Index

A
Action plan, 88–95
  alternative approaches in, 91–93
  implementation issues in, 93–95
  and overall business environment, 89
  and project environment, 89–91
  for project managers, 193–194
  questions for developing, 88–89
Adaptability:
  to achieve business outcomes, 58
  balancing control and, 191
  to fit projects and problems, 110–111
  as key change needed, 109
  in SDLC models, 191
Adaptive approaches, 5, 188–189
Adaptive life-cycle models, 168, 178–180
Adapt Phase (APM model), 122
Agile, 4
  agility vs., 4–5
  benefits and tradeoffs with, 57–61
  continuous improvement in, 35
  defining, 3–5
  general practices of, 47–56
  history and overview of, 38–44
  and Lean Software Development, 36–38
  levels of planning in, 50–51
  misconceptions about, 14–19
  movement toward, 67–70, 192, 195
  obstacles to becoming, 62–67
  patterns in adoption of, 71–73
  perceptions of, 44–47, 124–125
  as “program du jour,” 9
  and project management, 10–14. See also Agile project management
  quality improvement with, 62
  Sapient|Approach, 73–84
  traditional Waterfall vs., 7–9
  uncertainty reduction in, 152
  Agile Manifesto, 18, 36, 39, 55, 67–68
  Agile Manifesto Principles, 39–43
  Agile Modeling (AM), 219–221, 228
  Agile project management, 17–18, 101–130
  mindset for, 107–111
  models for, 119–123
  and PMBOK® Guide, 124–130
  practices in, 112–113
  principles of, 113–116
  roles in, 101–107
  skills for, 98, 111–112
  techniques used in, 117–119
  trend toward, 13–14
  “Agile Project Management” (Daniel J. and John D. Fernandez), 64
Agile Project Management (Jim Highsmith), 93
  agile tools and techniques, 48
  APM model, 120–123
  balancing planning and execution, 155–156
  better product development, 58
  capabilities, 112
  chaos and structure, 135
  focus on customer value, 108
  leadership, 157
  linearity vs. evolution, 68
  PMBOK® Guide, 124
  product development, 59
  repeatable and reliable processes, 136, 137
  risk management, 150
  “tailoring-down” syndrome, 126–127

235
Agile Project Management (APM) model, 120–123
Agile Project Management triangle, 13
Agile Unified Process (Agile UP), 221–224
defined, 229
as iterative emergent model, 176
other methods vs., 55–56
“Agile Unified Process” (ambysoft.com), 222–223
“Agile? Waterfall? How about WetAgile?” (Steve Pieczko), 69
Agility:
agile vs., 4–5
and control, 5, 8, 109
“Aligning PMBOK Agile” (Brian Bozzuto), 126
All-or-nothing thinking, 15, 85, 87
AM (Agile Modeling), 219–221, 228
Ambler, Scott, 145, 219, 221
AOL, 188–189
APM (Agile Project Management) model, 120–123
Architecture Envisioning (AM), 220

B
Balancing Agility and Discipline—A Guide for the Perplexed (Barry Boehm and Richard Turner), 14, 143, 158
Batch sizes, 31–33
“The Blending of Traditional and Agile Project Management” (Kathleen Hass), 169, 206
Boehm, Barry, 7, 14, 143, 158
Bozzuto, Brian, 124–126
BPR (business process reengineering), 94
Build by Feature (FDD), 204
“Burn down” chart (Scrum), 214
“Burning platform,” 90
Burns, Martin, 38
Business analysis:
as skill for agile project managers, 111
for traditional vs. agile projects, 105
Business Analysts, 103, 104, 106–107
Business environment, and readiness for agile, 89
Business Excellence model, 24–26
Business outcomes, 57–59, 86
Business processes, 141–142
Business process reengineering (BPR), 94
Business rules, 142
Business value:
failure to provide, 12
and traditional Project Management triangle, 11

C
Capability Maturity Model Integrated (CMMI), 76, 94
Capella, Jamie, 10
Change:
responsiveness to, 128–129
in traditional vs. learning organizations, 73
Change management:
with Scrum adoption, 62
success factors for, 90–91
for traditional vs. agile projects, 105
Chaos, 135
“CHAOS Summary 2009” (Standish Group), 10
Churchill, Winston S., 154
Close Phase (APM model), 122
CMMI (Capability Maturity Model Integrated), 76, 94
Code refactoring, 208–209, 229
Cohn, Mike, 62, 94
Collaboration:
customer, 114
in XP, 201
Collective ownership, 114–115, 136, 200
Commitment, organizational, 66
Communities of practice, 144
Competitive advantage, 60
Complexity:
- in requirements management, 142
- with SDLC models, 181
Conceptual integrity, 226
Concurrent processing (concurrent engineering), 33
Configuration management:
  - documentation for, 161
  - and flow, 33
Consensus building, 117–118
Construction Phase (AM), 223
Continuous improvement, 19, 35
  - in agile methodologies, 116
  - with SDLC, 181
  - in traditional vs. learning organizations, 73
  - in XP, 201
Continuous integration, 209–210, 229
Control:
  - balancing agility and, 5, 8, 109
  - balancing flexibility and adaptability with, 191
  - in PMBOK® vs. in practice, 128–129
  - for traditional vs. agile projects, 104
  - in traditional vs. learning organizations, 72
  - with Waterfall model, 6–7
Coordination of resources, 105
Corporate culture, 63–66
Corporate Executive Board, 10
Cost estimates, 12
Cost reductions, 60–61
COULD requirements, 54, 147
Cross-functional collaborative approach, 64–65
Cross-functional synergy, 60
Crossing the Chasm (Geoffrey A. Moore), 50
Customer (XP), 3
Customer collaboration, 114
Customer satisfaction, 60
Customer value:
  - agile focus on, 108–109
  - as lean principle, 24–26
Customization of projects, 134–135

D
Daily planning, 51
Daily Scrum Meetings, 214–215
Daily standup meetings, 117, 214
“Death march” projects, 16
Decision making, delaying, 225
Dell Computer, 31
DeMarco, Tom, 148
Design by Feature (FDD), 204
Developer-centric orientation, 36–37
Development teams, 95
Documentation:
  - in SDLC models, 160–162
  - in traditional plan-driven life cycle models, 170
Drucker, Peter F., 157
DSDM Consortium, 215
Dynamic Systems Development Method (DSDM), 215–219
  - comparison of other methods and, 55–56
  - defined, 229
  - principles of, 217

E
“Earth of chaos,” 135
Effective Project Management—Traditional, Agile, Extreme (Robert Wysocki), 109, 134, 145, 155, 164, 165, 172
80/20 rule, 218
Eisenhower, Dwight D., 154, 157
Elaboration Phase (AM), 222–223
Empire State Building, 31
Employees:
  - morale of, 60
  - respect for, 34, 35, 48
Empowerment, 48
  - in Lean Software Development, 226
  - in traditional vs. learning organizations, 72
Engagement, in traditional vs. learning organizations, 73

*The Enterprise and Scrum* (Ken Schwaber), 95

Enterprise transition team, 95

Enterprise Unified Process, 176

Envision Phase (APM model), 120–123

Epics, 53, 229

Executable Architecture Release phase (Sapient|Approach), 80

“Executive Brief | Are You Ready for Agile?” (Sapient Corporate White Paper), 89–90

Explore Phase (APM model), 122, 123

Extreme plan-drive Waterfall, 163–164

Extreme Programming (XP), 3–4

as adaptive approach, 5, 179
code refactoring with, 208
declared, 229–230
as development approach, 199–202
FDD vs., 203–205
other methods vs., 55–56
pair programming with, 207
in Sapient|Approach, 75, 76
with Scrum, 110

Failure, 73, 116, 210

Feature-Driven Development (FDD), 202–205
comparison of other methods and, 55–56
declared, 230
pair programming with, 207
XP vs., 203–205
“Feature-Driven Development” (wikipedia.org), 204

“Feature Driven Development (FDD) and Agile Modeling” (agilemodeling.com), 202

Feature list (FDD), 203–204

Feature sets, 32

Fernandez, Daniel J., 64

Fernandez, John D., 64

Fist Block, 118

“Fist of five” approach, 117–118

“The 5 Why’s Method,” 146

Flexibility:
to achieve business outcomes, 58
balancing control and, 191
as key change needed, 109
in SDLC models, 133–135, 191

Flow:
and batch size, 31–33
and concurrent processing, 33
as lean principle, 30–33
Focus, with timeboxing, 118

Force-fitting methodologies, 35, 86, 110

Functional department managers (Scrum), 104

Functional management, 105

Fusion™ phase (Sapient|Approach), 80

“Fuzzy front end,” 151, 153, 154

G

Generative approach, in *PMBOK®* vs.
in practice, 125–126

“‘Gotchas’” (Sapient Corporate White Paper), 14, 17

Gottesman, Erik, 32–33, 38, 74–77, 83, 107

“Growing Agility in a Large and Distributed Enterprise” (Erik Gottesman), 74–77

*A Guide to Project Management Body of Knowledge (PMBOK® Guide)*, see *PMBOK® Guide*

H

Hammer, Michael, 141

Hass, Kathleen, 169, 206

Highsmith, Jim:
on adaptive vs. optimizing approaches, 168
on agile, 55–56
on Agile Manifesto statements, 39
on Agile Project Management model, 120–123
on agile tools and techniques, 19, 58
on agile triangle, 13
on balancing planning and execution, 155–156
on better product development, 58
on business objectives, 57
on capabilities, 112
on chaos and structure, 135
on creators of Agile Manifesto, 55
on focus of agile methodologies, 141
on focus of project managers, 11, 12
on focus on customer value, 108
on leading, 157
on linearity vs. evolution, 68
on PMBOK® Guide, 124
on product development, 59
on repeatable and reliable processes, 136, 137
on risk management, 150
on “tailoring-down” syndrome, 126–127
“History: The Agile Manifesto” (Jim Highsmith), 55–56
Humanistic value orientation, 127–128
Hybrid approaches, 68–70
comparisons of, 55–56
designing and implementing, 194
reasons for adopting, 90–91

Implementation, 86. See also specific methodologies
as action plan issue, 93–95
agile vs. lean, 37
in PMBOK® vs. in practice, 125
of pure agile project management, 194–195
Inception Phase (AM), 222
INCOSE (International Council on Systems Engineering), 23–24
Incremental improvements, 91–92, 193–194
Incremental life-cycle model, 167, 173–174
Integrity, 226–227
Intent, in PMBOK® vs. in practice, 125
International Council on Systems Engineering (INCOSE), 23–24
“An Introduction to Agile Modeling” (agilemodeling.com), 219
“An Introduction to Feature-Driven Development” (agiledzone.com), 203–204
“Introduction to Test-Driven Design” (agiledata.org), 205–206
Iteration 0, 203
Iteration planning, 51, 200
Iterative approaches:
  factors favoring, 183
  implementing, 92–93
  in prioritizing requirements, 147
  uncertainty reduction in, 152
Iterative development processes, 32
Iterative emergent life-cycle model, 167–168, 176–178
Iterative plan-driven life-cycle model, 167, 174–176, 185–188

J
“Just Barely Good Enough” (JBGE), 145
Just-in-time planning, 49–50, 152

K
Kanban, 29–30
“Kanban Development Oversimplified,” 29–30
Kettering, Charles, 146
Krebs, Jochen (Joe), 188

L
“Last responsible moment,” 152
Leadership:
  agile project manager skills in, 111–112
  by business side of organization, 17
  and hybrid approach adoption, 90
  in moving toward agile, 192
  for organizational transformation, 192–193
  in SDLC models, 156–158
Leadership: (continued)
style of, 63–64
thought, 72
Lean (lean manufacturing, lean production), 21, 22
customer value in, 24–26
defined, 22, 230
flow in, 30–33
main principles of, 224–228
mapping the value stream in, 26–27
perfection in, 34–35
pull in, 27–30
respect for people in, 34
Lean Software Development, 21–38, 224–228
and agile, 36–38
defined, 230
principles behind, 21–35
Lean Software Development (Tom and Mary Poppendick), 159, 225–228
Lean Systems Engineering, 23
Learning, amplifying, 225
Learning environment, 65–66
Learning organizations, 72–73
Lister, Timothy, 148
Little’s Law of Queuing, 31

M
Manage Iterations phase
(Sapient Approach), 80
Management of uncertainty, 151–154
Management style, 63–64
Mapping the value stream, 26–27
MDD (Model-Driven Development), 82–83
Mindset, for agile project management, 67–68, 107–111
Models, agile project management, 119–123
Model-Driven Development (MDD), 82–83
“Model-driven Development Through the Agile Looking Glass” (Erik Gottesman), 83
Modeling with a Purpose (AM), 220
Model-Storming (AM), 220
Moltke the Elder, Helmuth von, 154
Moore, Geoffrey A., 50
MoSCoW technique, 54, 146–147
Multilevel testing, 143
Multiple-team projects, 119–120
MUST requirements, 54, 146

N
Needs, differentiating wants from, 145–147
Non-value-added work, 22

O
Operational management orientation, 36–37
Operational process performance, 26
Optimizing models, 168
Organizational change:
developing and leading, 192–193
need for, 86
Organizational commitment, 66
Organizational effectiveness, 60
Organizational maturity, 44, 94
Organizational practices, 48–49
Overall object model (FDD), 203
Ownership, collective, 114–115, 136, 200

P
Pair programming, 207–208
defined, 230
in XP, 200
Parkinson’s Law, 119
Patton, George S., 154, 157
“Paving Cowpaths” (Jim Highsmith), 141
People, respect for, 34, 35, 48
People management skills, 111–112
Perceived integrity, 226
Perception gap, 46
Perceptions of agile methodologies, 44–47
Perfection principle, 34–35
Phase “gates,” 171
Pieczko, Steve, 69
PIM (Platform-Independent Model), 82–84
Pizza box methodology, 14
Plan by Feature (FDD), 204
Plan-driven approaches, 4–5. See also Traditional plan-driven life-cycle models
factors favoring, 183
“tailoring-down” syndrome in, 126–127
Planning:
documentation for, 160–161
most common problems in, 155
in PMBOK® vs. in practice, 128–129
for risk reduction, 148–150
rolling wave, 113–114, 178
in SDLC models, 154–156
in traditional plan-driven life-cycle models, 70
Planning Poker, 52
Planning practices, 49–51
Platform-Independent Model (PIM), 82–84
PMBOK® Guide:
and agile project management, 124–130
applying principles from, 10
merging thinking of agile and, 130
stereotypical perceptions of, 124–129
PMI® (Project Management Institute), 191
PMO (Project Management Office), 19, 103
Poppendiek, Mary, 26, 27, 31, 36, 37, 159, 224
Poppendiek, Tom, 36, 37, 159, 224
Portfolio governance, 48
Post-Project Phase (DSDM), 218
Practices, agile, 47–54
organizational, 48–49
planning, 49–51
project management, 112–113
requirements definition, 51–54
technical, 48, 199–210
Predictive approaches, 4–5
Pre-Project Phase (DSDM), 218
Prescriptive approach, 125–126
Principles:
of Agile Manifesto, 68
of agile project management, 113–116
common to agile and lean, 37
of DSDM, 217
of Lean Software Development, 21–35
phases of, 218
in PMBOK® vs. in practice, 125
of SDLC models, 131–132
“Principles Behind the Agile Manifesto,” 68
The Principles of Product Flow (Don Reinertsen), 132
Prioritization, of requirements, 54, 220
Problems, adapting methodologies to, 110
Process design:
agile project managers’ capabilities in, 112
in SDLC models, 135–136
and training, 134–136
Process direction, for traditional vs. agile projects, 104
Process improvement, in XP, 201–202
Process orientation:
in lean and agile, 36
in PMBOK® vs. in practice, 127–128
Process standardization, in traditional vs. learning organizations, 72
Product backlog, 53, 213–214, 230
Productivity improvement, 60–61, 118
Product Owner (Scrum), 103, 104, 106, 212
Program du jour, 9–10, 85–86
Progress tracking, for traditional vs. agile projects, 105
Project, adapting methodologies to, 110
Project delivery frameworks, 211–233
  Agile Modeling, 219–221
  Agile Unified Process, 221–224
  Lean Software Development, 224–228
  Scrum, 211–215
Project environment, and readiness for agile, 89–91
Project Life-Cycle Phase (DSDM), 218
Project management:
  agile, see Agile project management documentation for, 160–161
effectiveness of, 10
fundamental problem with, 10–11
future direction of, 97
impact of agile on, 10–14
Project Management Institute (PMI®), 191
Project Management Office (PMO), 19, 103
Project Management triangle:
  agile, 13
  traditional, 11
Project managers:
  action plan for, 193–195
  as “chefs” vs. “cooks,” 97–98, 191–192
  key challenge for, 98
  skills for, 111–112
Project methodologies. See also specific methodologies
  “programs du jour,” 9–10
  selecting, 8–9
Project scope, in SDLC models, 181
Prototype models, 176, 230–231
Pull approach, 27–30
Pure agile, 163. See also Scrum
  comparisons of, 55–56
  defined, 5
  implementing, 93, 194–195
  levels of project management in, 102
  tradeoffs with, 163–164
Pure Waterfall approach, 184–185
Push approach, 27–29

Q
  Quality control, 34–35
  Quality improvement, 62

R
  Rapid Application Development (RAD), 215
  Rapid development, 225–226
  Rational Unified Process (RUP®):
    AM as simplified version of, 221
criticisms of, 224
defined, 231
  as iterative emergent model, 176–178
  and Sapient® Approach, 80
Realization of time spent, with timeboxing, 118–119
Regulatory environment, 66–67
Reinertsen, Don, 132
Release management, flow and, 33
Release planning, 51
Release Planning Meeting (Scrum), 214
Release Planning Process (XP), 200
Reliable processes, repeatable processes vs., 136–137
Repeatable processes:
  and flexibility, 134
  reliable processes vs., 136–137
Requirements definition, 51–54
  with story points, 62–54
  in traditional plan-driven life-cycle models, 70
  with user stories, 51–52
Requirements elicitation:
  in Scrum approach, 110
  in SDLC models, 138, 140
  for traditional vs. agile projects, 105
Requirements management:
  business process considerations in, 141–142
  complexity considerations in, 142
defining requirements, 138, 140
  feature sets in, 32
guidelines for, 147
in lean and agile approaches, 29
prioritization of requirements in,
   145–147
   push approach to, 27–29
   in SDLC models, 140–147, 181
   story pipelining in, 32–33
   supportability considerations in, 144
   testing considerations in, 142–144
Requirements traceability, 142, 161
Resources management, 105
Respect for people, 48
   and continuous improvement, 35
   as lean principle, 34
Responsiveness to change, 128–129
Rico, David, 4, 23
Rigidity, flexibility vs., 133–135
Risk, 59
Risk environment, 66–67
Risk identification stage, 148–149
Risk management:
   in SDLC models, 148–151, 181–182
   for traditional vs. agile projects, 106
Risk mitigation stage, 149
Risk tolerance, 150
Roadmap planning, 51, 180–181
Roles, project management, 101–107
   Business Analyst, 106–107
   traditional vs. agile, 103–106
Rolling wave planning, 113–114, 178
Roosevelt, Theodore, 157
RUP®, see Rational Unified Process

S
Sapient|Approach (S|A), 73–84
   methodology description, 78–84
   process methodology selection and
design, 74–78
   for Sapient’s unique challenges,
   73–74
Sapient Corporation, 73–84
Sapient Corporation White Papers:
   on business stakeholders, 17
   on discipline, 14
   on readiness for agile, 89–90
Schedule estimates, 12
Schwaber, Ken, 95, 128
Scrum, 3, 17
   as adaptive approach, 5, 179
   continuous improvement in, 35
   defined, 231
   levels of project management in, 102,
   112–113
   obstacles in transition to, 62–63
   other methods vs., 55–56
   as project delivery framework,
   211–215
   and project management, 110
   and requirements elicitation, 110
   in Sapient|Approach, 75
   self-organizing teams in, 102
   use of XP with, 110
ScrumMaster, 102, 103, 120, 192,
   211–212
   “Scrum of Scrums”
   (learnsoftwareprocesses.com), 120
Scrum-of-Scrums approach, 80, 110,
   119–120
Scrum rollout teams, 95
SDLC models, see Software
development life cycle models
Self-organization, 48, 102
SHOULD requirements, 54, 147
Shu Ha Ri, 131–132
Six Sigma, 9, 71
Sliger, Michelle, 130
Software development life cycle
   (SDLC) models, 131–189
   adaptive model, 178–180
   applying lean concepts to, 22–23
   categories of, 164–168
   defined, 132–133, 231
   documentation in, 160–162
   extreme plan-drive Waterfall, 163
   flexibility vs. rigidity in, 133–135
   guidelines for using, 180–182
   incremental model, 173–174
   iterative emergent model, 176–178
   iterative plan-driven model, 174–176
   leadership in, 156–158
Software development life cycle (SDLC) models, (continued)
management of uncertainty in, 151–154
planning in, 154–156
principles underlying, 131–132
process design in, 135–136
pure agile, 163
reliable vs. repeatable processes in, 136–137
requirements management in, 138, 140–147
risk management in, 148–151
selecting, 138, 139, 182–189
traditional plan-driven models, 168–173
training in, 135–136, 158–160
variations on, 164

The Software Project Manager's Bridge to Agility (Michelle Sliger), 130

Speculate Phase (APM model), 121–123
Spikes, 53–54, 232
“Sprints” (Scrum), 213
Sprint Backlog (Scrum), 214
Sprint Kickoff Meeting (Scrum), 214
Sprint Retrospective (Scrum), 215
Sprint Review (Scrum), 215
Stakeholder participation, in AM, 220
Standish Group, 10
Story pipelining, 32–33
Story points, 52, 232
Student Syndrome, 119
Subsequent Releases phase (Sapient/Approach), 80

Succeeding with Agile (Mike Cohn), 62, 94

Supportability, in requirements management, 144
Systems thinking, 107–108, 132

Team:
empowering, 226
Scrum, 212
Teamwork, 48
Technical practices, 48, 199–210
code refactoring, 208–209
continuous integration, 209–210
Extreme Programming, 199–202
Feature-Driven Development, 202–205
pair programming, 207–208
Test-Driven Development, 205–207

Technical skills, for agile project managers, 111
Test-Driven Development (TDD), 205–207
in AM, 220
defined, 232
in XP, 200

Testing:
and flow, 33
in requirements management, 142–144

Test management, 143–144
Thought leadership, 72
Timeboxing, 118–119, 199, 200
“Timeboxing” (agilehardware.com), 118–119

Time management, 119

Total Quality Management (TQM), 35
Toyota Production System, 29–30

Tradeoffs:
in adopting agile principles, 43–44, 57–61
with pure agile and extreme plan-drive Waterfall, 163–164
with unresolved uncertainties, 152–153

Traditional development approaches:
batch-size analogy in, 32
force-fitting methods in, 35
improving vs. discarding, 22
perceptions of, 8, 10, 45–46
truth about, 15–16

Tailoring, of projects, 134–135
“Tailoring-down” syndrome, 126–127

TDD, see Test-Driven Development
Traditional plan-driven life-cycle models, 168–173. See also Waterfall model
distinguishing characteristics of, 170
potential variations of, 170–171
risks and limitations with, 171–173
strengths and benefits of, 171
Traditional project management:
Agile Manifesto Principles vs. practices of, 39–43
reliability vs. repeatability of processes in, 137
roles in, 103–106
Traditional Project Management Office (PMO) organizations, 19
Traditional Project Management triangle, 11
Training:
and flexibility of processes, 133–134
and process design, 134–136
in SDLC models, 135–136, 158–160
Transition Phase (AM), 223
Transparency, 49, 66
Trust, 49
Turner, Richard, 7, 14, 143, 158

U

UCLA (University of California at Los Angeles), 16
Uncertainty, management of, 151–154
Unified Process, 177
University of California at Los Angeles (UCLA), 16
Upfront planning and control, in PMBOK® vs. in practice, 128–129
User Personas, 53, 232
User stories, 30, 199–200
defined, 232–233
form for, 52
in requirements definition, 51–54
story points in, 52
User Story Cards, 30

V

Validation, emphasis on, 116–117
Value-added work, 22
Value stream, mapping, 26–27
Verification, validation vs., 116–117
Vision, 90
Vision planning, 50
Vision statement, 146

W

Waltzing with Bears (Tom DeMarco and Timothy Lister), 148
Wants, differentiating needs from, 145–147
Waste(s), 22, 27
eliminating, 224–225
with large batch sizes, 31
Waterfall model, 168–173
agile vs., 7–9
control with, 6–7
defined, 5, 233
extreme plan-driven, 163–164
phases in, 5–6
pure, 184–185
uncertainty reduction in, 152
“What is DSDM?” (selectbs.com), 215–217
Whole Team (XP), 3
WON’T requirements, 54, 147
Wooden, John, 16
Wysocki, Robert (Bob):
on cooks vs. chefs, 97–98
on life-cycle models, 164, 165
on plan-driven approaches, 172
on planning, 155
on tailoring project management approaches, 109, 134
on wants vs. needs, 145

X

XP, see Extreme Programming

Y

Yourdon, Edward, 16