

Table of Contents

Rainer Drath, Pascal Habiger

1 What is AutomationML? — 1

- 1.1 Introduction and Motivation — 3
 - 1.1.1 About this book — 3
 - 1.1.2 Reading guidance — 4
- 1.2 Why AutomationML was developed — 5
- 1.3 Challenges for a neutral data format — 8
 - 1.3.1 Approaches for engineering data exchange — 8
 - 1.3.2 Iteration support: hidden but important and difficult — 13
 - 1.3.3 Management of heterogeneous semantics — 15
 - 1.3.4 Standardization deadlocks — 17
- 1.4 What is AutomationML? — 19
 - 1.4.1 Overview — 19
 - 1.4.2 Goals of AutomationML — 20
 - 1.4.3 Key innovations in AutomationML — 22
 - 1.4.4 Values of AutomationML — 24
- 1.5 Encoding information with AML – four layers of abstraction — 27
 - 1.5.1 Data model versus information model — 27
 - 1.5.2 The four-layer model of information encoding — 28
 - 1.5.3 AutomationML in the four-layers model — 31
- 1.6 Exclusions — 31
- 1.7 AutomationML Specifications — 32
 - 1.7.1 AutomationML IEC 62714 Series — 32
 - 1.7.2 Whitepapers — 35
 - 1.7.3 Application Recommendations — 35
 - 1.7.4 Best practice recommendations — 36
 - 1.7.5 Availability — 36
- 1.8 The AutomationML association — 37
 - 1.8.1 Initiators and members — 37
 - 1.8.2 Possibilities of participation — 38
- 1.9 References for Chapter 1 — 39

Rainer Drath

2 The CAEX and AutomationML guide — 45

- 2.1 AutomationML architecture and modelling philosophy — 47
 - 2.1.1 Introduction — 47
 - 2.1.2 AutomationML architecture — 47
 - 2.1.3 AutomationML submodels — 48
 - 2.1.4 Object-oriented modelling philosophy in plant engineering — 48
 - 2.1.5 Differences between AutomationML Edition 1 and 2 — 51
- 2.2 The CAEX 3.0 guide — 53
 - 2.2.1 About CAEX as meta data model — 53
 - 2.2.2 Persistence of a CAEX model: CAEX files — 54
 - 2.2.3 General CAEX modelling principles — 55

2.2.4	The instance hierarchy —	61
2.2.5	How to model a CAEX InternalElement —	63
2.2.6	CAEX library and class types —	69
2.2.7	How to model CAEX system unit classes —	72
2.2.8	How to model CAEX interface classes —	77
2.2.9	How to model CAEX role classes —	81
2.2.10	How to model CAEX attribute types —	85
2.2.11	How to model paths —	91
2.2.12	Modelling of relations —	92
2.2.13	Overriding inherited information —	102
2.2.14	Modelling version information —	104
2.3	AutomationML Edition 2 versus CAEX 3.0 —	110
2.3.1	Overview —	110
2.3.2	Additional general modelling rules —	111
2.3.3	Additional AutomationML versioning rules —	111
2.4	AutomationML base libraries —	113
2.4.1	General modelling rules —	113
2.4.2	Basic Role Class library —	113
2.4.3	Basic Interface Class Library —	118
2.4.4	Basic Attribute Type Library —	123
2.5	Referencing external documents —	126
2.5.1	Referencing COLLADA and behaviour documents —	126
2.5.2	Referencing external CAEX documents —	126
2.5.3	Referencing future external documents in the scope of AML —	129
2.5.4	Referencing additional documents outside the scope of AML —	129
2.5.5	Referencing CAEX attributes to items in external documents —	133
2.6	Extended AutomationML Concepts —	135
2.6.1	Overview —	135
2.6.2	The AML port concept - modelling of complex interfaces —	135
2.6.3	The AML facet concept - modelling of attribute filters —	138
2.6.4	The AML group concept - filtering of objects —	141
2.6.5	Combination of facets and groups —	143
2.6.6	Modelling of processes – the PPR concept —	146
2.6.7	Internationalization, multilingual attributes —	153
2.6.8	Modelling of lists and arrays —	155
2.6.9	The AML Container Format —	157
2.7	Best practice recommendations for AML modelling —	160
2.7.1	Modelling of Regular Expressions —	160
2.7.2	Modelling of Units —	161
2.8	Conclusion and summary —	163
2.9	References for Chapter 2 —	164

Rainer Drath, Steffen Lips

3 Modelling of Geometry and Kinematics — 167

3.1	Introduction —	169
3.2	Modelling of geometry and kinematics with COLLADA —	170
3.2.1	Overview of COLLADA 1.5 —	170

3.2.2	Geometry modelling	171
3.2.3	Kinematics description	187
3.3	Examples	200
3.3.1	BREP: Flange with hole	200
3.3.2	Facetted model: flange with hole	203
3.3.3	Kinematics of a screw with formula	204
3.4	Interlinking COLLADA and CAEX	206
3.4.1	Extensions of AML libraries for geometry and kinematics	206
3.4.2	How to reference a COLLADA document	207
3.4.3	Attributes refURI, Frame and refType	207
3.4.4	How to model attachments	208
3.4.5	Modelling rules	209
3.5	Summary	211
3.6	References for Chapter 3	212

Arndt Lüder, Nicole Schmidt, Rainer Drath

4 Modelling of Behaviour — 213

4.1	Introduction and Motivation	215
4.1.1	Who is using behaviour models?	216
4.1.2	What behaviour models are of interest?	217
4.2	AutomationML approach for representing behaviour models	218
4.3	How to reference external behaviour models	219
4.3.1	General: how to assign behaviour models to system objects	219
4.3.2	How to assign models of controlled good case and uncontrolled behaviour	219
4.3.3	How to assign interlocking models	222
4.4	How to model behaviour with IEC 61131-10	225
4.4.1	General approach	225
4.4.2	How to model sequences of controlled behaviour	226
4.4.3	How to model state-based models of uncontrolled behaviour	227
4.4.4	How to model interfaces	228
4.4.5	How to model interlocking models	229
4.5	Example	230
4.5.1	Modelling an example machine	230
4.5.2	Modelling the drives	230
4.5.3	Modelling the inputs and outputs of the drives	231
4.5.4	Modelling of interlocking behaviour	231
4.6	Modelling of complex behaviour or protected models via FMU	232
4.7	Discussion and Conclusion	233
4.8	References for Chapter 4	233

Lorenz Hundt, Josef Prinz, Rainer Drath

5 The AutomationML Editor — 235

5.1	Introduction	237
5.2	Purpose of Usage	238
5.3	User Interface	239

5.4	Editing Functions —	241
5.4.1	General UI concept: how to create, modify and delete objects —	241
5.4.2	How to create libraries —	242
5.4.3	How to create new classes —	242
5.4.4	How to model inheritance between classes —	243
5.4.5	How to associate roles to a <code>SystemUnitClass</code> —	243
5.4.6	How to modify inheritance relations —	244
5.4.7	How to create and edit attributes —	245
5.4.8	How to edit attribute constraints —	245
5.4.9	How to override an inherited attribute of a class —	247
5.4.10	How to override inherited elements of a class —	248
5.4.11	How to undo inheritance —	249
5.4.12	How to exclude inherited data —	249
5.4.13	How to create a <code>InstanceHierarchy</code> —	250
5.4.14	How to create an <code>ExternalInterface</code> —	250
5.4.15	How to create, delete and modify an <code>InternalLink</code> —	250
5.4.16	How to create instances —	251
5.4.17	How to associate roles to an <code>InternalElement</code> —	252
5.4.18	How to model <code>RoleRequirement</code> attributes or interfaces —	252
5.4.19	How to remove <code>RoleClass</code> associations —	253
5.4.20	How to model Mapping Objects —	253
5.5	Extended Functionality —	253
5.5.1	How to navigate between referenced classes or instances —	254
5.5.2	How to split and merge CAEX Documents —	254
5.5.3	How to convert between CAEX 2.15 and CAEX 3.0 —	255
5.5.4	How to validate an AutomationML document —	255
5.5.5	How to import AutomationML Libraries —	256
5.5.6	How to generate an AMLX Container —	257
5.5.7	How to save the CAEX-Schema —	259
5.5.8	How to create high-quality screenshots for documentation —	259
5.6	Customization of the AutomationML Editor —	260
5.6.1	Software updates, news, recent documents and libraries —	260
5.6.2	How to show-hide details —	260
5.6.3	How to use individual Themes and Layout's —	261
5.6.4	How to use Plugins —	262
5.7	How to send Feedback —	262
5.8	Summary —	263
5.9	References for Chapter 5 —	263

Abbreviations — 265

Trademarks — 267

Biographies — 273