

---

# Contents

<b>Preface</b> .....	VII
General .....	VII
Brief Guide to Volume 3 .....	VII
Acknowledgements .....	X

---

## Part I OPENING

---

<b>1 The Triptych Paradigm</b> .....	3
1.1 Delineations of Software Engineering .....	3
1.1.1 “Old” Delineations .....	3
1.1.2 Our View: What Is Software Engineering? .....	6
1.2 The Triptych of Software Engineering .....	7
1.2.1 On Universes of Discourse and Domains .....	7
1.2.2 Domain Engineering .....	9
1.2.3 Requirements Engineering .....	22
1.2.4 Software .....	24
1.2.5 Software Design .....	24
1.2.6 Discussion .....	30
1.3 Phases, Stages and Steps of Development .....	30
1.3.1 Phases of Software Development .....	31
1.3.2 Stages and Steps of Development .....	31
1.3.3 Domain Development .....	33
1.3.4 Requirements Development .....	36
1.3.5 Computing Systems Design .....	38
1.3.6 Discussion: Phases, Stages and Steps .....	40
1.4 The Triptych Process Model — A First View .....	42
1.4.1 The Concept of a Process Model .....	42
1.4.2 The Triptych Process Model .....	42
1.5 Conclusion to Chapter 1 .....	42
1.5.1 Summary .....	43

1.5.2	What Will Be Covered Later? .....	43
1.6	Bibliographical Notes .....	43
1.7	Exercises .....	44
1.7.1	On a Series of Software Developments .....	44
1.7.2	Introductory Remarks .....	49
1.7.3	The Exercises .....	50
<b>2</b>	<b>Documents .....</b>	<b>53</b>
2.1	Documentation Is All! .....	53
2.2	Kinds of Document Parts .....	54
2.2.1	General .....	54
2.2.2	What Is a Description? .....	54
2.3	Deliverables .....	56
2.4	Informative Document Parts .....	57
2.4.1	Name, Place and Date .....	57
2.4.2	Partners .....	57
2.4.3	Current Situation, Needs, Ideas and Concepts .....	59
2.4.4	Scope, Span and Synopsis .....	62
2.4.5	Assumptions and Dependencies .....	64
2.4.6	Implicit/Derivative Goals .....	65
2.4.7	Standards .....	65
2.4.8	Contracts and Design Briefs .....	68
2.4.9	Logbook .....	69
2.4.10	Discussion of Informative Documentation .....	69
2.5	Descriptive Document Parts .....	70
2.5.1	Rough Sketches .....	73
2.5.2	Terminologies .....	75
2.5.3	Narratives .....	78
2.5.4	Formal Descriptions .....	81
2.5.5	Discussion of Descriptive Documentation .....	84
2.6	Analytic Document Parts .....	84
2.6.1	Concept Formation .....	85
2.6.2	Validation .....	86
2.6.3	Verification, Model Checking, Testing .....	86
2.6.4	Theory Formation .....	87
2.6.5	Discussion of Analytic Documentation .....	87
2.7	Discussion .....	88
2.7.1	General .....	88
2.7.2	Summary of Chapter .....	88
2.8	Exercises .....	90
2.8.1	A Preamble .....	90
2.8.2	The Exercises .....	90

---

**Part II CONCEPTUAL FRAMEWORK**


---

<b>3</b>	<b>Methods and Methodology</b>	95
3.1	Method	95
3.2	Methodology	96
3.3	Method Constituents	97
3.3.1	Principle	97
3.3.2	Analysis	97
3.3.3	Construction (or Synthesis)	98
3.3.4	Techniques	98
3.3.5	Tools	98
3.4	Development Principles, Techniques and Tools	99
3.4.1	Some Metaprinciples	99
3.4.2	Some Principles, Techniques and Tools	100
3.5	Discussion	103
3.6	Exercises	104
<b>4</b>	<b>Models and Modelling</b>	105
4.1	Introductory, Context-Setting Remarks	105
4.1.1	Models Versus “Possible Worlds”	105
4.1.2	On Models of a Specification	106
4.1.3	Modelling	106
4.1.4	Universes of Discourse	107
4.2	Model Attributes	107
4.2.1	Analogic, Analytic and Iconic Models	107
4.2.2	Descriptive and Prescriptive Models	111
4.2.3	Extensional and Intensional Models	113
4.3	Roles of Models	115
4.4	The Modelling Principle	116
4.5	Discussion	116
4.6	Exercises	117

---

**Part III DESCRIPTIONS: THEORY AND PRACTICE**


---

<b>5</b>	<b>Phenomena and Concepts</b>	121
5.1	Introduction	121
5.2	Phenomena and Concepts	121
5.2.1	Physically Manifest Phenomena	122
5.2.2	Mentally Conceived Concepts	122
5.2.3	Categories of Phenomena and Concepts	122
5.2.4	Concrete and Abstract Concepts	123
5.2.5	Categories of Descriptions	123
5.2.6	What Is a Description?	124

5.3	Entities .....	125
5.3.1	Atomic Entities .....	125
5.3.2	Composite Entities .....	126
5.3.3	Subentities .....	126
5.3.4	Values, Mereology and Attributes .....	126
5.3.5	Entity Mereology .....	127
5.3.6	Mereologies and Attributes .....	128
5.3.7	Model-Oriented Mereologies .....	128
5.3.8	Model-Oriented Attributes — An Aside .....	128
5.3.9	Entity Properties .....	128
5.3.10	Real Examples and Our Type System .....	129
5.3.11	A Type System .....	135
5.3.12	Type Constraints .....	136
5.3.13	Summary: Principles, Techniques and Tools .....	137
5.4	Functions .....	138
5.4.1	Function Signatures .....	139
5.4.2	Function Definition .....	140
5.4.3	Algorithms .....	142
5.5	Events and Behaviours .....	144
5.5.1	States, Actions, Events and Behaviours .....	144
5.5.2	Synchronisation and Communication .....	145
5.5.3	Processes .....	147
5.5.4	Traces .....	148
5.5.5	Process Definition Languages .....	148
5.6	Choice on Modelling Phenomena and Concepts .....	149
5.6.1	Qualitative Characteristics .....	149
5.6.2	Quantitative Characteristics .....	149
5.6.3	Principles, Techniques and Tools .....	151
5.7	Discussion .....	153
5.7.1	Entities, Functions, Events and Behaviours .....	153
5.7.2	Intensity and Problem Frames .....	153
5.8	Bibliographical Notes .....	154
5.9	Exercises .....	154
5.9.1	A Preamble .....	154
5.9.2	The Exercises .....	154
<b>6</b>	<b>On Defining and on Definitions .....</b>	<b>155</b>
6.1	A Pragmatics of Definitions .....	157
6.1.1	Phenomena, Artifacts and Concepts .....	157
6.1.2	What Are Definitions? .....	158
6.1.3	The Nature of Concepts Being Defined .....	158
6.1.4	Mathematical Definitions .....	159
6.1.5	Physical World Definitions .....	159
6.1.6	Formal Definitions .....	160
6.2	Varieties of Philosophy Definitions .....	160

6.2.1	Six Varietal Characterisations of Art . . . . .	161
6.2.2	Discussion . . . . .	162
6.2.3	A Possible Objection . . . . .	163
6.3	Preliminary Discussion . . . . .	163
6.4	A Syntax of Formal Definitions . . . . .	163
6.4.1	Recognition and Reproduction . . . . .	165
6.4.2	Uniqueness and Identification . . . . .	166
6.4.3	Ontological Terms . . . . .	167
6.5	A Semantics of Formal Definitions . . . . .	167
6.6	Discussion . . . . .	168
6.6.1	General . . . . .	168
6.6.2	Principles, Techniques and Tools . . . . .	168
6.7	Exercises . . . . .	169
<b>7</b>	<b>Jackson's Description Principles . . . . .</b>	<b>173</b>
7.1	Phenomena, Facts and Individuals . . . . .	173
7.2	Designations . . . . .	174
7.2.1	Some Observations . . . . .	175
7.2.2	Formalisation . . . . .	177
7.2.3	Observer Functions and Identification . . . . .	178
7.2.4	Mathematical and Computing Entities . . . . .	179
7.2.5	Discussion: Designations . . . . .	183
7.3	Explicit Definitions . . . . .	184
7.3.1	Definitions: "The Narrow Bridge" . . . . .	184
7.3.2	Definition of Abstract, Intangible Concepts . . . . .	185
7.3.3	How Much, How Little to Define? . . . . .	186
7.3.4	Discussion: Definitions . . . . .	186
7.4	Refutable Assertions . . . . .	187
7.4.1	Designation and Definition Assertions . . . . .	187
7.4.2	Analysis . . . . .	188
7.4.3	"Dangling" Assertions . . . . .	188
7.4.4	Discussion: Refutable Assertions . . . . .	189
7.5	Discussion: Description Principles . . . . .	190
7.6	Bibliographical Notes . . . . .	190
7.7	Exercises . . . . .	190
7.7.1	A Preamble . . . . .	190
7.7.2	The Exercises . . . . .	190

---

**Part IV DOMAIN ENGINEERING**

---

<b>8</b>	<b>Overview of Domain Engineering</b>	193
8.1	Introduction	193
8.2	A Review of <i>Why Domain Engineering?</i>	194
8.3	Overview of Part and Chapter	194
8.4	Domain Stakeholders and Their Perspectives	195
8.5	Domain Acquisition and Validation	196
8.6	Domain Analysis and Concept Formation	196
8.7	Domain Facets	197
8.8	Auxiliary Stages of Domain Development	197
8.9	The Domain Model Document	198
8.9.1	A Preview of Things to Come	198
8.9.2	Contents of a Domain Model Document	198
8.10	Further Structure of This Part	199
8.11	Bibliographical Notes	199
8.12	Exercises	199
<b>9</b>	<b>Domain Stakeholders</b>	201
9.1	Introduction	201
9.2	Stakeholders	201
9.2.1	General Application Stakeholders	202
9.2.2	Software Development Stakeholders	202
9.2.3	Purpose of Listing Stakeholders	203
9.3	Stakeholder Perspectives	203
9.3.1	Perspectives of General Applications	204
9.3.2	Perspectives of Software Development	208
9.4	Discussion: Stakeholders and Their Perspectives	208
9.4.1	General	208
9.4.2	Principles, Techniques and Tools	208
9.5	Exercises	209
9.5.1	Preamble	209
9.5.2	Assignments	209
9.5.3	Postlude	210
<b>10</b>	<b>Domain Attributes</b>	211
10.1	Introduction	211
10.2	Continuity, Discreteness and Chaos	212
10.2.1	Time	212
10.2.2	Continuity	212
10.2.3	Discreteness	214
10.2.4	Chaos	220
10.2.5	Discussion	222
10.3	Statics and Dynamics	222

10.3.1	Static Phenomena and Concepts . . . . .	223
10.3.2	Dynamic Phenomena and Concepts . . . . .	225
10.4	Tangibility and Intangibility . . . . .	241
10.4.1	Humanly Tangible Phenomena . . . . .	241
10.4.2	Otherwise Physically Tangible Phenomena . . . . .	244
10.4.3	Intangible Phenomena . . . . .	245
10.4.4	Discussion . . . . .	245
10.5	One, Two, ..., Dimensionality . . . . .	246
10.5.1	Zero Dimensionality . . . . .	247
10.5.2	One Dimensionality . . . . .	247
10.5.3	Multidimensionality . . . . .	248
10.5.4	Discussion . . . . .	249
10.6	Discussion . . . . .	249
10.7	Bibliographical Notes . . . . .	249
10.8	Exercises . . . . .	250
10.8.1	A Preamble . . . . .	250
10.8.2	The Exercises . . . . .	250
<b>11</b>	<b>Domain Facets . . . . .</b>	<b>251</b>
11.1	Introduction . . . . .	251
11.1.1	Separation of Concerns . . . . .	253
11.1.2	Discussion of the Separation Principle . . . . .	253
11.1.3	Structure of Chapter . . . . .	253
11.2	Domain Facilitators: Business Processes . . . . .	253
11.2.1	Business Processes . . . . .	254
11.2.2	Overall Principles . . . . .	257
11.2.3	Informal and Formal Examples . . . . .	258
11.2.4	Discussion . . . . .	262
11.2.5	Summary . . . . .	263
11.2.6	Reminder . . . . .	263
11.3	Domain Intrinsic . . . . .	264
11.3.1	Overall Principles . . . . .	264
11.3.2	Conceptual Versus Actual Intrinsic . . . . .	268
11.3.3	Methodological Consequences . . . . .	270
11.3.4	Discussion . . . . .	270
11.3.5	Utter Barebones Intrinsic . . . . .	270
11.3.6	Reminder . . . . .	271
11.4	Domain Support Technologies . . . . .	271
11.4.1	Overall Principles . . . . .	272
11.4.2	Methodological Consequences . . . . .	275
11.4.3	Discussion . . . . .	276
11.4.4	Reminder . . . . .	276
11.5	Domain Management and Organisation . . . . .	276
11.5.1	Overall Principles . . . . .	277
11.5.2	A Conceptual Analysis, I . . . . .	278

11.5.3	Methodological Consequences, I+II . . . . .	279
11.5.4	Conceptual Analysis, II . . . . .	279
11.5.5	Methodological Consequences, III . . . . .	281
11.5.6	Discussion . . . . .	282
11.5.7	Reminder . . . . .	282
11.6	Domain Rules and Regulations . . . . .	282
11.6.1	Overall Principles . . . . .	283
11.6.2	Methodological Consequences . . . . .	284
11.6.3	Rules and Regulation Languages . . . . .	286
11.6.4	Principles and Techniques . . . . .	286
11.6.5	Reminder . . . . .	287
11.7	Domain Scripts . . . . .	287
11.7.1	The Description of Scripts . . . . .	287
11.7.2	Methodological Consequences . . . . .	307
11.7.3	Reminder + More . . . . .	308
11.8	Domain Human Behaviour . . . . .	308
11.8.1	Overall Principles . . . . .	309
11.8.2	Methodological Consequences . . . . .	313
11.8.3	Human Behaviour and Knowledge Engineering . . . . .	315
11.8.4	Discussion . . . . .	315
11.8.5	Reminder . . . . .	315
11.9	Other Domain Facets? . . . . .	316
11.10	Composition of Domain Models . . . . .	316
11.10.1	Collating Domain Facet Descriptions . . . . .	316
11.10.2	Technical Issues . . . . .	318
11.11	Exercises . . . . .	318
11.11.1	A Preamble . . . . .	318
11.11.2	The Exercises . . . . .	318
<b>12</b>	<b>Domain Acquisition . . . . .</b>	<b>321</b>
12.1	Introduction . . . . .	321
12.1.1	Domain Facts . . . . .	322
12.1.2	Elicitation of Domain Facts . . . . .	322
12.1.3	Recording Domain Facts . . . . .	322
12.1.4	Indexing Domain Description Sketches . . . . .	323
12.2	The Acquisition Process . . . . .	324
12.2.1	Stakeholder Liaison . . . . .	325
12.2.2	Elicitation Studies . . . . .	325
12.2.3	Elicitation Interviews . . . . .	327
12.2.4	Elicitation Questionnaires . . . . .	327
12.2.5	Elicitation Reports . . . . .	330
12.3	Discussion . . . . .	330
12.3.1	Concept and Process Review . . . . .	330
12.3.2	Process Iteration . . . . .	331
12.3.3	Delineation: Acquisition and Analysis . . . . .	331



12.3.4	Principles, Techniques and Tools . . . . .	331
12.4	Exercises . . . . .	332
12.4.1	A Preamble . . . . .	332
12.4.2	The Exercises . . . . .	332
<b>13</b>	<b>Domain Analysis and Concept Formation . . . . .</b>	<b>333</b>
13.1	Introduction . . . . .	333
13.2	Concept Formation . . . . .	333
13.2.1	Simply Abstracted Concepts . . . . .	334
13.2.2	Breakthrough Abstracted Concepts . . . . .	335
13.3	Consistencies, Conflicts and Completeness . . . . .	336
13.3.1	Inconsistencies . . . . .	337
13.3.2	Conflicts . . . . .	337
13.3.3	Incompleteness . . . . .	337
13.3.4	Looseness and Nondeterminism . . . . .	338
13.4	From Analysis to Synthesis . . . . .	338
13.5	Discussion . . . . .	339
13.5.1	General . . . . .	339
13.5.2	Principles, Techniques and Tools . . . . .	339
13.6	Bibliographical Notes . . . . .	340
13.7	Exercises . . . . .	340
13.7.1	A Preamble . . . . .	340
13.7.2	The Exercises . . . . .	341
<b>14</b>	<b>Domain Verification and Validation . . . . .</b>	<b>343</b>
14.1	Introduction . . . . .	343
14.2	Domain Verification . . . . .	344
14.2.1	Informal Reasoning . . . . .	345
14.2.2	Testing . . . . .	345
14.2.3	Formal Proofs . . . . .	345
14.2.4	Model Checking . . . . .	346
14.3	Domain Validation . . . . .	346
14.3.1	The Domain Validation Documents . . . . .	346
14.3.2	The Domain Validation Process . . . . .	347
14.3.3	Domain Development Iterations . . . . .	347
14.4	Discussion . . . . .	348
14.4.1	General . . . . .	348
14.4.2	Principles, Techniques and Tools . . . . .	348
14.5	Exercises . . . . .	348
14.5.1	A Preamble . . . . .	348
14.5.2	The Exercises . . . . .	349

<b>15</b>	<b>Towards Domain Theories</b>	351
15.1	Introduction	351
15.2	What Is a Domain Theory?	352
15.3	Example Statements of Domain Theories	352
15.4	Possible Domain Theories	354
15.5	How Do We Establish a Theory?	355
15.6	Purpose of a Domain Theory	356
15.7	Summary Principles, Techniques and Tools	356
15.8	Bibliographical Notes	356
15.9	Exercises	357
15.9.1	A Preamble	357
15.9.2	The Exercises	357
<b>16</b>	<b>The Domain Engineering Process Model</b>	359
16.1	Introduction	359
16.2	Review of Domain Development	359
16.3	Review of Domain Documents	361
16.4	Discussion	362
<hr/>		
<b>Part V REQUIREMENTS ENGINEERING</b>		
<hr/>		
<b>17</b>	<b>Overview of Requirements Engineering</b>	365
17.1	Introduction	368
17.1.1	Further Characterisation of ‘Requirement’	369
17.1.2	The “Machine”	369
17.2	Why Requirements, and for What?	370
17.2.1	Why Requirements?	370
17.2.2	Requirements for What?	370
17.2.3	What Does ‘Implements’ Mean?	370
17.3	Getting Started on Requirements Development	371
17.3.1	Initial Informative Documentation	371
17.3.2	Requirements Eureka	373
17.3.3	Pragmatic Prescriptive Documentation	374
17.3.4	Planning Requirements Development	375
17.4	On Domains, Requirements and the Machine	375
17.5	Overview: Requirements Engineering Stages	377
17.6	The Requirements Document	378
17.6.1	A Preview of Things to Come	378
17.6.2	Contents of a Requirements Document	378
17.6.3	Comments on Requirements Documents	379
17.7	The Structure of the Rest of the Part	379
17.8	Bibliographical Notes	380
17.9	Exercises	380
17.9.1	A Preamble	380

17.9.2	The Exercises .....	380
<b>18</b>	<b>Requirements Stakeholders .....</b>	<b>383</b>
18.1	Introduction .....	383
18.2	General Application Stakeholders .....	384
18.3	COTS Software House Stakeholders .....	384
18.3.1	General .....	384
18.3.2	“Corporate Knowledge” .....	385
18.3.3	Classes of Domain-Specific Requirements .....	385
18.3.4	Generic COTS Software Stakeholder Perspective ...	385
18.4	Discussion .....	385
18.4.1	General .....	385
18.4.2	Principles, Techniques and Tools .....	386
18.5	Exercises .....	386
18.5.1	Preamble .....	386
18.5.2	The Exercises .....	386
<b>19</b>	<b>Requirements Facets .....</b>	<b>389</b>
19.1	Introduction .....	390
19.2	Rough Sketching and Terminology .....	390
19.2.1	Initial Requirements Modelling .....	390
19.2.2	Rough-Sketch Requirements .....	391
19.2.3	Requirements Terminology .....	398
19.2.4	Systematic Narration .....	403
19.3	Business Process Reengineering Requirements .....	404
19.3.1	Michael Hammer’s Ideas on BPR .....	404
19.3.2	What Are <i>BPR Requirements</i> ? .....	405
19.3.3	Overview of BPR Operations .....	406
19.3.4	BPR and the Requirements Document .....	406
19.3.5	Intrinsics Review and Replacement .....	407
19.3.6	Support Technology Review and Replacement .....	407
19.3.7	Management and Organisation Reengineering .....	408
19.3.8	Rules and Regulations Reengineering .....	409
19.3.9	Human Behaviour Reengineering .....	409
19.3.10	Script Reengineering .....	410
19.3.11	Discussion: Business Process Reengineering .....	410
19.4	Domain Requirements .....	411
19.4.1	Domain-to-Requirements Operations .....	411
19.4.2	Domain Reqs. and the Reqs. Document .....	412
19.4.3	A Domain Example .....	413
19.4.4	Domain Projection .....	414
19.4.5	Domain Determination .....	419
19.4.6	Domain Instantiation .....	422
19.4.7	Domain Extension .....	423
19.4.8	Domain Requirements Fitting .....	426

19.4.9	Discussion: Domain Requirements . . . . .	429
19.5	Interface Requirements . . . . .	429
19.5.1	Shared Phenomena and Concept Identification . . . . .	430
19.5.2	Interface Requirements Facets . . . . .	430
19.5.3	Interface Reqs. and the Reqs. Document . . . . .	431
19.5.4	Shared Data Initialisation . . . . .	432
19.5.5	Shared Data Refreshment . . . . .	433
19.5.6	Computational Interface Requirements . . . . .	433
19.5.7	Man-Machine Dialogue . . . . .	434
19.5.8	Man-Machine Physiological Interface . . . . .	435
19.5.9	Machine-Machine Dialogue . . . . .	442
19.5.10	Discussion: Interface Requirements . . . . .	443
19.6	Machine Requirements . . . . .	445
19.6.1	Machine Requirements Facets . . . . .	445
19.6.2	Machine Reqs. and the Reqs. Document . . . . .	445
19.6.3	Performance Requirements . . . . .	446
19.6.4	Dependability Requirements . . . . .	448
19.6.5	Fault Tree Analysis . . . . .	454
19.6.6	Maintenance Requirements . . . . .	470
19.6.7	Platform Requirements . . . . .	471
19.6.8	Documentation Requirements . . . . .	473
19.6.9	Discussion: Machine Requirements . . . . .	473
19.7	Composition of Requirements Models . . . . .	474
19.7.1	General . . . . .	474
19.7.2	Collating Requirements Facet Prescriptions . . . . .	474
19.8	Discussion: Requirements Facets . . . . .	474
19.8.1	General . . . . .	474
19.8.2	Principles, Techniques and Tools . . . . .	474
19.9	Bibliographical Notes . . . . .	475
19.10	Exercises . . . . .	475
19.10.1	A Preamble . . . . .	475
19.10.2	The Exercises . . . . .	475
<b>20</b>	<b>Requirements Acquisition . . . . .</b>	<b>479</b>
20.1	Requirements Acquisition Versus Domain Models . . . . .	479
20.2	Domain Model-Based Requirements Acquisition . . . . .	480
20.2.1	Domain Requirements Acquisition, a Preview . . . . .	480
20.2.2	Remaining Requirements Acquisition, a Preview . . . . .	481
20.2.3	Further Issues . . . . .	482
20.3	Overview of Concepts . . . . .	482
20.3.1	Requirements . . . . .	482
20.3.2	Elicitation of Requirements . . . . .	483
20.3.3	Recording Requirements . . . . .	483
20.3.4	Indexing Requirements Prescription Sketches . . . . .	484
20.4	The Acquisition Process . . . . .	485

20.4.1	Stakeholder Liaison . . . . .	486
20.4.2	Elicitation Studies . . . . .	487
20.4.3	Elicitation Interviews . . . . .	487
20.4.4	Elicitation Questionnaires . . . . .	487
20.4.5	Elicitation Reports . . . . .	491
20.5	Discussion . . . . .	491
20.5.1	Concept and Process Review . . . . .	491
20.5.2	Process Iteration . . . . .	492
20.5.3	Delineation: Acquisition and Analysis . . . . .	492
20.5.4	Principles, Techniques and Tools . . . . .	492
20.6	Exercises . . . . .	493
20.6.1	A Preamble . . . . .	493
20.6.2	The Exercises . . . . .	493
<b>21</b>	<b>Requirements Analysis and Concept Formation . . . . .</b>	<b>495</b>
21.1	Introduction . . . . .	495
21.2	Concept Formation . . . . .	497
21.3	Consistencies, Conflicts, and Completeness . . . . .	497
21.3.1	Inconsistencies . . . . .	497
21.3.2	Conflicts . . . . .	498
21.3.3	Incompleteness . . . . .	498
21.3.4	Looseness and Nondeterminism . . . . .	499
21.4	From Analysis to Synthesis . . . . .	499
21.5	Discussion . . . . .	499
21.5.1	General . . . . .	499
21.5.2	Principles, Techniques and Tools . . . . .	499
21.6	Bibliographical Notes . . . . .	500
21.7	Exercises . . . . .	501
21.7.1	A Preamble . . . . .	501
21.7.2	The Exercises . . . . .	501
<b>22</b>	<b>Requirements Verification and Validation . . . . .</b>	<b>503</b>
22.1	Introduction . . . . .	503
22.2	Requirements Verification . . . . .	504
22.2.1	Informal Reasoning . . . . .	505
22.2.2	Testing . . . . .	505
22.2.3	Formal Proofs . . . . .	506
22.2.4	Model Checking . . . . .	506
22.3	Requirements Validation . . . . .	506
22.3.1	The Requirements Validation Documents . . . . .	507
22.3.2	The Requirements Validation Process . . . . .	507
22.3.3	Requirements Development Iterations . . . . .	508
22.4	Discussion . . . . .	508
22.4.1	General . . . . .	508
22.4.2	Principles, Techniques and Tools . . . . .	508

22.5	Bibliographical Notes .....	509
22.6	Exercises .....	509
22.6.1	Preamble .....	509
22.6.2	The Exercises .....	509
<b>23</b>	<b>Requirements Satisfiability and Feasibility .....</b>	<b>511</b>
23.1	Introduction .....	511
23.2	Satisfaction Study .....	512
23.2.1	Correct (Validated) Requirements Document .....	512
23.2.2	Unambiguous Requirements Document .....	512
23.2.3	Complete Requirements Document .....	512
23.2.4	Consistent Requirements Document .....	512
23.2.5	Stable Requirements Document .....	512
23.2.6	Verifiable Requirements Document .....	513
23.2.7	Modifiable Requirements Document .....	513
23.2.8	Traceable Requirements Document .....	513
23.2.9	Faithful Requirements Document .....	513
23.2.10	Discussion of Satisfiability .....	514
23.3	Technical Feasibility Study .....	514
23.3.1	Feasibility of Business Process Reengineering .....	514
23.3.2	Feasibility of Hardware .....	514
23.3.3	Feasibility of Software .....	514
23.3.4	Discussion of Technical Feasibility .....	515
23.4	Economic Feasibility Study .....	515
23.4.1	Feasible Development Costs .....	515
23.4.2	Feasible Write-off Costs .....	515
23.4.3	Gains Outweigh Costs? .....	515
23.4.4	Discussion of Economic Feasibility .....	516
23.5	Compliance with Implicit/Derivative Goals .....	516
23.5.1	Review of Implicit/Derivative Goals .....	516
23.5.2	Discussion of Implicit/Derivative Goals .....	516
23.6	Discussion .....	516
23.6.1	General .....	516
23.6.2	Principles, Techniques and Tools .....	517
23.7	Exercises .....	518
23.7.1	A Preamble .....	518
23.7.2	The Exercises .....	518
<b>24</b>	<b>The Requirements Engineering Process Model .....</b>	<b>521</b>
24.1	Introduction .....	521
24.2	Review of Requirements Development .....	521
24.3	Review of Requirements Documents .....	522
24.4	The Repeat Table of Contents Listing .....	522
24.5	Discussion .....	523

---

**Part VI COMPUTING SYSTEMS DESIGN**


---

<b>25</b>	<b>Hardware/Software Codesign</b> .....	527
25.1	Introduction — On Architecture .....	527
25.2	Hardware Components and Modules .....	528
25.3	Software Components and Modules .....	528
25.4	Hardware/Software Codesign .....	528
25.5	Stepwise Refinement of Architectures .....	529
25.6	Discussion .....	529
25.7	Principles, Techniques and Tools .....	529
<b>26</b>	<b>Software Architecture Design</b> .....	531
26.1	Introduction .....	531
26.2	Initial Domain Requirements Architecture .....	532
26.3	Initial Machine Requirements Architecture .....	534
26.4	Analysis of Some Machine Requirements .....	536
26.4.1	Performance .....	536
26.4.2	Availability .....	536
26.4.3	Accessibility .....	537
26.4.4	Adaptive Maintainability .....	537
26.5	Prioritisation of Design Decisions .....	537
26.6	Corresponding Designs .....	537
26.6.1	Design Decision wrt. Performance .....	538
26.6.2	Design Decision wrt. Availability .....	539
26.6.3	Design Decision wrt. Accessibility .....	540
26.6.4	Design Decision wrt. Adaptability .....	542
26.7	Discussion .....	543
26.7.1	General .....	543
26.7.2	Principles and Techniques .....	544
26.8	Bibliographical Notes .....	545
26.9	Exercises .....	545
26.9.1	A Preamble .....	545
26.9.2	The Exercises .....	545
<b>27</b>	<b>A Case Study in Component Design</b> .....	547
27.1	Overview Introduction .....	547
27.1.1	System Complexity .....	547
27.1.2	Proposed Remedies .....	548
27.1.3	Stepwise Development .....	548
27.1.4	Stagewise Iteration .....	549
27.2	Overview of Example .....	549
27.3	Methodology Overview .....	550
27.3.1	Principles .....	550
27.3.2	Techniques .....	551

27.4	Step 0: Files and Pages	552
27.4.1	A “Snapshot”	552
27.4.2	An Abstract Formal Model	552
27.4.3	Abstract Versus Concrete Basic Actions	553
27.4.4	Concrete Actions	555
27.5	Step 1: Catalogue, Disk and Storage	555
27.5.1	Catalogue Directories	555
27.5.2	Abstraction	558
27.5.3	Actions	559
27.5.4	Adequacy and Sufficiency	561
27.5.5	Correctness	562
27.6	Step 2: Disks	564
27.6.1	Data Refinement	564
27.6.2	Disk Type	564
27.6.3	FS0, FS1 and FS2 Types	564
27.6.4	Disk Type Invariant	565
27.6.5	Disk Type Abstraction	566
27.6.6	Adequacy, Sufficiency, Operations and Correctness	566
27.7	Step 3: Caches	566
27.7.1	Technology Considerations	566
27.7.2	Cached Directory and Page Access	567
27.7.3	Invariance	568
27.7.4	Abstraction	569
27.7.5	Actions	569
27.7.6	Adequacy, Sufficiency and Correctness	571
27.8	Step 4: Storage Crashes	571
27.8.1	Storage and Disk	571
27.8.2	Concrete Semantic Types	572
27.8.3	Invariance	572
27.8.4	Consistent Storage and Disks	572
27.8.5	Abstractions	574
27.8.6	Garbage Collection	574
27.8.7	New Actions	575
27.8.8	Some Previous Commands	575
27.9	Step 5: Flattening Storage and Disks	576
27.9.1	“Flat” Storage and Disk	576
27.9.2	“The Rest”	577
27.10	Step 6: Disk Space Management	577
27.10.1	The Issue	577
27.10.2	“The Rest”	578
27.11	Discussion	579
27.11.1	General	579
27.11.2	Principles and Techniques	579
27.12	Bibliographical Notes	580
27.13	Exercises	580



27.13.1	A Preamble .....	580
27.13.2	The Exercises .....	580
<b>28</b>	<b>Domain-Specific Architectures .....</b>	<b>583</b>
28.1	Introduction .....	583
28.1.1	General .....	583
28.1.2	Some Definitions .....	584
28.1.3	On Architectures .....	584
28.1.4	Problem Frames .....	585
28.1.5	Chapter Structure .....	586
28.2	Translator Architectures .....	586
28.2.1	Translator Domain .....	587
28.2.2	Translator Requirements .....	588
28.2.3	Translator Design .....	588
28.2.4	Process Graph for Translator Development .....	590
28.3	Information Repository Architectures .....	596
28.3.1	Information Repository Domain .....	597
28.3.2	Information Repository Requirements .....	598
28.3.3	Information Repository Design .....	599
28.4	Client/Server Architectures .....	610
28.4.1	Client/Server Domain/Requirements Models .....	610
28.4.2	Some Meta-RSL/CSP Constructs .....	613
28.4.3	Single-Client, Single-Server Model .....	615
28.4.4	Multiple-Client, Single-Server Model .....	619
28.4.5	Client/Server Event Manager Model .....	620
28.4.6	Discussion .....	628
28.5	Workpiece Architectures .....	628
28.5.1	Workpiece Domain .....	629
28.5.2	Workpiece Requirements .....	629
28.5.3	Workpiece Systems Design .....	632
28.6	Reactive System Architectures .....	632
28.6.1	Reactive Systems Domain .....	632
28.6.2	Reactive Systems Control Requirements .....	634
28.6.3	Reactive Systems Control Design .....	635
28.6.4	Discussion of Reactive Systems Design .....	636
28.7	Connection Frame .....	636
28.7.1	Connection Domain .....	637
28.7.2	Connection Requirements .....	638
28.7.3	Connection Systems Design .....	640
28.8	Discussion .....	640
28.8.1	General .....	640
28.8.2	Principles, Techniques and Tools .....	640
28.9	Exercises .....	641
28.9.1	A Preamble .....	641
28.9.2	The Exercises .....	641

<b>29</b>	<b>Etcetera: Coding and All That!</b> .....	645
29.1	From Formal Specification to Programming .....	645
29.1.1	From Specifications to Programs .....	646
29.1.2	From Abstract Types to Data Structures .....	646
29.1.3	From Applicative to Imperative Programs .....	646
29.1.4	Translations into Concurrent Programs .....	647
29.1.5	From RSL to SML, Java, C# and Other Languages ..	647
29.2	The Beauty of Programming .....	647
	Art, Discipline, Craft, Science, Logic and Practice .....	647
29.3	Programming Practices .....	648
29.3.1	Structured Programming .....	648
29.3.2	Extreme Programming .....	648
29.3.3	Object-Oriented cum UML Programming .....	649
29.3.4	Chief Programmer Programming .....	649
29.4	Confidence-Building Software Development .....	650
29.4.1	When to Verify, Model Check and Test .....	650
29.4.2	Demo $\rightarrow$ Skeleton $\rightarrow$ Prototype $\rightarrow$ System .....	652
29.5	Verification, Model Checking and Testing .....	655
29.5.1	Verification .....	656
29.5.2	Model Checking .....	658
29.5.3	Testing .....	658
29.5.4	Discussion .....	661
29.6	Discussion .....	661
29.7	Exercises .....	662
29.7.1	A Preamble .....	662
29.7.2	The Exercises .....	662
<b>30</b>	<b>The Computing Systems Design Process Model</b> .....	663
30.1	Introduction .....	663
30.2	Review of Software Design .....	663
30.2.1	A Process Model .....	664
30.2.2	Discussion .....	665
30.3	Review of Software Design Documents .....	666
30.4	Discussion .....	668

---

## Part VII CLOSING

---

<b>31</b>	<b>The Triptych Development Process Model</b> .....	671
31.1	Phase Process Models .....	671
31.2	Phase Documentation Table of Contents .....	675
31.3	Conclusion .....	678

<b>32</b>	<b>Finale</b>	679
32.1	Informal and Formal Software Engineering	679
32.1.1	Informal Software Engineering	680
32.1.2	Formal Software Engineering	680
32.1.3	Conclusion	680
32.2	Myths and Commandments of Formal Methods	680
32.2.1	First Seven Myths	681
32.2.2	Seven More Myths	682
32.2.3	Ten Formal Methods Commandments	683
32.3	FAQs: Frequently Asked Questions	685
32.3.1	General	685
32.3.2	Domains	686
32.3.3	Requirements	688
32.4	Research and Tool Development	688
32.4.1	Evolving Principles, Techniques and Tools	688
32.4.2	Grand Challenges	689
32.5	Application Areas	691
32.5.1	Additional Areas	691
32.5.2	The Examples	692
32.6	Closing Remarks	693
32.6.1	On Programming, Engineering and Management	693
32.6.2	Current Software Engineering Edifices	694
32.6.3	Current Software Engineering Jargon	694
32.6.4	A New View on Software Engineering	695

---

## Part VIII APPENDIXES

---

<b>A</b>	<b>An RSL Primer</b>	699
A.1	Types	699
A.1.1	Type Expressions	699
A.1.2	Type Definitions	701
A.2	The RSL Predicate Calculus	702
A.2.1	Propositional Expressions	702
A.2.2	Simple Predicate Expressions	702
A.2.3	Quantified Expressions	703
A.3	Concrete RSL Types	703
A.3.1	Set Enumerations	703
A.3.2	Cartesian Enumerations	704
A.3.3	List Enumerations	704
A.3.4	Map Enumerations	705
A.3.5	Set Operations	705
A.3.6	Cartesian Operations	707
A.3.7	List Operations	708
A.3.8	Map Operations	709

A.4	$\lambda$ -Calculus and Functions	711
A.4.1	The $\lambda$ -Calculus Syntax	711
A.4.2	Free and Bound Variables	712
A.4.3	Substitution	712
A.4.4	$\alpha$ -Renaming and $\beta$ -Reduction	712
A.4.5	Function Signatures	713
A.4.6	Function Definitions	713
A.5	Further Applicative Expressions	714
A.5.1	Let Expressions	714
A.5.2	Conditionals	715
A.5.3	Operator/Operand Expressions	716
A.6	Imperative Constructs	716
A.6.1	Variables and Assignment	716
A.6.2	Statement Sequences and <b>skip</b>	716
A.6.3	Imperative Conditionals	717
A.6.4	Iterative Conditionals	717
A.6.5	Iterative Sequencing	717
A.7	Process Constructs	717
A.7.1	Process Channels	717
A.7.2	Process Composition	718
A.7.3	Input/Output Events	718
A.7.4	Process Definitions	718
A.8	Simple RSL Specifications	719
<b>B</b>	<b>Glossary</b>	721
<b>C</b>	<b>Indexes</b>	723
C.1	Concepts Index	724
C.2	Characterisations and Definitions Index	741
C.3	Authors Index	745
	<b>References</b>	749



<http://www.springer.com/978-3-540-21151-8>

Software Engineering 3  
Domains, Requirements, and Software Design  
Bjørner, D.  
2006, XXX, 768 p., Hardcover  
ISBN: 978-3-540-21151-8