

A Practical Guide to SysML

The Systems Modeling Language

Sanford Friedenthal

Alan Moore

Rick Steiner



AMSTERDAM • BOSTON • HEIDELBERG • LONDON
NEW YORK • OXFORD • PARIS • SAN DIEGO
SAN FRANCISCO • SINGAPORE • SYDNEY • TOKYO

Morgan Kaufmann Publishers is an imprint of Elsevier



Contents

Preface	xvii
Acknowledgments.....	xxi
About the Authors	xxiii

PART I INTRODUCTION

CHAPTER 1 Systems Engineering Overview.....	3
1.1 Motivation for Systems Engineering.....	3
1.2 The Systems Engineering Process	4
1.3 Typical Application of the Systems Engineering Process.....	5
1.4 Multidisciplinary Systems Engineering Team	9
1.5 Codifying Systems Engineering Practice through Standards	10
1.6 Summary	13
1.7 Questions	14
 CHAPTER 2 Model-Based Systems Engineering.....	 15
2.1 Contrasting the Document-Based and Model-Based Approach.....	15
2.1.1 Document-Based Systems Engineering Approach.....	15
2.1.2 Model-Based Systems Engineering Approach.....	16
2.2 Modeling Principles.....	21
2.2.1 Model and MBSE Method Definition	21
2.2.2 The Purpose for Modeling a System.....	21
2.2.3 Establishing Criteria to Meet the Model Purpose.....	22
2.2.4 Model-Based Metrics.....	25
2.2.5 Other Model-Based Metrics	26
2.3 Summary	27
2.4 Questions	27
 CHAPTER 3 Getting Started with SysML	 29
3.1 SysML Purpose and Key Features	29
3.2 SysML Diagram Overview.....	29
3.3 Introducing SysML-Lite	31
3.3.1 SysML-Lite Diagrams and Language Features.....	31
3.3.2 SysML-Lite Air Compressor Example.....	34
3.3.3 SysML Modeling Tool Tips	38
3.4 A Simplified MBSE Method	44
3.5 The Learning Curve for SysML and MBSE.....	47
3.6 Summary	48
3.7 Questions	48

CHAPTER 4	An Automobile Example Using the SysML Basic Feature Set.....	51
4.1	SysML Basic Feature Set	51
4.2	Automobile Example Overview	51
4.2.1	Problem Summary	52
4.3	Automobile Model.....	52
4.3.1	Package Diagram for Organizing the Model	53
4.3.2	Capturing the <i>Automobile Specification</i> in a Requirement Diagram	55
4.3.3	Defining the <i>Vehicle</i> and Its External Environment Using a Block Definition Diagram	57
4.3.4	Use Case Diagram for <i>Operate Vehicle</i>	58
4.3.5	Representing <i>Drive Vehicle</i> Behavior with a Sequence Diagram	60
4.3.6	Referenced Sequence Diagram to <i>Turn On Vehicle</i>	60
4.3.7	<i>Control Power</i> Activity Diagram.....	62
4.3.8	State Machine Diagram for <i>Drive Vehicle States</i>	64
4.3.9	<i>Vehicle Context</i> Using an Internal Block Diagram	64
4.3.10	<i>Vehicle Hierarchy</i> Represented on a Block Definition Diagram.....	67
4.3.11	Activity Diagram for <i>Provide Power</i>	69
4.3.12	Internal Block Diagram for the <i>Power Subsystem</i>	69
4.3.13	Defining the Equations to Analyze Vehicle Performance	73
4.3.14	Analyzing Vehicle Acceleration Using the Parametric Diagram	75
4.3.15	Analysis Results from Analyzing Vehicle Acceleration.....	75
4.3.16	Defining the <i>Vehicle Controller</i> Actions to Optimize Engine Performance	77
4.3.17	Specifying the <i>Vehicle</i> and Its Components.....	78
4.3.18	Requirements Traceability	79
4.3.19	View and Viewpoint.....	81
4.4	Model Interchange.....	82
4.5	Summary	82
4.6	Questions	83

PART II LANGUAGE DESCRIPTION

CHAPTER 5	SysML Language Architecture	87
5.1	The OMG SysML Language Specification.....	87
5.2	The Architecture of the SysML Language	88
5.2.1	The General-Purpose Systems Modeling Domain.....	89
5.2.2	The Modeling Language (or Metamodel)	90
5.2.3	The System Model (or User Model)	91
5.2.4	Model Interchange	92
5.3	SysML Diagrams	93
5.3.1	Diagram Frames.....	94
5.3.2	Diagram Header	95

5.3.3	Diagram Description.....	96
5.3.4	Diagram Content.....	96
5.3.5	Additional Notations.....	99
5.4	The Surveillance System Case Study	100
5.4.1	Case Study Overview.....	100
5.4.2	Modeling Conventions.....	100
5.5	Organization of Part II.....	101
5.5.1	OCSMP Certification Coverage and SysML 1.3	101
5.6	Questions	102

CHAPTER 6 Organizing the Model with Packages..... 103

6.1	Overview	103
6.2	The Package Diagram	104
6.3	Defining Packages Using a Package Diagram.....	104
6.4	Organizing a Package Hierarchy.....	106
6.5	Showing Packageable Elements on a Package Diagram	107
6.6	Packages as Namespaces.....	109
6.7	Importing Model Elements into Packages	109
6.8	Showing Dependencies between Packageable Elements.....	112
6.9	Specifying Views and Viewpoints.....	114
6.10	Summary	115
6.11	Questions.....	116

CHAPTER 7 Modeling Structure with Blocks 119

7.1	Overview	119
7.1.1	Block Definition Diagram	120
7.1.2	Internal Block Diagram	121
7.2	Modeling Blocks on a Block Definition Diagram.....	121
7.3	Modeling the Structure and Characteristics of Blocks Using Properties.....	123
7.3.1	Modeling Block Composition Hierarchies Using Part Properties	123
7.3.2	Modeling Relationships between Blocks Using Reference Properties.....	130
7.3.3	Using Associations to Type Connectors between Parts.....	132
7.3.4	Modeling Quantifiable Characteristics of Blocks Using Value Properties	137
7.4	Modeling Flows	142
7.4.1	Modeling Items That Flow	143
7.4.2	Flow Properties	143
7.4.3	Modeling Flows between Parts on an Internal Block Diagram.....	144
7.5	Modeling Block Behavior	147
7.5.1	Modeling the Main Behavior of a Block	148
7.5.2	Specifying the Behavioral Features of Blocks.....	148
7.5.3	Modeling Block-Defined Methods	150
7.5.4	Routing Requests Across Connectors	151

7.6	Modeling Interfaces Using Ports.....	152
7.6.1	Full Ports.....	153
7.6.2	Proxy Ports.....	154
7.6.3	Connecting Ports.....	157
7.6.4	Modeling Flows between Ports	165
7.6.5	Using Interfaces with Ports	165
7.7	Modeling Classification Hierarchies Using Generalization.....	167
7.7.1	Classification and the Structural Features of a Block.....	169
7.7.2	Classification and Behavioral Features	170
7.7.3	Modeling Overlapping Classifications Using Generalization Sets	171
7.7.4	Modeling Variants Using Classification	172
7.7.5	Using Property-Specific Types to Model Context-Specific Block Characteristics	173
7.7.6	Modeling Block Configurations as Specialized Blocks	173
7.8	Modeling Block Configurations Using Instances	176
7.9	Deprecated Features	178
7.9.1	Flow Ports	179
7.10	Summary	180
7.11	Questions.....	182

CHAPTER 8 Modeling Constraints with Parametrics..... 185

8.1	Overview	185
8.1.1	Defining Constraints Using the Block Definition Diagram	185
8.1.2	The Parametric Diagram.....	186
8.2	Using Constraint Expressions to Represent System Constraints.....	187
8.3	Encapsulating Constraints in Constraint Blocks to Enable Reuse	188
8.3.1	Additional Parameter Characteristics	188
8.4	Using Composition to Build Complex Constraint Blocks.....	190
8.5	Using a Parametric Diagram to Bind Parameters of Constraint Blocks	191
8.6	Constraining Value Properties of a Block.....	193
8.7	Capturing Values in Block Configurations	195
8.8	Constraining Time-Dependent Properties to Facilitate Time-Based Analysis.....	195
8.9	Using Constraint Blocks to Constrain Item Flows	197
8.10	Describing an Analysis Context.....	198
8.11	Modeling Evaluation of Alternatives and Trade Studies.....	200
8.12	Summary	202
8.13	Questions.....	203

CHAPTER 9 Modeling Flow-Based Behavior with Activities 205

9.1	Overview	205
9.2	The Activity Diagram.....	206
9.3	Actions—The Foundation of Activities	208
9.4	The Basics of Modeling Activities.....	209

9.4.1	Specifying Input and Output Parameters for an Activity	209
9.4.2	Composing Activities Using Call Behavior Actions	211
9.5	Using Object Flows to Describe the Flow of Items between Actions	213
9.5.1	Routing Object Flows	213
9.5.2	Routing Object Flows from Parameter Sets	216
9.5.3	Buffers and Data Stores	219
9.6	Using Control Flows to Specify the Order of Action Execution	220
9.6.1	Depicting Control Logic with Control Nodes	220
9.6.2	Using Control Operators to Enable and Disable Actions	222
9.7	Handling Signals and Other Events	224
9.8	Structuring Activities	225
9.8.1	Interruptible Regions	225
9.8.2	Using Structured Activity Nodes	226
9.9	Advanced Flow Modeling	228
9.9.1	Modeling Flow Rates	228
9.9.2	Modeling Flow Order	229
9.9.3	Modeling Probabilistic Flow	230
9.10	Modeling Constraints on Activity Execution	231
9.10.1	Modeling Pre- and Post-conditions and Input and Output States	231
9.10.2	Adding Timing Constraints to Actions	233
9.11	Relating Activities to Blocks and Other Behaviors	234
9.11.1	Linking Behavior to Structure Using Partitions	234
9.11.2	Specifying an Activity in a Block Context	236
9.11.3	Relationship between Activities and Other Behaviors	239
9.12	Modeling Activity Hierarchies Using Block Definition Diagrams	240
9.12.1	Modeling Activity Invocation Using Composite Associations	240
9.12.2	Modeling Parameter and Other Object Nodes Using Associations	240
9.12.3	Adding Parametric Constraints to Activities	242
9.13	Enhanced Functional Flow Block Diagram	243
9.14	Executing Activities	243
9.14.1	The Foundational UML Subset (fUML)	244
9.14.2	The Action Language for Foundational UML (Alf)	245
9.14.3	Primitive Actions	246
9.14.4	Executing Continuous Activities	247
9.15	Summary	248
9.16	Questions	249

CHAPTER 10	Modeling Message-Based Behavior with Interactions	251
10.1	Overview	251
10.2	The Sequence Diagram	252
10.3	The Context for Interactions	252
10.4	Using Lifelines to Represent Participants in an Interaction	254
10.4.1	Occurrence Specifications	255

10.5	Exchanging Messages between Lifelines	256
10.5.1	Synchronous and Asynchronous Messages	256
10.5.2	Lost and Found Messages	258
10.5.3	Weak Sequencing	259
10.5.4	Executions	259
10.5.5	Lifeline Creation and Destruction	261
10.6	Representing Time on a Sequence Diagram	261
10.7	Describing Complex Scenarios Using Combined Fragments	264
10.7.1	Basic Interaction Operators.....	265
10.7.2	Additional Interaction Operators	266
10.7.3	State Invariants	268
10.8	Using Interaction References to Structure Complex Interactions	270
10.9	Decomposing Lifelines to Represent Internal Behavior	270
10.10	Summary	273
10.11	Questions	274

CHAPTER 11 Modeling Event-Based Behavior with State Machines 277

11.1	Overview	277
11.2	State Machine Diagram.....	278
11.3	Specifying States in a State Machine	278
11.3.1	Region.....	278
11.3.2	State	280
11.4	Transitioning between States	281
11.4.1	Transition Fundamentals	281
11.4.2	Routing Transitions Using Pseudostates.....	284
11.4.3	Showing Transitions Graphically.....	287
11.5	State Machines and Operation Calls.....	287
11.6	State Hierarchies	288
11.6.1	Composite State with a Single Region	289
11.6.2	Composite State with Multiple (Orthogonal) Regions.....	290
11.6.3	Transition Firing Order in Nested State Hierarchies.....	292
11.6.4	Using the History Pseudostate to Return to a Previously Interrupted State	293
11.6.5	Reusing State Machines	295
11.7	Contrasting Discrete and Continuous States	297
11.8	Summary	299
11.9	Questions	300

CHAPTER 12 Modeling Functionality with Use Cases 303

12.1	Overview	303
12.2	Use Case Diagram.....	303
12.3	Using Actors to Represent the Users of a System	304
12.3.1	Further Descriptions of Actors.....	305

12.4	Using Use Cases to Describe System Functionality	305
12.4.1	Use Case Relationships	307
12.4.2	Use Case Descriptions	309
12.5	Elaborating Use Cases with Behaviors	310
12.5.1	Context Diagrams	310
12.5.2	Sequence Diagrams	310
12.5.3	Activity Diagrams	311
12.5.4	State Machine Diagrams	313
12.6	Summary	314
12.7	Questions	315

CHAPTER 13 Modeling Text-Based Requirements and Their Relationship to Design

317

13.1	Overview	317
13.2	Requirement Diagram	318
13.3	Representing a Text Requirement in the Model	320
13.4	Types of Requirements Relationships	322
13.5	Representing Cross-Cutting Relationships in SysML Diagrams	322
13.5.1	Depicting Requirements Relationships Directly	323
13.5.2	Depicting Requirements Relationships Using Compartment Notation	324
13.5.3	Depicting Requirements Relationships Using Callout Notation	324
13.6	Depicting Rationale for Requirements Relationships	325
13.7	Depicting Requirements and Their Relationships in Tables	326
13.7.1	Depicting Requirement Relationships in Tables	326
13.7.2	Depicting Requirement Relationships as Matrices	327
13.8	Modeling Requirement Hierarchies in Packages	328
13.9	Modeling a Requirements Containment Hierarchy	328
13.9.1	The Browser View of a Containment Hierarchy	329
13.10	Modeling Requirement Derivation	329
13.11	Asserting That a Requirement is Satisfied	331
13.12	Verifying That a Requirement is Satisfied	332
13.13	Reducing Requirements Ambiguity Using the Refine Relationship	335
13.14	Using the General-Purpose Trace Relationship	338
13.15	Reusing Requirements with the Copy Relationship	338
13.16	Summary	339
13.17	Questions	340

CHAPTER 14 Modeling Cross-Cutting Relationships with Allocations.....

343

14.1	Overview	343
14.2	Allocation Relationship	343
14.3	Allocation Notation	345
14.4	Types of Allocation	347

14.4.1	Allocation of Requirements	347
14.4.2	Allocation of Behavior or Function	347
14.4.3	Allocation of Flow	348
14.4.4	Allocation of Structure.....	348
14.4.5	Allocation of Properties	348
14.4.6	Summary of Relationships Associated with the Term “Allocation”	349
14.5	Planning for Reuse: Specifying Definition and Usage in Allocation	349
14.5.1	Allocating Usage	350
14.5.2	Allocating Definition.....	351
14.5.3	Allocating Asymmetrically	351
14.5.4	Guidelines for Allocating Definition and Usage	351
14.6	Allocating Behavior to Structure Using Functional Allocation.....	352
14.6.1	Modeling Functional Allocation of Usage	354
14.6.2	Modeling Functional Allocation of Definition	354
14.6.3	Modeling Functional Allocation Using Allocate Activity Partitions (Allocate Swimlanes)	357
14.7	Connecting Functional Flow with Structural Flow Using Functional Flow Allocation.....	358
14.7.1	Options for Functionally Allocating Flow.....	358
14.7.2	Allocating an Object Flow to a Connector.....	358
14.7.3	Allocating Object Flow to Item Flow	359
14.8	Modeling Allocation between Independent Structural Hierarchies.....	361
14.8.1	Modeling Structural Allocation of Usage.....	362
14.8.2	Allocating a Logical Connector to a Physical Structure	362
14.8.3	Modeling Structural Allocation of Definition	363
14.9	Modeling Structural Flow Allocation.....	364
14.10	Evaluating Allocation across a User Model.....	366
14.10.1	Establishing Balance and Consistency	366
14.11	Taking Allocation to the Next Step	366
14.12	Summary	367
14.13	Questions	367

CHAPTER 15 Customizing SysML for Specific Domains..... 369

15.1	Overview	369
15.1.1	A Brief Review of Metamodeling Concepts	370
15.2	Defining Model Libraries to Provide Reusable Constructs.....	373
15.3	Defining Stereotypes to Extend Existing SysML Concepts.....	374
15.3.1	Adding Properties and Constraints to Stereotypes	376
15.4	Extending the SysML Language Using Profiles	379
15.4.1	Referencing a Metamodel or Metaclass from a Profile	380
15.5	Applying Profiles to User Models in Order to Use Stereotypes.....	381
15.6	Applying Stereotypes when Building a Model	382
15.6.1	Specializing Model Elements with Applied Stereotypes	384

15.7	Summary	388
15.8	Questions	389

PART III MODELING EXAMPLES

CHAPTER 16	Water Distiller Example Using Functional Analysis	393
16.1	Stating the Problem – The Need for Clean Drinking Water.....	393
16.2	Defining the Model-Based Systems Engineering Approach.....	394
16.3	Organizing the Model	394
16.4	Establishing Requirements.....	396
16.4.1	Characterizing Stakeholder Needs.....	396
16.4.2	Characterizing System Requirements.....	399
16.4.3	Characterizing Required Behaviors	400
16.4.4	Refining Behavior.....	406
16.5	Modeling Structure.....	409
16.5.1	Defining Distiller's Blocks in the Block Definition Diagram.....	409
16.5.2	Allocating Behavior	412
16.5.3	Defining the Ports on the Blocks	414
16.5.4	Creating the Internal Block Diagram with Parts, Ports, Connectors, and Item Flows.....	414
16.5.5	Allocation of Flow	417
16.6	Analyze Performance	417
16.6.1	Item Flow Heat Balance Analysis	417
16.6.2	Resolving Heat Balance	420
16.7	Modify the Original Design.....	420
16.7.1	Updating Behavior	420
16.7.2	Updating Allocation and Structure	421
16.7.3	Controlling the Distiller and the User Interaction.....	425
16.7.4	Developing a User Interface and a Controller.....	426
16.7.5	Startup and Shutdown Considerations	427
16.8	Summary	429
16.9	Questions	429

CHAPTER 17	Residential Security System Example Using the Object- Oriented Systems Engineering Method	431
17.1	Method Overview.....	431
17.1.1	Motivation and Background.....	431
17.1.2	System Development Process Overview	432
17.1.3	OOSEM System Specification and Design Process	435
17.2	Residential Security Example Overview	437
17.2.1	Problem Background.....	437
17.2.2	Project Startup	437

17.3	Applying OOSEM to Specify and Design the Residential Security System.....	438
17.3.1	Setup Model	439
17.3.2	Analyze Stakeholder Needs	444
17.3.3	Analyze System Requirements	453
17.3.4	Define Logical Architecture	465
17.3.5	Synthesize Candidate Physical Architectures.....	472
17.3.6	Optimize and Evaluate Alternatives	501
17.3.7	Manage Requirements Traceability	507
17.3.8	OOSEM Support to Integrate and Verify System	513
17.3.9	Develop Enabling Systems	515
17.4	Summary	518
17.5	Questions	519

PART IV TRANSITIONING TO MODEL-BASED SYSTEMS ENGINEERING

CHAPTER 18	Integrating SysML into a Systems Development Environment.....	523
18.1	Understanding the System Model's Role in the Broader Modeling Context	523
18.1.1	The System Model as an Integrating Framework	523
18.1.2	Types of Models and Simulations.....	523
18.1.3	Using the System Model with Other Models	526
18.2	Tool Roles in a Systems Development Environment.....	530
18.2.1	Use of Tools to Model and Specify the System.....	530
18.2.2	Use of Tools to Manage the Design Configuration and Related Data....	531
18.2.3	Use of Tools to View and Document the Data.....	534
18.2.4	Verification and Validation Tools.....	535
18.2.5	Use of Project Management Tools to Manage the Development Process	535
18.3	An Overview of Information Flow between Tools.....	535
18.3.1	Interconnecting the System Modeling Tool with Other Tools.....	535
18.3.2	Interface with Requirements Management Tool	536
18.3.3	Interface with SoS/Business Modeling Tools.....	538
18.3.4	Interface with Simulation and Analysis Tools.....	538
18.3.5	Interface with Verification Tools.....	539
18.3.6	Interface with Development Tools.....	539
18.3.7	Interface with Documentation & View Generation Tool	540
18.3.8	Interface with Configuration Management Tool.....	540
18.3.9	Interface with Project Management Tool	542
18.4	Data Exchange Mechanisms	542
18.4.1	Considerations for Data Exchange	542
18.4.2	File-Based Exchange.....	544
18.4.3	API-based Exchange	546
18.4.4	Performing Transformations	547

18.5	Data Exchange Applications.....	548
18.5.1	SysML to Modelica (bidirectional transformation).....	548
18.5.2	Interchanging SysML Models and Ontologies.....	552
18.5.3	Document Generation from Models (unidirectional transformation)	552
18.6	Selecting a System Modeling Tool.....	553
18.6.1	Tool Selection Criteria	553
18.6.2	SysML Compliance.....	554
18.7	Summary	554
18.8	Questions	555
CHAPTER 19	Deploying SysML into an Organization.....	557
19.1	Improvement Process	557
19.1.1	Monitor and Assess	558
19.1.2	Plan the Improvement	559
19.1.3	Define Changes to Process, Methods, Tools, and Training.....	559
19.1.4	Pilot the Approach	560
19.1.5	Deploy Changes Incrementally	561
19.2	Summary	563
19.3	Questions	563
	Appendix A.....	565
	References.....	595
	Index	599