abbreviations. See acronyms/abbreviations
accept or ignore option, 69
acceptance (user-acceptance) testing, 2, 7–8,
500–501
defined, 601
jump-start for, 598
project life cycle and, 500–501
quality risk categories and, 21–24
Access (Microsoft), 154, 271, 271n4, 272
accountability
bug-tracking database and, 160–161
test lab and, 305
tragedy of the commons and, 305
accuracy
audience and, 407–409
bug reports and, 151
bug-tracking database and, 194
acoustics tests, 560–561
acronyms/abbreviations
defining, 53
Omninet Marketing Requirements Document, 569
action components, 82–84
action verbs, test plans and, 76
Actual Date column, 207
Adams, Scott, 320n2, 533
“AddValidUser.case”, 135
ad-hoc testing, 4, 394, 395, 601
Advanced Software Testing (Black)
Volume I, 105n4, 335n6, 348n7
Volume II, 179n7, 335n6, 348n7, 538n19
Aesop’s fables, 68n4
After the Gold Rush (McConnell), 336
Agile methodologies, 5, 16, 401, 401n5,
450, 502n1, 506–507, 536–547. See also
sprints
case study, 536–547
challenges, 401, 536–547
Extreme Programming, 5, 506, 507
online information, 401n5
Scrum, 450, 506, 536, 538, 539, 540, 541, 542,
543, 546
alarm card, 18
all things held equal (ceteris paribus), 189
allocation of effort, 15
all-pairs tables, 115, 115n7. See also orthogonal
arrays
alpha tests, 8
altitude, 556
American Society for Quality (ASQ), 335,
382n1
American Software Testing Qualifications
Board, 335n6. See also ISTQB
analytical risk-based testing. See risk-based
testing
Anderson, Hans Christian, 405
Anna Karenina (Tolstoy), 514, 515n7
anomaly, 147
anomaly reports (bug reports), 75, 177, 177n6,
223. See also bug reports
anomaly-report (bug-report) template,
176–178
anonymization process, 136–138
APIs (application programming interfaces), 6, 19
scope creep and, 358
test harnesses, 81, 237
application domain skills, 328
application programming interfaces. See APIs
arbitrary sprint durations, 542–543
architecture. See Omninet; system architecture
test-system architecture
Arrhenius equation, 559
artifacts. See test artifacts
The Art of Software Testing (Myers), 1n1, 134, 187, 322n4, 329n5
aspiring hero, 323
ASQ (American Society for Quality), 335, 382n1
assertiveness, milquetoast v., 324–325
assessments (RBCS). See also Critical Testing Processes; RBCS
quality methodology, 597
test processes, 597
test tools, 597
test-assessment and -improvement framework, 454n5
asset tags, 306, 307
audience. See target audience
audition interviewing, 348–349
automation. See test automation
automobile, Windows CE computer for, 267
availability/stability/reliability, 22

**B**
bad bug reports, 151–152
bad coverage decisions, 110–111
bad experiences, learning from, 351, 351n9
bad news, good ways for delivering, 405–406
balanced curiosity, 321–322
balanced scorecards, 228. See also dashboards
Basic Library. See also Black, Rex; RBCS
“Case Study AddValidUser.case”, 135
“Case Study Functionality.suite”, 135
“Case Study Loan Processing Test Cases.xls”, 135
“ISTQB Certification”, 335n6
“Measuring Defect Potentials and Defect Removal Efficiency.”, 539n20
“Mission Made Possible,” 70n5, 507n5
“Quality Risk Analysis.”, 29n7
“Software Test Estimation.”, 41n10
basic test-case template, 92–96
Batini, Carlo, 269n3
battery power, 555
Beck, Kent, 543n22
behavioral interviewing, 348
behavioral test coverage, 610–611
behavioral (black-box) tests, 2, 3
defined, 601
functional tests v., 604–605
doing functionality, 104
of functionality, 104
of non-functional elements, 104
Beizer, Boris, 6n3, 86n2, 171n4. See also bug taxonomy
belling the cat, 68n4
best practices
quality assurance, 90
quality risk analysis, 17
for quality test systems, 90–91
reactive testing, 108
research literature reading as, 133–134
test systems, 90–91
testing process, 534–536
beta testing, 2, 4, 8, 24, 115, 294, 304, 329
Beyond Good and Evil (Nietzsche), 351, 351n9, 352
Black, Rex. See also RBCS
Advanced Software Testing
Volume I, 105n4, 335n6, 348n7
Volume II, 179n7, 335n6, 348n7, 538n19
CA Network and Voice Management case study, 239
Critical Testing Processes, 190n8, 454–455, 454n5, 530n13, 531, 535, 535n16, 597
Foundations of Software Testing: ISTQB Certification, 335n6, 348n7
ISTQB and, 334–335, 596
“ISTQB Certification”, 335n6
“Mission Made Possible,” 70n5, 507n5
“No More Bad MBOs,” 264n2
Pragmatic Software Testing, 5n2, 6n3, 47n11, 105n4, 115n7, 134n9
“Quality Risk Analysis.”, 29n7
“Software Test Estimation.”, 41n10
black-bead/red-bead experiment, 91, 91n3
black-box tests. See behavioral tests
blended testing strategy, 536. See also Agile methodologies
blind men/elephant parable, 11–12
blind spots, in sprint silos, 543–545
blindfolded children/piñata metaphor, 108
body of knowledge/syllabus, 331, 334, 335
body shops, 366, 428, 429, 430
bonuses, schedule-based, 97–98, 360
bouncing-ball bug report, 160
bounds section (test-plan template), 53–55
brainstorming session, 27, 35, 46
brake pedals, 266, 517–518, 521–525, 549
bridge keeper, knights and, 349
Brooks, Andrew, 239
Brooks, Frederick, 382n1, 520, 520n11
Brook’s Law, 520
brownouts, 555
brute-force technique (regression-risk mitigation strategy), 131, 510
BS antennae set, 350
buddy tests, 5
budgets. See also investment creating, 43–45
logistics database and, 271–281
SpeedyWriter, 45, 489–491
bugs
challenging, 192–194
chart. See open/closed chart
deferred, 193
defined, 146, 601–602
endless bug discovery nightmare, 182
exercises, 195–197
features v., 192–193
field-reported, 604
I am a Bug! 164n2
IEEE 829 standard and, 177, 177n6
ignored bugs nightmare, 182, 183
irreproducible, 193
isolation of. See isolation
definition cycle, 161–163, 164n2
mismanaged bugs nightmare, 182–183
priority, 156–158, 193, 606, 608
ranking of, 156–158
root causes, 169–174, 608
RPNs for, 158, 212, 213, 214
severity, 156–158, 606, 608
sticky wickets, 192–194
bug chart. See open/closed chart
bug clusters, 132, 193, 234, 236
bug coverage, 116–118
bug crawl, 602. See also bug triage process;
change-control boards
bug data
misuse of, 190–192
quality control charts and, 179
bug hunting, 16, 70, 79, 106
Bug ID column, 202
bug metrics. See metrics
bug report ping-pong, 163
bug reports. See also bug-detail report; failure description
accurate, 151
anomaly reports v., 75, 177, 177n6, 223
anomaly-report template (IEEE 829), 176–178
bad, 151–152
bouncing ball, 160
classifying, 64–65
complete, 151
concise, 151, 153
defined, 146–147
exercises, 195–197
flexible, 154–156
good
10-step process for, 152–154, 154n1
bonuses and, 360
writing, 150–152
life cycle, 158–160
Omninet and, 195
SpeedyWriter, 152
bug scrub meeting, 181
bug taxonomy, 171–174, 171n4
“Bug Taxonomy and Statistics” (Beizer), 171n4
bug tracking, 64, 145–148. See also
bug-tracking database
IEEE 829 standard and, 148
managing, 190–194
bug triage process, 65, 163–165, 164n2, 602
bug-detail report, 155, 156
SpeedyWriter, 155, 156, 166
bug-entry form, 155, 156, 158, 165, 166, 175, 176
buggy deliverables, 523
bug-lifetime metric, 175
bug-review committee, 164
bugs reports, rice sacks v., 360, 362
bug/test coverage chart, 226–228
bug-tracking database, 145–197
accountability and, 160–161
accuracy and, 194
benefits of, 147–148
bug-entry form, 155, 156, 158, 165, 166, 175, 176
charts and, 179–190
Close Date field, 174
commercial, 154
configuration field, 168–169
constructing, 154–156
defined, 147
design for, 155
detection phase, 175
dynamic fields for, 165–166
dynamic information for, 158–166
failure description, 148–154
finalized, 175–176
bug-tracking database, (continued)
finishing touches, 167–168
flexible bug reporting, 154–156
IEEE 829 standard and, 148
injection phase, 175
linkages and, 261–262
managing bug tracking, 190–194
metrics and, 179–190
ownership and, 160–161
quality risks field, 169, 226
ranking of bugs, 156–158. See also priority; severity
reasons for, 146–148
removal phase, 175
resolution field, 169
root cause analysis and, 169–174
subsystem field, 167–168
build plan, 6
burndown charts, 542
Burnstein, Ilene, 530, 530n13
can test section, 36–46
Capability Maturity Model. See CMM
The Capability Maturity Model (Paulk, et al.), 532n14
capacities/volumes (quality risk category), 19–20, 22
Capitalism, 448
career development, 359–360
cascading update feature, 271n4
case studies. See also Agile methodologies; information-appliance project
Agile, 536–547
bug metrics, 194–195
CA Network and Voice Management, 239–248
Collard and Company, 138–142
FMEA, 46–47
IVR, 87, 288–290, 563
loan-processing-application project, 76–77, 98–99, 135, 236–237
production-data anonymization, 136–138
ROI, 547–549
SpeedyWriter, 478–481
test-case documentation (details/precision continuum), 108, 109, 138–142
testers. See also testers
testing service providers, 460–469
test-tracking spreadsheet, 217–221, 236–248
widgets, 469–472
“Case Study AddValidUser.case”, 135
“Case Study Functionality.suite”, 135
“Case Study Info Appliance Bug Metrics.xls”, 256
“Case Study Info Appliance Client Test Tracking.xls”, 255, 565
“Case Study Info Appliance Server Test Tracking.xls”, 255
“Case Study IVR Network Test Change Management.mdb”, 290
“Case Study IVR Network Test Logistics.mdb”, 288
“Case Study Loan Processing Bug Metrics.xls”, 194, 237
“Case Study Loan Processing Test Cases.xls”, 76, 135, 237, 255, 256
“Case Study Loan Processing Test Coverage Tracking,” 237
“Case Study Loan Processing Test Plan.doc”, 76
“Case Study Loan Processing Test Tracking,” 237, 255, 256
Caspar Milquetoast, assertiveness v., 324–325
cat, belling the, 68n4
CCBs. See change-control boards
Ceri, Stefano, 269n3
certainty, progress v., 258
certification, 334–336
ASQ and, 335, 382n1
Foundations of Software Testing: ISTQB Certification, 335n6, 348n7
ISTQB, RBCS employees and, 596
“ISTQB Certification”, 335n6
QAI and, 335, 382n1
RBCS, 599
Sogeti and, 335
Cervantes, Miguel, 377, 378
ceteris paribus (all things held equal), 189
challenging bugs, 192–194
change, volume/speed of, 537
change history (test-plan template), 71
change management, 284–285
change management database, 257, 284–288
Change Type field, 287–288
change-control boards (CCBs), 65, 164, 164n3, 285, 285n5, 504, 602
charts. See also dashboards
bug-tracking database, 179–190
notes on, 189–190, 229–232
test-tracking spreadsheet, 223–232
checklists (quality risk analysis), 17–25
choosing partners (distributed testing), 422–433
clarity
bug reports and, 153
test plans and, 76
cleanliness, test environment and, 309–310
clients, of RBCS, 536, 596–597. See also RBCS
clogged filter, low test effectiveness and efficiency phenomenon, 498–499
close date, fix date and, 181
Close Date field (bug-tracking database), 174
closed-loop corrective actions, 529
closure gap. See closure period
closure periods, 184, 185, 513, 602
clusters, bug, 132, 193, 234, 236
CMM (Capability Maturity Model), 454–455, 477, 516, 530, 532, 533
The Capability Maturity Model, 532n14
test maturity assessment and, 454–455
CMM in Practice (Jalote), 532n14
CMMI (Compatibility Maturity Model Integration), 454, 455, 532, 533
code and fix process, 508–509
code coverage, 18
code swaps, 5
code-based tests. See structural tests
cognitive dissonance, 546
Cold War, 448
Collard, Ross, 138
Collard and Company case study, 138–142
columns. See also specific columns
in DataRocket FMEA, 33–35
in SpeedyWriter quality risk analysis, 30–32
test-case summary worksheet, 200–203
Comment column, 202
communication
distributed test effort and, 443–444
outsourced projects and, 459
upper management and, 390–393
compatibility, configuration options and, 23
Compatibility Maturity Model Integration. See CMMI
completeness, of bug reports, 151
The Complete Guide to Software Testing (Hetzel), 320n1, 382n1
cost component (subsystem/unit) testing, 5–6
defined, 602, 612
getting done/completing of, 10
inconsistent/inadequate, 538–539, 543n22
jump-start for, 598
quality risk categories and, 17–19
components, hardware testing and, 563–564
component/system coupling, 427
comp.risks Usenet newsgroup, 265
computer hardware testing. See hardware testing
computer-controlled MRI medical systems, 562
Computer-Related Risks (Neumann), 265
The Computer Consultant’s Guide (Ruhl), 363n11
Conceptual Database Design (Batini, et al.), 269n3
conductivity/resistance testing, 556
configuration field (bug-tracking database), 168–169
configuration-coverage problem, 113–114
configuration-lookup table, 201, 202
configurations. See test configurations
confirmation tests, 131, 210–211, 602
congruent (test-system architecture), 602
consultants. See also RBCS
experts v., 372–374
original meaning of, 372
RBCS and, 372, 597
consumables (Inventory template), 301–302
Context field (anomaly-report template), 177–178
contexts. See global view of testing
contingencies strategy (risk handling), 69–71
continuation criteria, 59, 602–603
continuum. See also spectrum; test granularity
deployment/test case, 466
distributed testing effort and, 435
outsourcing and, 46, 460
of quality, 477–487
of testing, 477–478
“cost, quality, schedule—pick two,” 37

cost of quality model, 477–478

cost-of-quality graphs, 493–494

cost-of-quality/ROI analysis
  for information-appliance project, 487–488, 548
  for SpeedyWriter, 478–481
couch potato, 541
coverage. See test coverage
Craig, Rick, 6n3, 354n10
“Credible Estimation for Small Projects”
  (Hadden), 41n9
Crispin, Lisa, 507n4
criteria
  continuation, 59, 602–603
  entry, 58–59, 603
  exit, 58, 60, 603
  rating, 57
Critical Testing Processes (Black), 190n8,
  454–455, 454n9, 530n13, 531, 535, 535n16, 597
Critical Testing Processes (CTP), 530, 531
  framework, 454–455
Crosby, Phillip, 11n5, 477, 478
cross-functional brainstorming session, 27, 35, 46

crunch mode, 84, 257–291
certainty v. progress, 258
distributed test effort and, 442–443
tips for, 257–267
  tools for, 257
follow-up and, 258–259
data-flow coverage, 18
cross-functional brainstorming session, 27, 35, 46
daily closure period, 184
dark, testing in, 400–402
dashboards, 179
death, 228–229
data configuration, test platform and, 308
databases. See also bug-tracking database
  Access and, 154, 271, 271n4, 272
  change management database, 257, 284–288
  entity-relationship diagrams, 268–271, 269n3, 307, 318
  logistics database, 257, 267–284
data-flow coverage, 18
data-generation tools, 136
datal, 18
  alarm card, 18
  behavioral testing, 3
  component testing, 5
  documentation risk, 24
  dynamic priority approach and, 122, 125–127
  FMEA chart, 32–35, 216
  hardware-allocation plan, 62, 63
  integration testing, 6, 20
  open/closed chart for, 228
  operations risk, 21
  railroading approach and, 122, 127–129, 607
  screen-oriented registration test case, 99, 100
  shotgunning approach and, 115, 122, 127, 608–609
  static priority approach and, 122, 123–125
  stress-test case for, 96–98
  structural testing, 9
  subsystems of, 167
  system testing example, 7, 122–131
  test-tracking spreadsheet case studies, 217–221
  transaction handling, 21
date and time handling, 22
Date Anomaly Discovered field, 177
deadly test escapes, 265
debugging
  defined, 603
  isolation e., 603, 605
  in test environment, 523
  /testing task continuum, 466
defect detection effectiveness. See defect
detection percentage
defect detection percentage (DDP), 109, 110,
  118, 188–189, 264, 360, 375, 376, 398, 480,
  549
defect removal model, 175
defect-based tests, 104
defects. See bugs
Defense Department, United States, 454n4,
  532n15
defered bug, 193
definitions. See also glossaries
good, 53, 501
test-plan template and, 53
delays, in schedules, 393–395
DeMarco, Tom, 354n10, 520n10
Deming, W. Edwards, 91, 528, 533
Deming/Shewhart cycle, 528–529
The Deming Management Method (Walton),
  91n3, 533
dependencies, follow-up and, 258–259
deployment (cut-over, migration, release). See also releases
defined, 603
  first customer ship, 604, 606
  “depraved minds .... can be usefully employed,” 321
Description of Anomaly field, 178
Description of the Corrective Action section, 178
design models (Omninet System
  Requirements Document), 585–589
design-based tests. See structural tests
details of the Master Test Plan section (IEEE
  829 test-plan template), 74–75
details section (IEEE 829)
  Level Interim Test Status Report template,
    233
  Level Test Report template, 234–235
  Master Test Report template, 236
test-log template, 222–223
detection costs, 478
detection phase (bug-tracking database), 175
development, test organization as part of,
  383–384
development peers, 395–397, 486
development services group (model), 387, 388
development sprints, 545
development-project resource, test
  organization as, 384–385
diamonds, 268
Dilbert cartoons, 320n2, 533
directions, of test management, 388–400
discussion, test plan drafts and, 51
distributed environment, 22
distributed test team organizational chart,
  460–461
distributed testing, 421–447. See also
  outsourcing
  advantage of, 437
  communication and, 443–444
cost and, 435
crunch mode and, 442–443
culture clashes and, 445–446
defined, 421, 603
disadvantages of, 438–439
flavors of, 421–422
logistics and, 437–439
management of, 422, 442–447
mapping issues and, 439–441
outsourcing and, 421, 448–460
  need for organization in, 450–451
  partners and, 422–433
  perspectives and
    inside looking out, 448
    outside looking in, 449
  planning of, 422, 433–441
  politics and, 444–445
  sales offices and, 421, 431–432
  steps for, 422
  test setting diagram and, 54, 55
  testing service providers and, 421–422,
    428–431
  case studies, 460–469
  trust and, 446–447
  unified test program and, 436–437
  users/user-surrogates and, 421, 432–433
  vendors and, 421, 424–428, 564–565
  divergent experiences, of quality, 12–14
  Document Identifier section (IEEE 829
  test-plan template), 72
documentation
  no, testing and, 400–402
  test-case (details/precision), 108, 109,
    138–142
test-plan, 76
dog-and-pony shows, 391
Don Quixote, 377, 378, 405, 482
drops/shocks/vibrations, 23–24, 312,
  555–556, 560, 564
Drucker, Peter, 515, 515n8
due diligence, 427, 430, 603
dulcinea, 377, 378
dynamic information, for bug-tracking database, 158–166
dynamic priority approach, 122, 125–127
dynamic testing. See reactive testing

early adoption, testing and, 409–412
early starting, of test phases, 9, 10
earned value, 207–208
economic context (global view of testing), 475–495
education/training/certification (testers), 331–336. See also certification
Effective Project Management (Wysocki, et al.), 38n8
efficiency. See test efficiency
efficiency system requirements (Omninet System Requirements Document), 582–583
effort, allocation of, 15
80/20 rule of management, 511
electrical conductivity/resistance testing, 554–555
electrical testing, 554–555
elephant/blind men parable, 11–12
emperor’s new clothes scenario, 405
employees, at RBCS, 596. See also RBCS
endless bug discovery nightmare, 182
engineering prototypes, 63, 67, 500, 513, 557
entities, 268
entity-relationship diagrams, 268–271, 269n3, 307, 318. See also relationships
entomology. See bug-tracking database
equipment damage (test labs), 312–313
failure description, 148–154
failure mode and effect analysis. See FMEA
failure weights, 212, 213, 214
false customer problem, 264–265
false negatives, 264–265
false positives, 263–265
fault injection
faults, bugs v., 192–193
Faulkner, William, 150, 151
faulty test plans, 195–197
features, bugs v., 507n3
fiduciary (of test systems), 12–13, 86, 87, 110, 604
field-reported bugs, 604
file manipulation, SpeedyWriter, 5
fire extinguishers, 296, 298, 311
first customer ship, 604, 606. See also deployment
fiscal context (global view of testing), 475–495
fishing analogies
  low-fidelity test system and, 110
  mature test processes and, 497–498
  regression-test gaps and, 110, 120
  test granularity and, 4, 498, 499
“five Cs,” 320n1
fix date, close date and, 181
flexibility (test component), 604
flexible bug reporting, 154–156
floor plan (test lab), 296
FMEA (failure mode and effect analysis), 29, 32–35
case study, 46–47
DataRocket and, 32–35, 216
defined, 604
pitfalls, 35
focus, space cadets, 322–323
follow-up, 258–259
foundation, of test projects, 1–47
Foundations of Software Testing: ISTQB Certification (Black), 335n6, 348n7
found/fixed chart. See open/closed chart
Franklin, Will, 547n12
frequently asked questions (test-plan template), 71
functional tests, 598, 604–605
functionality
  behavioral tests of, 104
  quality risk, 19, 21
functionality system requirements (Omninet System Requirements Document), 576–579
“Functionalitysuite”, 135
Fundamental Concepts for the Software Quality Engineer: Volume 2 (Daughtry), 29n7
funding (test funding), 481–487
furnishings (Inventory template), 302

G

GA (general availability), 604. See also first customer ship
Gantt charts, 42, 43, 408, 423, 424
garden-variety test labs, 294
Gartner Group, 410, 410n7, 411, 450, 545, 545n23
gas pedals, 266, 517–521, 549
Gaudreau, Steven, 239
geek-culture immersion studies, 349n8
Gelperin, David, 531
general availability (GA), 604. See also first customer ship
General section (IEEE 829)
  anomaly-report template, 178
  Level Interim Test Status Report template, 233
  Level Test Report template, 235
  Master Test Report template, 236
test-design template, 101
test-log template, 223
test-plan template, 75
general view of testing. See also outside looking in perspective
economic context, 475–495
process context, 497, 514–527
project context, 497–514
glossaries, 601–612
definitions and, 53
General section and, 75, 223
IEEE 610, 53
ISTQB, 53
goals/positions/experiences, 336–338
goat rodeos, 257, 605
Goldstein, John, 274, 275, 276, 293, 439
Graham, Dorothy, 85n1, 507n3
guessing, error, 4, 134, 322n4
GUI (graphical user interface)
  automation at, 43, 50, 81, 85, 89, 341, 368, 391, 537
  file manipulation component and, 6
  SpeedyWriter, 167
test tools, 50, 82, 88, 89, 341
Guide to Quality Control (Ishikawa), 514n6, 553

H

hacker types, 337
Hadden, Rita, 41n9
HALT, HASS & HASA Explained: Accelerated Reliability Techniques (McLean), 300n1, 559n1
HALT chambers, 300
HALTs (highly accelerated life testings), 300n1, 559, 559n1
handicap accessibility, 298
hard work, sloth n., 324
hardware (Inventory template), 300–301
Hardware Assignment report, 280
hardware testing, 553–566
components/subsystems and, 563–564
electrical testing, 554–555
environmental tests, 555–557
fundamental, 554
integrated software and, 564
mechanical life, 19, 63, 84, 358, 554, 557, 563, 564
MTBF and, 208, 559
pilot testing and, 565
reliability and, 554, 558–559
self-tests, 554
shocks/vibrations/drops, 23–24, 312, 555–556, 560, 564
test conditions and, 554
test management, 553–554
test tools and, 553–554
hardware-allocation plan, 62, 276
DataRocket, 62, 63
expansion of, 277
straw-man plan and, 609
target audience, 408
hardware/infrastructure side, logistics
database and, 276–281
hardware/software system projects, 513–514
having a req, 341
help-desk, 397–398
Hemingway, Ernest, 150, 151
hero, aspiring, 323
Hetzel, Bill, 320n1, 382n1
high-fidelity test system, 12
hiring process, 341–353, 368–372
Hispanic piñata metaphor, 108
“hobgoblin of small minds,” 261
holidays/cultures, crunch mode and, 266–267
home equity loan project. See
loan-processing-application project
How to Break Software (Whittaker), 106n5
How to Break Software Security (Whittaker), 106n5
How to Break Web Software (Whittaker), 106n5
HP’s Quality Center, 200
human factor. See people side
Huntford, Roland, 411
Hype Cycle, 410, 410n7, 545, 545n23
I am a Bug! (Sabourin), 164n2
IEEE 610 glossary, 53
IEEE 829 standard
anomaly-report (bug-report) template, 176–178
bug tracking and, 148
bugs and, 177, 177n6
sections. See specific sections
test-log template, 221–223
test-plan template, 52, 71–75
test-results reporting templates, 232–236
test-system templates, 99–104
test-case template, 101–103, 610
test-design template, 100–101
test-procedure template, 103–104
IFIRs (initial field incident reports), 529
ignore or accept option, 69
ignored bugs nightmare, 182, 183
IIBA (International Institute of Business Analysts), 334. See also certification improvements
independent test organization, 385–386
“Info Appliance Bug Metrics.xls.”, 256
“Info Appliance Client Test Tracking.xls, 255, 565
“Info Appliance Server Test Tracking.xls,” 255
informal quality risk analysis technique, 29–32
information-appliance (Internet-appliance) project, 374–375, 460–461, 547–549
cost-of-quality/ROI analysis for, 487–488, 548
distributed test team organizational chart, 460–461
hardware tests, 565–566
modem issues and, 457–458
system architecture, 374
test maturity assessment, 547, 549
test team, 374
test team organization, 375
in-house test teams, 462–466, 472. See also testing service providers
initial field incident reports (IFIRs), 529
injection phase (bug-tracking database), 175
inside looking out perspective
distributed testing and, 448
testing process and, 534
institutionalizing dashboards, 406–407
insurance model, testing and, 476, 494
integrated software, 564
integration, test projects and, 512–514
integration (product) testing, 6–7. See also
system testing
DataRocket, 6, 20
defined, 329, 605
project life cycle and, 499–500
quality risk categories and, 19–21
SpeedyWriter, 6, 20
interactive-voice-response. See IVR
interfaces, 19. See also GUI
International Institute of Business Analysts
(IIBA), 334. See also certification
International Requirements Engineering
Board (IREB), 334
International Software Testing Qualifications
Board. See ISTQB
Internet kiosks. See kiosk entries; Omninet
Internet-appliance project. See
information-appliance project
interview approach (stakeholder groups), 27,
35
interview processes, 347–350. See also specific
interviews
Introduction section (IEEE 829)
anomaly-report template, 177
Level Interim Test Status Report template,
233
Level Test Report template, 234
Master Test Report template, 236
test-log template, 222
test-plan template, 72
inventory, test lab, 299–305
Inventory template, 300–303
investment (testing investment), 475–487
inward managing, 388, 389. See also staffing;
test teams
IREB (International Requirements
Engineering Board), 334
Iron Box and Triangle, 37
irreproducible bugs, 193
“Is Quality Negotiable?” (Crispin), 507n4
Ishikawa, Kaoru, 514n6, 553
is/is not format
test group organizational boundaries, 381,
387
test-project scope, 53, 54, 73
island, test team and, 89, 517–525
ISO 9000 for Software Developers (Schmauch),
532n14
ISO 9000/14000 family of standards, 532
ISO 9126 quality model, 24–25
isolation (of bugs), 64–65, 150, 153, 603, 605
ISTQB (International Software Testing
Qualifications Board), 334–335. See also
RBCS
Black and, 334–335, 596
certifications, 335n6
RBCS employees and, 596
exams, RBCS and, 599
Foundations of Software Testing: ISTQB
Certification, 335n6, 348n7
glossary, 53
“ISTQB Certification” (Black), 335n6
iterations
metrics and iterative process, 232
short, 537–538
IVR (interactive-voice-response)
case study, 87, 288–290, 563
server, dropping of, 560
“IVR Network Test Change Management.mdb”,
290
“IVR Network Test Logistics.mdb,” 288
J
Jalote, Pankaj, 532n14
Jaskiel, Stefan, 6n3, 354n10, 530n13
job descriptions, hiring and, 341–345
Jones, Capers, 10n4, 41, 503, 538, 539, 539n20,
539n21, 543n22
Jorgensen, Paul, 105n4
jump-starts, 364, 598. See also RBCS
Juran, J. M., 11, 11n5, 146, 477, 478
Juran on Planning for Quality (Juran), 11n5
K
Kamen, Henry, 406n6
Kan, Stephen, 10n4, 118n8, 175n5, 179n7,
208n1
Kaner, Cem, 56n1
key, 268
kiddie soccer approach, 385, 385n2
killing the messenger, 546
King Arthur’s knights, bridge keeper and, 349
kiosk module flow, 586, 587
kiosk state-transition diagram, 588
kiosk state-transition table, 588
kludges, 258–259, 359, 605
knights, bridge keeper and, 349
known unknowns, 528
Koomen, Tim, 531n13, 547, 547n24
Kubaczkowski, Greg, 70n5, 507n5

L
labs. See test labs
lack of constancy of purpose, 533
The Last Place on Earth (Huntford), 411
lateness, effects of, 393–395
late-night testing, 337
layoffs, 403–404
layouts, test lab, 315, 317
Lessons Learned in Software Testing (Kaner, et al.), 56n1
Level Interim Test Status Report template, 232–233
level of risk, 14–15
Level Test Report template, 233–235
levels 0–5 (test-specification levels of detail), 138–142
levels of testing. See test phases
library. See Basic Library; test-case library
life cycle. See also cycles; test cycles
bug, 161–163, 164n2
bug report, 158–160
project/software, 498–514
acceptance testing and, 500–501
informal/formal, 497
integration testing and, 499–500
pilot testing and, 500–501
test integration into, 380–381
test phases and, 498–499
themes of, 498–501
test-case, 209–212
linkages, test-tracking/bug-tracking, 261–262
liquidation, 403–404
live tests, 2, 4. See also acceptance testing;
ad-hoc testing; alpha tests; beta testing;
exploratory testing
“Loan Processing Bug Metrics.xls,” 194, 237
“Loan Processing Test Cases.xls,” 76, 135, 237, 255, 256
“Loan Processing Test Coverage Tracking,” 237
“Loan Processing Test Plan.doc”, 76
“Loan Processing Test Tracking,” 237, 255, 256
loan-processing-application project, 76–77, 98–99, 135, 236–237
local regression, 118, 211
localization, 22
Locations by Test and Tester report, 277
Locations table (test lab), 299
logical/concrete test-case continuum (documentation details/precision), 108, 109, 138–142
logistics, distributed testing and, 437–439
logistics database, 257, 267–284
budgeting and, 271–281
test cycle. See test management
entity-relationship modeling, 268–271
management, 532, 533
Hardware Assignment report, 280
implementing, 271
Locations by Test and Tester report, 277
management peers and, 397, 486
planned releases for targets, 281–282
planning and, 271–281
hardware/infrastructure side, 276–281
people side, 272–276
SpeedyWriter test assignments, 273
SpeedyWriter testers, 273
SpeedyWriter tests, 273
tables/relationships, Access view of, 271, 272
Test Hardware report, 280
test lab and, 299, 307
Tested Configurations report, 283, 284
Tester Assignments report, 272, 274
Tests by Location and Tester report, 276
tracking software configurations, 281–284
long-term temporary workers, 364–368
low-fidelity test systems, 13, 86, 110

M
macro view. See global view of testing
magnetic resonance imaging (MRI) medical systems, computer-controlled, 562
maintainability
risks, 24, 25
test data and, 137
test system, 605
test-system architecture and, 85, 88, 519
maintainability system requirements
(Omninet System Requirements Document), 583–584
maintenance releases, 509–512
make interesting new mistakes, 526, 527
Malcolm Baldridge National Quality Award, 532, 533
management. See also test management
bug tracking, 190–194
bug triage process, 65, 163–165, 164n2, 602
doing distributed test effort, 422, 442–447
Index

Effective Project Management, 38n8
80/20 rule, 511
equipment/configurations, for test lab, 306–309
management by objectives strategy, 361–362
managing testing and, 134
quality-risk, 607
test cases. See test-tracking spreadsheet

test lab, 293–318. See also test labs
test release, 65–68
of test teams, 352–376
management considerations. See politics
management peers, 397, 486
Management’s New Paradigms’’ (Drucker), 515, 515n8
manager titles, test teams and, 381–383
Managing the Test People (McKay), 320n1
many meetings, 541–542
many-to-many relationships, 82, 268, 269, 270, 271, 318
mapping issues, 439–441
Mariana, Juan de, 405–406
marketing, 398–400
Marketing Requirements Document. See
Omninet Marketing Requirements Document
master test plan, 51
Master Test Plan section (IEEE 829 test-plan
template), 74–75
Master Test Report template, 235–236
mature test process, 497–498, 528–529. See also testing process
McConnell, Steve, 19n6, 61n3, 71n6, 164n3, 285, 285n5, 336, 386n3
McLean, Harry, 300n1, 599n1
meals, overtime and, 360–361
mean time between failures. See MTBF
mean time to repair. See MTTR
measure twice, cut once, 312–313
‘Measuring Defect Potentials and Defect
Removal Efficiency’’ (Jones), 539n20
mechanical life, testing and, 19, 63, 84, 358, 554, 557, 563, 564
meetings, many, 541–542
MEGO (my eyes glazed over), 53, 264, 605–606
Mencken, H. L., 108
metrics
bug-tracking database, 179–190
case study, 194–195
iterative process and, 232
notes on, 189–190, 229–232
surrogate, 230
test-tracking spreadsheet, 223–232
Metrics and Models in Software Quality
Engineering (Kan), 10n4, 118n8, 175n5, 179n7, 208n1
Microsoft Access, 154, 271, 271n4, 272
midstream releases, 67
might test, should test, can test approach, 1–46, 326
migration. See deployment
milestones schedule, 56, 57
milquetoast, assertiveness v., 324–325
minimalist test-tracking spreadsheet, 200–205
minimizing false positives/false negatives,
263–265
misinterpretation, test results, 263–265
mismanged bugs nightmare, 182–183
‘Mission Made Possible’’ (Black &
Kubaczkowski), 70n5, 507n5
mission statement, 378–379
mistakes
hiring, 351–352
make interesting new mistakes motto, 526, 527
regression-test gaps and, 129–131
test coverage, 110–111
tester, 524–525
misuse of bug data, politics and, 190–192
mitigation strategies. See risk-mitigation
strategies
models. See specific models
modern issues, information-appliance project
and, 457–458
mono-team approach, 383–384
Monty Python and the Holy Grail, 349, 349n8
moonlighters, 337
‘more haste, less speed,’ 359
motivating test teams, 353–362
mouse, belling the cat and, 68n4
MRI (magnetic resonance imaging) medical
systems, computer controlled, 562
MTBF (mean time between failures), 208, 559, 559n1, 606
MTTR (mean time to repair), 559, 606
Mullins, Jodi, 239
multiple test plans, 50–51
Murray, Bill, 527
my eyes glazed over (MEGO), 53, 264, 605–606
Myers, Glenford, 1n1, 134, 187, 322n4, 329n5
The Mythical Man Month (Brooks, F.), 382n1, 520, 520n11
N
Navathe, Shamkant, 269n3
NEBS test, 561, 562
networked environments (quality risk category), 22
Neumann, Peter, 265
new technology, 290, 409–412
new-feature regression, 119, 211
Nietzsche, Friedrich, 351, 351n9, 352
nightmares (open/closed chart), 181–183
“No More Bad MBOs” (Black), 264n2
non sequitur interview, 349
non-functional elements, behavioral tests of, 104
non-functional tests, RBCS and, 598
non-prescriptive test-process-assessment models, 529, 530
nonrequirements failure, 64
normalized value, 213
countermeasures, 265
nuclear bomb simulated testing, 562
nuclear-medicine device (Therac-25), 265, 562
O
objectives, management by, 361–362
O’Connor, Patrick, 86n2, 208n1, 514n6, 559n1
office politics. See politics
oil-exploration software client, 333
Omninet, 569
bug reporting and, 195
kiosk module flow, 586, 587
kiosk OS/browser/connection speed configuration orthogonal array, 588–589
countermeasures, 265
kiosk state-transition diagram, 588
kiosk state-transition table, 588
payment processing decision table, 586, 587
risk analysis for, 47
system architecture, 585–586
test team, skills for, 375
test tracking for, 249
Omninet Marketing Requirements Document, 47, 567–573
acronyms/abbreviations, 569
applicable documents, 570
release date, 570
requirements, 570–573
scope, 569
test plan (rough draft) and, 77
Omninet Screen Prototype Document, 570
Omninet System Requirements Document, 47, 575–589
design models, 585–589
efficiency system requirements, 582–583
functionality system requirements, 576–579
maintainability system requirements, 583–584
portability system requirements, 584–585
reliability system requirements, 580–581
requirements, 576–585
usability system requirements, 581–582
one-to-many relationships, 237, 268
one-to-one relationships, 171, 268
onion-peeling, 251, 252, 253
on-site interview processes, 347–350
open/closed chart, 179–183
ceteris paribus and, 189
DataRocket, 228
good, 181
nightmares in, 181–183
SpeedyWriter, 180
operations (quality risk category), 21
“optimistic on the outside, pessimistic on the inside”, 392
oracles
defined, 606
output, 606
test, 539–540, 606
organization, outsourcing and, 450–451
organizational challenges for test managers, 377–419
testing service providers and, 466–468
organizational context, test group in, 383–386
organizational models, 338–341
organizational setting diagram, 54, 55
orthogonal, 606
orthogonal arrays, 115
all-pairs tables v., 115, 115n7
kiosk OS/browser/connection speed configuration, 588–589
OS/browser/connection speed configuration orthogonal arrays, 588–589
output oracles, 606
outside looking in perspective. See also global view of testing
distributed testing and, 449
test process and, 534–536
outsourcing, 448–460, 512. See also distributed testing
case study, 469–472
communication issues and, 459
cost and, 46, 460
distributed testing and, 421, 448–460
exercises, 473
planning/preparing testing and, 455–459
RBCS and, 422n1, 449, 599
temporary workers and, 362–374
test execution and, 459–460
test team selection and, 452–455
trust and, 458–459
outward managing, 389, 395–400
overhead/facilities category, 44
overtasked/understaffed, 357
overtime, meals and, 360–361
overview section (test-plan template), 52–53
ownership, bug-tracking database and,
160–161

P
packaging testing, 560
pair programming technique, 5
Pareto chart, 186, 291, 529, 549
Pareto principle, 173
partners (distributed testing), 422–433
Pass state, 201, 208
Paulk, Mark, 532n14
payment processing decision table
(Omninet), 586, 587
Peak of Inflated Expectations, 545, 546
pedals. See brake pedals; gas pedals
peer review, 606
people side (human factor), 7
logistics database planning and, 272–276
test labs and, 310–314
widgets/outsourcing case study and,
469–472
Perfect (test process outline), 534, 535
Perform (test process outline), 534, 535
performance, quality risk category, 20, 22
peripheral duties, added to testing, 386–388
Perry, Bill, 264n1, 389n4
pertinence, test plans and, 76
pessimism
‘‘pessimistic on the inside, optimistic on the
outside’’, 392
polite, 410
professional, 320–321, 375, 396, 405, 410, 533
Pettichord, Bret, 154n1. See also 10-step
process for good bug reports
phased test approach. See test phases
pilot testing, 8
defined, 606
hardware testing and, 565
project life cycle and, 500–501
piñata metaphor, 108
ping-pong
bug report ping-pong, 163
red-bead/black-bead experiment and, 91, 91n3
pink slips, 403–404
pins, testing and, 563
pizza, 360, 361
Plan (test process outline), 534, 535
Plan Date column, 207
planned releases for targets (logistics
database), 281–282
planning
for distributed test effort, 422, 433–441
for testing in outsourced projects, 455–459
planning phase, 38. See also test plans
Plateau of Productivity, 545
platforms. See reference platforms; test
platforms
Pol, Martin, 530n13, 531
polite pessimism, 410. See also professional
pessimism
political instability, 458, 458n6
politics (office politics), 377–419
bug data misuse and, 190–192
Byzantine sense of, 397
defined, 377
distributed test effort and, 444–445
main theme, 411
pollutants, 556
portability system requirements (Omninet
System Requirements Document),
584–585
positions/experiences/goals, 336–338
power input/consumption/output (quality
risk category), 23
Practical Programmer column, 521n12
Practical Reliability Engineering (O’Connor),
86n2, 208n1, 514n6, 553, 559n1
Practical Software Testing (Burnstein), 530,
530n13
Pragmatic Software Testing (Black), 5n2, 6n3,
47n11, 105n4, 115n7, 134n9
precision continuum (test-case
documentation), 108, 109, 138–142
Prepare (test process outline), 534, 535
prescriptive test-process-assessment models,
529, 530
Pressman, Roger, 502
prevention costs, 477
Principles of Quality Costs (Campanella), 10n4,
477, 478, 493n2
prioritizing test cases/test suites, 212–213
priority (bug), 156–158
anomaly-report template and, 178
consensus on, 193
priority (bug), (continued)
defined, 157–158, 606
severity of, 156–158, 606, 608
probe effects, 86, 87, 121
process brake pedals, 521–525
process context (global view of testing), 497, 514–527
process gas pedals, 518–521
process-centric view, of testing process, 535n16. See also testing process
processes. See also testing process; specific processes
defined, 528
improvement of, 514–527
process maturity, 169, 170
product testing. See integration testing
production data, 136–138
production-data anonymization, 136–138
productivity
external effects on, 517–525
test labs and, 310–314
professional pessimism, 320–321, 375, 396, 405, 410, 533
professionalism skills, 327–328
progress, certainty of, 258
project context (global view of testing), 497–514
project life cycle. See life cycle
project post-mortems, 285
project post-partums, 285
project resource approach, 544
project retrospectives, 285, 525–527
project risks, 14, 69–71
examples of, 69–70
risk mitigation strategies, 69–71
project team, diversity in, 290
project test services (RBCS), 598–599
project-based test organization, 340
project-centered test lab layout, 315
projection, 546
projects (test projects). See also case studies;
specific projects
foundation of, 1–47
Gantt chart, 408
hardware/software system, 513–514
integration and, 512–513
negotiating, 45–46
retrospectives, 285, 525–527
scope, 53, 54
sequential, 501–504
uniqueness/newness of, 514
proposed schedule of milestones, 56, 57
publications, online/hard-copy, 595
puzzle interview, 349
pyramids, skills, 433–454

Q
QA. See quality assurance
QAI (Quality Assurance Institute), 335, 382n1
qualification interviewing, 347–348
quality
ASQ and, 335, 382n1
blind men/elephant parable and, 11–12
cost of, 477–487
customers/users and, 11–12
defined, 11
experience of
defined, 604
divergences in, 12–14
investment in, 475–476
Juran and, 11, 146
professional pessimism and, 320–321, 375, 396, 405, 410, 533
term, manager title and, 382
for test systems, 85–88
testing process and, 10–36
test-system coverage and, 13–14
vendor, 564–565
''quality, schedule, cost,—pick two,'' 37
quality assurance (QA)
best practices, 90
quality control v., 382, 412, 412n8
quality-assurance manager, 377, 382, 383
Quality Assurance Institute (QAI), 335, 382n1
Quality Center, 200
quality control, 382
charts, bug data and, 179
Guide to Quality Control (Ishikawa), 514n6, 553
quality assurance v., 382, 412, 412n8
statistical, 514
test processes and, 528–529
Quality Control Handbook (Gryna & Juran), 477
quality engineering, supplier, 564–565
Quality Is Free: The Art of Making Quality Certain (Crosby), 11n5, 477, 478
quality methodology, RBCS and, 397. See also RBCS
quality risk analysis
best practices, 17
challenges, 36
checklists
ISO 9126 quality model, 24–25
usual suspects, 17–24
informal technique, 29–32
Omninet, 47
process options for, 25–29
RPNs and, 31, 34, 36
techniques, 29, 29n7
test lab inventory and, 303–304
tips, 35–36
“Quality Risk Analysis” (Black), 29n7
quality risks, 11, 13, 14
categories, 17–24. See also specific quality risk
categories
defined, 14, 607
field, bug-tracking database and, 169, 226
how to analyze, 14–15
management, 607
test cases and, 113–114
test-plan template, 55–56
quality risks table, 56
Quality Week 2001 conference, 543n22
quality-assurance manager, 377, 382, 383

R
radiation, 561–562
railroading approach, 122, 127–129, 607
ramp-up, defined, 607
ranking of bugs, 156–158. See also priority;
severity
Rapid Application Development, 506, 537
Rational Unified Process, 537
Rayleigh chart, 407
Raytheon’s Electronic Systems group, 477, 533
RBCS (Rex Black Consulting Services), 595–600. See also Basic Library; Black, Rex
about, 595–596
assessments, 597. See also Critical Testing
Processes
quality methodology, 597
test processes, 597
test tools, 597
test-assessment and -improvement
framework, 454n5
bottom line, 599–600
certification, 599
IIBA and IREB, 334
clients of, 536, 596–597
consulting and, 372, 597
contact information, 600
employees at, 596
ISTQB exams and, 599
jump-starts, 598
outsourcing and, 422n1, 449, 599
project test services, 598–599
staff augmentation and, 599
test management skills and, 599
test recruiting and, 599
training/training materials and, 333, 475, 536, 599
value of, 596
reactive testing (dynamic testing), 16, 106–108. See also Agile methodologies;
bug hunting; exploratory testing; software
attacks
as best practice, 108
blended testing strategy and, 536
error guessing and, 134
piñata metaphor and, 108
software attacks and, 609
red-bead/black-bead experiment, 91, 91n3
reference materials (Inventory template), 302–303
reference platforms, 61, 301, 401–402
referenced documents (test-plan template), 71
References section (IEEE 829 test-plan
template), 73
regression, 118–120, 210
confirmation testing and, 210–211
defined, 607
existing-feature, 119–120, 211
exposed, 211
geological-modeling software client and,
131–132, 510–511
local, 118, 211
maintenance releases and, 509–510
new-feature, 119, 211
remote, 118, 211
risk, 607
risk-mitigation strategies, 131–132
brute-force technique, 131, 510
Zen, 118
regression testing, 60, 121, 160, 536, 607. See also Agile methodologies
regression-test gaps, 110, 118–132
avoiding mistakes and, 129–131
defined, 607–608
minimization, 122–131
dynamic priority approach, 122, 125–127
railroading approach, 122, 127–129, 607
shotgunning approach, 122, 127, 608–609
static priority approach, 122, 123–125
test automation and, 120–122
test escapes and, 110
regulations. See standards/regulations
relationships. See also entity-relationship
diagrams
many-to-many, 82, 268, 269, 270, 271, 318
relationships. See also entity-relationship diagrams (continued)
one-to-many, 237, 268
one-to-one, 171, 268
release processes/versions, crunch mode and, 259–260
releases. See also deployment; first customer ship; maintenance releases
midstream, 67
spinning-disk, 609
test, 39, 65–66, 204, 611
reliability /availability/stability, 22
hardware testing and, 554, 558–559
reliability demonstration, 559
reliability system requirements (Omninet System Requirements Document), 580–581
reminders, follow-up and, 258–259
remote regression, 118, 211
removal phase (bug-tracking database), 175 reports/reporting, target audience and, 407–409. See also bug reports; specific reports
Report Wizard, 155
reporting logs, 608
reporting tools, 608
reproducibility, 149–150, 153
requirements
Omninet Marketing Requirements Document, 570–573
Omninet System Requirements Document, 576–585
requirements coverage, 113n6
requirements failure, 64
research literature reading (best practice), 133–134
resistance/conductivity testing, 556
resolution field
anomaly-report template and, 178
bug-tracking database, 169
resources
estimating, 43–45
over-allocation, SpeedyWriter, 369
scarce, 41, 69, 148, 161, 468, 513, 557
test-plan template, 62, 64
responsibilities. See roles/responsibilities
result-interpretation errors, 91, 110
automated tests and, 264
false positives/false negatives and, 265
results reporting, See test results reporting résumés, 345–347
resumption/suspension criteria. See continuation criteria
retrospectives, project, 285, 525–527
return on investment. See ROI
reviewing
bug report, 153
review meetings, 75–76
revisions/release processes, crunch mode and, 259–260
Rex Black Consulting Services. See RBCS
Rice, Randy, 264n1, 389n4
rice sacks, bug reports v., 360, 362
risk analysis. See quality risk analysis
risk priority numbers. See RPNs
risk-based testing, 14–17, 536, 538. See also
Agile methodologies
benefits of, 16–17
blended testing strategy and, 536
jump-start for, 364, 598
properties of, 15–16
risk-mitigation strategies
project risks and, 69–71
regression, 131–132
brute-force technique, 131, 510
risks. See also project risks; quality risks
defined, 14
level of, 14–15
of new technology, 290, 409–412
regression and. See regression
Risks Digest, 265
rodeos, goat, 257, 605
ROI (return on investment), 390–391
/cost-of-quality analysis, for information-appliance project, 548
roles/responsibilities
test managers, 377–378
test team, additional activities and, 386–388
Roll Up columns, 202, 213–214
rolling closure period, 184
root cause analysis, 169–174, 235
aim of, 171
bug taxonomy, 171–174, 171n4
root cause chart, 184
root causes (for bugs), 169–174
bug-tracking database and, 169–174
defined, 170, 608
rough draft, test plan, 77
RPNs (risk priority numbers)
for bugs, 158, 212, 213, 214
quality risk analysis and, 31, 34, 36
test lab inventory and, 303
Ruhl, Janet, 363n11
rules of thumb, 41, 165
SaaS (Software as a Service), 448, 465
Sabourin, Robert, 164n2
sacks of rice, bug reports v., 360, 362
safety
  hardware testing and, 561
  radiation and, 561–562
  safety-critical systems, 133, 265, 562
test labs and, 298, 310–313
sags, 555
Saint Paul’s Cathedral, 84
sales managers, 398–400
sales offices, 421, 431–432
Sancho Panza, 377
sanity checks, 41, 44, 69, 561
  estimation rules and, 41
  subsystem, 554
scalable, defined, 608
scarcely resources, 41, 69, 148, 161, 468, 513, 557.
  See also engineering prototypes
schedule-based bonuses, 97–98, 360
schedules.  See also test schedules
delays in, 393–395
follow-up, 258–259
milestones, 56, 57
"schedule, cost, quality—pick two," 37
work, reasonable, 356–359
Schmauch, Charles, 532n14
scientific wild-ass guesses (SWAGs), 199, 609
scope, 53, 54
scope creep, 325, 358, 524
Scope section (IEEE 829)
  test-log template, 222
  test-plan template, 72
scope shrink, 358, 394, 524
scorecards.  See dashboards
Screen Prototype Document, Omninet, 570
screen-and-field test-case template, 98–99
screening résumés, 345–347
screen-oriented registration test case, 99, 100
scripted testing, defined, 608
Scrum, 450, 506, 536, 538, 539, 540, 541, 542,
  543, 546.  See also Agile methodologies
security
  quality risks category, 23
test lab and, 305–306
self-correcting element (test execution phase), 16, 106
self-tests, hardware testing and, 554
Sell, Alan, 547n24
sequential projects, 501–504
setting (test-plan template), 54–55
severity (bug), 156–158
  defined, 156–157, 608
priority v., 156–158, 606, 608
Shapiro, Eileen, 533
shearing force, 557
shelfware, 76
Shewhart, Walter A., 528, 529.  See also
  Deming/Shewhart cycle
shifting test basis, 540–541
shocks/vibrations/drops, 23–24, 312,
  555–556, 560, 564
shoehorning test schedules, 37–43
short iterations, 537–538
shotgunning approach, 115, 122, 127, 608–609
should test section, 10–36
signal quality, 19
signal-quality testing, 555, 563
simulated nuclear bomb testing, 562
sink or swim, 353
Six Sigma, 532
ski slope and cliff variant, 503, 521
skills
categories
  application domain, 328
  professionalism, 327–328
  technology, 328
testing, 328–329
soft.  See politics
test management, RBCS and, 599
tester, 327–331
Omninet, 375
skill level, 264
skills pyramids, in developing/developed regions, 453–454
skills-assessment worksheet, 330–333
skills-based test organization, 339
Slope of Enlightenment, 545
sloth, hard work v., 324
smoke detectors, 296, 298, 311
smoke tests, 58, 59, 69, 392, 500, 504, 519, 520
soccer, 385, 385n2
soft skills.  See politics
software (Inventory template), 300
Software as a Service (SaaS), 448, 465
software attacks, 16, 56, 56n1, 70, 79, 106, 380,
  609.  See also exploratory testing
Software Cafeteria, 274, 279, 294, 423, 439
Software Engineering: A Practitioner’s Approach
  (Pressman), 502
Software Engineering Institute, 454, 516, 532.
  See also CMM
Software Project Survival Guide (McConnell),
  19n6, 61n3, 71n6, 164n3, 285n5, 386n3
Software Quality Engineering (SQE), 531
Software Quality Professional, 532n15
Software System Testing and Quality Assurance
(Beizer), 603, 86n2
Software Test Automation (Fewster & Graham),
85n1, 507n3
“Software Test Estimation” (Black), 41n10
Software Testing: A Craftman’s Approach
(Jorgensen), 105n4
Software Testing Techniques, Second Edition
(Beizer), 171n4. See also bug taxonomy
Sogeti, 335, 530, 531
space cadets, focus v., 322–323
The Spanish Inquisition (Kamen), 406n6
spectrum
logical/concrete test-case, 108, 109, 138–142
test granularity, 4, 8
skills perspective, 329
speed/volume, of change, 537
SpeedyWriter
behavioral testing, 3
budgets for, 45, 489–491
bug report, 152
bug-detail report, 155, 156, 166
case study, 478–481
closure period chart, 185
component testing, 5
continuation criteria for, 59
entry criteria for, 58–59
exit criteria for, 60
failure description, 149–150
GUI, 167
informal quality risk analysis, 29–32
integration testing, 6, 20
live testing, 4
logistics database and. See logistics database
open/closed chart for, 180
resource over-allocation, 369
root cause chart for, 184
skills-assessment worksheet, 330–333
string testing, 7
subsystem chart, 186–187
subsystems of, 167–168
system testing, 7
test automation engineer job description, 343–344
test environment, 317–318
work-breakdown structure, Gantt-chart
view of, 42, 43
SPICE, 532
spinning-disk release, 609
spiral model, 504–505
springs, 538, 539, 542, 543, 544, 545
development, 545
durations, arbitrary, 542–543
silos, blind spots in, 543–545
test, 545
SQE (Software Quality Engineering), 531
stability/reliability/availability, 22
staffing
staff augmentation, RBCS and, 599
staff category, 43
staffing phase, 38
test teams, 319–353
standards/regulations. See also IEEE 829
standard
compliance, 23
hardware testing, 562–563
State column, 200, 201
state machines, 17–18, 21, 390
states, 17–18, 158–160
static mats, 298
static priority approach, 122, 123–125
status reports. See test status reports
STC Lab, 272, 274, 281, 423, 424, 437
STEP (Systematic Test and Evaluation
Process), 530, 531
steps to reproduce (failure description),
149–150
sticky wickets (bugs), 192–194
stocking test labs, 299–305
stopping criteria. See continuation criteria
“A Story about User Stories and Test Driven
Development,’’ (Coplien, et al.), 538n18
story-point velocity, 542
straw-man plan, 609
stress (quality risk category), 22
stress interview, 349
stress-test case, for DataRocket, 96–98
string testing, 7, 609
structural test coverage, 610–611
structural (code-based/design-based/
glass-box/white-box) tests, 2–3, 8–9, 104
students, testers and, 337
subsystem chart, 186–188
subsystem combo-box definition, 175, 176
subsystem field (bug-tracking database),
167–168
subsystem lookup table, 175, 176
subsystems. See also component testing
of DataRocket, 167
hardware testing and, 563–564
of SpeedyWriter, 167–168
subsystem interfaces, 19
subsystem sanity test, 554
Summary field
anomaly-report template, 177
failure description, 149
summary worksheets. See test-case summary worksheets; test-suite summary worksheets
supplier quality engineering, 564–565
Suri, Deepti, 255n2
surrogate keys, 271
surrogate metric, 230
Surviving the Top Ten Challenges of Software Testing (Perry & Rice), 264n1, 389n4
suspension/resumption criteria. See continuation criteria
SUT. See system under test
SWAGs (scientific wild-ass guesses), 199, 609
syllabus/body of knowledge, 331, 334, 335
system architecture. See also test-system architecture
information-appliance project, 374
Omninet, 585–586
System Config column, 201
System Config worksheet, 201, 202
System Cookers, 63, 217, 221, 299–300, 436, 437
system diagnostics, 554
System Overview and Key Features section (IEEE 829 test-plan template), 73
System Requirements Document. See Omninet System Requirements Document system testing, 7. See also integration testing
DataRocket and, 7, 122–131
defined, 329, 609
partial, test-case summary worksheets and, 200, 201
quality risk categories and, 21–24
system under test (SUT), 86, 147, 294, 609
Systematic Software Testing (Craig & Jaskiel), 6n3, 354n10, 530n13
Systematic Test and Evaluation Process (STEP), 530, 531

target audience
accuracy and, 407–409
hardware-allocation plan and, 408
planned releases (logistics database) for, 281–282
reports and, 407–409
TBD (to be determined), 50, 51, 610
technical support, 397–398
technical tests, jump-start for, 599
technology
new/cutting edge, 290, 409–412
technology skills, 328
technology-centered test lab layout, 315
Technology Trigger, 545
templates. See IEEE 829 standard; test-case templates; test-plan template
temporary workers, 362–374. See also contractors; outsourcing
temps. See temporary workers
tests/testing. See also Agile methodologies; distributed testing; management; test management; testing process; specific tests
blended testing strategy, 536
cost of, 477–478
in dark, 400–402
Dilbert cartoons on, 320n2
early adoption and, 409–412
global view
economic context, 475–495
process context, 497–514, 514–527
project context, 497–514
insurance model and, 476, 494
integration, into software life cycle, 380–381
managing, 134
might test, should test, can test approach, 1–46, 326
peripheral duties added to, 386–388
skills, 328–329. See also skills
“We test everything that could break,” 507, 508
without documentation, 400–402
Test Administration Requirements section (IEEE 829 test-plan template), 74
test architecture, 82
test artifacts, 610. See also test systems
test automation, 264
jump-start for, 598
RBCS and, 598
regression-test gaps and, 120–122
result-interpretation errors and, 264
test automation engineer (SpeedyWriter job description), 343–344
test basis, shifting, 540–541
test cases, 91–109. See also test-case templates
defined, 610
documenting
continuum, 108, 109, 138–142
precision/details, 105–109, 138–142
test cases, (continued)
incremental improvement, 133–134
managing. See test-tracking spreadsheet
prioritizing, 212–213
quality risks and, 112–113
screen-oriented registration, 99, 100
with serious problems, 219
stress (DataRocket), 96–98
test conditions and, 91–92
test procedures v., 223
test specifications v., 138, 610
test systems and, 91
test charts, 263, 604
test conditions, 83–84
creating, 91–92
defined, 610
hardware testing and, 554
test configurations
options, compatibility and, 23
section, 38
test coverage, 113–115
test environment/crunch mode and,
260–261
test-plan template, 61–62
test coverage, 109–118, 610–611
behavioral, 610–611
bug, 116–118
code, 18
configuration, 113–115
data-flow, 18
decisions, bad, 110–111
defined, 610–611
mistakes, 110–111
regression-test gaps and, 110, 118–132. See
also regression-test gaps
structural, 610–611
test-system, quality and, 13–14
test cycles, 39, 68, 203–204. See also cycles; life
cycle
defined, 68, 611
test release process, 66
test-plan template, 68
test dashboards. See dashboards
test data, 136–138, 137
test development
phase, 38–39
test-plan template, 60–61
Test Documentation Requirements section
(IEEE 829 test-plan template), 74
test efficiency, 379
test engineering, 337, 338
jump-start for, 598
test engineer’s role, 336
test environments
category, 44
cleanliness of, 309–310
configuration
crunch mode and, 260–261
test labs and, 309–310
debugging in, 523
defined, 611
SpeedyWriter, 317–318
test-plan template, 61–62
test escapes, 67, 109–110
deadly, 265
deferred bugs v., 193
defined, 611
learning from, 133
low-fidelity test systems and, 110
nuclear-medicine device and, 265, 562
regression-test gaps and, 110
SUT and, 609
test execution, 38, 39
Gantt chart, 423, 424
managing. See test-tracking spreadsheet
monitoring, distributed test effort and,
442–443
outsourced projects and, 459–460
period, for test phases, 9–10
process, defining, 263
self-correcting element in, 16, 106
test-plan template, 62–69
bug isolation, 64–65
bug report classification, 64–65
resources, 62, 64
test cycles, 68
test hours, 69
test release management, 65–68
test-case tracking, 64
test failure, 65
test funding. See funding
test ghetto, 387
test granularity, 2–4, 498, 499, 605
defined, 605
fishnet analogy and, 4, 498, 499
skills perspective and, 329
spectrum, 4, 8
test group, in organizational context, 383–386
Test Hardware report, 280
test hours, 69, 224, 225
test item transmittal reports, 66
test labs, 293–318
accountability and, 305
case study, 314–317
configuration management, 306–309
defined, 293
equipment damage, 312–313
equipment management, 306–309
garden-variety, 294
human factors and, 310–314
inventory, 299–305
Inventory template, 300–303
layouts, 315, 317
Locations table, 299
logistics database and, 299, 307
need for, 294–295
planning for, 295–299
productivity and, 310–314
safety and, 298, 305–306
security and, 305–306
selecting location for, 295
STC, 272, 274, 281, 423, 424, 437
System Cookers, 63, 217, 221, 299–300, 436, 437
test environment cleanliness, 309–310
third-party, 304
tracking concerns, 305–306
test levels, 73. See also test phases
test logging, 221. See also test-log template
test management. See also management
directions of, 388–400
hardware testing and, 553–554
jump-start for, 598
managing testing and, 134
skills, RBCS and, 599
Test Management Approach (TMap), 530, 531
test managers
development peers and, 395–397, 486
management peers and, 397, 486
organizational challenges for, 377–419
/other managers, work relations, 388–400
roles/responsibilities of, 377–378
test maturity assessment
CMM, 454–455
information-appliance project, 547, 549
test maturity models (TMMs), 530, 549
test missions, 82, 378–381
test oracles, 539–540, 606
test organization
as development-project resource, 384–385
independent, 385–386
models, 338–341, 383–386
as part of development, 383–384
Test Overview section (IEEE 829 test-plan template), 73–74
test passes, 39, 66, 203–204, 611
test phases, 4–10, 498–501. See also life cycle
benefits of, 8–9
configuration section, 38
defined, 611
early starting of, 9, 10
planning section, 38
project life cycle and, 498–501
staffing section, 38
test-execution period and, 9–10
test plans, 49–77. See also test-plan template
action verbs and, 76
approval, 75–76
clarity and, 76
documentation, 76
drafts, 51
exercise, 77
floor-plan sketch for, 296
master, 51
multiple, 50–51
pertinence and, 76
quantity of, 50–51
reasons for, 49–50
rough draft, 77
selling, 75–76
in testware, 82
test platforms
configuration dimensions, 308–309
data configuration, 308
defined, 294, 611
test policies, 378–381
test procedures, test cases v., 223
Test Process Improvement (Koomen & Pol), 530n13
Test Process Improvement (TPI), 530, 531, 547, 549
Test Process Improvement (TPI), 530, 531, 547, 549
Test Progress chart, 224–225, 238
test projects. See projects
test recruiting, RBCS and, 599
test releases, 39, 65–68, 204, 611
Test Reporting Requirements section (IEEE 829 test-plan template), 74
test results
auditing/updating, 261–262
misinterpretation, 263–265
proper delivery of, 404–409
test results reporting, 15, 16. See also
test-results reporting templates
Test Schedule report, 274, 275
test schedules, 37–43
test script languages, 139, 608
test scripts, 6, 60, 61, 281, 544, 608. See also
scripted testing
test sequencing, 15
test services, project, 598–599. See also RBCS
test set. See test suites
test setting diagram, for distributed test effort, 54, 55
test specifications, 105. See also test cases
test cases v., 138, 610
test-specification levels of detail, 139–142
test status reports (exercise), 412–419
test strategy, 82
Test Suite/Case column, 200
test suites (test set), 50, 82, 83, 100
defined, 611
prioritizing, 212–213
role of, 84
test systems, 1
architecture. See test-system architecture
best practices for, 90–91
composition of, 80
coverage, quality and, 13–14
defined, 60, 79–80, 612
fidelity, 12–13, 86, 110, 604
IEEE 829 test-system templates, 99–104
maintainable, 605
principles for, 90–91
quality, 85–88
test artifacts v., 610
test cases and, 91. See also test cases
testers and, 89
test teams. See also testers; testing service providers
defining, 325–338
equal treatment of, 354–356
information-appliance project, 374
in-house, 462–466, 472
island and, 89, 517–525
management of, 352–357
manager titles and, 381–383
motivating, 353–362
Omninet, skills for, 375
roles/responsibilities, 386–388
selection, outsourcing and, 452–455
size, 325–327
staffing, 319–353
testing service providers v., 462–466
virtual, 215, 422, 468, 473
test technicians, 336
test to fail, 513, 612
test to pass, 513, 612
test tools, 81–82, 83
assessment of, RBCS and, 597
defined, 612
hardware testing and, 553–554
resource category, 43
scrimping on, 522
test tracking. See also test-tracking spreadsheet
"Case Study Info Appliance Client Test Tracking.xls, 255, 565
Omninet, 249
test triage, 15, 16
test yield, 612
test-assessment and -improvement framework. See Critical Testing Processes
test/bug coverage chart, 226–228
test-case library, 83–84, 610
test-case life cycle, 209–212
test-case setup, 83, 610
test-case summary worksheets, 223
columns in, 200–203
extended, 206
halfway into first cycle of system testing,
200, 201
major problems, 220
worrisome, 217
test-case teardown, 84, 610
test-case templates, 79
basic, 92–96
IEEE 829, 101–103, 610
screen-and-field, 98–99
test-design techniques and, 104–105, 105n4
test-case tracking, 64
TestCases table, 237
test-centric test lab layout, 317
TestConditions table, 237
test-design techniques, 104–105, 105n4
test-design template (IEEE 829), 100–101
Test-Driven Development, 538, 538n18
Tested Configurations report, 283, 284
Tester Assignments report, 272, 274
testers. See also test teams
assigning, 264–265
certification. See certification
education/training/certification, 331–336.
See also certification
hiring, 341–353
mistakes, 524–525
new, training and, 352–353
skills, 264, 327–331
test system and, 89
tester failure and, 612
traits for, 320–325
test-fulfillment chart, 225–226
Testing as a Service, 422
testing certification. See certification
testing investment. See investment
testing process, 534–536
assessment of, RBCS and, 597
best practices, 534–536
*Critical Testing Processes*, 190n8, 454–455,
454n5, 530n13, 531, 535, 535n16,
597
defined, 528
improving, 527–534
*inside looking out* perspective and, 534
management, 534–536
mature, 497–498, 528–529
overview, 534–536
*outside looking in* perspective and, 534–536
process-centric view of, 535n16
quality and, 10–36
quality control and, 528–529
Test Process section (IEEE 829 test-plan
template), 74
testing service providers, 421–422, 428–431
advantages of, 429
case studies and, 460–469
disadvantages of, 430
effective use of, 468–469
in-house test teams vs., 462–466
organizational challenges and, 466–468
test tasks for, 463–464, 472
testing skills, 328–329
testing/debugging task continuum, 466
test-log template (IEEE 829), 221–223
test-management tools, 200, 262. See also
test-tracking spreadsheet
test-phase sequencing, 9–10
test-plan template, 52–71
bounds section, 53–55
change history, 71
frequently asked questions, 71
IEEE 829, 52, 71–75
milestones schedule, 56, 57
overview section, 52–53
parts (list) of, 52
project risks/contingencies, 69–71
quality risks, 55–56
referenced documents, 71
test configurations, 61–62
test development, 60–61
test environments, 61–62
test execution, 62–69
transitions, 56–60, 602–603
test-policy document, 379–380
test-procedure template (IEEE 829), 103–104
test-process-assessment models, 529–530,
532

test-result interpretation errors. See
result-interpretation errors
test-results reporting templates (IEEE 829),
232–236
Tests by Location and Tester report, 276
test-suite setup, 610
test-suite summary worksheets, 224
extended, 206
major problems, 220
no problems, 221
worrisome, 218
test-suite yield, 212
test-system architecture, 79–91
congruent, 602
defined, 79
ingoing and, 79–84
good, 519
maintainability and, 85, 88, 519
principles for, 84–91
test-system failure, 612
test-tracking spreadsheet, 199–256
case studies and, 217–221, 236–248
charts, 223–232
efforts, 205–216
exercises, 249–256
extending, test-case details and, 215
linkages and, 261–262
minimalist, 200–205
testware, 81–84
Therac-25 nuclear-medicine device, 265,
562
thermal tests, 554, 558
third-party test labs, 304
TickIT, 532
time and date handling, 22
titles, test teams and, 381–382
*T-Map: Next* (Koomen et al.), 530n13
TMap (Test Management Approach), 530,
531
TMMs (test maturity models), 530, 531,
549
to be determined. See TBD
 Tolstoy, Leo, 514, 515, 515n7
tools (Inventory template), 302. See also test
tools
top-down approach, 38, 43
torque force, 557
Total Fixed curve, 181
Total Opened curve, 180, 181
Total Resolved curve, 180, 181
TPI (Test Process Improvement), 530, 531,
547, 549
tracking. See also bug-tracking database
software configurations, logistics database
and, 281–284
test lab and, 305–306
tragedy of the commons, 305
training. See also certification; RBCS
/education/certification, testers and,
331–336
new testers and, 352–353
RBCS and, 333, 475, 536, 599
traits (for testers), 320–325
transaction handling (quality risk category),
18, 21
transfer strategy, 69
transitions (test-plan template), 56–60
continuation criteria, 59, 602–603
entry criteria, 58–59, 603
exit criteria, 58, 60, 603
triage. See bug triage process; test triage
Triangle, Iron Box and, 37
Trough of Disillusionment, 545
trust
bug-tracking efforts and, 190–191
distributed test effort and, 446–447
outsourced projects and, 458–459

V
V model, 451, 456, 501–504, 547
value, of RBCS, 596
vendor testing, 512–513
vendor-quality problems, 427
vendors, 421, 424–428, 564–565. See also
distributed testing
vibrations/drops/shocks, 23–24, 312,
555–556, 560, 564
virtual test teams, 215, 422, 468, 473
vision statements, 378
voltages, temporary losses in, 555
volumes/capacities (quality risk category),
19–20, 22
volume/speed, of change, 537

W
waiver requested, 64
Walton, Mary, 91n3, 533
War and Peace (Tolstoy), 515n7
“We test everything that could break,” 507,
508
“welcome changing requirements, even late
in development”, 401, 537
white-box tests. See structural tests
Whittaker, James, 106, 106n5
widgets case study, 469–472
Windows CE computer, for automobile,
267
446
work schedules, reasonable, 356–359
work smarter, not harder, 379
work-breakdown structures, 38, 38n8, 42,
43
workers. See outsourcing; staffing; temporary
workers; test teams
Wren, Christopher, 84
Wysocki, Robert, 38n8

Y
“You can tell the pioneers...”, 409

Z
Zen regression, 118
zero-sum game, 351, 486, 544