
Contents

Preface	v
----------------------	---

Part I Models and Software Development

1 Introduction	3
1.1 The Challenge of Software Development	3
1.2 Mastering Complexity: The Role of Modelling	4
1.3 The Place of Modelling in Software Development	5
1.4 Modelling Structure: Object-oriented Systems	6
1.5 Modelling Data and Functionality: VDM++	10
1.6 Using VDM++ in Object-oriented Design	14
1.7 Book Organisation and Content	15
1.8 Summary	18
2 Building a Model in VDM++: An Overview	19
2.1 Constructing a Model	19
2.2 A Chemical Plant Example	20
2.3 Choosing a Structure Using UML Class Diagrams	21
2.4 Making the Model More Comprehensive	29
2.5 Validating the Model	38
2.6 Generating Code	41
2.7 Summary	41
3 VDM++ Tool Support	43
3.1 Introduction	43
3.2 Getting Hold of the Software	44
3.3 Using Rose	44
3.4 Using VDMTools	47
3.5 Summary	60

Part II Modelling Object-oriented Systems in VDM++

4	Defining Data	63
4.1	Introduction	63
4.2	Object-oriented Structuring	64
4.3	Instance Variables and Invariants	68
4.4	Types and Type Definitions	71
4.5	Values and Value Definitions	83
4.6	Summary	84
5	Defining Functionality	85
5.1	Introduction	85
5.2	Function Definitions	86
5.3	Operation Definitions	91
5.4	Expressions	97
5.5	Pattern Binding	100
5.6	Bindings	101
5.7	Statements	102
5.8	Summary	105
6	Modelling Unordered Collections	107
6.1	Introduction	107
6.2	Overview of Sets	107
6.3	The Robot Controller	111
6.4	Summary	135
7	Modelling Ordered Collections	137
7.1	Introduction	137
7.2	Overview of Sequences	137
7.3	The Congestion Warning System (CWS)	142
7.4	Summary	165
8	Modelling Relationships	167
8.1	Introduction	167
8.2	Overview of Mappings	167
8.3	The Congestion Warning System Revisited	171
8.4	Summary	188

Part III Modelling in Practice: Three Case Studies

9	Model Structuring: The Enigma Cipher	191
9.1	Introduction	191
9.2	The Historic Significance of Enigma	192
9.3	The Enigma Cipher Algorithm	193
9.4	Building the Enigma Model	195
9.5	The VDMUnit Framework	214
9.6	Testing the Enigma Model	224
9.7	Summary	229
10	Combining Views: The CSLaM System	231
10.1	Introduction	231
10.2	The CSLaM System	232
10.3	The On-board CSL System	233
10.4	Member Declarations	234
10.5	The Cab Display	236
10.6	The Beacon Classes	238
10.7	Announcements and Restrictions	240
10.8	Monitoring the Performance of Drivers	241
10.9	Summary	246
11	TradeOne: From Enterprise Architecture to Business Application	249
11.1	Introduction	249
11.2	Enterprise Architectures	251
11.3	The TradeOne Architecture	255
11.4	The Validation Approach	264
11.5	Metrics from TradeOne	265
11.6	Lessons Learned from the TradeOne Project	270
11.7	Summary	271
<hr/>		
Part IV From Models to Code		
<hr/>		
12	Concurrency in VDM++	275
12.1	Introduction	275
12.2	Modelling Concurrency: Threads and Permission Predicates	276
12.3	A POP3 Server	282
12.4	Synchronising Communication in the POP3 Server	291
12.5	Summary	294
13	Model Quality	295
13.1	Introduction	295
13.2	Assessing Internal Consistency: Integrity Checking	296
13.3	Assessing External Consistency Using an API	303
13.4	Validating the POP3 Example through the API	307
13.5	Summary	317

14 Implementing in Java	319
14.1 Introduction	319
14.2 Overview of Java Code Generation	320
14.3 Basic Code Generation	322
14.4 Code Generation and Concurrency	334
14.5 Code Generation of POP3	339
14.6 Summary	343

Appendices

Solutions to Exercises	349
References	383
List of Acronyms	387
Subject Index	389
Definitions Index	399

Validated Designs for Object-oriented Systems

Fitzgerald, J.; Larsen, P.G.; Mukherjee, P.; Plat, N.;
Verhoef, M.

2005, XII, 404 p. 65 illus., Hardcover

ISBN: 978-1-85233-881-7