

First edition
2017-03

Information technology — Object management group systems modeling language (OMG SysML)

*Technologies de l'information — Langage de modélisation de systèmes
OMG (OMG SysML)*



Reference number
ISO/IEC 19514:2017(E)

© ISO/IEC 2017



COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2017, Published in Switzerland

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
Ch. de Blandonnet 8 • CP 401
CH-1214 Vernier, Geneva, Switzerland
Tel. +41 22 749 01 11
Fax +41 22 749 09 47
copyright@iso.org
www.iso.org

Table of Contents

FOREWORD	xix
INTRODUCTION	xx
1 Scope	1
1.1 General	1
2 Normative References	1
3 Additional Information	2
3.1 Relationships to Other Standards	2
3.2 How to Read this International Standard	2
3.2.1 Organization	3
3.3 Acknowledgments	4
4 Language Architecture	7
4.1 General	7
4.2 Design Principles	10
4.3 Architecture	10
4.4 Extension Mechanisms	13
4.5 SysML Diagrams	13
5 Conformance	15
5.1 Overview	15
5.2 Conformance Types	15
6 Language Formalism	17
6.1 Levels of Formalism	17
6.2 Clause Structure	17
6.2.1 Overview	17
6.2.2 Diagram Elements	17
6.2.3 UML Extensions	17
6.2.4 Usage Examples	18
6.3 Conventions and Typography	18
STRUCTURAL CONSTRUCTS	19

7 Model Elements	21
7.1 Overview.....	21
7.1.1 View and Viewpoint.....	21
7.2 Diagram Elements.....	22
7.3 UML Extensions.....	25
7.3.1 Diagram Extensions.....	25
7.3.1.1 UML Diagram Elements not Included in SysML.....	25
7.3.2 Stereotypes.....	26
7.3.2.1 Conform.....	26
7.3.2.2 ElementGroup.....	27
7.3.2.3 Expose.....	28
7.3.2.4 Problem.....	28
7.3.2.5 Rationale.....	29
7.3.2.6 Stakeholder.....	29
7.3.2.7 View.....	29
7.3.2.8 Viewpoint.....	30
7.4 Usage Examples.....	30
8 Blocks	33
8.1 Overview.....	33
8.2 Diagram Elements.....	34
8.2.1 Block Definition Diagram.....	34
8.2.2 Internal Block Diagram.....	40
8.3 UML Extensions.....	42
8.3.1 Diagram Extensions.....	42
8.3.1.1 Block Definition Diagram.....	42
8.3.1.2 Internal Block Diagram.....	44
8.3.1.3 UML Diagram Elements not Included in SysML Block Definition Diagrams.....	46
8.3.1.4 UML Diagram Elements not Included in SysML Internal Block Diagrams.....	46
8.3.2 Stereotypes.....	47
8.3.2.1 AdjunctProperty.....	49
8.3.2.2 Binding Connector.....	50
8.3.2.3 Block.....	51
8.3.2.4 Bound Reference.....	53
8.3.2.5 ClassifierBehaviorProperty.....	54
8.3.2.6 ConnectorProperty.....	54
8.3.2.7 DirectedRelationshipPropertyPath.....	55
8.3.2.8 DistributedProperty.....	56
8.3.2.9 ElementPropertyPath.....	56
8.3.2.10 EndPathMultiplicity.....	56
8.3.2.11 NestedConnectorEnd.....	57
8.3.2.12 ParticipantProperty.....	57
8.3.2.13 PropertySpecificType.....	58
8.3.2.14 ValueType.....	58
8.3.3 Model Libraries.....	59

8.3.3.1 Package PrimitiveValueTypes	59
8.3.3.2 Package UnitAndQuantityKind	60
8.4 Usage Examples	62
8.4.1 Wheel Hub Assembly.....	62
8.4.2 Example Value Type Definitions	64
8.4.3 Design Configuration for SUV EPA Fuel Economy Test.....	65
8.4.4 Water Delivery	65
8.4.5 Constraining Decomposition	65
8.4.6 Units and Quantity Kinds	67
9 Ports and Flows.....	71
9.1 Overview	71
9.1.1 Ports.....	71
9.1.2 Flow Properties, Provided and Required Features, and Nested Ports	71
9.1.3 Proxy Ports and Full Ports	71
9.1.4 Item Flows.....	72
9.1.5 Deprecation of Flow Ports and Flow Specifications.....	72
9.2 Diagram Elements.....	73
9.2.1 Block Definition Diagram.....	73
9.2.2 Internal Block Diagram.....	76
9.3 UML Extensions	78
9.3.1 Diagram Extensions	78
9.3.1.1 DirectedFeature	78
9.3.1.2 FlowProperty	78
9.3.1.3 FullPort	78
9.3.1.4 InvocationOnNestedPortAction	78
9.3.1.5 ItemFlow	78
9.3.1.6 Port.....	78
9.3.1.7 ProxyPort	79
9.3.1.8 TriggerOnNestedPort	79
9.3.2 Stereotypes.....	79
9.3.2.1 AcceptChangeStructuralFeatureEventAction	81
9.3.2.2 Block	82
9.3.2.3 ChangeStructuralFeatureEvent	82
9.3.2.4 DirectedFeature	82
9.3.2.5 FeatureDirection	83
9.3.2.6 FlowDirection.....	84
9.3.2.7 FlowProperty	84
9.3.2.8 FullPort	85
9.3.2.9 InterfaceBlock	86
9.3.2.10 InvocationOnNestedPortAction	86
9.3.2.11 ItemFlow	86
9.3.2.12 ProxyPort	87
9.3.2.13 TriggerOnNestedPort	88

9.4 Usage Examples	89
9.4.1 Ports with Required and Provided Features	89
9.4.2 Flow Ports and Item Flows	89
9.4.3 Ports with Flow Properties	90
9.4.4 Proxy and Full Ports	90
9.4.5 Association and Port Decomposition	91
9.4.6 Item Flow Decomposition	95
10 Constraint Blocks.....	97
10.1 Overview.....	97
10.2 Diagram Elements	98
10.2.1 Block Definition Diagram.....	98
10.2.2 Parametric Diagram	98
10.3 UML Extensions	99
10.3.1 Diagram Extensions	99
10.3.1.1 Block Definition Diagram	99
10.3.1.2 Parametric Diagram	101
10.3.2 Stereotypes	100
10.3.2.1 ConstraintBlock	101
10.4 Usage Examples	101
10.4.1 Definition of Constraint Blocks on a Block Definition Diagram.....	101
10.4.2 Usage of Constraint Blocks on a Parametric Diagram.....	101
BEHAVIORAL CONSTRUCTS.....	103
11 Activities	105
11.1 Overview.....	105
11.1.1 Control as Data	105
11.1.2 Continuous Systems	105
11.1.3 Probability	105
11.1.4 Activities as Blocks.....	106
11.1.5 Timelines.....	106
11.2 Diagram Elements.....	107
11.2.1 Activity Diagram	105
11.3 UML Extensions	114
11.3.1 Diagram Extensions	114
11.3.1.1 Activity	114
11.3.1.2 CallBehaviorAction	115
11.3.1.3 ControlFlow	116
11.3.1.4 ObjectNode, Variables, and Parameters	116
11.3.2 Stereotypes.....	117
11.3.2.1 Continuous	118

11.3.2.2 ControlOperator	119
11.3.2.3 Discrete	119
11.3.2.4 NoBuffer	119
11.3.2.5 Overwrite	120
11.3.2.6 Optional	120
11.3.2.7 Probability	120
11.3.2.8 Rate	121
11.3.3 Model Libraries	121
11.3.3.1 Package ControlValues	121
11.4 Usage Examples	122
12 Interactions	127
12.1 Overview	127
12.2 Diagram Elements	128
12.2.1 Sequence Diagram	128
12.3 UML Extensions	133
12.3.1 Diagram Extensions	133
12.3.1.1 Exclusion of Communication Diagram, Interaction Overview Diagram, and Timing Diagram	133
12.3.1.2 Interactions and Parameters	133
12.4 Usage Examples	134
12.4.1 Sequence Diagrams	134
13 State Machines	135
13.1 Overview	135
13.2 Diagram Elements	135
13.2.1 State Machine Diagram	135
13.3 UML Extensions	140
13.3.1 Diagram Extensions	140
13.3.1.1 State Machines and Parameters	140
13.4 Usage Examples	140
13.4.1 State Machine Diagram	140
14 Use Cases	141
14.1 Overview	141
14.2 Diagram Elements	142
14.2.1 Use Case Diagram	142
14.3 UML Extensions	143
14.4 Usage Examples	143
CROSSCUTTING CONSTRUCTS	145

15	Allocations	147
15.1	Overview	147
15.2	Diagram Elements	147
15.2.1	Representing Allocation on Diagrams	148
15.3	UML Extensions	149
15.3.1	Diagram Extensions	149
15.3.1.1	Tables	149
15.3.1.2	Allocate Relationship Rendering	149
15.3.1.3	Allocation Compartment Format	149
15.3.1.4	Allocation Callout Format	149
15.3.1.5	AllocatedActivityPartition Label	149
15.3.2	Stereotypes	150
15.3.2.1	Allocate(from Allocations)	150
15.3.2.2	AllocateActivityPartition(from Allocations)	151
15.4	Usage Examples	152
15.4.1	Behavior Allocation of Actions to Parts and Activities to Blocks	152
15.4.2	Allocate Flow	153
15.4.2.1	Allocating Structure	154
15.4.2.2	Automotive Example	154
15.4.3	Tabular Representation	155
16	Requirements	157
16.1	Overview	157
16.2	Diagram Elements	159
16.2.1	Requirement Diagram	159
16.3	UML Extensions	162
16.3.1	Diagram Extensions	162
16.3.1.1	Requirement Diagram	162
16.3.1.2	Requirement Notation	162
16.3.1.3	Requirement Property Callout Format	162
16.3.1.4	Requirements on Other Diagrams	162
16.3.1.5	Requirements Table	163
16.3.2	Stereotypes	164
16.3.2.1	Copy	164
16.3.2.2	DeriveReq	165
16.3.2.3	Refine	165
16.3.2.4	Requirement	165
16.3.2.5	TestCase	167
16.3.2.6	Satisfy	167
16.3.2.7	Trace	167
16.3.2.8	Verify	168
16.4	Usage Examples	168
16.4.1	Requirement Decomposition and Traceability	168
16.4.2	Requirements and Design Elements	169

16.4.3 Requirements Reuse	171
16.4.4 Verification Procedure (Test Case)	172
17 Profiles & Model Libraries.....	175
17.1 Overview	175
17.2 Diagram Elements.....	176
17.2.1 Profile Definition in Package Diagram	176
17.2.1.1 Extension	178
17.2.2 Stereotypes Used On Diagrams	178
17.2.2.1 StereotypeInNode	179
17.2.2.2 StereotypeInComment	180
17.2.2.3 StereotypeInCompartment	180
17.3 UML Extensions	180
17.4 Usage Examples	180
17.4.1 Defining a Profile.....	180
17.4.2 Adding Stereotypes to a Profile	181
17.4.3 Defining a Model Library that Uses a Profile.....	182
17.4.4 Guidance on Whether to Use a Stereotype or Class	183
17.4.5 Using a Profile.....	183
17.4.6 Using a Stereotype	184
17.4.7 Using a Model Library Element.....	184
ANNEXES	187
Annex A: Diagrams.....	189
Annex B: SysML Diagram Interchange	195
Annex C: Deprecated Elements	205
Annex D: Sample Problem	213
Annex E: Non-normative Extensions.....	251
Annex F: Requirements Traceability	319
Annex G: Model Interchange.....	321
Annex H: Legal Information	325

This is a preview of "ISO/IEC 19514:2017". [Click here to purchase the full version from the ANSI store.](#)

List of Figures

Figure 4.1 - Overview of SysML/UML Interrelationship.....	7
Figure 4.2 - SysML Extension of UML.....	11
Figure 4.3 - SysML Package Structure.....	12
Figure 4.4 - Non-normative Package Structure.....	13
Figure 7.1 - Stereotypes defined in package ModelElements.....	26
Figure 7.2 - Rationale and Problem examples.....	31
Figure 8.1 - Nested property reference.....	45
Figure 8.2 - Abstract syntax extensions for SysML blocks.....	47
Figure 8.3 - Abstract syntax extensions for SysML properties.....	47
Figure 8.4 - Abstract syntax extensions for SysML value types.....	47
Figure 8.5 - Abstract syntax extensions for SysML property paths.....	48
Figure 8.6 - Abstract syntax extensions for SysML connector ends.....	48
Figure 8.7 - Abstract syntax extensions for SysML property-specific types.....	48
Figure 8.8 - Abstract syntax extensions for SysML bound references.....	49
Figure 8.9 - Abstract syntax extensions for SysML adjunct properties and classifier behavior properties.....	49
Figure 8.10 - Model library for primitive value types.....	59
Figure 8.11 - Model library for Unit and QuantityKind.....	60
Figure 8.12 - Block diagram for the Wheel Package.....	63
Figure 8.13 - Internal Block Diagram for WheelHubAssembly.....	64
Figure 8.14 - Defining Value Types with units of measure from the International System of Units (SI).....	64
Figure 8.15 - Vehicle decomposition.....	65
Figure 8.16 - Vehicle internal structure.....	66
Figure 8.17 - Vehicle specialization.....	66
Figure 8.18 - Example of Unit, QuantityKind and ValueType definitions.....	67
Figure 8.19 - Instance-level view of the Unit, QuantityKind and ValueType definitions.....	68
Figure 8.20 - Example of equivalent Unit representations.....	68
Figure 8.21 - Instance-level representation of equivalent Unit definitions.....	69
Figure 9.1 - Port Stereotypes.....	79
Figure 9.2 - Stereotypes for Actions on Nested Ports.....	80
Figure 9.3 - Stereotypes for Property Value Change Events.....	80
Figure 9.4 - Provided and Required Features.....	80
Figure 9.5 - ItemFlow Stereotype.....	81
Figure 9.6 - Usage example of ports with provided and required features.....	89
Figure 9.7 Usage example of proxy and full ports.....	91
Figure 9.8 - Water Delivery association block.....	92
Figure 9.9 - Internal structure of Water Delivery association block.....	92
Figure 9.10 - Two views of Water Delivery connector within House block.....	93
Figure 9.11 - Specializations of Water Client in house example.....	93
Figure 9.12 - Plumbing association block.....	94
Figure 9.13 - Internal structure of Plumbing association block.....	94
Figure 9.14 - Water Delivery association block with internal Plumbing connector.....	94

Figure 9.15 - Usage example of item flows in internal block diagrams	95
Figure 9.16 - Usage example of item flow decomposition	96
Figure 9.17 - Usage example of item flow decomposition	96
Figure 10.1 - Stereotypes defined in SysML ConstraintBlocks package.....	100
Figure 11.1 - Block definition diagram with activities as blocks.....	115
Figure 11.2 - CallBehaviorAction notation.with behavior stereotype	115
Figure 11.3 - CallBehaviorAction notation.with action name	115
Figure 11.4 - Control flow notation.....	116
Figure 11.5 - Block definition diagram with activities as blocks associated with types of object nodes, variables, and parameters	116
Figure 11.6 - ObjectNode notation in activity diagrams	117
Figure 11.7 - ObjectNode notation in activity diagrams	117
Figure 11.8 - Abstract Syntax for SysML Activity Extensions	118
Figure 11.9 - Control values.....	121
Figure 11.10 - Continuous system example 1	123
Figure 11.11 - Continuous system example 2	124
Figure 11.12 - Continuous system example 3	124
Figure 11.13 - Example block definition diagram for activity decomposition	125
Figure 11.14 - Example block definition diagram for object node types.....	125
Figure 12.1 - Block definition diagram with interactions as blocks associated with used interactions and types of parameters.....	133
Figure 13.1 - Block definition diagram with state machines as blocks associated with submachines and types of parameters.....	140
Figure 15.1 - Abstract syntax extensions for SysML Allocation.....	150
Figure 15.2 - Abstract syntax expression for AllocatedActivityPartition.....	150
Figure 15.3 - Generic Allocation, including /from and /to association ends	152
Figure 15.4 - Behavior allocation.....	152
Figure 15.5 - Example of flow allocation from ObjectFlow to Connector	153
Figure 15.6 - Example of flow allocation from ObjectFlow to ItemFlow	153
Figure 15.7 - Example of flow allocation from ObjectNode to FlowProperty	154
Figure 15.8 - Example of Structural Allocation	154
Figure 15.9 - Allocation Matrix showing Allocation for Hybrid SUV Accelerate Example.....	155
Figure 16.1 - Non-normative Examples of Tabular Representations of Requirements	163
Figure 16.2 - Abstract Syntax for Requirements Stereotypes	164
Figure 16.3 - Requirements Derivation.....	169
Figure 16.4 - Links between requirements and design.....	170
Figure 16.5 - Requirement satisfaction in an internal block diagram	171
Figure 16.6 - Use of the copy dependency to facilitate reuse	171
Figure 16.7 - Linkage of a Test Case to a requirement: This figure shows the Requirement Diagram	172
Figure 16.8 - Linkage of a Test Case to a requirement: This figure shows the Test Case as a State Diagram.....	173
Figure 17.1 - Defining a stereotype.....	178
Figure 17.2 - Using a stereotype	179

Figure 17.3 - Using stereotypes and showing values.....	180
Figure 17.4 - Other notational forms for showing values	180
Figure 17.5 - Definition of a profile.....	181
Figure 17.6 - Profile Contents.....	181
Figure 17.7 - Two model libraries.....	182
Figure 17.8 - A model with applied profile and imported model library.....	183
Figure 17.9 - Using two stereotypes on a model element.....	184
Figure 17.10 - Using model library elements.....	184
Figure A.1 - SysML Diagram Taxonomy	190
Figure A.2 - Diagram Frame.....	191
Figure A.3 - Diagram Usages.....	193
Figure A.4 - Optional Form of Line Crossing.....	194
Figure C.1 - Deprecated Stereotypes	208
Figure D.1 - Establishing the User Model by Importing and Applying SysML Profile & Model Library (Package Diagram).....	214
Figure D.2 - Defining valueTypes and units to be Used in the Sample Problem	215
Figure D.3 - Establishing Structure of the User Model using Packages and Views (Package Diagram)	216
Figure D.4 - Establishing the Context of the Hybrid SUV System using a User-Defined Context Diagram. (Internal Block Diagram) Completeness of Diagram Noted in Diagram Description.....	217
Figure D.5 - Establishing Top Level Use Cases for the Hybrid SUV (Use Case Diagram).....	218
Figure D.6 - Establishing Operational Use Cases for “Drive the Vehicle” (Use Case Diagram).....	219
Figure D.7 - Elaborating Black Box Behavior for the “Drive the Vehicle” Use Case (Sequence Diagram).....	220
Figure D.8 - Finite State Machine Associated with “Drive the Vehicle” (State Machine Diagram).....	221
Figure D.9 - Black Box Interaction for “StartVehicle,” referencing White Box Interaction (Sequence Diagram).....	221
Figure D.10 - White Box Interaction for “StartVehicle” (Sequence Diagram).....	222
Figure D.11 - Establishing HSUV Requirements Hierarchy (containment) - (Requirements Diagram).....	223
Figure D.12 - Establishing Derived Requirements and Rationale from Lowest Tier of Requirements Hierarchy (Requirements Diagram).....	224
Figure D.13 - Acceleration Requirement Relationships (Requirements Diagram)	225
Figure D.14 - Requirements Relationships Expressed in Tabular Format (Table)	226
Figure D.15 - Defining the Automotive Domain (compare with Figure D.4) - (Block Definition Diagram)	227
Figure D.16 - Defining Structure of the Hybrid SUV System (Block Definition Diagram)	227
Figure D.17 - Internal Structure of Hybrid SUV (Internal Block Diagram).....	228
Figure D.18 - Defining Structure of Power Subsystem (Block Definition Diagram).....	229
Figure D.19 - Internal Structure of the Power Subsystem (Internal Block Diagram).....	230
Figure D.20 - Blocks Typing Ports in the Power Subsystem (Block Definition Diagram)	230
Figure D.21 - Initially Defining Port Types with Flow Properties for the CAN Bus (Block Definition Diagram).....	231
Figure D.22 - Consolidating Connectors into the CAN Bus. (Internal Block Diagram).....	232
Figure D.23 - Elaborating Definition of Fuel Flow. (Block Definition Diagram).....	232
Figure D.24 - Defining Fuel Flow Constraints (Parametric Diagram)	233
Figure D.25 - Detailed Internal Structure of Fuel Delivery Subsystem (Internal Block Diagram)	234
Figure D.26 - Defining Analyses for Hybrid SUV Engineering Development (Block Definition Diagram)	235
Figure D.27 - Establishing a Performance View of the User Model (Package Diagram)	236
Figure D.28 - Defining Requirements and VnV viewpoints (Package Diagram).....	237

Figure D.29 - Requirements and VnV views exposing elements from the model (Package Diagram).....	238
Figure D.30 - The Requirements and VnV views with supporting views (Package Diagram)	239
Figure D.31 - Defining Measures of Effectiveness and Key Relationships (Parametric Diagram).....	240
Figure D.32 - Establishing Mathematical Relationships for Fuel Economy Calculations (Parametric Diagram)..	241
Figure D.33 - Straight Line Vehicle Dynamics Mathematical Model (Parametric Diagram).....	242
Figure D.34 - Defining Straight-Line Vehicle Dynamics Mathematical Constraints (Block Definition Diagram).....	243
Figure D.35 - Results of Maximum Acceleration Analysis (Timing Diagram).....	244
Figure D.36 - Behavior Model for “Accelerate” Function (Activity Diagram).....	245
Figure D.37 - Decomposition of “Accelerate” Function (Block Definition diagram).....	246
Figure D.38 - Detailed Behavior Model for “Provide Power” (Activity Diagram) Note hierarchical consistency with Figure D.36.....	247
Figure D.39 - Flow Allocation to Power Subsystem (Internal Block Diagram).....	248
Figure D.40 - Tabular Representation of Allocation from “Accelerate” Behavior Model to Power Subsystem (Table).....	248
Figure D.41 - Special Case of Internal Block Diagram Showing Reference to Specific Properties (serial numbers).....	249
Figure E.1 - Example activity with «effbd» stereotype applied.....	253
Figure E.2 - Example activity with «streaming» and «nonStreaming» stereotypes applied to subactivities	253
Figure E.3 - Example extensions to Requirement.....	256
Figure E.4 - Example Parametric Diagram using Stereotypes for Measures of Effectiveness	257
Figure E.5 - QUDV Concepts diagram	259
Figure E.6 - QUDV Units diagram	260
Figure E.7 - QUDV Quantity Kinds diagram	260
Figure E.8 - Base Unit and Quantity Kinds of the SI and ISQ respectively	278
Figure E.9 - Example of a derived unit and derived quantity kind	278
Figure E.10 - Spring Length Example	279
Figure E.11 - Model libraries of SysML Quantity Kinds and Units for the covered content of ISO 80000 parts 3,4,5,6,7,9,10 and 13.....	280
Figure E.12 - Organization of the definitions of units and quantities from the normative parts of ISO 80000 covered in SysML 1.4, which includes all the normative content of parts 3,4,5,6; the subset of parts 7,9,10 corresponding to the content from SysML 1.3 and the subset of part 13 pertaining to commonly used units of information. Parts 8,11 and 12 are not covered because none of their units and quantities were referenced in previous versions of SysML nor in the summary tables in ISO 80000-1	281
Figure E.13 - Content relationships for the systems of units and quantities in from the different parts of ISO 80000 in relation to ISO 80000 as a whole and to the International System of Units (SI) and quantities (ISQ)	282
Figure E.14 - Table 1 (from ISO 80000-1) SI base units for the ISQ base quantities	283
Figure E.15 - Table 2 (from ISO 80000-1) ISQ derived quantities and SI derived units with special names (1) ..	284
Figure E.16 - Table 2 (from ISO 80000-1) ISQ derived quantities and SI derived units with special names (2) ..	285
Figure E.17 - Table 2 (from ISO 80000-1) ISQ derived quantities and SI derived units with special names (3) ..	286
Figure E.18 - Table 3 (from the SI brochure) SI derived units with special names and symbols.....	287
Figure E.19 - Constant numbers used throughout the SysML ISO 80000 library.	289

This is a preview of "ISO/IEC 19514:2017". [Click here to purchase the full version from the ANSI store.](#)

Figure E.20 - Example of value type definitions for a quantity and applicable units and prefixed units.....	290
Figure E.21 - Basic distribution stereotypes	316
Figure E.22 - Distribution Example	317
Figure G.1 - SysML/AP233 Data Overlaps.....	322

This is a preview of "ISO/IEC 19514:2017". [Click here to purchase the full version from the ANSI store.](#)

List of Tables

Table 4.1 - UML 2 metaclasses excluded from the UML4SysML subset	8
Table 4.2 - UML 2 metaclasses and datatypes included in the UML4SysML subset	9
Table 4.3 - SysML stereotypes, blocks, valuetypes, and datatypes	10
Table 7.1 - Graphical nodes defined by ModelElements package	22
Table 7.2 - Graphical paths defined by ModelElements package	24
Table 8.1 - Graphical nodes defined in Block Definition diagrams	34
Table 8.2 - Graphical paths defined by in Block Definition diagrams	37
Table 8.3 - Graphical nodes defined in Internal Block diagrams	40
Table 8.4 - Graphical paths defined in Internal Block diagrams	41
Table 9.1 - Graphical nodes defined in Block Definition diagrams	73
Table 9.2 - Graphical nodes defined in Internal Block diagrams	76
Table 10.1 - Graphical nodes defined in Block Definition diagrams	98
Table 10.2 - Graphical nodes defined in Parametric diagrams	99
Table 11.1 - Graphical nodes included in activity diagrams	107
Table 11.2 - Graphical paths included in activity diagrams	112
Table 11.3 - Other graphical elements included in activity diagrams	113
Table 12.1 - Graphical nodes included in sequence diagrams	128
Table 12.2 - Graphical paths included in sequence diagram	132
Table 12.3 - Other graphical elements included in sequence diagram	132
Table 13.1 - Graphical nodes included in state machine diagrams	135
Table 13.2 - Graphical paths included in state machine diagrams	139
Table 13.3 - Other graphical elements included in state machine diagram	139
Table 14.1 - Graphical nodes included in Use Case diagrams	142
Table 14.2 - Graphical paths included in Use Case diagrams	143
Table 15.1 - Extension to graphical nodes included in diagrams	148
Table 16.1 - Graphical nodes included in Requirement diagrams	159
Table 16.2 - Graphical paths included in Requirement diagrams	160
Table 17.1 - Graphical nodes used in profile definition	176
Table 17.2 - Graphical paths used in profile definition	177
Table 17.3 - Notations for Stereotype Use	178
Table 17.4 - Notations for Stereotype Use (continued)	179
Table B.1 - SysML Diagram Elements	201
Table C.1 - Graphical nodes defined in block definition diagrams	206
Table C.2 - Graphical nodes defined in internal block diagrams	207
Table E.1 - Addition stereotypes for EFFBDs	251
Table E.2 - Streaming options for activities	252
Table E.3 - Additional Requirement Stereotypes	254
Table E.4 - Requirement property enumeration types	255
Table E.5 - Stereotypes for Measures of Effectiveness	257

Table E.6 - The decimal and binary prefixes in scope of the International System of Units (SI) which uses the ISO 80000 system of units and its included systems of units such as ISO 80000-13	287
Table E.7 - Normative units in ISO 80000-3 (1 of 2)	291
Table E.8 - Normative units in ISO 80000-3 (2 of 2)	292
Table E.9 - Normative quantity kinds in ISO 80000-3 (1 of 2)	292
Table E.10 - Normative quantity kinds in ISO 80000-3 (2 of 2)	293
Table E.11 - Normative units in ISO 80000-4 (1 of 2).....	294
Table E.12 - Normative units in ISO 80000-4 (2 of 2)	295
Table E.13 - Normative quantity kinds in ISO 80000-4 (1 of 4)	296
Table E.14 - Normative quantity kinds in ISO 80000-4 (2 of 4)	297
Table E.15 - Normative quantity kinds in ISO 80000-4 (3 of 4)	298
Table E.16 - Normative quantity kinds in ISO 80000-4 (4 of 4)	299
Table E.17 - Normative units in ISO 80000-5 (1 of 2)	300
Table E.18 - Normative units in ISO 80000-5 (2 of 2)	301
Table E.19 - Normative quantity kinds in ISO 80000-5 (1 of 5)	302
Table E.20 - Normative quantity kinds in ISO 80000-5 (2 of 5)	303
Table E.21 - Normative quantity kinds in ISO 80000-5 (3 of 5)	303
Table E.22 - Normative quantity kinds in ISO 80000-5 (4 of 5)	304
Table E.23 - Normative quantity kinds in ISO 80000-5 (5 of 5)	305
Table E.24 - Normative units in ISO 80000-6 (1 of 5)	306
Table E.25 - Normative units in ISO 80000-6 (2 of 5)	307
Table E.26 - Normative units in ISO 80000-6 (3 of 5)	307
Table E.27 - Normative units in ISO 80000-6 (4 of 5)	308
Table E.28 - Normative units in ISO 80000-6 (5 of 5)	309
Table E.29 - Normative quantity kinds in ISO 80000-6 (1 of 4)	310
Table E.30 - Normative quantity kinds in ISO 80000-6 (2 of 4)	311
Table E.31 - Normative quantity kinds in ISO 80000-6 (3 of 4)	312
Table E.32 - Normative quantity kinds in ISO 80000-6 (4 of 4)	313
Table E.33 - Units in ISO 80000-7.....	314
Table E.34 - Quantity Kinds in ISO 80000-7.....	314
Table E.35 - Units in ISO 80000-9.....	314
Table E.36 - Quantity Kinds in ISO 80000-9.....	315
Table E.37 - Units in ISO 80000-10.....	315
Table E.38 - Quantity Kinds in ISO 80000-10.....	315
Table E.39 - Units in ISO 80000-13.....	315
Table E.40 - Quantity Kinds in ISO 80000-13.....	316
Table E.41 - Distribution Stereotypes	317

FOREWORD

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation on the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see the following URL: www.iso.org/iso/foreword.html.

This document was prepared by the Object Management Group (OMG) and was adopted, under the PAS procedure, by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, in parallel with its approval by national bodies of ISO and IEC.

This document is related to:

- ITU-T Recommendation X.902 (1995) | ISO/IEC 10746-2:1995, *Information Technology - Open Distributed Processing - Reference Model: Foundations*
- ITU-T Recommendation X.903 (1995) | ISO/IEC 10746-3:1995, *Information Technology - Open Distributed Processing - Reference Model: Architecture*
- ITU-T Recommendation X.920 (1997) | ISO/IEC 14750:1997, *Information Technology - Open Distributed Processing - Interface Definition Language*

Apart from this Foreword, the text of this document is identical with that for the OMG specification for Systems Modeling Language, v1.4.1.

INTRODUCTION

The rapid growth of distributed processing has led to a need for a coordinating framework for this standardization and ITU-T Recommendations X.901-904 | ISO/IEC 10746, the Reference Model of Open Distributed Processing (RM-ODP) provides such a framework. It defines an architecture within which support of distribution, interoperability and portability can be integrated.

RM-ODP Part 2 (ISO/IEC 10746-2) defines the foundational concepts and modeling framework for describing distributed systems. The scopes and objectives of the RM-ODP Part 2 and the UML, while related, are not the same and, in a number of cases, the RM-ODP Part 2 and the UML specification use the same term for concepts which are related but not identical (e.g., interface). Nevertheless, a specification using the Part 2 modeling concepts can be expressed using UML with appropriate extensions (using stereotypes, tags, and constraints).

RM-ODP Part 3 (ISO/IEC 10746-3) specifies a generic architecture of open distributed systems, expressed using the foundational concepts and framework defined in Part 2. Given the relation between UML as a modeling language and Part 3 of the RM-ODP standard, it is easy to show that UML is suitable as a notation for the individual viewpoint specifications defined by the RM-ODP.

This International Standard for OMG Systems Modeling Language is a standard for the technology specification of an ODP system. It defines a technology to provide the infrastructure required to support functional distribution of an ODP system, specifying functions required to manage physical distribution, communications, processing and storage, and the roles of different technology objects in supporting those functions.

This International Standard defines a general-purpose language for systems engineering applications, called the OMG Systems Modeling Language (OMG SysML™). Throughout the rest of this International Standard the language will be referred to as SysML.

SysML supports the specification, analysis, design, verification, and validation of a broad range of complex systems. These systems may include hardware, software, information, processes, personnel, and facilities.

It is common practice for engineers to use a wide range of modeling languages, tools, and techniques on large systems projects. SysML is intended to unify diverse modeling languages used by systems engineers and can be used with a wide variety of discipline- and domain-specific modeling languages.

SysML reuses a subset of UML 2.5 and provides additional extensions needed to address the requirements in UML for SE. SysML uses the UML 2.5 extension mechanisms as further elaborated in Clause 17 as the primary mechanism to specify the extensions to UML 2.5. This revision of SysML relies on several new features incorporated into UML 2.5. Any use of the term “UML 2” or “UML” in this International Standard, unless otherwise noted, will refer to UML 2.5 in general and the UML 2.5 specification in particular.

Since SysML uses UML 2.5 as its foundation, systems engineers modeling with SysML and software engineers modeling with UML 2.5 will be able to collaborate on models of software-intensive systems. This will improve communication among the various stakeholders who participate in the systems development process and promote interoperability among modeling tools. It is anticipated that SysML will be customized to model domain-specific applications, such as automotive, aerospace, communication, and information systems.