

Contents

Part I Refining Z Specifications

1	An Introduction to Z	3
1.1	Z: A Language for Specifying Systems	3
1.2	Predicate Logic and Set Theory	4
1.3	Types, Declarations and Abbreviations	7
1.3.1	Types	8
1.3.2	Axiomatic Definitions	9
1.3.3	Abbreviations	10
1.4	Relations, Functions, Sequences and Bags	11
1.4.1	Relations	11
1.4.2	Functions	15
1.4.3	A Pitfall: Undefined Expressions	16
1.4.4	Sequences	17
1.4.5	Bags	19
1.5	Schemas	19
1.5.1	Schema Syntax	20
1.5.2	Schema Inclusion	21
1.5.3	Decorations and Conventions	22
1.5.4	States, Operations and ADTs	23
1.5.5	The Schema Calculus	27
1.5.6	Schemas as Declarations	35
1.5.7	Schemas as Predicates	37
1.5.8	Schemas as Types	38
1.5.9	Schema Equality	39
1.6	Example Refinements	40
1.7	What Does a Z Specification Mean?	45
1.8	The Z Standard	46
1.9	Tool Support	47
1.10	Bibliographical Notes	48
	References	49

2	Simple Refinement	53
2.1	What Is Refinement?	53
2.2	Operation Refinement	56
2.3	From Concrete to Abstract Data Types	61
2.4	Establishing and Imposing Invariants	61
2.4.1	Establishing Invariants	62
2.4.2	Imposing Invariants	63
2.5	Example: London Underground	64
2.6	Bibliographical Notes	67
	References	68
3	Data Refinement and Simulations	69
3.1	Programs and Observations for ADTs	70
3.2	Upward and Downward Simulations	73
3.3	Dealing with (Partial) Relations	80
3.3.1	Partial Relations and Totalisations	81
3.3.2	Soundness and Completeness for Embeddings	88
3.3.3	Partial Relations, Directly	89
3.4	Bibliographical Notes	89
	References	90
4	Refinement in Z	93
4.1	The Relational Basis of a Z Specification	93
4.2	Deriving Downward Simulation in Z	96
4.3	Deriving Upward Simulation in Z	100
4.4	Embedding Inputs and Outputs	102
4.5	Deriving Simulation Rules in Z —Again	105
4.6	Examples	110
4.7	Reflections on the Embedding	120
4.7.1	Alternative Embeddings	120
4.7.2	Programs, Revisited	121
4.8	Proving and Disproving Refinement	122
4.9	Some Pitfalls	123
4.9.1	Name Capture	123
4.9.2	Incompleteness of Behavioural Rules	125
4.9.3	Interaction with Underspecification	125
4.10	Bibliographical Notes	126
	References	126
5	Calculating Refinements	129
5.1	Downward Simulations	132
5.1.1	Non-functional Retrieve Relations	136
5.2	Upward Simulations	141
5.3	Calculating Common Refinements	146
5.4	Bibliographical Notes	149
	References	150

6	Promotion	151
6.1	Example: Multiple Processors	151
6.2	Example: A Football League	153
6.3	Free Promotions and Preconditions	156
6.4	The Refinement of a Promotion	158
6.4.1	Example: Refining Multiple Processors	159
6.4.2	Refinement Conditions for Promotion	161
6.5	Commonly Occurring Promotions	163
6.6	Calculating Refinements	170
6.7	Upward Simulations of Promotions	170
6.8	Bibliographical Notes	171
	References	172
7	Testing and Refinement	173
7.1	Deriving Tests from Specifications	174
7.2	Testing Refinements and Implementations	180
7.2.1	Calculating Concrete Tests	181
7.2.2	Calculating Concrete States	183
7.3	Building the Concrete Finite State Machine	184
7.3.1	Using a Functional Retrieve Relation	184
7.3.2	Using a Non-functional Retrieve Relation	190
7.4	Refinements Due to Upward Simulations	193
7.5	Bibliographical Notes	198
	References	199
8	A Single Simulation Rule	201
8.1	Powersimulations	203
8.2	Application to Z	207
8.3	Calculating Powersimulations	212
8.4	Bibliographical Notes	215
	References	215

Part II Interfaces and Operations: ADTs Viewed in an Environment

9	Refinement, Observation and Modification	219
9.1	Grey Box Data Types	221
9.2	Refinement of Grey Box Data Types	225
9.3	Display Boxes	229
9.4	Bibliographical Notes	235
	References	235
10	IO Refinement	237
10.1	Examples of IO Refinement: “Safe” and “Unsafe”	238
10.2	An Embedding for IO Refinement	241
10.3	Intermezzo: IO Transformers	243
10.4	Deriving IO Refinement	249
10.4.1	Initialisation	249

10.4.2	Finalisation	250
10.4.3	Applicability	251
10.4.4	Correctness	252
10.5	Conditions of IO Refinement	254
10.6	Further Examples	259
10.7	Bibliographical Notes	262
	References	263
11	Weak Refinement	265
11.1	Using Stuttering Steps	267
11.2	Weak Refinement	269
11.2.1	The Relational Context	269
11.2.2	The Schema Calculus Context	271
11.2.3	Further Examples	275
11.3	Properties	282
11.3.1	Weak Refinement Is Not a Pre-congruence	284
11.3.2	Internal Operations with Output	285
11.4	Upward Simulations	286
11.5	Removing Internal Operations	289
11.6	Divergence	294
11.7	Bibliographical Notes	299
	References	300
12	Non-atomic Refinement	303
12.1	Non-atomic Refinement via Stuttering Steps	304
12.1.1	Semantic Considerations	305
12.1.2	Using the Schema Calculus	307
12.2	Non-atomic Refinement Without Stuttering	309
12.3	Using IO Transformations	312
12.3.1	Semantic Considerations Again	314
12.3.2	The Z Characterisation	316
12.4	Further Examples	318
12.5	Varying the Length of the Decomposition	322
12.6	Upward Simulations	326
12.7	Properties	327
12.7.1	No Effect Elsewhere	327
12.7.2	Non-interference	328
12.7.3	Interruptive	331
12.7.4	Using Coupled Simulations	332
12.7.5	Further Non-atomic Refinements	335
12.8	Bibliographical Notes	336
	References	337
13	Case Study: A Digital and Analogue Watch	341
13.1	The Abstract Design	341
13.2	Grey Box Specification and Refinement	343

13.3	An Analogue Watch: Using IO Refinement	345
13.4	Adding Seconds: Weak Refinement	346
13.5	Resetting the Time: Using Non-atomic Refinement	347
	References	351
14	Further Generalisations	353
14.1	Alphabet Extension	353
14.2	Alphabet Translation	355
14.3	Compatibility of Generalisations	356
14.4	Approximate Refinement	357
14.5	Bibliographical Notes	359
	References	359
 Part III Object-Oriented Refinement		
15	An Introduction to Object-Z	363
15.1	Classes	364
15.1.1	The Object-Z Schema Calculus	367
15.1.2	Late Binding of Operations	368
15.1.3	Preconditions	369
15.2	Inheritance	369
15.2.1	Example: A Bank Account	369
15.3	Object Instantiation	372
15.3.1	Modelling Aggregates	373
15.3.2	Example: A Bank	375
15.3.3	Object Containment	376
15.4	Communicating Objects	377
15.5	Semantics	379
15.5.1	Polymorphism and Generic Parameters	380
15.6	Example: A Football League	380
15.7	Bibliographical Notes	382
	References	383
16	Refinement in Object-Z	385
16.1	The Simulation Rules in Object-Z	385
16.2	Weak Refinement in Object-Z	391
16.3	IO and Non-atomic Refinement in Object-Z	393
16.3.1	Non-atomic Refinement	394
16.4	Refinement, Subtyping and Inheritance	399
16.5	Bibliographical Notes	400
	References	401
17	Class Refinement	403
17.1	Objects and References	403
17.1.1	Understanding Object Evolution	406
17.1.2	Verifying the Bank Refinement	407
17.2	Class Simulations	409

17.3	Issues of Compositionality	421
17.3.1	Promoting Refinements Between Object-Z Classes	425
17.4	Bibliographical Notes	428
	References	428
Part IV Modelling State and Behaviour		
18	Combining CSP and Object-Z	431
18.1	An Introduction to CSP	431
18.1.1	The Semantics of CSP	435
18.2	Integrating CSP and Object-Z	438
18.2.1	A Common Semantic Model	439
18.3	Combining CSP Processes and Object-Z Classes	440
18.4	Combining Object-Z Classes Using CSP Operators	444
18.4.1	Semantic Considerations—Models of Operations and Outputs	448
18.5	Bibliographical Notes	451
	References	453
19	Refining CSP and Object-Z Specifications	457
19.1	Refinement in CSP	457
19.2	The Correspondence Between Data Refinement and Refinement in CSP	459
19.2.1	Adding Outputs to the Model	461
19.3	Refinement of CSP Components	463
19.4	Refinement of Object-Z Components	464
19.4.1	Example: The Lift Specification	466
19.5	Structural Refinements	469
19.6	Bibliographical Notes	470
	References	472
20	Conclusions	475
	References	478
Glossary of Notation		481
	Logic	481
	Sets	482
	Numbers	482
	Relations	483
	Functions	483
	Sequences	484
	Bags	484
	Z Notation	484
	Object-Z Notation	485
	References	486
Index		487

Refinement in Z and Object-Z
Foundations and Advanced Applications
Derrick, J.; Boiten, E.A.
2014, XVIII, 492 p., Hardcover
ISBN: 978-1-4471-5354-2