

Barry W. Boehm

*Director, Software Research and Technology
TRW, Inc.*



SOFTWARE ENGINEERING ECONOMICS

Technische Universität Darmstadt	
FACHBEREICH INFORMATIK	
B I B L I O T H E K	
Inventar-Nr.:	<u>109-00519</u>
Sachgebiete:	_____
Standort:	_____

Prentice-Hall, Inc., Englewood Cliffs, New Jersey 07632



CONTENTS

PREFACE	xix
PART I INTRODUCTION: MOTIVATION AND CONTEXT	1
Chapter 1 Case Study 1: <i>Scientific American</i> Subscription Processing	3
1.1 The Old System	3
1.2 The Programming Solution: Top-Down Stepwise Refinement	4
1.3 The Programming Solution: Results	5
1.4 The Economic-Programming Approach	6

1.5	Results of the Economic-Programming Approach	7
1.6	General Discussion	8
1.7	Questions	8

Chapter 2 Case Study 2: An Urban School Attendance System **10**

2.1	Programming Aspects	10
2.2	Economic Aspects	10
2.3	Human Relations Aspects	11
2.4	Lessons Learned	11
2.5	General Discussion	12
2.6	Questions	13

Chapter 3 The Goals of Software Engineering **14**

3.1	Introduction	14
3.2	Software Engineering: A Definition	16
3.3	Software Trends: Cost	17
3.4	Software Trends: Social Impact	18
3.5	The Plurality of Goals	20
3.6	An Example: Weinberg's Experiment	20
3.7	The Plurality of Software Engineering Means	21
3.8	The Software Engineering Goal Structure	23
3.9	The GOALS Approach to Software Engineering	23
3.10	Questions	26

PART II THE SOFTWARE LIFE-CYCLE: A QUANTITATIVE MODEL **29**

Chapter 4 The Software Life-Cycle: Phases and Activities **35**

4.1	Introduction	35
4.2	The Waterfall Model	35
4.3	Economic Rationale for the Waterfall Model	38
4.4	Refinements of the Waterfall Model	41
4.5	Detailed Life-Cycle Phase Definitions	46

4.6	Detailed Phase/Activity Definitions	46
4.7	The Software Work Breakdown Structure (WBS)	47
4.8	Software Maintenance	54
4.9	Questions	55

Chapter 5 The Basic COCOMO Model 57

5.1	Introduction	57
5.2	Definitions and Assumptions	58
5.3	Development Effort and Schedule	61
5.4	Phase Distribution	64
5.5	Nominal Project Profiles	65
5.6	The Rayleigh Distribution	67
5.7	Interpolation	69
5.8	Basic Software Maintenance Effort Estimation	71
5.9	Questions	71

Chapter 6 The Basic COCOMO Model: Development Modes 74

6.1	Introduction	74
6.2	Basic Effort and Schedule Equations	75
6.3	The Three COCOMO Modes of Software Development	78
6.4	Discussion of the Basic COCOMO Effort and Schedule Equations	83
6.5	Phase Distribution of Effort and Schedule	89
6.6	Questions	94

Chapter 7 The Basic COCOMO Model: Activity Distribution 97

7.1	Introduction	97
7.2	Activity Distribution by Phase	98
7.3	Basic COCOMO Case Study: The Hunt National Bank EFT System	103
7.4	Deriving Basic Project Organization Charts	104
7.5	Discussion of Basic COCOMO Phase and Activity Distributions	110
7.6	Limitations of Basic COCOMO	111
7.7	Questions	111

Chapter 8	The Intermediate COCOMO Model: Product Level Estimates	114
8.1	Introduction	114
8.2	Intermediate COCOMO: Software Development Effort Estimation	117
8.3	A Pricing Example: Microprocessor Communications Software	125
8.4	A Management Example: Reduced Cost-to-Complete	127
8.5	Adjusted Estimate of Annual Maintenance Effort	129
8.6	Example: Microprocessor Communications Software Maintenance	130
8.7	Interpolation and Extrapolation	132
8.8	Estimating the Effects of Adapting Existing Software	133
8.9	Discussion of the Intermediate COCOMO Effort Equations	138
8.10	Questions	141
Chapter 9	Intermediate COCOMO: Component Level Estimation	145
9.1	Introduction	145
9.2	The Component Level Estimating Form (CLEF)	146
9.3	Using the CLEF with Adapted Software	151
9.4	Transaction Processing System (TPS) Example: Basic Development Estimate	153
9.5	TPS Component Level Maintenance Estimate and Phase Distribution	156
9.6	Questions	160
PART III	FUNDAMENTALS OF SOFTWARE ENGINEERING ECONOMICS	165
PART IIIA	COST-EFFECTIVENESS ANALYSIS	169
Chapter 10	Performance Models and Cost-Effectiveness Models	170
10.1	Performance Models	170
10.2	Optimal Performance	173
10.3	Sensitivity Analysis	176
10.4	Cost-Effectiveness Models	178
10.5	Questions	181

Chapter 11 Production Functions: Economies of Scale 187

- 11.1 Example 187
- 11.2 General Discussion: Definitions 188
- 11.3 Discrete Production Functions 189
- 11.4 Basic Production Functions for Software Development 189
- 11.5 Economies and Diseconomies of Scale 189
- 11.6 Diseconomies of Scale on Large Software Projects 190
- 11.7 The Best Way to Combat Diseconomies of Scale 191
- 11.8 Questions 195

Chapter 12 Choosing Among Alternatives: Decision Criteria 197

- 12.1 Example: Minimum Available Budget 197
- 12.2 Minimum Performance Requirement 198
- 12.3 Maximum Effectiveness/Cost Ratio 199
- 12.4 Maximum Effectiveness-Cost Difference 199
- 12.5 Composite Options 201
- 12.6 General Discussion 202
- 12.7 Questions 202

PART IIIB MULTIPLE-GOAL DECISION ANALYSIS 205

Chapter 13 Net Value and Marginal Analysis 207

- 13.1 Example 207
- 13.2 General Discussion: Marginal Analysis 208
- 13.3 Illustration 210
- 13.4 Some Caveats in Dealing with Net Value and Profit 212
- 13.5 Value of Information Processing Products 212
- 13.6 Questions 213

Chapter 14 Present versus Future Expenditure and Income 215

- 14.1 Example: An Oversimplified Cost Analysis 215
- 14.2 Interest Calculations 216

14.3	Present Value Calculations	216
14.4	Present Value of a Series of Cash Flows	217
14.5	Summary of Rental versus Purchase Analysis	218
14.6	General Discussion: Summary of Present Value Concepts and Formulas	219
14.7	Present Value Characteristics	220
14.8	Sensitivity to Interest Rate or Discount Rate	220
14.9	Applications to Software Engineering	221
14.10	Questions	221

Chapter 15 Figures of Merit

223

15.1	Example: Software Package Selection	223
15.2	Net Value Analysis	224
15.3	Figure of Merit Analysis	225
15.4	General Discussion: A Weighted Sum Analysis for Hardware and Software Selection: A Case Study	227
15.5	Case Study: Activity Description	228
15.6	Case Study: Problems with the Evaluation Function	234
15.7	Case Study: Problems with Weights and Ratings	235
15.8	Case Study: Summary	236
15.9	The Delivered System Capability (DSC) Figure of Merit	236
15.10	Properties of the DSC Figure of Merit	237
15.11	The TPS Example Revisited	238
15.12	Comparison of Weighted-Sum and DSC Figures of Merit	239
15.13	Questions	241

Chapter 16 Goals as Constraints

243

16.1	Example: TPS Option A Failure Modes	243
16.2	System Reliability and Availability	243
16.3	Figure of Merit Evaluation	244
16.4	Expressing Goals as Constraints	245
16.5	Goals as Constraints: Feasible Sets and Cost-Value Contours	246
16.6	General Discussion: Decision Problems with Constraints	248
16.7	Software Engineering Applications	250
16.8	Mathematical Optimization Techniques	250
16.9	Capabilities and Limitations of Mathematical Optimization Techniques	255
16.10	Questions	256

Chapter 17	Systems Analysis and Constrained Optimization	258
17.1	Example	258
17.2	General Discussion	262
17.3	Questions	263
Chapter 18	Coping with Unreconcilable and Unquantifiable Goals	265
18.1	Example: TPS Option B: Special-Purpose Operating System Development	266
18.2	In-House versus Vendor Development Considerations	266
18.3	Presentation Methods	267
18.4	General Discussion: Unquantifiable Criteria	269
18.5	Presentation Techniques for Unquantifiable Criteria	270
18.6	Presentation Techniques for Mixed Quantifiable and Unquantifiable Criteria	272
18.7	Some Cautions in Presenting and Interpreting Multivariate Data	275
18.8	Questions	276
PART IIIC	DEALING WITH UNCERTAINTIES, RISK, AND THE VALUE OF INFORMATION	278
Chapter 19	Coping with Uncertainties: Risk Analysis	279
19.1	Example: Operating System Development Options	279
19.2	Decision Rules for Complete Uncertainty	280
19.3	Subjective Probabilities	282
19.4	General Discussion: Decision Rules Under Complete Uncertainty	283
19.5	The Value of Information	283
19.6	Subjective Probabilities	284
19.7	Utility Functions	284
19.8	Software Engineering Implications	285
19.9	Questions	286

Chapter 20	Statistical Decision Theory: The Value of Information	289
20.1	Example: The Prototype Approach	289
20.2	Expected Value of Perfect Information	290
20.3	Working with Imperfect Information	290
20.4	Example	291
20.5	Bayes' Formula	291
20.6	Maximizing the Net Expected Value of the Prototype	293
20.7	General Discussion: Expected Value of Perfect Information	294
20.8	Expected Value of Imperfect Information	295
20.9	The Value-of-Information Procedure	296
20.10	Use of the Value-of-Information Procedure in Software Engineering	297
20.11	Value-of-Information Decision Guidelines	298
20.12	Pitfalls Avoided by Using the Value-of-Information Approach	299
20.13	Value of Information: Wrap-Up	300
20.14	Questions	300
PART IV	THE ART OF SOFTWARE COST ESTIMATION	305
PART IVA	SOFTWARE COST ESTIMATION METHODS AND PROCEDURES	308
Chapter 21	Seven Basic Steps in Software Cost Estimation	309
21.1	Step 1: Establish Objectives	310
21.2	Step 2: Plan for Required Data and Resources	313
21.3	Step 3: Pin Down Software Requirements	315
21.4	Step 4: Work Out as Much Detail as Feasible	316
21.5	Step 5: Use Several Independent Techniques and Sources	323
21.6	Step 6: Compare and Iterate Estimates	323
21.7	Step 7: Followup	326
21.8	Questions	328
Chapter 22	Alternative Software Cost Estimation Methods	329
22.1	Algorithmic Models	330
22.2	Expert Judgment	333

22.3	Estimation by Analogy	336
22.4	Parkinsonian Estimation	336
22.5	Price-to-Win Estimating	337
22.6	Top-Down Estimating	337
22.7	Bottom-Up Estimating	338
22.8	Summary Comparison of Methods	341
22.9	Questions	342

PART IVB THE DETAILED COCOMO MODEL 344

Chapter 23 Detailed COCOMO: Summary and Operational Description 347

23.1	Introduction	347
23.2	The Software Hierarchy Estimating Form (SHEF)	348
23.3	The Software Hierarchy Estimating Form (SHEF) Procedures	351
23.4	Detailed COCOMO Example: Student Job Information System	355
23.5	Schedule Adjustment Calculations	359
23.6	Discussion	361
23.7	Questions	367

Chapter 24 Detailed COCOMO Cost Drivers: Product Attributes 371

24.1	RELY: Required Software Reliability	372
24.2	DATA: Data Base Size	386
24.3	CPLX: Software Product Complexity	390
24.4	Questions	395
24.5	Topics of Research	397

Chapter 25 Detailed COCOMO Cost Drivers: Computer Attributes 400

25.1	TIME: Execution Time Constraint	400
25.2	STOR: Main Storage Constraint	410
25.3	VIRT: Virtual Machine Volatility	413
25.4	TURN: Computer Turnaround Time	415
25.5	Questions	421
25.6	Topics for Further Research	424

Chapter 26	Detailed COCOMO Cost Drivers: Personnel Attributes	426
26.1	ACAP: Analyst Capability	427
26.2	AEXP: Applications Experience	431
26.3	PCAP: Programmer Capability	435
26.4	VEXP: Virtual Machine Experience	439
26.5	LEXP: Programming Language Experience	442
26.6	General Discussion of Personnel Attributes	444
26.7	Questions	448
26.8	Topics for Further Research	449
Chapter 27	Detailed COCOMO Cost Drivers: Project Attributes	451
27.1	MODP: Modern Programming Practices	451
27.2	TOOL: Use of Software Tools	459
27.3	SCED: Schedule Constraint	466
27.4	Questions	472
27.5	Topics for Further Research	473
Chapter 28	Factors Not Included in COCOMO	475
28.1	Type of Application	476
28.2	Language Level	477
28.3	Other Size Measures: Complexity, Entities, Specifications	479
28.4	Requirements Volatility	484
28.5	Personnel Continuity	486
28.6	Management Quality	486
28.7	Customer Interface Quality	488
28.8	Amount of Documentation	488
28.9	Hardware Configuration	489
28.10	Security/Privacy Restrictions	489
28.11	Topics for Further Research	490
Chapter 29	COCOMO Evaluation	492
29.1	Introduction	492
29.2	The COCOMO Project Data Base	494
29.3	COCOMO Estimates versus Actuals: Development Effort	495

29.4	COCOMO Estimates versus Actuals: Development Schedule	495
29.5	COCOMO Estimates versus Actuals: Phase Distribution	504
29.6	COCOMO Estimates versus Actuals: Activity Distribution	507
29.7	Other Software Cost-Estimation Models	510
29.8	COCOMO Evaluation with Respect to Model Criteria	520
29.9	Tailoring COCOMO to a Particular Installation	524
29.10	Topics for Further Research	530

PART IVC SOFTWARE COST ESTIMATION AND LIFE-CYCLE MANAGEMENT **532**

Chapter 30 Software Maintenance Cost Estimation **533**

30.1	Introduction	533
30.2	The COCOMO Software Maintenance Model	534
30.3	Comparison with Project Results	539
30.4	Other Software Maintenance Cost-Estimation Models	540
30.5	Software Maintenance Phenomenology	545
30.6	Software Maintenance Project Data	550
30.7	Questions	553
30.8	Topics for Further Research	554

Chapter 31 Software Life-Cycle Cost Estimation **556**

31.1	Introduction	556
31.2	Software Conversion Cost-Estimating Relationships	557
31.3	Software Conversion Estimates versus Actuals	561
31.4	Software Installation and Training Cost Estimation	565
31.5	Computer Cost Estimation for Software Development	566
31.6	Amount of Software Documentation	571
31.7	Other Software-Related Life-Cycle Costs	576
31.8	An Example Software Life-Cycle Cost-Benefit Analysis	577
31.9	Topics for Further Research	590

Chapter 32 Software Project Planning and Control **591**

32.1	Introduction	591
32.2	A Software Project Planning and Control Framework	594

32.3	Project Scheduling Techniques	597
32.4	Detailed Software Planning and Control: The Unit Development Folder	607
32.5	Monitoring Project Expenditures versus Progress: The Earned Value System	612
32.6	Software Project Planning and Control Example	618
32.7	Building a Software Cost Data Base	637
32.8	Software Planning and Control: Summary Discussion	638
32.9	Questions	638

Chapter 33 Improving Software Productivity

641

33.1	Introduction	641
33.2	Non-Programming Options: Software Packages	647
33.3	Non-Programming Options: Program Generators	654
33.4	Software Productivity Controllables: Product Attributes	658
33.5	Software Productivity Controllables: Computer Attributes	662
33.6	Software Productivity Controllables: Personnel Attributes	666
33.7	Software Productivity Controllables: Project Attributes	676
33.8	Establishing a Software Productivity Improvement Program	682
33.9	Conclusions	688

INDEX

751