

Contents

Foreword	xiii
Preface	xv
1 Introduction	1
1.1 The need for communication	1
1.1.1 Software and people	1
1.1.2 Structure information to prepare communication	2
1.1.3 Retrieve system information from people	4
1.1.4 Communicate system information to people	5
1.1.5 Support communication	6
1.1.6 Structure teamwork	7
1.2 The FMC Idea	8
1.2.1 Requirements for a modeling technique supporting communication	8
1.2.2 FMC	8
1.2.3 Three aspects – Three diagram types	9
1.3 Outline of this book	10
2 Compositional Structures	13
2.1 An example: The travel agency	13
2.1.1 The scenario	13
2.1.2 A dynamic system	13
2.1.3 An information-processing system	15
2.1.4 A block diagram showing the system	16
2.1.5 A refined model	17
2.2 Modeling the structure of a system	18
2.3 Agents accessing storages	18
2.3.1 Read access	19
2.3.2 Write access	20

2.3.3	Modifying access	21
2.3.4	Concurrent use of storages	22
2.4	Agents communicate via channels	23
2.4.1	Channels	23
2.4.2	Communication scenarios	23
2.4.3	Broadcast communication	25
2.4.4	Using storages for communication	26
2.5	Summary	27
2.6	Exercises	28
3	Dynamic Structures	29
3.1	Petri nets: Basic principles	29
3.2	Conflicts and conditions	33
3.3	Basic patterns	34
3.4	Responsibilities and scope boundaries	36
3.5	Summary	38
3.6	Exercises	40
4	Value Structures and Mind Maps	41
4.1	Entity sets and relationships	42
4.2	Cardinalities	45
4.3	Predicates and roles	46
4.4	Partitions	48
4.5	Reification	49
4.6	Summary	50
4.7	Exercises	52
5	FMC Basics: Summary	53
6	Reinforcing the Concepts	57
6.1	The meta model: A mind map to FMC	57
6.2	Operational versus control state	59
6.2.1	Introduction	59
6.2.2	Problem and solution	60
6.2.3	Criteria of distinction	61
6.2.4	Discrete control loop	69
6.2.5	Summary	71
6.3	Block diagrams: Advanced concepts	72
6.3.1	Nested nodes: Different views versus layout	72
6.3.2	From conceptual to implementation structures	75

6.3.3	Syntax and meaning of refinement	77
6.3.4	Structure variant systems	80
6.4	Petri nets: Advanced concepts	85
6.4.1	Transition-bordered and transition-like sub-nets	85
6.4.2	Safe Petri nets	86
6.4.3	Multitoken places	89
6.4.4	Recursive Petri nets	90
6.4.5	Timing constraints	95
6.5	Non-hierarchical transformations and semantic layers	97
6.5.1	Encoding of values	98
6.5.2	Implementation of channels and storages	100
6.5.3	Relevance of non-hierarchical transformations	101
6.6	Exercises	104
7	Towards Implementation Structures	107
7.1	System structure versus software structure	107
7.2	From Processor to processes	111
7.2.1	Role and processor view	111
7.2.2	Role piggyback	113
7.2.3	Role multiplex, interrupt handling and context switch	116
7.3	Distribution, concurrency and synchronization	120
7.3.1	Physical versus conceptual distribution	120
7.3.2	Concurrency versus distribution	123
7.3.3	Inter-task communication	125
7.3.4	Modeling task and memory management activities	128
7.3.5	Mapping concurrent behavior to tasks	130
7.3.6	Synchronized access to shared data	132
7.3.7	Deriving transactions from FMC models	135
7.4	From FMC to objects and classes	143
7.4.1	Common views	143
7.4.2	Low-level mappings	147
7.4.3	High-level mappings	149
7.4.4	Additional views	153
7.4.5	Guidelines for choosing a view	153
7.5	Conceptual patterns versus software patterns	154
8	Applying FMC in Your Daily Work	157
8.1	Becoming comfortable with FMC	157
8.1.1	Your first steps with FMC	158
8.1.2	The next steps	158

8.1.3	Things to consider	159
8.2	Describing existing systems with FMC	160
8.2.1	How to get information	161
8.2.2	Documenting systems	162
8.2.3	Preparing integration	163
8.2.4	Preparing migration and re-engineering	164
8.3	Using FMC in construction	164
8.3.1	Supporting requirements engineering with ‘Mental Prototyping’	165
8.3.2	Architecture-based design: The System Map	167
8.3.3	Integrate new team members	171
8.3.4	Evolution and migration of systems	172
8.4	Using FMC diagrams to support communication	173
8.4.1	Using FMC in interviews, meetings and discussions	174
8.4.2	Presentations	176
8.4.3	Using FMC in documentation and reports	177
8.4.4	Technical training and teaching	179
8.5	Guidelines for didactical modeling	182
8.5.1	What is the purpose of the models?	182
8.5.2	Who is my audience?	183
8.5.3	Which aspects do I want to show?	183
8.5.4	Which granularity and level of abstraction should I choose?	184
8.6	Cost and benefit of modeling	185
9	Modeling and Visualization Guidelines	189
9.1	Introduction	189
9.2	Increasing the reader’s perception	190
9.2.1	Linestyles	191
9.2.2	Arcs and arc trees	192
9.2.3	Shape and arrangement of nodes	194
9.2.4	Colors and shading	196
9.2.5	Fonts	198
9.3	Increasing comprehension	198
9.3.1	Labeling	199
9.3.2	Checking plausibility	201
9.3.3	Component hierarchies versus semantic layers	204
9.3.4	From naive models to abstractions	206
9.4	Secondary notation, patterns and pitfalls	208

9.4.1	Block diagrams	209
9.4.2	Petri nets	211
9.4.3	E/R diagrams	217
9.4.4	Annotations	217
10	Relationship with Other Modeling Approaches	221
10.1	Comparing FMC with Structured Analysis	221
10.1.1	Background of Structured Analysis	221
10.1.2	Modeling and description concepts	222
10.1.3	Document management and data dictionary	226
10.1.4	Development process	227
10.1.5	Comparing FMC and SA	227
10.2	FMC and the Unified Modeling Language	232
10.2.1	Differences in focus	232
10.2.2	Differences in notation	233
10.2.3	Differences in concept	235
10.2.4	Complementary usage of FMC and UML	236
10.2.5	Model Driven Architecture	243
11	A System of Server Patterns	247
11.1	Application domain	247
11.1.1	The system	247
11.1.2	Requests and sessions	248
11.1.3	Forces	250
11.1.4	Conceptual focus of the pattern language	250
11.2	A pattern language for request processing servers	251
11.2.1	Overview	251
11.2.2	Guideline for choosing an architecture	254
11.2.3	Patterns for request processing servers in the literature	255
11.2.4	The LISTENER / WORKER pattern	255
11.2.5	The FORKING SERVER pattern	258
11.2.6	The WORKER POOL pattern	262
11.2.7	The WORKER POOL MANAGER pattern	265
11.2.8	The JOB QUEUE pattern	268
11.2.9	The LEADER / FOLLOWERS pattern	272
11.2.10	The SESSION CONTEXT MANAGER pattern	275
11.3	Example applications	278
11.3.1	Internet Daemon	278
11.3.2	Apache HTTP server	279
11.3.3	SAP R/3	283

11.3.4 Related applications at object level	284
11.4 Conclusion and further research	284
Epilogue	287
A Solutions	289
B Reference Sheets	299
C Glossary	309
References	323
Index	331