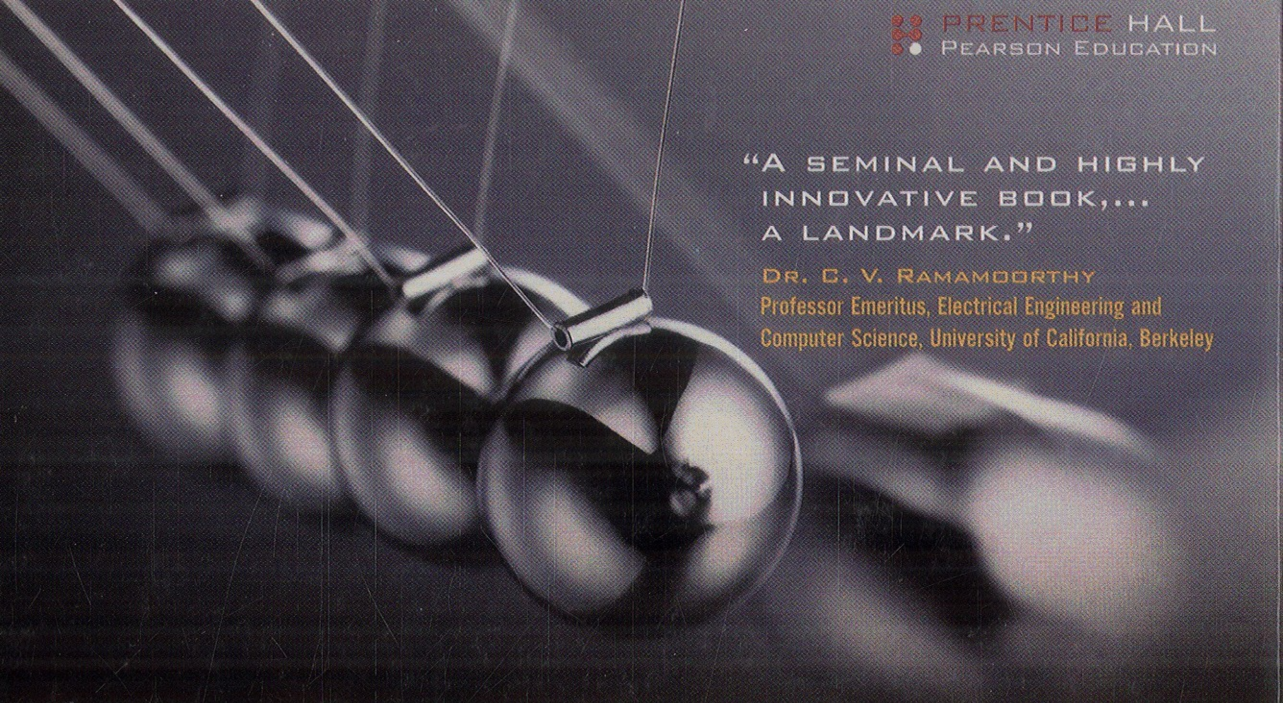
PRENTICE HALL
PEARSON EDUCATION



"A SEMINAL AND HIGHLY INNOVATIVE BOOK,... A LANDMARK."
DR. C. V. RAMAMOORTHY
Professor Emeritus, Electrical Engineering and Computer Science, University of California, Berkeley

# DESIGN FOR TRUSTWORTHY SOFTWARE

## TOOLS, TECHNIQUES, AND METHODOLOGY OF DEVELOPING ROBUST SOFTWARE

BIJAY K. JAYASWAL PETER C. PATTON

# Contents

Foreword xxv

Preface xxvii

Acknowledgments xxxiii

About the Authors xxxv

PART I Contemporary Software Development Process, Their Shortcomings, and the Challenge of Trustworthy Software

CHAPTER 1 Software Development Methodology Today 3

Software Development: The Need for a New Paradigm 5
Sidebar 1.1: Computer Complexity 7
Software Development Strategies and Life-Cycle Models 8
- Build-and-Fix Model 10
- Waterfall Model 11
- Rapid Prototyping Model 12
- Incremental Model 13
- Extreme Programming 14
- Spiral Model 16
- Object-Oriented Programming 17
- Iterative Development or Evolutionary Model 19
- Comparison of Various Life-Cycle Models 20

Software Process Improvement 20
- Rational Unified Process 21
- Capability Maturity Model 22
- ISO 9000-3 Software Development Guidance Standard 23
- Comparison of RUP, CMM, and ISO 9000 26

ADR Method 27
Seven Components of the Robust Software Development Process 27
Robust Software Development Model 29
*Sidebar 1.2: Mission-Critical Aircraft Control Software* 29
Key Points 30
Additional Resources 31
Internet Exercises 31
Review Questions 31
Discussion Questions and Projects 32
Endnotes 32

**CHAPTER 2 The Challenge of Trustworthy Software: Robust Design in Software Context 35**

Software Reliability: Myth and Reality 37
  Similarities and Differences Between Software and Manufactured Products 37
  Comparing Software and Hardware Reliability 39
  Causes of Software Unreliability 41
Limitations of Traditional Quality Control Systems 43
Japanese Quality Management Systems and the Taguchi Approach 43
Sidebar 2.1: The Life and Times of Dr. Genichi Taguchi 43
Sidebar 2.2: Quality Engineering Methodology at a Glance 45
Sidebar 2.3: Taguchi on Taguchi Methods 46
Sidebar 2.4: The Essence of Deming's 14 Points 48
The Nitty-Gritty of Taguchi Methods for Robust Design 51
  The Concept of Signal-to-Noise Ratio 52
  The Concept of Quality Loss Function 53
  The Concept of Robust Design 55
The Challenge of Software Reliability: Design for Trustworthy Software 56
A Robust Software Development Model: DFTS Process in Practice 61
Key Points 63
Additional Resources 65
Internet Exercises 65
Review Questions 66
Discussion Questions and Projects 67
Endnotes 67

CHAPTER 3 Software Quality Metrics 69
Measuring Software Quality 71
Classic Software Quality Metrics 71
Total Quality Management 73
Generic Software Quality Measures 74
Metrics Methodology 74
In-Process Quality Metrics for Software Testing 76
Software Complexity Metrics 77
Software Science 78
Cyclomatic Complexity 79
Function Point Metrics 80
Availability and Customer Satisfaction Metrics 82
Sidebar 3.1: A Software Urban Legend 83
Current Metrics and Models Technology 84
New Metrics for Architectural Design and Assessment 86
Common Architectural Design Problems 87
Pattern Metrics in OOAD 89
Key Points 90
Additional Resources 91
Internet Exercises 91
Review Questions 91
Discussion Questions and Projects 92
Endnotes 92

CHAPTER 4 Financial Perspectives on Trustworthy Software 95
Why DFTS Entails Different Financial Analyses 97
Cost and Quality: Then and Now 98
Cost of Software Quality 102
Benefits of Cost-of-Quality Analysis 102
Cost of Quality Tasks 103
Classification of Cost of Software Quality 105
Establishing a CoSQ Reporting System 109
Payback from Investment in Quality 116
Value of CoSQ Analysis 117
Pitfalls of a CoSQ Program 118
Cost of Software Quality Over the Life Cycle 118

Case Study 4.1: CoSQ at Intents Software 121
CoSQ and Activity-Based Costing 125
ABC in a Software Organization 126
Starting ABC in a Software Organization 127
Benefits of ABC 127
Sidebar 4.1: ABC for Service Industries 128
Quality Loss Function in Software 128
Financial Evaluation of a DFTS Investment 129
Metrics for DFTS Evaluation 130
Establishing a Financial Evaluation Framework for a DFTS Initiative 130
Key Points 132
Additional Resources 134
Internet Exercises 134
Review Questions 134
Discussion Questions 135
Problems 136
Endnotes 138

**CHAPTER 5 Organizational Infrastructure and Leadership for DFTS 139**

Organizational Challenges of a DFTS Deployment 141
DFTS Implementation Framework 141
Step 1: Creating Management Awareness and Buy-in 144
Step 2: Communicating Top Management's Consensus and Commitment 147
Step 3: Recognizing Potential Pitfalls of a DFTS Initiative 147
Sidebar 5.1: Virtuous Teaching Cycle and TPOV 156
Step 4: Laying Foundations for a Quality-Focused Enterprise 157
Step 5: Building the Organizational Infrastructure 160
Step 6: Understanding the Roles of the Key Players 161
Step 7: Designing a Supportive Organizational Structure 170
Step 8: Establishing Effective Communication 172
Step 9: Creating an Appropriate Reward System 174
Step 10: Establishing Cost of Software Quality 175
Step 11: Planning and Launching Organization-Wide Learning 176
Step 12: Implementing the DFTS Model 177
Step 13: Monitoring and Feedback for Learning and Improvement 180
Step 14: Freezing the Improvements and Gains 180
Step 15: Integrating and Expanding the Initiative 181

Putting It All Together 181
Key Points 182
Additional Resources 186
Internet Exercises 186
Review Questions 187
Discussion Questions and Projects 188
Endnotes 189

## PART II Tools and Techniques of Design for Trustworthy Software

**CHAPTER 6 *The Seven Basic (B7) Tools of Quality*** **193**
The Seven Basic (B7) Tools 196
Sidebar 6.1: Kaoru Ishikawa: Developing a SpecificallyJapanese Quality Strategy 198
B7 in a DFTS Context 200
Other DFTS Tools, Techniques, and Methodologies 201
Flowcharts 202
High-Level Flowcharts 204
Detailed Flowcharts 204
Swim Lane Flowcharts 205
Pareto Charts 205
Cause-and-Effect Diagrams 206
Creating Cause-and-Effect-Diagrams to Identify Causes 208
Cause-and-Effect-Diagrams for Process Classification 210
Scatter Diagrams 212
Check Sheets 214
Histograms 215
Determining the Distribution Pattern 216
Determining Whether Specifications Are Satisfied 217
Comparing Data by Stratifying 217
Graphs 218
Control Charts 219
Key Points 222
Additional Resources 223
Review Questions 223
Discussion Questions 224
Endnotes 225

**CHAPTER 7 The 7 MP Tools: Analyzing and Interpreting Qualitative and Verbal Data 227**

The N7 and 7 MP Tools 230
Typical Applications of 7 MP Tools 231
Affinity Diagram 234
Interrelationship Diagraph (I.D.) 238
Tree Diagram 240
Prioritization Matrices 244
Matrix Diagram 244
Process Decision Program Chart (PDPC) 245
Activity Network Diagram 246
Behavioral Skills for 7 MP Tools 247
Key Points 248
Additional Resources 249
Review Questions 249
Discussion Questions and Projects 250
Endnotes 250

**CHAPTER 8 The Analytic Hierarchy Process 253**

Prioritization, Complexity, and the Analytic Hierarchy Process 255
Multiobjective Decision-Making and AHP 256
  Terminology 258
  Structuring an Objectives Hierarchy 258
  Decision Hierarchy 261
Case Study 8.1: MIS Director's IT Dilemma 261
Case Study 8.1 Solution Using Expert Choice 262
  Step 1: Brainstorm and Construct a Hierarchical Model of the Problem 263
  Step 2: Derive Ratio Scale Priorities for the Objectives 264
  Step 3: Derive Priorities for the Alternatives with Respect to Each Objective 267
  Step 4: Synthesis 273
Approximations to AHP with Manual Calculations 276
  Approximate Solution Method 1 276
  Approximate Solution Method 2: Brassard's Full Analytical Criteria Method for Prioritization 284

Conclusion 289
Key Points 289
Additional Resources 290
Internet Exercises 290
Review Questions 290
*Discussion Questions and Projects* 291
*Problems* 292
Problem 1: Managing Complexity in System Conversion 292
Problem 2: Managing Software Complexity in a High-TechStart-up Enterprise 294
Problem 3: Complexity in Patient Record Systems 296
Problem 4: Oil Well Drilling Decision System 297
Problem 5: The ROI Issue 299
Problem 6: An Abstract Complexity Analysis 299
Problem 7: Sensitivity to Complexity 300
Endnotes 300

**CHAPTER 9 Complexity, Mistakes, and Poka Yoke in Software Development Processes 303**

Poka Yoke as a Quality Control System 305
Principles of Poka Yoke 306
Causes of Defects: Variation, Mistakes, and Complexities 307
Situations in Which Poka Yoke Works Well 309
Mistakes as Causes of Defects 310
Controlling Complexity in Software Development 312
Mistakes, Inspection Methods, and Poka Yoke 316
Deploying a Poka Yoke System 317
Identifying a Poka Yoke Solution 321
Key Points 322
Additional Resources 324
Internet Exercises 325
Review Questions 325
Discussion Questions and Projects 326
Endnotes 326

CHAPTER 10 5S for Intelligent Housekeeping in Software Development 329
5S: A Giant Step Toward a Productive Workplace Environment 331
Implementation Phases of the 5S System 332
Phase 1: Sorting/Cleaning Up 332
Phase 2: Straightening/Orderliness 332
Phase 3: Shine/Cleanliness 333
Phase 4: Standardize 333
Phase 5: Sustain/Discipline 333
The 5S System and the DFTS Process 334
*Sidebar 10.1: From 5S to the Lean DFTS Process* 335
Overcoming Resistance 338
Implementing 5S 339
Step 1: Management Buy-in 340
Step 2: Training and Implementation 340
Step 3: Link to a Reward System 340
Step 4: Follow-up and Continuous Improvement 340
Key Points 341
Additional Resources 342
Internet Exercises 342
Review Questions 343
Discussion Questions and Projects 343
Endnotes 344

CHAPTER 11 Understanding Customer Needs: Software QFD and the Voice of the Customer 345
QFD: Origin and Introduction 347
What's Different about QFD as a Quality System? 348
The History of QFD 350
The History of Software QFD 350
So, What Is QFD and Why Do We Need It? 352
A Focus on Priority 354
QFD Defined 355
QFD Deployments 356
The Four-Phase Model of QFD 357
The "House of Quality" Matrix 359

Problems with Traditional QFD Applied to Software 363
- Traditional QFD Failures 363
- "The Matrix Is Too Big" 364
- "It Takes Too Long" 365
- "We Knew That Already" 365

Modern QFD for Software 367
- Blitz QFD 368
- The Seven Management and Planning (7 MP) Tools 368
- Customer Satisfaction and Value 369

The Blitz QFD Process 370
- Step 1: Key Project Goal 371
- Step 2: Key Customer Segment 371
- Step 3: Key Process Steps 372
- Step 4: Go to Gemba 372
- Step 5: What Are the Customer Needs? 374
- Step 6: Structure the Customer Needs 377
- Step 7: Analyze Customer Needs Structure 378
- Step 8: Prioritize the Customer Needs 378
- Step 9: Deploy Prioritized Customer Needs 380
- Downstream Deployments: Analyze (Only) Important Relationships in Detail 382
- The "House of Quality" and Beyond 383
- Six Sigma Projects 385
- Follow-Up: Apply, Evolve, and Improve the Process 385
- *Rapid Development* 385
- Schedule Deployment with Critical Chain Project Management 386

Implementing Software QFD 386
- The People Side of QFD 386
- QFD Challenges and Pitfalls 387
- How to Implement Software QFD 390

Conclusion 391
- Modern QFD in the DFTS Process 391

Key Points 393

Additional Resources 394

Internet Exercises 395

*Review Questions* 396

Discussion Questions 397

Endnotes 399

About the Author 404

CHAPTER 12 Creativity and Innovation in the Software Design Process: TRIZ and Pugh Concept Selection Methodology 405
The Need for Creativity in DFTS 407
Creativity and TRIZ 407
Sidebar 12.1: What Is Serendipity? 408
Sidebar 12.2: Being There When the Page Was Blank 411
TRIZ in Software Development 411
Sidebar 12.3: Lingua Latina Non Mortus Est 412
TRIZ, QFD, and Taguchi Methods 419
Brainstorming 421
Pugh Concept Selection Methodology 423
Software as Intellectual Property 425
Sidebar 12.4: A Picture Is Worth... 427
Key Points 428
Additional Resources 428
Internet Exercises 428
Review Questions 429
Discussion Questions and Projects 429
Endnotes 429

CHAPTER 13 Risk Assessment and Failure Modes and Effects Analysis in Software 431
FMEA: Failure Modes and Effects Analysis 433
Upstream Application of FMEA 437
Software Failure Tree Analysis 440
Software Failure Modes and Their Sources 443
Risk Assignment and Evaluation at Each Stage of DFTS 445
Key Points 446
Additional Resources 447
Internet Exercises 447
Review Questions 447
Discussion Questions and Projects 447
Endnotes 448

**CHAPTER 14 Object and Component Technologies and Other Development Tools** **449**

Major Challenges in Enterprise Business Applications 450

Object-Oriented Analysis, Design, and Programming 451

Sidebar 14.1: The Birth of Object-Oriented Programming 451

Sidebar 14.2: The Power of Java Middleware 458

Component-Based Software Development Technology 459

Extreme Programming for Productivity 462

N-Version Programming for Reliability 463

Advantages of NVP 464

Disadvantages of NVP 465

Modern Programming Environments 465

Trends in Computer Programming Automation 469

Key Points 472

Additional Resources 472

Internet Exercises 472

Review Questions 473

Discussion Questions and Projects 473

Endnotes 473

**PART III DESIGNING FOR TRUSTWORTHY SOFTWARE**

**CHAPTER 15 Quality Measures and Statistical Methods for Trustworthy Software** **479**

Trustworthy Software 481

Microsoft's Trustworthy Computing Initiative 482

Statistical Process Control for Software Development Processes 484

Statistical Methods for Software Architects 491

Key Points 494

Additional Resources 495

Internet Exercises 495

Review Questions 496

Discussion Questions and Projects 496

Problems 496

Endnotes 496

**CHAPTER 16 Robust Software in Context 499**

The Software Specification Process 501
Sidebar 16.1: A Precise Functional Specification 503
What Is Robust Software? 504
Requirements for Software to Be Robust 505
Sidebar 16.2: Getting the End User's Input 506
Specifying Software Robustness 506
Sidebar 16.3: An Example of Parameter Design 508
Key Points 508
Additional Resources 509
Internet Exercises 509
Review Questions 509
Discussion Questions and Projects 509
Problems 510
Endnotes 510

**CHAPTER 17 Taguchi Methods and Optimization for Robust Software 511**

*Taguchi Methods for Robust Software Design* 513
An Example from Engineering Design 517
An Example from Software Design and Development 521
Orthogonal Matrices for Taguchi Parameter Design Experiments 526
Applications to the Design of Trustworthy Software 529
Key Points 529
Additional Resources 530
Internet Exercises 530
Review Questions 530
Discussion Questions 530
Problems 531
Endnotes 531

**CHAPTER 18 Verification, Validation, Testing, and Evaluation for Trustworthiness 533**

*Continuing the Development Cycle* 535
Sidebar 18.1: An Urban Legend About Business Software 536
Verification 537
Case Study 18.1: Taguchi Methods for RTOS Design Verification 537
Validation 541
Case Study 18.2: Taguchi Methods for Software Validation 541
Testing and Evaluation 544
Sidebar 18.2: Testing and Debugging Anomalies 545
Key Points 549
Additional Resources 550
Internet Exercises 550
Review Questions 550
Discussion Questions and Projects 550
Problems 551
Endnotes 551

**CHAPTER 19 Integration, Extension, and Maintenance for Trustworthiness 553**

Completing the Development Cycle 555
Integration 555
Sidebar 19.1: The Supermarine Spitfire 556
Extension 556
Case Study 19.1: Extending the Capability of an Electronic Warfare System 557
Maintenance 558
Case Study 19.2: Field Maintenance of Software Systems 559
Sidebar 19.2: Maintaining Sophisticated Software Functionality Out of Existence 560
Key Points 561
Additional Resources 561
Internet Exercises 562
Review Questions 562
Discussion Questions and Projects 562
Problems 562
Endnotes 563

## PART IV Putting It All Together: Deployment of a DFTS Program

**CHAPTER 20 Organizational Preparedness for DFTS** **567**

Time to Ponder 569

Case Study 20.1: Striving for a Perfect Production Process 569

Case Study 20.2: Institutionalizing Six Sigma at GE 572

Leadership Challenges for Transformational Initiatives 577

Assessing Key Organizational Elements 577

Creating Leadership Commitment 578

Understanding the Leadership Role 579

Assessing Strategic Linkages 580

Ensuring Organization-Wide Participation 580

Understanding the Need for Customer Focus 581

Assessing Current Quality Management Capability 582

Key Points 583

*Additional Resources* 584

Internet Exercises 585

Review Questions 585

Discussion Questions and Prọjects 585

Endnotes 586

**CHAPTER 21 Launching a DFTS Initiative** **587**

DFTS and the PICS Framework 589

Plan 589

Implement 592

Step 11: Launching Organization-Wide Learning 592

Designing Learning Curricula: Customization and Differentiation 593

Training Support Personnel 593

Step 12: Implementing DFTS Technology: Learning and Application Process 595

Control 600

Step 13: Feedback Control Systems 603

Case Study 21.1: GE's Operating System for Continual Learning and Enrichment 606

Project Management 610

Secure 611

Step 14: Freezing the Improvements and Gains 611

Step 15: Integrating and Expanding the Initiative 612

Case Study 21.2: Quality Initiatives and Their Integration at TCS 618
Application in Small Software Firms and e-Cottages 620
What's Next? 620
Key Points 621
Additional Resources 623
Internet Exercises 623
Review Questions 624
Discussion Questions 625
Endnotes 625

## PART V Six Case Studies

**CHAPTER 22 Cost of Software Quality (CoSQ) at Raytheon's Electronic Systems (RES) Group 633**

Introduction 634
RES and Its Improvement Program 634
Cost of Software Quality 635
- RES's CoSQ Model 635
- CoSQ Data Gathering 636

Experiences and Lessons Learned 636
- CoSQ Model Usage Lessons 636
- Using the CoSQ Data to Understand the Impact of Improvement 637
- CoSQ Costs and Benefits 640
- Institutionalization of CoSQ Tracking 641

Case Study Implications 641
Endnotes 642

**CHAPTER 23 Information Technology Portfolio Alignment 643**

Part One—The Challenge 644
- The Five Phases of an Iterative Process 645
- Objectivity, Subjectivity, and Quality 648

Part Two—A New, Rational Approach 649
- Step 1: Design 649
- Step 2: Structuring Complexity—Focusing on Objectives 650
- Step 3: Measurement 651
- Step 4: Synthesis 656
- Step 5: Optimization 657

Risk 660
Extensions 662
Summary 664
Endnote 664

**CHAPTER 24 Defining Customer Needs for Brand-New Products: QFD for Unprecedented Software 665**

Introduction 667
Definition of Value 667
Why Not Ask? 668
Unprecedented Products 669
Defining Brand-New Needs 669
*Methods for Defining Customer Needs* 669
Tools 674
QFD's Seven Management and Planning (7MP) Tools 675
Sidebar 24.1: What Is the Theory of Constraints (TOC)? 676
TOC's Thinking Processes 677
Last Steps 678
Marketing Brand-New Products 678
Layers of Resistance 679
Conclusion 682
Acknowledgments 682
References 682
About the Author 684

**CHAPTER 25 Jurassic QFD: Integrating Service and Product Quality Function Deployment 685**

Company Profile of MD Robotics 686
Why QFD? 687
History of QFD 687
Kano's Requirements 688
Triceratops Encounter at Universal Studios Florida Island of Adventure 690
QFD Template 691
Voice of Customer Analysis 692
Emotion Deployment 696

Body Deployment 697
Engineering Requirements Deployment 700
Summary 702
About the Authors 703
References 704

## CHAPTER 26 Project QFD: Managing Software Development Projects Better with Blitz QFD 707

Introduction 709
Failure 709
Partial Success 710
QFD Defined 710
Starting Right 710
Problems with New Development 710
Incoherent Development Is Inefficient 711
Coherent Development Is Efficient 713
Focus on Value with Project QFD 714
Seven Steps to Better Projects 715
Summary 725
Acknowledgments 726
References 726
About the Author 728

## CHAPTER 27 QFD 2000: Integrating QFD and Other Quality Methods to Improve the New-Product Development Process 729

Demand for New Products 730
Quality and New-Product Development 730
Modern Quality Tools 732
New-Product Development Process 734
Resources for QFD and Other Quality Methods 737
Analytic Hierarchy Process (AHP) and Analytic Network Process (ANP) 737
Balanced Scorecard 737
Blitz QFD 738
Conjoint Analysis 738
Consumer Encounters 738
Customer Integrated Decision Making (CIDM) 738

- de Bono 738
- Deming 738
- Gemba Visit/Voice of Customer Analysis 738
- Hoshin Planning 739
- Kano Model 739
- Kansei Engineering 739
- Lead User Research 739
- Lean Manufacturing 740
- New Lanchester Strategy 740
- Neural Linguistic Programming (NLP) 740
- Project Management 740
- Pugh Concept Selection 740
- QFD (Comprehensive) 740
- Reliability 741
- Seeds to Needs QFD 741
- Seven Management and Planning (7 MP) Tools 741
- Seven Product Planning (7PP) Tools 741
- Seven Quality Control (7QC) Tools 741
- Six Sigma, SPC 742
- Software Engineering 742
- Stage-Gate 742
- Strategic Information Systems (SIS) 742
- Supply Chain Management 742
- Taguchi Methods 742
- Theory of Constraints 742
- Total Quality Management (TQM) 743
- TRIZ 743
- Value Engineering 743

About the Author 743

References 744

**Glossary of Technical Terms 745**

**Name Index 753**

**Index 759**