
 consistent 234
 timely 13
 urgency of 38
 Demings, Edward 244
 demonstrating 242
 at Acme Media 267
 co-presenting 269
 incomplete features 269
 demonstrations
 and team member
 personalities 268
 and user acceptance 270
 impromptu 263
 schedule 267
 structured 263
 technique, choosing 263
 User Acceptance Testing 263
 deployment
 and holidays 292
 blackout windows 292

- deployment (*continued*)
 - celebrating 294
 - collaborative 294
 - complexity 277
 - considerations 291
 - instant 291
 - pilot 294
 - plan 292
 - example at Acme Media 292
 - safe window 292
 - steps, planning 291
 - Deployment phase 102
 - design options, reviewing 240
 - design testing 247
 - developers
 - hiding 261
 - working with business people 7, 232
 - development
 - adapting to change during 29
 - and defects 242
 - and unit tests 241
 - beginning 237–239
 - distributed 40
 - environment, preparing 225
 - evolutionary approach 328
 - iterative 231
 - sustainable 8
 - what it really looks like 240
 - Development phase 100–101
 - development process, phases 91
 - disaster recovery 291
 - discipline, addressing 321
 - distractions, minimizing during iterations 236
 - documentation
 - location of 286
 - necessary 239
 - of design decisions 321
 - required vs. optional 31
 - done, defining 204
 - Dr. Agile 48
- E**
-
- early adopters 323
 - mapping to pilots 324
 - early majority 323
 - hurdles 325
 - elevator statement
 - creating 140–141
 - for Acme Media 141
 - email hell 232
 - environment
 - physical, and face-to-face communication 233
 - preparation 225
 - escalation process 287
 - estimating, more efficient 29
 - estimation
 - and planning poker 190
 - at Acme Media 189
 - evolutionary change 325
 - executive
 - buy-in, getting via readiness assessment 47
 - rewards for 62
 - showing agile value to 59
 - sponsor 63
 - support 58–65
 - executive team, communicating with 62
 - exploratory testing 248
 - Extreme Programming (XP)
 - 9, 34–35
 - characteristics 35
 - strengths 35
 - weaknesses 35
- F**
-
- feasibility 115–135
 - alternate methods 134
 - analysis 124
 - and technical analysis 131
 - feedback, reacting to 127
 - go/no go decision 132
 - in the big picture 116–118
 - modifying original idea 126
 - summarizing pros and cons 132
 - team
 - at Acme Media 119
 - choosing 119–121
 - feedback from 126
 - initial reaction 124
 - reviewing known requirements 121–132
 - team checklist 119
 - team review of the modified idea 131
 - tools to support 121
 - using wiki 134
 - when it isn't an option 134
 - Feasibility phase 97–99
 - feature
 - adding screen details 208
 - complete, defining 204
 - completing 239–240
 - labeling as complete 238
 - larger than expected 256
 - prioritizing 28
 - workflow 206
 - feature card 28
 - correct amount of information 156
 - example 154
 - example acceptance test 159
 - exercise 157
 - fields 154
 - structure 154–157
 - vs. technical design 160
 - feature cards 145, 153–169
 - and being lean 153
 - and common understanding 157
 - and compliance 165
 - and customer availability 165
 - and customer discussions 153
 - and early risk identification 156
 - and project complexity 165
 - compared to functional specifications 163
 - compared to use cases 162
 - compared to user stories 161
 - creating 157–161
 - creating at Acme Media 158
 - customer interview
 - example 159
 - customer-focused 156
 - hard copy vs. electronic 166
 - limitations 164–166
 - reviewing as a group 160
 - secondary benefits 156
 - sorting 175
 - Feature Description
 - Document 142
 - Feature Driven Development (FDD) 9
 - feature modeling 205–209
 - value of 209
 - feature shell 154
 - features 13, 117
 - and market impact 181
 - assigning to iterations 199
 - building in parallel 240
 - common understanding 144
 - considering dependencies 200
 - dependencies 175
 - features (*continued*)
 - discovered during an

- iteration 273
- festimating in detail 209
- evaluating risk 175–177
- grouping 178
- high-priority 200
- high-risk 200
- identifying hidden 207
- identifying tasks 209
- incomplete 260
- incomplete,
 - demonstrating 269
- initial value assessment 174
- introducing to team 141–142
- minimum needed 175
- outlining screens for 207
- prioritizing 180
 - at Acme Media 172
 - by value 174
- prioritizing, sequencing, and grouping 171
- removing from the project 179
- reviewing completeness at end of iteration 266
- sideline 275
- technical vs. customer facing 181
- testing 241
- work sequence 238
- feedback
 - converting into action 307
 - early 235
- Fibonacci scale 188
- fixed-bid contract 41
- functional specifications 164
 - compared to feature cards 163
 - identifying 239
- functional testing 247, 283
- functional tests, creating 242
- functionality, bonus 234
- funding 226

G

- Gladwell, Malcolm 328
- go/no go decision 132
- go-live checklist 290
- go-live decision 289
- groupthink 84

H

- help materials, finalizing 287

- Highsmith, Jim 30, 77
 - light-touch leadership 77
- holidays and deployment 292
- human-centric methods 13
- human-centric practices 320–321

I

- indicator 50
 - computing weight 54
 - weighted interval 54
- individual performance 84–86
- InfoQ 37
- innovation 13
- innovators, mapping to pilot projects 323
- integration testing 246
- integration, continuous 235
- interface testing, during iteration 0 228
- issue, deferring 265
- iteration
 - adapting at the end 262–265
 - adapting during 260
 - assigning features to 199
 - detailed estimates 212
 - end, as milestone 318
 - enhancing urgency 80
 - features discovered during 273
 - incomplete work 259
 - initializing 197
 - keeping length constant 275
 - length 196
 - number, during pilot 199
 - reviewing work completed 266
 - status, tracking 214
 - stretching 260
 - time between 197
 - typical tasks between 197
 - work sequence within 238
- iteration 0 223
 - and completing contracts 224
 - and environment preparation 225
 - and project funding 226
 - and the communication plan 229
 - and the marketing plan 229
 - architectural assessment 224
 - cheating during 228
 - finalizing team members 227
 - length 196

- preparing test content 226
- project tools 226
- starting the work early 228
- iteration capacity 211
- iteration plan and team ownership 210
- iteration planning
 - and acceptance tests 204
 - comparing estimates to capacity 212
 - sharing information 212
- iterative development 231

J

- JUnit 246

K

- kickoff meeting 201
 - core team 69–71
 - objectives 201
 - presenters 202

L

- laggards 325
- late majority 325
- launch, soft 281
- Lean 9
- lean manufacturing 244
- light-touch leadership 77
- load balancer 291
- load testing 226
 - requiring 93
- LoadRunner 226

M

- maintenance and support worksheet 286
- maintenance process, enabling 288
- manager
 - agile role model 79
 - in an agile environment 77–81
 - leading team to ownership 81
 - soft skills 78
 - vs. ScrumMaster 82

- manager (*continued*)
 working with other managers 78
 working with stakeholders 79
- marketing plan and iteration 0 229
- McConnell, Steve 37
- mentality, plan-driven 10–11
- methods, human-centric 13
- Microsoft Project 212
- migration
 confusion about process during 314
 cost of 60
 going too fast 41
 less progress during 314
 planning 292
 risks of 61
 starting to feel agile during 315
 team polarization during 315
 your role 71
- minimum functionality 170
- Moore, Geoffrey 322
- motivated team members 7
- motivation, of team members 84
- N**
-
- nominal value, mapping result range to 56
- nonfunctional requirements, validating 284–286
- NUnit 246
- O**
-
- objective statement 149
- optimistic range 55
- organizational characteristic 49
 result range 55
- Osherove, Roy 245
- ownership
 by the team 321
 leading team to 81
- P**
-
- people-dependent tasks 237
- personal performance and retrospective 306
- personality types, and demonstrations 268
- pessimistic range 55
- pilot
 mapping to early adopters 324
 mapping to innovators 323
 reducing risk with 294
- pilot project 106
 and external customers 110
 and team ownership 321
 breadth and depth 109
 common findings after 316
 confusion during 314
 customer involvement, evaluation 317
 delivering frequently, evaluation 318
 duration 107
 embracing change evaluation 316
 good, characteristics of 107–110
 human-centric evaluation 320–321
 measured against the agile enablers 316
 next steps 322
 number of iterations 199
 priority 108
 scoring to evaluate 316
 selecting at Acme Media 110–112
 size 107
 slower than legacy projects 314
 starting to feel agile 315
 subset of larger project 108
 team polarization 315
 technical excellence, evaluation 319
 too large 107
 too small 107
 using to identify weaknesses 106
 what Acme Media learned 316–321
- pilot team
 concerns 139
 feedback to the core team 139
 identifying 137–138
 preparing 138–140
 training on agile 139
- plan-based approach vs. value-driven approach 11
- plan-driven mentality 10–11
- planning
 "just enough" 79
 more efficient 29
- Planning phase 99–100
- planning poker 190
- post-mortem. *See* retrospective
- power distance 45
- practice
 conditional 91
 existing, to keep 92
 tying to agile principles 92
 valuable, identifying 91–93
- priorities
 business 174
 changing, consequences of 28
 clarifying 28
 re-evaluating 263
- problem, working 240
- process
 current
 deciding what to keep 95–91
 enhancing 94–103
 understanding 88–94
 defined vs. empirical 11
 designing more agile 95
 perfect, using for reference 93
 virtual phases 97–103
- processes, required vs. optional 31
- product
 envisioning 140
 highlights 141
 testing, during release 281
- product backlog 171–181
- production support 198
- productivity, increasing during transition 46
- progress matrix 216
- progress, working software as primary measure 8
- project
 approving, approaches 133
 delivering 30
 feasibility, reevaluating 257
 funding 226
 kickoff meeting 202
 medium-priority, examples 108
 objective statement 149

project (*continued*)
 status 29
 tools for tracking 217–219
 team, size and expense 133
 time constraints 198
 timeline, determining 198
 project backlog 110
 project cheating, starting work
 early 228
 Project Management Lifecycle
 (PMLC) 30
 project structure, correct level
 of 30
 project team
 common vision 136
 dedicated during an
 iteration 227
 defining and finalizing 227
 large 40
 minimizing distractions 236
 project worksheet 146–150
 feature highlights 149
 issues and risks 149
 just enough 148
 milestones 149
 objective statement 149
 sections 148
 stakeholders 149
 technical considerations 149
 user/customer benefits 149
 proof of concept (POC) 319
 prototyping 319
 in iteration 0 228
 purging requirements,
 validating 285

Q

quality level 282
 quality, building into
 product 244
 Quecreek 3–5
 QuickTest 251

R

readiness assessment
 and organizational
 characteristics 49
 calculating results 54–57
 conducting 49–57
 importance of 44
 reviewing results 52–57
 to reduce risk 44

to support productivity 47
 using to get executive
 buy-in 47
 readiness-assessment table
 49–52
 Recovery Point Objective
 (RPO) 285
 Recovery Time Objective
 (RTO) 285
 redesign 265
 regulatory deadline,
 supporting 279
 release 289–295
 status, tracking 216
 release plan 117
 and iteration 0 length 196
 delivery range 203
 highly visible 199
 pieces, defining 196–199
 tools for creating 203
 release planning
 and development iteration
 length 196
 and feature priority 200
 and overall timeline 198
 and support groups 202
 assigning features to
 iterations 199
 kickoff meeting 201
 time between iterations 197
 value in each iteration 200
 release schedule 279
 releasing software
 and deployment
 planning 292
 and quality level 282
 and resource deadlines 279
 and support groups 286–288
 every iteration 280
 final product approval 284
 for a compliance
 deadline 279
 reducing risk with a limited
 pilot 294
 to support a constraint 278
 to test the product 281
 triggers 278
 validating performance 284
 verifying non-functional
 requirements 284–286
 when there is enough
 value 280
 when to 278–282
 replanning 265
 at Acme Media 272–276

requirements
 amount of
 documentation 156
 changing, welcoming 7, 234
 customer refinement 257
 evolving or volatile 38
 fixed 11
 modifying 264
 nonfunctional,
 validating 284–286
 resource pool 137
 resources, consistent 39
 result range, mapping to nomi-
 nal value 56
 retrospective
 action items 307
 and anonymity 300
 and complaints 297
 and personal
 performance 306
 and team dynamics 305
 attendee thoughts at Acme
 Media 305
 between iterations 266
 ensuring effectiveness 300
 facilitator 304
 frequency 304
 meeting, what to expect
 304–305
 objectives of 298
 prioritizing issues 307
 related to project success 298
 root-cause analysis 307
 setting expectations for 298
 survey questions 300
 survey results 300
 surveys 300
 the meeting 302
 revenue, increasing 12
 reverse engineer 88–94
 sticky notes for areas to
 improve 95
 time boxing 89
 when to skip 103
 with index cards 88
 reward structure, for teams 84
 risk
 evaluating, for features
 175–177
 identifying early 156
 risk management 30
 roadblocks, overcoming 40
 Rothman, Johanna 76

S

-
- Sage Analyst 286
 - SAMI. *See* Sidky Agile Measurement Index
 - Sarbanes-Oxley 279
 - SOX 165
 - Satir, Virginia 47
 - scalability, considering 81
 - schedule
 - highly visible 199
 - modifying 318
 - predetermined, meeting 279
 - scope
 - continuing during development 240
 - revisiting after customer refinement 257
 - scope creep 41
 - in an agile environment 236
 - scope drift 234
 - scoping, during iteration 0 228
 - screens
 - adding details 208
 - for features, outlining 207
 - Scrum 9, 32–34
 - characteristics 33
 - iteration, initializing 197
 - strengths 33
 - weaknesses 33
 - ScrumMaster
 - role 82
 - vs. manager 82
 - SDLC. *See* Software Development Lifecycle
 - search support 286
 - self-organizing teams 9
 - side discussions,
 - eliminating 321
 - sideline features 275
 - Sidky Agile Measurement Index (SAMI) 314, 325–330
 - agile levels 328
 - agile practices 330
 - agile principles 329
 - compared to CMMI 326
 - sign-off 10
 - silo 39
 - simplicity 8
 - focusing on 234
 - simplicity in design 81
 - Six Sigma 64
 - smoke test 81
 - smoke testing 93
 - soft launch 281
 - soft skills 78
 - software
 - delivering frequently 13, 318
 - delivery, and agile principles 231–237
 - working, as primary progress measure 8
 - working, delivering frequently 7
 - Software Development Lifecycle (SDLC) 30
 - solution, purchasing 265
 - sponsor
 - executive 63
 - obtaining 63
 - role of 63
 - sprint, retrospective 197
 - stakeholders 149
 - working with managers 79
 - status
 - measuring 213–219
 - tools 226
 - tools for tracking 217–219
 - tracking 216
 - visibility 213–219
 - story points 187
 - at Acme Media 189
 - definition 187
 - related to tasks 189
 - related to work estimates 188
 - success, defining 204, 295
 - support groups 286–288
 - Surowiecki, James 185
 - survey
 - for retrospective 300
 - scoring 300
 - sustainable development 8
 - system availability,
 - validating 284
 - system monitoring,
 - enabling 287
 - system response time 285
-
- T**
 - tasks
 - assignments 238
 - not permanent 210
 - estimating 209
 - identifying 209
 - people-dependent 237
 - TDD. *See* Test Driven Development
 - team 20–23
 - aligning 136–150
 - and groupthink 84
 - buy-in 28
 - capacity during iteration 198
 - capacity, calculating 273
 - culture and roles 83
 - immature 41
 - individual performance 84–86
 - introducing to project features 141–142
 - large 40
 - leading to ownership 81
 - losing a member 258
 - minimizing distractions 236
 - motivation and rewards 84
 - one-time 41
 - polarization 315
 - production support for 198
 - reviewing performance and velocity 265
 - selecting for feasibility vs. regular projects 120
 - self-organizing 9
 - throughput, lower than expected 259
 - tuning behavior of 9
 - with agile mindset 82–86
 - with specialized skill sets 41
 - team members
 - career stage 85
 - motivated 7
 - teamwork 39
 - technical constraint, adapting to 258
 - technical excellence 8, 13, 233
 - best practices 80
 - during pilot 319
 - Technology Adoption Lifecycle 322–325
 - categories 322
 - test
 - automation 226
 - content 226
 - creating content for 240
 - test cases, defining, during iteration 0 228
 - Test Driven Development (TDD) 235
 - test harness 269
 - TestDirector 205
 - testing
 - and usability 248
 - automating 80

testing (*continued*)
 automation 249
 complete system 282
 early and often 235
 exploratory testing 248
 final 282–286
 functional testing 247
 functional/usability 283
 integration testing 246
 unit testing 245
 when to automate 250
 third party
 completing contracts
 with 224
 that doesn't deliver 259
 time constraint 198
 projects without 200
 time tracking, tools for 226
 time-boxing 319
 timeline, determining 198
Tipping Point, The 328
 tracking time 226
 tradeoff matrix 145–146
 and adapting 263
 and feature scope 146
 and flexibility 146
 and priorities 145
 and resources 146
 different spin 147
 training
 and iteration 0 228

need for 77
 plan 288
 suggested outline 76
 transition, increasing productivity during 46
 triage 264
 meeting 242
 Turner, Richard 44

U

UAT. *See* User Acceptance Testing
 unit test
 and development 241
 creating 240
 creating incrementally 242
 example at Acme Media 246
 unit testing 245
 requiring 93
 uptime percentage,
 increasing 284
 urgency to delivery 38
 usability testing 283
 use cases
 compared to feature cards 162
 types 162
 usefulness, delivering in every iteration 200
 User Acceptance Testing (UAT) 263, 270–271

user stories
 compared to feature cards 161
 defined 117
 users, maximum number supported 285

V

value, unrelenting pursuit of 80
 value-driven approach 10
 velocity
 evaluating at Acme Media 272
 reviewing 265
 version control 225
 VersionOne, agile survey 11
 Virginia Satir change curve 46
 and SAMI 327
 virtual phases 116

W

waterfall lifecycle 20
 waterfall model 100
 weight of indicator 54
 weighted interval 54
 wiki, using in feasibility 134
 work backlog 237
 work, delaying after customer refinement 257
 workaround 264
 workforce, motivated 13
 workstation, setting up 226