# Contents


**Preface** v
**Foreword to the second edition** xiii
**About the accompanying CD-ROM** xiv
**On the bibliography, Internet sources and exercises** xv
**Contents** xvii

**PART A: THE ISSUES** 1

**Chapter 1: Software quality** 3
1.1 EXTERNAL AND INTERNAL FACTORS 3
1.2 A REVIEW OF EXTERNAL FACTORS 4
1.3 ABOUT SOFTWARE MAINTENANCE 17
1.4 KEY CONCEPTS INTRODUCED IN THIS CHAPTER 19
1.5 BIBLIOGRAPHICAL NOTES 19

**Chapter 2: Criteria of object orientation** 21
2.1 ON THE CRITERIA 21
2.2 METHOD AND LANGUAGE 22
2.3 IMPLEMENTATION AND ENVIRONMENT 31
2.4 LIBRARIES 33
2.5 FOR MORE SNEAK PREVIEW 34
2.6 BIBLIOGRAPHICAL NOTES AND OBJECT RESOURCES 34

**PART B: THE ROAD TO OBJECT ORIENTATION** 37

**Chapter 3: Modularity** 39
3.1 FIVE CRITERIA 40
3.2 FIVE RULES 46
3.3 FIVE PRINCIPLES 53
3.4 KEY CONCEPTS INTRODUCED IN THIS CHAPTER 64
3.5 BIBLIOGRAPHICAL NOTES 64
EXERCISES 65

**Chapter 4: Approaches to reusability** 67
4.1 THE GOALS OF REUSABILITY 68
4.2 WHAT SHOULD WE REUSE? 70
4.3 REPETITION IN SOFTWARE DEVELOPMENT 74
4.4 NON-TECHNICAL OBSTACLES 74
4.5 THE TECHNICAL PROBLEM 81
4.6 FIVE REQUIREMENTS ON MODULE STRUCTURES 83
4.7 TRADITIONAL MODULAR STRUCTURES 89
4.8 OVERLOADING AND GENERICITY 93
4.9 KEY CONCEPTS INTRODUCED IN THIS CHAPTER 98
4.10 BIBLIOGRAPHICAL NOTES 99

**Chapter 5: Towards object technology** 101
5.1 THE INGREDIENTS OF COMPUTATION 101
5.2 FUNCTIONAL DECOMPOSITION 103
5.3 OBJECT-BASED DECOMPOSITION 114
5.4 OBJECT-ORIENTED SOFTWARE CONSTRUCTION 116
5.5 ISSUES 117
5.6 KEY CONCEPTS INTRODUCED IN THIS CHAPTER 119
5.7 BIBLIOGRAPHICAL NOTES 119

**Chapter 6: Abstract data types** 121
6.1 CRITERIA 122
6.2 IMPLEMENTATION VARIATIONS 122
6.3 TOWARDS AN ABSTRACT VIEW OF OBJECTS 126
6.4 FORMALIZING THE SPECIFICATION 129
6.5 FROM ABSTRACT DATA TYPES TO CLASSES 142
6.6 BEYOND SOFTWARE 147
6.7 SUPPLEMENTARY TOPICS 148
6.8 KEY CONCEPTS INTRODUCED IN THIS CHAPTER 159
6.9 BIBLIOGRAPHICAL NOTES 160
EXERCISES 161

**PART C: OBJECT-ORIENTED TECHNIQUES** 163

**Chapter 7: The static structure: classes** 165
7.1 OBJECTS ARE NOT THE SUBJECT 165
7.2 AVOIDING THE STANDARD CONFUSION 166
7.3 THE ROLE OF CLASSES 169
7.4 A UNIFORM TYPE SYSTEM 171
7.5 A SIMPLE CLASS 172
7.6 BASIC CONVENTIONS 177

7.7 THE OBJECT-ORIENTED STYLE OF COMPUTATION 181
7.8 SELECTIVE EXPORTS AND INFORMATION HIDING 191
7.9 PUTTING EVERYTHING TOGETHER 194
7.10 DISCUSSION 203
7.11 KEY CONCEPTS INTRODUCED IN THIS CHAPTER 213
7.12 BIBLIOGRAPHICAL NOTES 215
EXERCISES 216

**Chapter 8: The run-time structure: objects 217**

8.1 OBJECTS 218
8.2 OBJECTS AS A MODELING TOOL 228
8.3 MANIPULATING OBJECTS AND REFERENCES 231
8.4 CREATION PROCEDURES 236
8.5 MORE ON REFERENCES 240
8.6 OPERATIONS ON REFERENCES 242
8.7 COMPOSITE OBJECTS AND EXPANDED TYPES 254
8.8 ATTACHMENT: REFERENCE AND VALUE SEMANTICS 261
8.9 DEALING WITH REFERENCES: BENEFITS AND DANGERS 265
8.10 DISCUSSION 270
8.11 KEY CONCEPTS INTRODUCED IN THIS CHAPTER 276
8.12 BIBLIOGRAPHICAL NOTES 277
EXERCISES 277

**Chapter 9: Memory management 279**

9.1 WHAT HAPPENS TO OBJECTS 279
9.2 THE CASUAL APPROACH 291
9.3 RECLAIMING MEMORY: THE ISSUES 293
9.4 PROGRAMMER-CONTROLLED DEALLOCATION 294
9.5 THE COMPONENT-LEVEL APPROACH 297
9.6 AUTOMATIC MEMORY MANAGEMENT 301
9.7 REFERENCE COUNTING 302
9.8 GARBAGE COLLECTION 304
9.9 PRACTICAL ISSUES OF GARBAGE COLLECTION 309
9.10 AN ENVIRONMENT WITH MEMORY MANAGEMENT 312
9.11 KEY CONCEPTS INTRODUCED IN THIS CHAPTER 315
9.12 BIBLIOGRAPHICAL NOTES 315
EXERCISES 316

**Chapter 10: Genericity 317**

10.1 HORIZONTAL AND VERTICAL TYPE GENERALIZATION 317
10.2 THE NEED FOR TYPE PARAMETERIZATION 318
10.3 GENERIC CLASSES 320

10.4 ARRAYS 325
10.5 THE COST OF GENERICITY 328
10.6 DISCUSSION: NOT DONE YET 329
10.7 KEY CONCEPTS INTRODUCED IN THIS CHAPTER 329
10.8 BIBLIOGRAPHICAL NOTES 330
EXERCISES 330

**Chapter 11: Design by Contract: building reliable software 331**

11.1 BASIC RELIABILITY MECHANISMS 332
11.2 ABOUT SOFTWARE CORRECTNESS 333
11.3 EXPRESSING A SPECIFICATION 334
11.4 INTRODUCING ASSERTIONS INTO SOFTWARE TEXTS 337
11.5 PRECONDITIONS AND POSTCONDITIONS 338
11.6 CONTRACTING FOR SOFTWARE RELIABILITY 341
11.7 WORKING WITH ASSERTIONS 348
11.8 CLASS INVARIANTS 363
11.9 WHEN IS A CLASS CORRECT? 369
11.10 THE ADT CONNECTION 373
11.11 AN ASSERTION INSTRUCTION 378
11.12 LOOP INVARIANTS AND VARIANTS 380
11.13 USING ASSERTIONS 389
11.14 DISCUSSION 398
11.15 KEY CONCEPTS INTRODUCED IN THIS CHAPTER 406
11.16 BIBLIOGRAPHICAL NOTES 407
EXERCISES 408
POSTSCRIPT: THE ARIANE 5 FAILURE 410

**Chapter 12: When the contract is broken: exception handling 411**

12.1 BASIC CONCEPTS OF EXCEPTION HANDLING 411
12.2 HANDLING EXCEPTIONS 414
12.3 AN EXCEPTION MECHANISM 419
12.4 EXCEPTION HANDLING EXAMPLES 422
12.5 THE TASK OF A RESCUE CLAUSE 427
12.6 ADVANCED EXCEPTION HANDLING 431
12.7 DISCUSSION 435
12.8 KEY CONCEPTS INTRODUCED IN THIS CHAPTER 437
12.9 BIBLIOGRAPHICAL NOTES 438
EXERCISES 438

**Chapter 13: Supporting mechanisms 439**

13.1 INTERFACING WITH NON-O-O SOFTWARE 439
13.2 ARGUMENT PASSING 444

13.3 INSTRUCTIONS 447
13.4 EXPRESSIONS 452
13.5 STRINGS 456
13.6 INPUT AND OUTPUT 457
13.7 LEXICAL CONVENTIONS 457
13.8 KEY CONCEPTS INTRODUCED IN THIS CHAPTER 458
EXERCISES 458

**Chapter 14: Introduction to inheritance 459**

14.1 POLYGONS AND RECTANGLES 460
14.2 POLYMORPHISM 467
14.3 TYPING FOR INHERITANCE 472
14.4 DYNAMIC BINDING 480
14.5 DEFERRED FEATURES AND CLASSES 482
14.6 REDECLARATION TECHNIQUES 491
14.7 THE MEANING OF INHERITANCE 494
14.8 THE ROLE OF DEFERRED CLASSES 500
14.9 DISCUSSION 507
14.10 KEY CONCEPTS INTRODUCED IN THIS CHAPTER 516
14.11 BIBLIOGRAPHICAL NOTES 517
EXERCISES 517

**Chapter 15: Multiple inheritance 519**

15.1 EXAMPLES OF MULTIPLE INHERITANCE 519
15.2 FEATURE RENAMING 535
15.3 FLATTENING THE STRUCTURE 541
15.4 REPEATED INHERITANCE 543
15.5 DISCUSSION 563
15.6 KEY CONCEPTS INTRODUCED IN THIS CHAPTER 566
15.7 BIBLIOGRAPHICAL NOTES 567
EXERCISES 567

**Chapter 16: Inheritance techniques 569**

16.1 INHERITANCE AND ASSERTIONS 569
16.2 THE GLOBAL INHERITANCE STRUCTURE 580
16.3 FROZEN FEATURES 583
16.4 CONSTRAINED GENERICITY 585
16.5 ASSIGNMENT ATTEMPT 591
16.6 TYPING AND REDECLARATION 595
16.7 ANCHORED DECLARATION 598
16.8 INHERITANCE AND INFORMATION HIDING 605
16.9 KEY CONCEPTS INTRODUCED IN THIS CHAPTER 609

16.10 BIBLIOGRAPHICAL NOTE 610
EXERCISES 610

**Chapter 17: Typing 611**

17.1 THE TYPING PROBLEM 611
17.2 STATIC TYPING: WHY AND HOW 615
17.3 COVARIANCE AND DESCENDANT HIDING 621
17.4 FIRST APPROACHES TO SYSTEM VALIDITY 628
17.5 RELYING ON ANCHORED TYPES 630
17.6 GLOBAL ANALYSIS 633
17.7 BEWARE OF POLYMORPHIC CATCALLS! 636
17.8 AN ASSESSMENT 639
17.9 THE PERFECT FIT 640
17.10 KEY CONCEPTS STUDIED IN THIS CHAPTER 641
17.11 BIBLIOGRAPHICAL NOTES 641

**Chapter 18: Global objects and constants 643**

18.1 CONSTANTS OF BASIC TYPES 643
18.2 USE OF CONSTANTS 645
18.3 CONSTANTS OF CLASS TYPES 646
18.4 APPLICATIONS OF ONCE ROUTINES 648
18.5 CONSTANTS OF STRING TYPE 653
18.6 UNIQUE VALUES 654
18.7 DISCUSSION 656
18.8 KEY CONCEPTS INTRODUCED IN THIS CHAPTER 659
18.9 BIBLIOGRAPHICAL NOTES 660
EXERCISES 660

**PART D: OBJECT-ORIENTED METHODOLOGY: APPLYING THE METHOD WELL 661**

**Chapter 19: On methodology 663**

19.1 SOFTWARE METHODOLOGY: WHY AND WHAT 663
19.2 DEVISING GOOD RULES: ADVICE TO THE ADVISORS 664
19.3 ON USING METAPHORS 671
19.4 THE IMPORTANCE OF BEING HUMBLE 673
19.5 BIBLIOGRAPHICAL NOTES 674
EXERCISES 674

**Chapter 20: Design pattern: multi-panel interactive systems 675**

20.1 MULTI-PANEL SYSTEMS 675
20.2 A SIMPLE-MINDED ATTEMPT 677

20.3 A FUNCTIONAL, TOP-DOWN SOLUTION 678
20.4 A CRITIQUE OF THE SOLUTION 682
20.5 AN OBJECT-ORIENTED ARCHITECTURE 684
20.6 DISCUSSION 693
20.7 BIBLIOGRAPHICAL NOTE 694

**Chapter 21: Inheritance case study: "undo" in an interactive system 695**

21.1 PERSEVERARE DIABOLICUM 695
21.2 FINDING THE ABSTRACTIONS 699
21.3 MULTI-LEVEL UNDO-REDO 704
21.4 IMPLEMENTATION ASPECTS 707
21.5 A USER INTERFACE FOR UNDOING AND REDOING 711
21.6 DISCUSSION 712
21.7 BIBLIOGRAPHICAL NOTES 715
EXERCISES 715

**Chapter 22: How to find the classes 719**

22.1 STUDYING A REQUIREMENTS DOCUMENT 720
22.2 DANGER SIGNALS 726
22.3 GENERAL HEURISTICS FOR FINDING CLASSES 731
22.4 OTHER SOURCES OF CLASSES 735
22.5 REUSE 740
22.6 THE METHOD FOR OBTAINING CLASSES 741
22.7 KEY CONCEPTS INTRODUCED IN THIS CHAPTER 743
22.8 BIBLIOGRAPHICAL NOTES 744

**Chapter 23: Principles of class design 747**

23.1 SIDE EFFECTS IN FUNCTIONS 748
23.2 HOW MANY ARGUMENTS FOR A FEATURE? 764
23.3 CLASS SIZE: THE SHOPPING LIST APPROACH 770
23.4 ACTIVE DATA STRUCTURES 774
23.5 SELECTIVE EXPORTS 796
23.6 DEALING WITH ABNORMAL CASES 797
23.7 CLASS EVOLUTION: THE OBSOLETE CLAUSE 802
23.8 DOCUMENTING A CLASS AND A SYSTEM 803
23.9 KEY CONCEPTS INTRODUCED IN THIS CHAPTER 806
23.10 BIBLIOGRAPHICAL NOTES 806
EXERCISES 807

**Chapter 24: Using inheritance well** 809
24.1 HOW NOT TO USE INHERITANCE 809
24.2 WOULD YOU RATHER BUY OR INHERIT? 812
24.3 AN APPLICATION: THE HANDLE TECHNIQUE 817
24.4 TAXOMANIA 820
24.5 USING INHERITANCE: A TAXONOMY OF TAXONOMY 822
24.6 ONE MECHANISM, OR MORE? 833
24.7 SUBTYPE INHERITANCE AND DESCENDANT HIDING 835
24.8 IMPLEMENTATION INHERITANCE 844
24.9 FACILITY INHERITANCE 847
24.10 MULTIPLE CRITERIA AND VIEW INHERITANCE 851
24.11 HOW TO DEVELOP INHERITANCE STRUCTURES 858
24.12 A SUMMARY VIEW: USING INHERITANCE WELL 862
24.13 KEY CONCEPTS INTRODUCED IN THIS CHAPTER 863
24.14 BIBLIOGRAPHICAL NOTES 863
24.15 APPENDIX: A HISTORY OF TAXONOMY 864
EXERCISES 869

**Chapter 25: Useful techniques** 871
25.1 DESIGN PHILOSOPHY 871
25.2 CLASSES 872
25.3 INHERITANCE TECHNIQUES 873

**Chapter 26: A sense of style** 875
26.1 COSMETICS MATTERS! 875
26.2 CHOOSING THE RIGHT NAMES 879
26.3 USING CONSTANTS 884
26.4 HEADER COMMENTS AND INDEXING CLAUSES 886
26.5 TEXT LAYOUT AND PRESENTATION 891
26.6 FONTS 900
26.7 BIBLIOGRAPHICAL NOTES 901
EXERCISES 902

**Chapter 27: Object-oriented analysis** 903
27.1 THE GOALS OF ANALYSIS 903
27.2 THE CHANGING NATURE OF ANALYSIS 906
27.3 THE CONTRIBUTION OF OBJECT TECHNOLOGY 907
27.4 PROGRAMMING A TV STATION 907
27.5 EXPRESSING THE ANALYSIS: MULTIPLE VIEWS 914
27.6 ANALYSIS METHODS 917
27.7 THE BUSINESS OBJECT NOTATION 919
27.8 BIBLIOGRAPHY 922

**Chapter 28: The software construction process** 923

28.1 CLUSTERS 923
28.2 CONCURRENT ENGINEERING 924
28.3 STEPS AND TASKS 926
28.4 THE CLUSTER MODEL OF THE SOFTWARE LIFECYCLE 926
28.5 GENERALIZATION 928
28.6 SEAMLESSNESS AND REVERSIBILITY 930
28.7 WITH US, EVERYTHING IS THE FACE 933
28.8 KEY CONCEPTS COVERED IN THIS CHAPTER 934
28.9 BIBLIOGRAPHICAL NOTES 934

**Chapter 29: Teaching the method** 935

29.1 INDUSTRIAL TRAINING 935
29.2 INTRODUCTORY COURSES 937
29.3 OTHER COURSES 941
29.4 TOWARDS A NEW SOFTWARE PEDAGOGY 942
29.5 AN OBJECT-ORIENTED PLAN 946
29.6 KEY CONCEPTS STUDIED IN THIS CHAPTER 948
29.7 BIBLIOGRAPHICAL NOTES 948

**PART E: ADVANCED TOPICS** 949

**Chapter 30: Concurrency, distribution, client-server and the Internet** 951

30.1 A SNEAK PREVIEW 951
30.2 THE RISE OF CONCURRENCY 953
30.3 FROM PROCESSES TO OBJECTS 956
30.4 INTRODUCING CONCURRENT EXECUTION 964
30.5 SYNCHRONIZATION ISSUES 977
30.6 ACCESSING SEPARATE OBJECTS 982
30.7 WAIT CONDITIONS 990
30.8 REQUESTING SPECIAL SERVICE 998
30.9 EXAMPLES 1003
30.10 TOWARDS A PROOF RULE 1022
30.11 A SUMMARY OF THE MECHANISM 1025
30.12 DISCUSSION 1028
30.13 KEY CONCEPTS INTRODUCED IN THIS CHAPTER 1032
30.14 BIBLIOGRAPHICAL NOTES 1033
EXERCISES 1035

**Chapter 31: Object persistence and databases** **1037**

31.1 PERSISTENCE FROM THE LANGUAGE 1037
31.2 BEYOND PERSISTENCE CLOSURE 1039
31.3 SCHEMA EVOLUTION 1041
31.4 FROM PERSISTENCE TO DATABASES 1047
31.5 OBJECT-RELATIONAL INTEROPERABILITY 1048
31.6 OBJECT-ORIENTED DATABASE FUNDAMENTALS 1050
31.7 O-O DATABASE SYSTEMS: EXAMPLES 1055
31.8 DISCUSSION: BEYOND O-O DATABASES 1058
31.9 KEY CONCEPTS STUDIED IN THIS CHAPTER 1060
31.10 BIBLIOGRAPHICAL NOTES 1061
EXERCISES 1062

**Chapter 32: Some O-O techniques for graphical interactive applications** **1063**

32.1 NEEDED TOOLS 1064
32.2 PORTABILITY AND PLATFORM ADAPTATION 1066
32.3 GRAPHICAL ABSTRACTIONS 1068
32.4 INTERACTION MECHANISMS 1071
32.5 HANDLING THE EVENTS 1072
32.6 A MATHEMATICAL MODEL 1076
32.7 BIBLIOGRAPHICAL NOTES 1076

**PART F: APPLYING THE METHOD IN VARIOUS LANGUAGES AND ENVIRONMENTS** 1077

**Chapter 33: O-O programming and Ada** **1079**

33.1 A BIT OF CONTEXT 1079
33.2 PACKAGES 1081
33.3 A STACK IMPLEMENTATION 1081
33.4 HIDING THE REPRESENTATION: THE PRIVATE STORY 1085
33.5 EXCEPTIONS 1088
33.6 TASKS 1091
33.7 FROM ADA TO ADA 95 1092
33.8 KEY CONCEPTS INTRODUCED IN THIS CHAPTER 1097
33.9 BIBLIOGRAPHICAL NOTES 1097
EXERCISES 1098

**Chapter 34: Emulating object technology in non-O-O environments** 1099

34.1 LEVELS OF LANGUAGE SUPPORT 1099

34.2 OBJECT-ORIENTED PROGRAMMING IN PASCAL? 1100

34.3 FORTRAN 1102

34.4 OBJECT-ORIENTED PROGRAMMING AND C 1106

34.5 BIBLIOGRAPHICAL NOTES 1112

EXERCISES 1112

**Chapter 35: Simula to Java and beyond: major O-O languages and environments** 1113

35.1 SIMULA 1113

35.2 SMALLTALK 1126

35.3 LISP EXTENSIONS 1130

35.4 C EXTENSIONS 1131

35.5 JAVA 1136

35.6 OTHER O-O LANGUAGES 1137

35.7 BIBLIOGRAPHICAL NOTES 1138

EXERCISES 1139

**PART G: DOING IT RIGHT** 1141

**Chapter 36: An object-oriented environment** 1143

36.1 COMPONENTS 1143

36.2 THE LANGUAGE 1144

36.3 THE COMPILATION TECHNOLOGY 1144

36.4 TOOLS 1148

36.5 LIBRARIES 1150

36.6 INTERFACE MECHANISMS 1152

36.7 BIBLIOGRAPHICAL NOTES 1160

**Epilogue, In Full Frankness Exposing the Language** 1161

**PART H: APPENDICES** 1163

**Appendix A: Extracts from the Base libraries** **1165**

**Appendix B: Genericity versus inheritance** **1167**

B.1 GENERICITY 1168
B.2 INHERITANCE 1173
B.3 EMULATING INHERITANCE WITH GENERICITY 1175
B.4 EMULATING GENERICITY WITH INHERITANCE 1176
B.5 COMBINING GENERICITY AND INHERITANCE 1184
B.6 KEY CONCEPTS INTRODUCED IN THIS APPENDIX 1187
B.7 BIBLIOGRAPHICAL NOTES 1188
EXERCISES 1188

**Appendix C: Principles, rules, precepts and definitions** **1189**

**Appendix D: A glossary of object technology** **1193**

**Appendix E: Bibliography** **1203**

E.1 WORKS BY OTHER AUTHORS 1203
E.2 WORKS BY THE AUTHOR OF THE PRESENT BOOK 1221

**Index** **1225**