
INDEX

- abstraction 204
 - level, selecting 83
 - problems 205
- accessible plans 110
- accounting, size 42
 - added, definition 42
 - base, definition 42
 - deleted, definition 42
 - example 44
 - modified, definition 42
 - new reusable, definition 43
 - reused, definition 42
 - total, definition 43
- accuracy
 - estimation 80, 125
 - improving 80
 - plans 111
- Ackerman, A. F. 139, 161
- added
 - and modified size, definition 42
 - size, definition 42
- advantages and disadvantages, debugging 196
- A/FR, appraisal to failure ratio 146, 147
 - versus unit test defects 147
- alternate PROBE methods 95
- analytical verification 271
 - issues 277
 - methods 277
- appraisal
 - cost of quality 145, 146
 - to failure ratio, A/FR 146
- Artemis Views 66
- assignments, part time 130
- ASTREE 279
- attributes, FST 236
- automated verification 278
- average defect fix times 139, 140

- Barnes, John 278, 280, 285
- base
 - additions 87
 - size, definition 42
- baseline
 - personal process 11
 - process 295

330 Index

- Bencher, D. L. 154, 161
- benefits, defined process 6
- bias
 - estimating 81
 - knowledge 82
- Blanchet, Bruno 279, 285
- Boehm, B. W. 119, 132, 135, 161
- Booch, Grady 226, 250
- BPS/360 257
- Brooks, F. P. 128, 132
- building checklists 176
- Bush, M. 139, 161
- Butler, R. W. 255, 285

- C++
 - coding standard 50
 - compile defects 198
 - defect
 - distribution 179
 - fix times 139
 - development time estimates 76
 - part size table 77
 - review rate vs. yield 193
 - size data 75, 77
- calculation
 - correlation 36
 - cost vs. quality 146
 - planned value 119
 - prediction interval 128
 - PROBE worksheet 91, 99
 - regression parameters 101, 102
 - significance 37
 - yield 188, 190
- Capability Maturity Model (CMM)
 - level versus defect density 134, 156
- challenges, development 309
- change 312
 - control 312, 320
 - managing 320
 - requirements, handling 312
- characteristics, successful teams 314
- characterizing a process 295
- checklists
 - building 176
 - code review, C++ 175
 - design review 184, 283
 - updating 177
 - using 170, 180
- classes and proxies 74
- clear
 - designs 185
 - plans 110
- ClearSpaces trace table 262, 264
- CMM, Capability Maturity Model
 - level versus defect density 134, 156
- coach, TSP 317
- COCOMO 66
- code
 - intensive development 288
 - quality, PQI 151
 - review 163
 - checklist 175
 - quality, PQI 151
 - script 174
 - time range 167
- coding standards 179
 - C++ 50
- coefficient, correlation 36
- combining estimates 80
- commitments 2
 - and plans 60
 - negotiating 311
- comparison, UML and PSP design 247
- compile
 - defects 167
 - found and missed 198
 - versus test defects 167
 - phase 31
 - PQI 151
 - review before 196
 - time 165, 166
- complete
 - and orthogonal functions 267
 - designs 221
 - functions 266
- completeness of verification 277
- completion date, estimating 124
- conceptual design 63, 70, 87
- considerations
 - design 210
 - estimating 80

- planning 129
 - PROBE 105
 - process development 306
 - verification 280
- constructive verification 278
- contents
 - defined process 12
 - software plan 60
- context switching 181
- control
 - change 312, 320
 - maintaining project 312
- conventions, product 257
- coordination
 - design 287
 - team 131
- COQ, cost of quality 145, 146
- correlation 36
 - calculating 36
 - definition 36
 - formula 36
- cost
 - finding and fixing defects 139
 - of quality (COQ) 145, 146
- countable
 - proxy 73
 - size measures 40
- counters, size 48
 - LOC 49
 - logical 49
 - physical 49
- counting
 - deletions and modifications 52
 - program elements 49
 - standards
 - database 40
 - LOC 40
- creativity 5
- creep, requirements 65
- criteria for selecting a proxy 71
- Crosby, P. B. 135, 145, 161
- current process 295
- Curtis, Bill 207, 224
- customizable proxy 74
- customizing the development process
 - 289
- data
 - database estimating 86
 - defects per hour 140, 193
 - DRL 193
 - estimating with 85
 - limited 107
 - outliers 106, 126
 - recording 21
 - using limited 107
- database
 - counting standard 40
 - elements as proxy 86
 - estimating data 86
 - size measurement 37, 39
- Davis, N. 133, 162, 314, 327
- death march project 1
- debugging
 - advantages and disadvantages of 196
 - definition 195
- deChampeaux, D. 229, 250
- decision table 269
- defect
 - C++ fix times 139
 - compile vs. test 167, 168
 - cost of finding and fixing 139
 - data recording 20, 24
 - density 46, 191, 226
 - versus CMM level 134, 156
 - description 24
 - escapes 189
 - filtering 144
 - fix reference 24
 - fix time 27, 140
 - average 139, 140
 - example 28
 - relative 165
- found
 - and missed by compile 198
 - per hour 140
- injection
 - 810 developers 173
 - and removal rates (3240 programs)
 - 150
- levels, CMM and TSP 156
- multiple 27
- Pareto distribution (C++) 179

332 Index

- defect (*continued*)
 - per function point 141
 - per hour
 - (3240 programs) 140
 - data 193
 - per KLOC (810 developers) 141
 - versus design/code time 149
 - prevention 158, 172
 - recording 24
 - recording log 25
 - example 25
 - instructions 26
 - removal leverage (DRL) 192
 - removal rate 193
 - repair time, Xerox 138
 - type standard 27, 177
 - types 141
 - versus design time 149
- defined process
 - benefits 6
 - contents 14
- defining a
 - personal process 7, 13
 - process 291
 - reasons for 290
 - steps in 292
- definition
 - added and modified size 42
 - added size 42
 - base size 42
 - code
 - quality, PQI 151
 - review quality, PQI 151
 - compile PQI 151
 - correlation 36
 - debugging 195
 - defect
 - escapes 189
 - removal leverage (DRL) 192
 - deleted size 42
 - delta time 22
 - design 204
 - conceptual 63
 - quality, PQI 151
 - review quality, PQI 151
 - DRL 192
 - earned value 119
 - escapes 189
 - estimated proxy size 92
 - inspection 164
 - interruption time 22
 - LPI 128
 - modified size 42
 - new reusable size 43
 - part 70
 - plan 60
 - planned value 119
 - prediction interval 126
 - process 12, 292
 - product 70
 - program quality, PQI 151
 - projected size 92
 - quality 135
 - measures, PQI 151
 - regression line 86
 - requirements 212
 - reusable size, new 43
 - reused size 42
 - review 164
 - quality, PQI 151
 - yield 188
 - self-directed team 315
 - significance 37
 - size accounting types 42
 - software quality 135
 - task time 130
 - team 314
 - total size 43
 - unit test PQI 151
 - unprecedented product 84
 - UPI 128
 - yield 143, 188
 - deleted size 42
 - deletions, counting 52
 - delivered defects vs. CMM level 134
 - delivering quality produces 313
 - delta time 22
 - Deming, W. E. 7, 10
 - density, defect 46, 191
 - depth of verification 277
 - Derby, Kentucky 172
 - description, defect 24

- design 203
 - as a learning process 209
 - by contract 278
 - clear and understandable 185
 - completeness 221
 - conceptual 63, 70, 87
 - considerations 210
 - coordination 287
 - definition 204
 - detailed 215
 - example, state machine 241
 - for security 187, 289
 - framework 208
 - hierarchy 248
 - high-level (HLD) 215
 - intensive development 288
 - large-scale systems 248
 - levels 210
 - notation 228
 - objective 226
 - PQI 151
 - precision 220
 - process 207
 - product standards 257
 - pyramid 211
 - quality 220
 - reasons to 206
 - representation 226
 - review 163, 181
 - checklist 184, 283
 - PQI 151
 - principles 183
 - script 183, 282
 - strategy 185
 - reviewable 185
 - secure 289
 - specification 214
 - structure 229
 - standards 257
 - strategies 216
 - templates 225–248
 - comparison with UML 247
 - structure 230
 - using 246
 - time versus
 - defects 149
 - development time (810 developers) 288
 - product size (810 developers) 289
 - productivity (810 developers) 206
 - verification 253
 - why do? 206
- detailed-level design (DLD) 215
- developer view of project success 322
- development
 - challenges 309
 - code-intensive 288
 - design-intensive 288
 - estimates 76
 - incremental 32
 - methods, generally used 4
 - practices, disciplined 6
 - principles 290
 - process, customizing 289
 - script, PSP0 19
 - strategies 290
 - example 217
 - selecting 220
 - time versus design time (810 developers) 288
- diagram, Ishikawa 159
- disadvantages, debugging 196
- discipline
 - logic for 4
 - PSP 6
- disciplined practices 6
- distribution, *t*, values 38
- DLD, detailed-level design 215
- domain-specific verification 278
- DRL, *see* defect-removal leverage
- Dyer, M. 271, 285, 314, 327
- earned value (EV) 119
 - definition 119
 - example 120
 - limitations 119
 - usefulness 119
- economics
 - review 194
 - software quality 136

334 Index

- effective
 - methods 324
 - teamwork 313
- efficiency
 - review 195
 - verification 278
- elements, process 292
- end date, estimating 124
- environment, TSP 323
- equation
 - beta parameters 102
 - correlation 36
 - prediction interval 128
 - regression 86
 - significance 37
- error, estimating 80, 125
- escapes, defined 189
- estimated
 - development time data 76
 - proxy size (E) 91, 92
- estimating 69
 - abstraction level 83
 - accuracy 80, 125
 - improving 80
 - base additions 87
 - bias 81
 - combining estimates 80
 - completion date 124
 - considerations 80
 - data, size 75
 - database projects 86
 - error 80, 125
 - example 100
 - from data 70, 85
 - judgment 105
 - knowledge bias 82
 - large systems 83
 - limited data 95
 - nonprogramming tasks 102, 104
 - overcompensation 107
 - parts additions 89
 - parts size 87
 - practice 105
 - PROBE method for 88
 - procedure
 - size 92
 - time 93
 - project completion 124
 - proxy-based 71, 87
 - reused parts 89
 - script, PROBE 96
 - size 69
 - example 100
 - software 69
 - software 69
 - task hours 113
 - time, with PROBE 93
 - tools 66
 - unprecedented products 84
 - with data 85
 - with limited data 95
 - with proxies 75
 - yield 190
- EV, earned value 119
- evolution
 - process 294
 - PSP process 8
- example
 - ClearSpaces trace table 262, 264
 - defect fix time 28
 - Defect Recording Log 25
 - development strategies 217
 - earned value 120
 - estimating 100
 - execution table 260
 - for-loop verification 271
 - FST attributes 236
 - Functional Specification Template 234
 - Logical Specification Template 240
 - LogIn state machine 241, 269
 - execution table 260
 - Operational Specification Template 231
 - Project Plan Summary
 - PSP0 31
 - PSP1 94
 - regression calculation 101
 - repeat-until verification 276
 - report-writing data 103
 - Schedule Planning Template 116, 121
 - search state machine 242, 270

- size
 - accounting 44
 - estimating 100
- state
 - machine design 241
 - machine verification 268–270
 - Specification Template 238
 - Task Planning Template 118, 123
 - trace table 262
 - while-loop verification 272, 273
- execution table
- example 260
- LogIn 260
- verification 258

- Fagan, Michael 164, 199, 202
- failure cost of quality 145, 146
- filter view of yield 144
- firm requirements 321
- fix reference, defect 24
- fix time
 - defects 27, 139, 140
 - relative 165
- Flaherty, M. J. 48, 55
- for-loop verification 271
- forms
 - TSP Weekly Summary Report 124
 - why helpful 16
- formula for
 - beta parameters 102
 - correlation 36
 - prediction interval 128
 - regression 86
 - significance 37
- framework for
 - design 208
 - project planning 63, 112
 - understanding personal work 3
- FST, *see* Functional Specification Template
- function points 53
 - defects per 141
- Functional Specification Template (FST) 233
 - example 234
 - instructions 235
- functions
 - complete 266
 - complete and orthogonal 267
 - orthogonal 267
- future, personal 326

- Gage, D. 254, 285
- Gale, J. L. 159, 162
- Garmus, D. 53, 55
- general purpose verification 278
- Gilb, Tom 199, 202
- goals, process 293
- Greene, Maurice 7
- guidelines
 - design strategy 216
 - planning 129

- handling requirements changes 320
- help, from manager 129
- hierarchy
 - design 248
 - implementation 249
- high-
 - level design (HLD) 215
 - quality parts 310
- histogram, compile vs. test defects 168
- history, learning from 325
- HLD, high-level design 215
- how a process helps 11, 15
- Humphrey, W. S. 47, 55, 60, 67, 119, 120, 132, 164, 199, 202, 295, 296, 308, 314, 327

- IBM 47, 57, 129, 141, 257, 258, 320
- IFPUG, International Function Point Users Group 53
- implementation
 - hierarchy 249
 - sensitive proxies 74
- improvement
 - estimating accuracy 80
 - personal 15
 - practices, PSP 157
 - strategies 296, 335
- IMS 141
- incomplete knowledge bias 82
- increases, workload 65

336 Index

- incremental development 32
- index, process quality, *see* process quality index
- industrial project performance 310
- information
 - in plans 61
 - needs, quality management 279
- initial process 297
- injection, defect, 810 developers 173
- inspections
 - definition 164
 - vs. reviews 199
- Instructions
 - Defect Recording Log 26
 - Functional Specification Template 235
 - Operational Specification Template 232
 - PSP0 Project Plan Summary 32
 - Size Estimating Template 98
 - State Specification Template 239
 - Time Recording Log 23
- International Function Point Users Group, IFPUG 53
- interruption time 22
- interruptions 23
- interval, prediction 93, 126, 128
- intuitive methods 4
- Ishikawa diagram 159
- issues
 - analytical verification 277
 - planning 65
 - review 194
 - safety 187
 - security 187
- Jacky, Jonathan 214, 224, 227, 251, 280, 285
- Jones, Capers 53, 55, 141, 142, 162
- judgment, estimating 105
- Juran, J. M. 145, 162
- Karnaugh map 266
- Kentucky Derby 172
- Knight, J. C. 199, 202
- knowledge bias, in estimating 82
- large
 - projects 310
 - scale design 248
 - systems estimating 83
- Larus, J. R. 280, 285
- launch, TSP 316
- learning
 - from history 325
 - process, design as a 209
 - to use the PSP 8
- levels
 - design 210
 - verification 278
- Levendel, Y. 154, 162
- limitations, EV 119
- limited data
 - estimating with 95
 - using 107
- Linberg, K. R. 322, 327
- line of code
 - counters 49
 - counting standard 40
- linear regression 86
 - parameters, calculating 101, 102
 - requirements for 95
- Littlewood, Bev 255, 285
- LOC, *see* line of code
- Log
 - Defect Recording 25
 - Time Recording 22
- logic
 - for the PSP 157
 - for the TSP 314
 - notation 227, 228
 - software engineering discipline 4
 - Specification Template (LST) 240
 - example 240
- logical LOC counters 49
- LogIn state machine
 - design 234, 238, 240
 - example 241
 - execution table 260
 - State Specification Template 238
 - verification 269
- loop verification 271

- LPI, lower prediction interval 128
- LST, *see* Logic Specification Template
- maintaining the plan 312, 318
- making the schedule 115
- managers, getting help from 129
- managing
 - change 312, 320
 - personal
 - quality 154
 - work 324
 - product quality 153, 156
 - quality 153, 321
 - information needed 279
 - steps in 279
 - task time 114
 - your own project 318
- Mandeville, W. A. 145, 162
- map, Karnaugh 266
- Martin, D. 315, 327
- Mays, R. G. 159, 162
- maze, testing 255
 - possible paths 256
- McConnell, Steve 241, 251
- measures
 - countable 40
 - precise 40
 - process 15
 - PSP0 20
 - quality 143
 - review 187
 - size 35, 53
 - specific 40
 - yield 143
- measuring software size 35
- methods
 - effective 324
 - generally used 4
 - intuitive 4
 - PROBE 85, 88
 - scalable 203
 - verification 279
- Microsoft 280
 - Project 66
- mile, time for man and horse 171
- milestones 119
- Mills, H. D. 271, 285
- Modarress, B. M. 145, 162
- modifications, counting 52
- modified size, definition 42
- motivation 129
- multiple defects 27
- needs, process 293
- negotiating commitments 311
- new reusable size, definition 43
- new tasks, estimating 102
- nonprogramming tasks, estimating 102, 104
- notation
 - design 228
 - logic 227, 228
- objectives
 - design 226
 - review 199
- O'Neill, D. 139, 162
- operational
 - processes 6
 - Specification Template (OST) 230
 - example 231
 - Instructions 232
- orthogonal functions 267
- OS/360 57, 141
- Osprey aircraft 254
- OST, *see* Operational Specification Template
- other size measures 53
- outliers 106, 126
- overcompensation in estimating 83, 107
- overview, PSP 8
- parameters, regression 102
- Pareto distribution, defects 179
- part
 - additions 89
 - base 87
 - definition 70
 - LOC table
 - C++ 77
 - Pascal 77
 - quality 310

338 Index

- part (*continued*)
 - reused 89
 - size
 - determining 87
 - estimated 87, 91
 - table 77
 - time assignments 130
 - type, determining 87
- Pascal
 - defect fix times 140
 - development time estimates 76
 - part size table 77
 - size data 73, 77
- paths, through testing maze 256
- PDL, programming design language 240
- performance, industrial software projects 310
- period plans 111
- personal
 - goals 326
 - improvement 15
 - methods, effective 324
 - planning 62
 - process
 - baseline 11
 - defining and using 7
 - how it helps 11
 - learning to use 8
 - why needed? 12
 - quality
 - management 153
 - practices 142
 - strengths and weaknesses 324
 - work, planning and managing 324
- Personal Software Process (PSP)
 - defect-type standard 27
 - discipline 6
 - evolution 8
 - exercise program size ranges 64
 - improvement practices 157
 - logic for 157
 - overview 8
 - principles 157, 290
 - process evolution 8
 - purpose 3
 - quality strategy 135
 - size counting 49, 50
 - strategy 5, 291
 - Time Recording Log 22
 - tool support 34
 - using 309
- personal
 - strengths and weaknesses 324
 - work, planning and managing 324
 - working style 287
- PERT 118
- philosophy, PROBE method 82
- physical LOC counters 49
- PIP, *see* Process Improvement Proposal
- planning 57, 109
 - a software project 62
 - considerations 129
 - framework 63, 112
 - guidelines 129
 - issues 65
 - personal work 62, 324
 - process 58
 - requirements 61
 - script, PSP0 19
 - steps 62
 - the schedule 116
 - tools 66
 - why first in the PSP course 58
- planned value (PV) 119
 - calculating 119
- plans
 - accessible 110
 - accurate 111
 - and commitments 60
 - clear 110
 - contents of 60
 - definition 60
 - information provided by 61
 - maintaining 318
 - period 111
 - precise 111
 - project 111
 - quality 65
 - requirements for 61, 109
 - accessible 110
 - accurate 111
 - clear 110
 - precise 111
 - specific 110

- what they are 60
 - why make? 59
- points, outlier 126
- possible proxies 74
- postmortem phase
 - PSP0 Script 21
 - why important 17
- PQI, *see* process quality index
- practice 325
 - estimating 105
 - verification 281
- practices
 - PSP improvement 157
 - quality 142
 - review 194
 - trace table 264
- precise
 - designs 220
 - plans 111
 - size measures 40
- prediction interval 93, 126
 - calculating 128
 - defined 126
- preparation for TSP 9
- prevention, defect 158, 172
 - strategies 158
- Primavera 66
- principles
 - design review 183
 - development 290
 - process development 290
 - PSP 157, 290
 - requirements uncertainty 207, 208
 - review 168
 - size estimating 69
- PROBE 85
 - alternate methods 95
 - Calculation Worksheet 91
 - Instructions 99
 - considerations 105
 - estimating considerations 105
 - Estimating Script 96
 - limited data 95
 - method 82, 85, 88
 - philosophy 82
 - script 96
 - time estimating 93
 - with limited data 95
- problems, abstractions 205
- procedure, estimating 87, 88, 92, 93
- process
 - baseline 11, 295
 - benefits of 6
 - characterizing 295
 - contents 12
 - PSP0 14
 - current 295
 - customizing 289
 - defined 12
 - defining 291
 - personal 7, 13
 - definition 12, 292
 - design 207
 - development
 - considerations 306
 - principles 290
 - strategy 290, 296
 - development Process (PDP) 299
 - customizing 289
 - script 299
 - elements 292
 - evolution 8, 294
 - goals 293
 - how helps 11, 15
 - Improvement Proposal (PIP) 200, 292, 299
 - initial 297
 - measures 15
 - needs and goals 293
 - operational 6
 - planning 58
 - Product Maintenance (PMP) 302
 - script 304
 - Prototype Experimental (PEP) 302
 - script 303
 - PSP0 14
 - development script 19
 - planning script 19
 - process script 18
 - PSP1 Project Plan Summary 94
 - PSP3 Development 300
 - quality index (PQI) 150–153
 - code quality 151

340 Index

- process (*continued*)
 - code review quality 151
 - definition 150
 - design quality 151
 - design review quality 151
 - profiles 152
 - program quality 151
 - sample programs 152
 - versus test defects 153
- script, PSP0 18
- self-improvement 3
- strategies, improvement 296, 335
- target 296
- TSP launch 316
- why
 - a process is needed 12
 - define a process 13
- producing the
 - relative-size table 78
 - schedule 113
- product
 - conventions 257
 - definition 70
 - design standards 257
 - Maintenance Process (PMP) 302
 - script 304
 - quality 136, 313
 - managing 156
 - reviewable 172
 - size vs. design time 289
 - standards 257
 - unprecedented 84
- productivity 47, 130
 - calculating 47
 - variations 48
 - versus design time 206
- program
 - elements, counting 49
 - quality, in PQI 151
 - size ranges 64
 - size with and without design (810 developers) 289
- programming
 - design language (PDL) 240
 - style 32
- progress tracking and reporting 319
- project
 - completion, projecting 124
 - control, maintaining 312
 - death march 1
 - large 310
 - managing your own 318
 - performance, industrial 310
 - Plan Summary
 - PSP0 30, 31
 - PSP1 94
 - planning framework 63, 112
 - plans 111
 - scale 287
 - success, developer view 322
 - success vs. size 310
 - tracking and reporting 319
 - troubled 310
- projected size (P) 92
 - definition 92
- projecting completion dates 124
- proper state machine 268
- Prototype Experimental Process (PEP) 302
 - script 303
- prototyping 209
- proxy
 - based estimating 71, 87
 - classes as 74
 - countable 73
 - criteria for 71
 - customizable 74
 - database programming 86
 - easily visualized 73
 - in estimating 75
 - possible 74
 - relative size table 77–79
 - selecting 71
 - selection criteria 71
 - sensitive to implementation 74
 - size
 - estimated 91
 - ranges 77–79
 - visualized 73
- pseudocode 240
- PSP, *see* Personal Software Process

- PSP0 14
 - contents 14
 - development script 19
 - flow 14
 - measures 20
 - planning script 19
 - postmortem script 21
 - process flow 14
 - process script 18
 - Project Plan Summary 30
 - example 31
 - Instructions 32
- PSP1 Project Plan Summary example 94
- PSP2
 - Design Review Checklist 184
 - Design Review Script 183
- PSP3.0
 - development script 301
 - process 300
- purpose
 - PSP 3
 - TSP 314
- PV, planned value 119
- pyramid, design 211
- quality
 - cost of (COQ) 145
 - definition 135
 - design 220
 - economics of 136
 - index, PQI 150
 - large projects 310
 - management 153
 - in TSP 321
 - information needs 279
 - personal 154
 - steps 279
 - measures 143
 - parts, need for 310
 - plan 65
 - PQI 150–153
 - practices, personal 142
 - product 136, 156, 313
 - profiles 152
 - software 133, 135
 - strategy, PSP 135
- r*, correlation value 36
- Ragland, B. 139, 162
- ranges, program size 64
- rate
 - defects injection and removal (3240 programs) 140, 150
 - review 192
- record, the mile 171
- recording
 - defect data 21, 24
 - time data 21
- redesign 247
- regression
 - calculation example 101
 - equation 86
 - line, definition 86
 - linear 86
 - parameters, calculating 102
- relative size table, producing 78–79
- Remus, H. 139, 162
- repair time, Xerox defects 138
- repeat-until verification 276
- report, TSP weekly 124
- reporting project progress 319
- representation, design 226
- requirements 2, 186
 - changes, handling 320
 - creep 65
 - definition 212
 - firm 321
 - for linear regression 95
 - for plans 61, 109
 - for size measures 40
 - instability, managing 320
 - planning 61
 - size measures 40
 - specification-design cycle 213
 - uncertainty principle 207, 208
- results, TSP 322
- retrospective verification 278
- reusable size, new, definition 43
- reuse standards 258
- reused
 - parts 89
 - size, definition 42
- REVIC 66

342 Index

- review 163
 - before or after compile 196
 - code 163
 - definition 164
 - design 163, 181
 - checklist 184, 283
 - principles 183
 - script 183, 282
 - economics 194
 - efficiency 195
 - issues 194
 - measures 187
 - objectives 199
 - on the screen 194
 - practices 194
 - principles 168
 - rate 192
 - vs. yield 148, 193
 - strategy 180
 - versus inspection 199
 - why do 164
 - yield 188
- reviewable
 - designs 185
 - products 172
- rewards of teamwork 322
- Rumbaugh, James 233, 251
- running, time for mile 171
- Russell, G. W. 139, 162

- safety 187
- Santayana, G. 325, 327
- scalable development methods 203
- scale, project 287
- schedule
 - making 115
 - planning 116
 - planning Template 121
 - example 121
 - producing 113, 115
 - steps in 115
- Schneider, Geri 230, 251
- Schulmeyer, G. 159, 162
- screen, reviewing on 194
- script
 - code review 174
 - design review 183, 282

- PROBE 96
- Process Development Process (PDP) 299
- Product Maintenance Process (PMP) 304
- Prototype Experimental Process (PEP) 303
- PSP0
 - development 19
 - planning 19
 - postmortem 21
 - process 18
- PSP3 development 301
- search state machine
 - design strategy 243
 - example 242
 - state diagram 246
 - State Specification Template 245
 - verification 270
- SEER 66
- security 187, 289
- SEI, Software Engineering Institute 1, 309
- selecting
 - a development strategy 220
 - a proxy 71
 - an abstraction level 83
- self-
 - directed team 315
 - improvement process 3
- Shooman, M. L. 139, 162
- significance 37
 - calculating 37
 - definition 37
- size
 - accounting 42
 - elements 42
 - example 44
 - counters 48, 49
 - data
 - C++ 75
 - Pascal 73
 - database programs 37, 39
 - estimated part (E) 87
 - estimating 69, 96, 100
 - principles 69
 - procedure 92

- Template 90
- Template Instructions 98
- measures 35, 53
 - countable 40, 73
 - other than LOC 53
 - precise 40
 - requirements 40
 - specific 40
- measuring 35, 48
- part, determining 77, 87
- projected part (P) 92
- proxy, estimated 91
- ranges, PSP programs 64
- relative, table 78–79
- versus
 - design time (810 developers) 289
 - project success 310
- skill 3, 5
 - required for TSP 314
- software
 - design 203
 - discipline, logic for 4
 - engineering discipline 4
 - Engineering Institute 1, 309
 - estimating 69
 - plan, contents 60
 - planning 109
 - quality 133, 135
 - definition 135
 - economics 136
 - size, measuring 35
- soundness, verification 277
- SOW, statement of work 66
- SPARK 280
- specific
 - plans 110
 - size measures 40
- specification, design 214, 229
- Splint 280
- SST, *see* State Specification Template
- stability, requirements 320
- standard
 - coding 50, 179
 - database counting 40
 - defect 27, 177
 - design 257
 - line-of-code counting 40
 - reuse 258
 - size counting 40
- Standish Group 310, 327
- state
 - diagram
 - LogIn 237
 - search 246
 - search design strategy 243
 - machine
 - design example 241
 - LogIn 268
 - proper 268
 - verification 265, 268, 269
- Specification Template (SST) 236
 - instructions 239
 - LogIn example 237, 238
 - search example 245
- statement of work, SOW 66
- status report, TSP 124, 320
- steps in
 - defining a personal process 292
 - planning 62
 - quality management 279
 - scheduling 115
- strategy
 - defect prevention 158
 - design 216, 217
 - design review 185
 - guidelines, design 216
 - improvement 296, 335
 - process
 - development 296
 - improvement 290
 - PSP 5
 - quality, PSP 135
 - review 180
 - search state machine 243
 - selecting 220
 - testing 5
 - verification 281
- strengths, personal 324
- structure, PSP design templates
 - 229, 230
- style
 - programming 32
 - working 287
- success, project, vs. size 310

344 Index

- successful
 - project, developer view 322
 - team characteristics 314
- support, tool 34
- switching tasks, cost of 131, 181
- t*-distribution, table of values 38
- table
 - decision 269
 - t*-distribution 38
 - trace 262
 - ClearSpaces 264
- target process 296
- task
 - hours, estimating 113
 - Planning Template 118, 123
 - switching 131
 - time 130
 - definition 130
 - estimating 113, 130
 - managing 114
- team
 - characteristics, successful 314
 - coordination 131
 - definition 314
 - of one, TSP 323
 - self-directed 315
 - Software Process (TSP) 9, 309, 313
 - coach 317
 - defect levels 156
 - environment 323
 - launch process 316
 - logic for 314
 - preparation for 9
 - purpose 314
 - quality management 321
 - results 322
 - skills required 314
 - team of one 323
 - Weekly Summary Status form 124, 320
 - successful, characteristics 314
- teambuilding 314
- teamwork
 - rewards of 322
 - sustaining 313
- template
 - design 225
 - Functional Specification (FST) 233
 - example 234
 - Instructions 235
 - Logic Specification (LST) 240
 - example 240
 - Operational Specification (OST) 230
 - example 231
 - Instructions 232
 - Schedule Planning 116, 121
 - Size Estimating 90
 - Instructions 98
 - State Specification (SST) 236
 - example 238, 245
 - Instructions 239
 - Task Planning 118, 123
- test
 - defects 167, 168
 - time 165
- testing
 - maze paths 255, 256
 - strategy 5
- Thayer, T. A. 198, 202
- time
 - code-review 167
 - compile 165, 166
 - delta 22
 - estimating, with PROBE 93
 - for mile, men and horses 171
 - interruption 22
 - record, for mile 171
 - recording 21
 - Log 22
 - Log Instructions 23
 - task 113, 114, 130
 - test 165
- tool
 - estimating and planning 66
 - support, PSP 34
- total
 - defect injection, 810 developers 173
 - size, definition 43
- trace tables 262
 - ClearSpaces 262, 264
 - practices 264

- tracking project progress 319
- trend line, regression 86
- troubled projects 310
- TSP, *see* Team Software Process
- types, defect 141

- UML, Universal Modeling Language 226, 247
 - compared with PSP templates 247
- uncertainty principle, requirements 207, 208
- undercompensation 107
- understandable designs 185
- understanding, frameworks for 3
- unit test
 - defects vs. A/FR 147
 - PQI 151
- unprecedented products
 - definition 84
 - estimating 84
- updating checklists 177
- UPI, upper prediction interval 128
- usefulness, EV 119
- using
 - a checklist 170, 180
 - design templates 246
 - effective methods 324
 - limited data 107
 - a personal process 7
 - PSP 309

- values, *t*-distribution 38
- van Genuchten, M. 139, 162
- VB.NET 31
- verification
 - analytical 277
 - automated 278
 - completeness 277
 - considerations 280
 - constructive 278
 - depth 277
 - design 253
 - efficiency 278
 - execution table 258
 - for-loop 271
 - issues 277
 - level 278
 - loop 271
 - methods 279
 - practice 281
 - proper state machine 268
 - examples 268–271
 - repeat-until 276
 - retrospective 278
 - soundness 277
 - state machine 265, 269, 270
 - strategy 281
 - trace-table 262
 - while-loop 272, 273
 - why do? 254
- verifying designs 253
- visualizing, proxy 73

- weaknesses, personal 324
- Weekly Summary Report, TSP 124, 320
- Weller, E. F. 139, 162
- what plans are 60
- while-loop verification 272, 273
- why
 - define a process? 13
 - design? 206
 - do reviews? 164
 - forms are helpful 16
 - make plans 59
 - planning first in PSP 58
 - postmortem important 17
 - processes are needed 12
 - verify designs? 254
- workload increases 65
- Worksheet, PROBE Calculation 91
 - Instructions 99
- World record, mile 171

- Xerox 138

- Yang, Jinlin 280, 285
- yield 143
 - calculation 188, 190
 - definition 143
 - estimates of 190
 - filter view of 144
 - for 810 developers 191

346 Index

yield (*continued*)

PSP class 145

review 188

versus review rate

for 25 C++ programs 193

for 810 developers 148

Zells, Lois 118, 132