Christof Ebert • Reiner Dumke

Software Measurement

Establish - Extract - Evaluate - Execute

With 157 Figures and 50 Tables

fyj Springer

Contents

1. Introduction 1
- 1.1. The Purpose of the Book 1
- 1.2. Measurement Standards 4
- 1.3. How this Book Is Organized 7
- 1.4. Who Should Read this Book? 10
- 1.5. Authors 12

2. Making Measurement a Success - A Primer 17
- 2.1. Why Measurement? 17
- 2.2. The Need for Measurement 20
- 2.3. A Simple and Effective Measurement Process 23
  - 2.3.1 The E4—Measurement Process 23
  - 2.3.2 Establish 25
  - 2.3.3 Extract 28
  - 2.3.4 Evaluate 33
  - 2.3.5 Execute 34
- 2.4. Hints for the Practitioner 37
- 2.5. Summary 39

3. Measurement Foundations 41
- 3.1. Introduction to Measurement Foundations 41
- 3.2. Foundations of Software Measurement 42
- 3.3. Theoretical Foundations 54
- 3.4. Software Engineering with Measurement 63
- 3.5. Analyzing Measurement Data 67
- 3.6. Empirical Validation: Avoiding the Shotgun Approach 71
- 3.7. Hints for the Practitioner 72
- 3.8. Summary 72

4. Planning the Measurement Process 73
- 4.1. Software Measurement Needs Planning 73
- 4.2. Planning with Measurement Frameworks 73
- 4.3. Holistic Planning of Measurement: The CAME Approach 84
- 4.4. Hints for the Practitioner 89
- 4.5. Summary 90

5. Performing the Measurement Process 91
- 5.1. Measurement Needs Tools 91
- 5.2. Instrumenting the Measurement Process 92
  - 5.2.1 Process Measurement and Evaluation 92
  - 5.2.2 Product Measurement and Evaluation 93
  - 5.2.3 Resource Measurement and Evaluation 95

5.2.4 Measurement Presentation and Analysis 95
5.2.5 Measurement Training 96
5.3. Case Study: Static Code Analysis 96
5.4. Solutions and Directions in Software e-Measurement 102
5.5. A Service-Oriented Measurement Infrastructure 104
5.6. Hints for the Practitioner 107
5.7. Summary 108

6. Introducing a Measurement Program 109
6.1. Towards Useful and Used Measurements 109
6.2. The Measurement Life-Cycle 109
6.3. Setting up the Measurement Program 110
6.4. Roles and Responsibilities in a Measurement Program 113
6.5. Using Measurements: Success Recipes 117
6.5.1 Measurement and Analysis 117
6.5.2 Measurement Definition and Collection 120
6.5.3 Data Quality 126
6.5.4 Analyzing, Visualizing, and Presenting Measurements 129
6.5.5 Statistical Traps and How to Avoid Them 133
6.5.6 Performance Indicators and Scorecards 141
6.5.7 Storing Measurements: The History Database 145
6.5.8 The People Impact 148
6.5.9 The Dark Side of Measurement 154
6.6. The Cost and the ROI of Software Measurement 158
6.7. Hints for the Practitioner 160
6.8. Summary 163

7. Estimation of Size, Effort and Cost 165
7.1. The Importance of Estimation 165
7.2. An Overview on Estimation Techniques 165
7.3. Using the COSMIC Full Function Point Approach 169
7.4. Case Study: Feasibility Study with COSMIC FFP 180
7.5. Case Study: Estimation for IT Systems 183
7.6. The Software Estimation Crisis 194
7.7. Hints for the Practitioner 197
7.8. Summary 198

8. Project Management 199
8.1. Measurement and Project Management 199
8.2. Software Project Management 200
8.3. Measurements for Project Control 205
8.3.1 Basic Project Planning and Tracking 205
8.3.2 Earned Value Analysis 212
8.3.3 Measurements for Requirements 217
8.3.4 Measurements for Testing 222
8.4. Agile Projects and Lean Measurement 228

8.5. Risk Management ... 232
8.6. The Project Outlook: Forecasts and Predictions ... 238
8.7. Hints for the Practitioner ... 240
8.8. Summary ... 243

9. Quality Control and Assurance ... 245
9.1. Assuring the Quality of Software Systems ... 245
9.2. Fundamental Concepts ... 247
9.2.1 Defect Estimation ... 247
9.2.2 Defect Detection, Quality Gates, and Reporting ... 251
9.2.3 Case Study: Quality Gates ... 252
9.3. Early Defect Removal ... 258
9.3.1 Reducing Cost of Non-Quality ... 258
9.3.2 Planning Early Defect Detection Activities ... 260
9.3.3 Identifying Error-Prone Components ... 262
9.3.4 Defect Predictions in Embedded Systems ... 268
9.4. Validation and Testing ... 273
9.4.1 Test Measurement ... 273
9.4.2 Analyzing Defects ... 283
9.5. Software Reliability Prediction ... 287
9.5.1 Practical Software Reliability Engineering ... 287
9.5.2 Applying Reliability Growth Models ... 291
9.6. The Return on Investment from Better Quality ... 294
9.7. Hints for the Practitioner ... 299
9.8. Summary ... 300

10. Measuring Software Systems ... 301
10.1. Measurement Beyond the Component or Project ... 301
10.2. Performance Engineering ... 301
10.3. Measuring SOA-Based Systems ... 309
10.4. Measurement of Agent-Based Systems ... 317
10.5. Hints for the Practitioner ... 327
10.6. Summary ... 328

11. Improving Processes and Products ... 329
11.1. The Need for Process Excellence ... 329
11.2. Objective-rDriven Process Improvement ... 332
11.2.1 Usable and Useful Processes ... 332
11.2.2 Managing Change ... 336
11.2.3 CMMI for Process Improvement ... 342
11.2.4 Setting Improvement Objectives ... 349
11.2.5 Implementing Process Improvement ... 353
11.2.6 Measuring Process Improvement ... 356
11.2.7 Critical Success Factors to Process Improvement ... 358
11.3. Measurement within Process Frameworks ... 359
11.3.1 Measurement Processes in CMMI, SPICE and ISO9000 ... 359

11.3.2 CMMI: Practical Measurements for each Process Area 365
11.4. Productivity, Efficiency and Effectiveness 372
11.5. Quantitative Process Management 392
11.5.1 Process Excellence 392
11.5.2 Techniques for Quantitative Process Management 395
11.5.3 Six Sigma in Software Engineering 403
11.5.4 Case Study: Quantitative Process Management 410
11.5.5 Challenges in Applying SPC to Software 413
11.6. Empirical and Experimental Software Engineering 418
11.7. The Return on Investment from Better Processes 422
11.8. Hints for the Practitioner 428
11.9. lSummary 432

12. Controlling for IT and Software 435
12.1. Managing Software as a Business 435
12.2. The Business Case 436
12.3. The Return on Investment (ROI) 441
12.4. Cost Controlling 443
12.4.1 Cost Controlling in Software Projects 443
12.4.2 Cost Controlling in IT Services 446
12.4.3 Financial Analysis and Business Analysis 450
12.5. Strategic and Operational Management 450
12.5.1 Portfolio Management 450
12.5.2 Technology Management 456
12.5.3 Product and Release Management 458
12.5.4 Distributed Teams and Global Software Engineering 461
12.5.5 Supplier Management 465
12.6. Flints for the Practitioner 468
12.7. Summary 469

13. Measurement Repositories 471
13.1. Access to Measurement Results 471
13.2. Building the Measurement Database 471
13.2.1 Motivation and Requirements 471
13.2.2 Architecture of a Measurement database 474
13.2.3 Details of the Implementation 477
13.3. Benchmarking Based on the ISBSG Repository 478
13.4. Measurement Database Services for COTS Software 481
13.5. Hints for the Practitioner 485
13.6. Summary 486

14. Empirical Laws and Rules of Thumb 487
14.1. Applying Software Rules of Thumb 487
14.2. Project Planning 487
14.3. Global Software Engineering 490
14.4. Requirements Engineering 491

14.5. Quality ... 492
14.6. Software Pareto Laws ... 494
14.7. Productivity and Process Improvement ... 495
14.8. Hints for the Practitioner and Summary ... 497

15. Getting yet more Information ... 499
15.1. Access to Information Beyond this Book ... 499
15.2. Further Reading ... 500
15.3. Measurement Communities ... 500
15.4. Internet Resources ... 505
15.4.1 Internet URLs for Software Measurement ... 505
15.4.2 Internet URLs of Measurement Communities ... 509
15.5. Hints for the Practitioner and Summary ... 510

Glossary ... 511

Literature ... 539

Index ... 555