

ArchiMate® 3.1 Specification



ArchiMate® 3.1 Specification

The Open Group Publications available from Van Haren Publishing

The TOGAF Series:

The TOGAF® Standard Version 9.2
The TOGAF® Standard Version 9.2 – A Pocket Guide
TOGAF® 9 Foundation Study Guide, 4th Edition
TOGAF® 9 Certified Study Guide, 4th Edition
TOGAF® Business Architecture Level 1 Study Guide

The Open Group Series:

The IT4IT™ Reference Architecture, Version 2.1
IT4IT™ for Managing the Business of IT – A Management Guide
IT4IT™ Foundation Study Guide, 2nd edition
The IT4IT™ Reference Architecture, Version 2.1 – A Pocket Guide
Cloud Computing for Business – The Open Group Guide
ArchiMate® 3.1 Specification – A Pocket Guide
ArchiMate® 3.1 Specification

The Open Group Security Series:

O-TTPS - A Management Guide
Open Information Security Management Maturity Model (O-ISM3)
Open Enterprise Security Architecture (O-ESA)
Risk Management – The Open Group Guide
The Open FAIR™ Body of Knowledge – A Pocket Guide

All titles are available to purchase from:

www.opengroup.org

www.vanharen.net

and also many international and online distributors.

The Open Group Standard

ArchiMate® 3.1 Specification



Title: **ArchiMate 3.1 Specification**

Series: The Open Group Series

A Publication of: The Open Group

Publisher: Van Haren Publishing, 's-Hertogenbosch,
www.vanharen.net

ISBN Hard copy: 978 94 018 0512 4
ISBN eBook (pdf): 978 94 018 0513 1
ISBN ePUB: 978 94 018 0514 8

Edition: Sixth edition, first impression, November 2019

DTP: Coco Bookmedia, Amersfoort –NL

Copyright: © 2012-2019, The Open Group.
All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior permission of the copyright owner.

It is fair use of this specification for implementers to use the names, labels, etc. contained within the specification. The intent of publication of the specification is to encourage implementations of the specification.

Document Number: B197
Published by The Open Group, November 2019.

For information on licensing refer to www.opengroup.org/legal.

Comments relating to the material contained in this document may be submitted to:
The Open Group
Apex Plaza
Reading
Berkshire, RG1 1AX
United Kingdom
or by electronic mail to: ogspecs@opengroup.org

Contents

1	Introduction.....	1
1.1	Objective.....	1
1.2	Overview.....	1
1.3	Conformance.....	1
1.4	Normative References.....	2
1.5	Terminology.....	2
1.6	Future Directions.....	2
2	Definitions.....	3
2.1	ArchiMate Core Framework.....	3
2.2	ArchiMate Core Language.....	3
2.3	Architecture View.....	3
2.4	Architecture Viewpoint.....	3
2.5	Aspect.....	3
2.6	Attribute.....	4
2.7	Composite Element.....	4
2.8	Concept.....	4
2.9	Conformance.....	4
2.10	Conforming Implementation.....	4
2.11	Core Element.....	4
2.12	Element.....	4
2.13	Layer.....	4
2.14	Model.....	4
2.15	Relationship.....	5
3	Language Structure.....	6
3.1	Language Design Considerations.....	6
3.2	Top-Level Language Structure.....	6
3.3	Layering of the ArchiMate Language.....	7
3.4	The ArchiMate Core Framework.....	8
3.5	The ArchiMate Full Framework.....	9
3.6	Abstraction in the ArchiMate Language.....	10
3.7	Concepts and their Notation.....	11
3.8	Use of Nesting.....	11
3.9	Use of Colors and Notational Cues.....	11
4	Generic Metamodel.....	13
4.1	Behavior and Structure Elements.....	13
4.1.1	Active Structure Elements.....	14
4.1.2	Behavior Elements.....	15
4.1.3	Passive Structure Elements.....	16
4.2	Specializations of Structure and Behavior Elements.....	16
4.3	Summary of Structure and Behavior Elements.....	18
4.4	Motivation Elements.....	19
4.5	Composite Elements.....	19

	4.5.1	Grouping.....	20
	4.5.2	Location.....	21
5		Relationships.....	22
	5.1	Structural Relationships.....	23
	5.1.1	Composition Relationship.....	23
	5.1.2	Aggregation Relationship.....	24
	5.1.3	Assignment Relationship.....	25
	5.1.4	Realization Relationship.....	26
	5.1.5	Semantics of Structural Relationships.....	27
	5.2	Dependency Relationships.....	28
	5.2.1	Serving Relationship.....	28
	5.2.2	Access Relationship.....	29
	5.2.3	Influence Relationship.....	30
	5.2.4	Association Relationship.....	31
	5.2.5	Semantics of Dependency Relationships.....	32
	5.3	Dynamic Relationships.....	33
	5.3.1	Triggering Relationship.....	33
	5.3.2	Flow Relationship.....	33
	5.3.3	Semantics of Dynamic Relationships.....	34
	5.4	Other Relationships.....	34
	5.4.1	Specialization Relationship.....	34
	5.4.2	Semantics of Other Relationships.....	35
	5.5	Relationship Connectors.....	35
	5.5.1	Junction.....	35
	5.6	Summary of Relationships.....	37
	5.7	Derivation of Relationships.....	38
6		Motivation Elements.....	40
	6.1	Motivation Elements Metamodel.....	40
	6.2	Stakeholder, Driver, and Assessment.....	41
	6.2.1	Stakeholder.....	41
	6.2.2	Driver.....	41
	6.2.3	Assessment.....	42
	6.2.4	Example.....	42
	6.3	Goal, Outcome, Principle, Requirement, and Constraint.....	43
	6.3.1	Goal.....	43
	6.3.2	Outcome.....	44
	6.3.3	Principle.....	44
	6.3.4	Requirement.....	45
	6.3.5	Constraint.....	45
	6.3.6	Example.....	46
	6.4	Meaning and Value.....	47
	6.4.1	Meaning.....	47
	6.4.2	Value.....	47
	6.4.3	Example.....	48
	6.5	Summary of Motivation Elements.....	49
	6.6	Relationships with Core Elements.....	50
7		Strategy Elements.....	51
	7.1	Strategy Elements Metamodel.....	51

7.2	Structure Elements	51
7.2.1	Resource	51
7.3	Behavior Elements	52
7.3.1	Capability	52
7.3.2	Value Stream	53
7.3.3	Course of Action	54
7.4	Example	54
7.5	Summary of Strategy Elements	56
7.6	Relationships with Motivation and Core Elements	57
8	Business Layer	58
8.1	Business Layer Metamodel	58
8.2	Active Structure Elements	58
8.2.1	Business Actor	59
8.2.2	Business Role	60
8.2.3	Business Collaboration	60
8.2.4	Business Interface	61
8.2.5	Example	61
8.3	Behavior Elements	62
8.3.1	Business Process	63
8.3.2	Business Function	64
8.3.3	Business Interaction	65
8.3.4	Business Event	65
8.3.5	Business Service	66
8.3.6	Example	66
8.4	Passive Structure Elements	67
8.4.1	Business Object	67
8.4.2	Contract	68
8.4.3	Representation	68
8.4.4	Example	69
8.5	Composite Elements	69
8.5.1	Product	70
8.5.2	Example	71
8.6	Summary of Business Layer Elements	71
9	Application Layer	73
9.1	Application Layer Metamodel	73
9.2	Active Structure Elements	73
9.2.1	Application Component	74
9.2.2	Application Collaboration	75
9.2.3	Application Interface	75
9.2.4	Example	76
9.3	Behavior Elements	76
9.3.1	Application Function	77
9.3.2	Application Interaction	77
9.3.3	Application Process	78
9.3.4	Application Event	78
9.3.5	Application Service	79
9.3.6	Example	79
9.4	Passive Structure Elements	80
9.4.1	Data Object	80

9.4.2	Example.....	81
9.5	Summary of Application Layer Elements.....	81
10	Technology Layer	83
10.1	Technology Layer Metamodel.....	83
10.2	Active Structure Elements	83
10.2.1	Node	84
10.2.2	Device.....	85
10.2.3	System Software.....	85
10.2.4	Technology Collaboration	86
10.2.5	Technology Interface.....	86
10.2.6	Path.....	87
10.2.7	Communication Network	87
10.2.8	Example.....	88
10.3	Behavior Elements.....	89
10.3.1	Technology Function.....	90
10.3.2	Technology Process.....	90
10.3.3	Technology Interaction.....	91
10.3.4	Technology Event.....	91
10.3.5	Technology Service.....	92
10.3.6	Example.....	92
10.4	Passive Structure Elements	93
10.4.1	Artifact	94
10.4.2	Example.....	94
10.5	Summary of Technology Layer Elements	95
11	Physical Elements	97
11.1	Physical Elements Metamodel.....	97
11.2	Active Structure Elements	97
11.2.1	Equipment	98
11.2.2	Facility.....	98
11.2.3	Distribution Network.....	99
11.3	Behavior Elements.....	99
11.4	Passive Structure Elements	99
11.4.1	Material	99
11.5	Example	100
11.6	Summary of Physical Elements	100
12	Relationships Between Core Layers	102
12.1	Alignment of the Business Layer and Lower Layers.....	102
12.2	Alignment of the Application and Technology Layers.....	103
12.3	Example	103
13	Implementation and Migration Elements.....	105
13.1	Implementation and Migration Elements Metamodel	105
13.2	Implementation and Migration Elements.....	105
13.2.1	Work Package.....	105
13.2.2	Deliverable	106
13.2.3	Implementation Event	106
13.2.4	Plateau	107
13.2.5	Gap	107

13.2.6	Example.....	108
13.2.7	Summary of Implementation and Migration Elements	108
13.3	Relationships.....	109
13.4	Relationships with Other Aspects and Layers	109
14	Stakeholders, Architecture Views, and Viewpoints.....	111
14.1	Introduction.....	111
14.2	Stakeholders and Concerns	111
14.3	Architecture Views and Viewpoints	112
14.4	Viewpoint Mechanism.....	113
14.4.1	Defining and Classifying Viewpoints	114
14.4.2	Creating the View.....	115
14.5	Example Viewpoints.....	115
15	Language Customization Mechanisms	116
15.1	Adding Attributes to ArchiMate Elements and Relationships.....	116
15.2	Specialization of Elements and Relationships	117
15.2.1	Examples of Specializations of Business Layer Elements (Informative).....	118
15.2.2	Examples of Specializations of Application Layer Elements (Informative).....	119
15.2.3	Examples of Specializations of Technology Layer Elements (Informative).....	119
15.2.4	Examples of Specializations of Physical Elements (Informative)	120
15.2.5	Examples of Specializations of Motivation Elements (Informative)	120
15.2.6	Examples of Specializations of Strategy Elements (Informative)	121
15.2.7	Examples of Specializations of Implementation and Migration Elements (Informative).....	122
15.2.8	Examples of Specializations of Composite Elements (Informative)	122
15.2.9	Examples of Specializations of Relationships (Informative)	123
A	Summary of Language Notation.....	124
A.1	Core Elements.....	125
A.2	Motivation, Strategy, Implementation and Migration Elements.....	126
A.3	Relationships.....	127
B	Relationships (Normative).....	128
B.1	Specification of Derivation Rules.....	128
B.2	Derivation Rules for Valid Relationships	128
B.2.1	Valid Derivations for Specialization Relationships	128
B.2.2	Valid Derivations for Structural Relationships	129
B.2.3	Valid Derivations for Dependency Relationships	130
B.2.4	Valid Derivations for Dynamic Relationships	131
B.3	Derivation Rules for Potential Relationships.....	133
B.3.1	Potential Derivation for Specialization Relationships.....	134

B.3.2	Potential Derivation for Structural and Dependency Relationships	137
B.3.3	Potential Derivation for Dependency Relationships	137
B.3.4	Potential Derivation for Dynamic Relationships.....	138
B.4	Restrictions on Applying Derivation Rules	139
B.5	Relationship Tables.....	140
B.6	Grouping, Plateau, and Relationships Between Relationships	151
C	Example Viewpoints	152
C.1	Basic Viewpoints in the ArchiMate Language	152
C.1.1	Organization Viewpoint	154
C.1.2	Application Structure Viewpoint.....	155
C.1.3	Information Structure Viewpoint	155
C.1.4	Technology Viewpoint	156
C.1.5	Layered Viewpoint	157
C.1.6	Physical Viewpoint.....	157
C.1.7	Product Viewpoint.....	158
C.1.8	Application Usage Viewpoint	159
C.1.9	Technology Usage Viewpoint	160
C.1.10	Business Process Cooperation Viewpoint.....	160
C.1.11	Application Cooperation Viewpoint.....	161
C.1.12	Service Realization Viewpoint.....	162
C.1.13	Implementation and Deployment Viewpoint	163
C.2	Motivation Viewpoints	163
C.2.1	Stakeholder Viewpoint	164
C.2.2	Goal Realization Viewpoint	165
C.2.3	Requirements Realization Viewpoint.....	165
C.2.4	Motivation Viewpoint	166
C.3	Strategy Viewpoints.....	167
C.3.1	Strategy Viewpoint.....	167
C.3.2	Capability Map Viewpoint	168
C.3.3	Value Stream Viewpoint	168
C.3.4	Outcome Realization Viewpoint	169
C.3.5	Resource Map Viewpoint	169
C.4	Implementation and Migration Viewpoints.....	170
C.4.1	Project Viewpoint.....	170
C.4.2	Migration Viewpoint	171
C.4.3	Implementation and Migration Viewpoint	171
D	Relationship to Other Standards, Specifications, and Guidance Documents.....	173
D.1	The TOGAF Framework	173
D.2	The BIZBOK Guide.....	174
D.3	BPMN	174
D.4	UML	175
D.5	BMM.....	176
E	Changes from Version 2.1 to Version 3.1.....	177
E.1	Changes from Version 2.1 to Version 3.0.1.....	177
E.2	Changes from Version 3.0.1 to Version 3.1.....	178

List of Figures

Figure 1: Top-Level Hierarchy of ArchiMate Concepts	7
Figure 2: ArchiMate Core Framework	8
Figure 3: ArchiMate Full Framework	9
Figure 4: Hierarchy of Behavior and Structure Elements	13
Figure 5: Behavior and Structure Elements Metamodel	14
Figure 6: Generic Internal Active Structure Element Notation	15
Figure 7: Generic External Active Structure Elements (Interface) Notation	15
Figure 8: Generic Internal Behavior Element Notation.....	15
Figure 9: Generic External Behavior Element (Service) Notation	15
Figure 10: Generic Event Notation.....	16
Figure 11: Generic Passive Structure Element Notation	16
Figure 12: Specializations of Core Elements	16
Figure 13: Generic Process Notation.....	17
Figure 14: Generic Function Notation.....	17
Figure 15: Generic Collaboration Notation	17
Figure 16: Generic Interaction Notation.....	17
Figure 17: Generic Motivation Element Notation	19
Figure 18: Composite Elements	20
Figure 19: Grouping Notation	20
Figure 20: Location Notation	21
Figure 21: Overview of Relationships.....	22
Figure 22: Composition Notation.....	24
Figure 23: Aggregation Notation.....	24
Figure 24: Assignment Notation	25
Figure 25: Realization Notation	26
Figure 26: Serving Notation	28
Figure 27: Access Notation	29
Figure 28: Influence Notation	31
Figure 29: Association Notation.....	31
Figure 30: Triggering Notation	33
Figure 31: Flow Notation	33
Figure 32: Specialization Notation.....	34
Figure 33: Junction Notation	35
Figure 34: Motivation Elements Metamodel.....	40
Figure 35: Stakeholder Notation	41
Figure 36: Driver Notation	42
Figure 37: Assessment Notation.....	42
Figure 38: Goal Notation.....	44
Figure 39: Outcome Notation.....	44
Figure 40: Principle Notation	45
Figure 41: Requirement Notation.....	45
Figure 42: Constraint Notation	46
Figure 43: Meaning Notation	47
Figure 44: Value Notation	48
Figure 45: Relationships Between Motivation Elements and Core Elements.....	50

Figure 46: Strategy Elements Metamodel 51

Figure 47: Resource Notation 52

Figure 48: Capability Notation 53

Figure 49: Value Stream Notation 54

Figure 50: Course of Action Notation 54

Figure 51: Relationships Between Strategy Elements and Motivation and Core Elements 57

Figure 52: Business Layer Metamodel 58

Figure 53: Business Internal Active Structure Elements 59

Figure 54: Business Actor Notation 60

Figure 55: Business Role Notation 60

Figure 56: Business Collaboration Notation 61

Figure 57: Business Interface Notation 61

Figure 58: Business Internal Behavior Elements 63

Figure 59: Business Process Notation 64

Figure 60: Business Function Notation 65

Figure 61: Business Interaction Notation 65

Figure 62: Business Event Notation 66

Figure 63: Business Service Notation 66

Figure 64: Business Passive Structure Elements 67

Figure 65: Business Object Notation 68

Figure 66: Contract Notation 68

Figure 67: Representation Notation 69

Figure 68: Product Metamodel 70

Figure 69: Product Notation 71

Figure 70: Application Layer Metamodel 73

Figure 71: Application Internal Active Structure Elements 74

Figure 72: Application Component Notation 75

Figure 73: Application Collaboration Notation 75

Figure 74: Application Interface Notation 76

Figure 75: Application Internal Behavior Elements 77

Figure 76: Application Function Notation 77

Figure 77: Application Interaction Notation 78

Figure 78: Application Process Notation 78

Figure 79: Application Event Notation 79

Figure 80: Application Service Notation 79

Figure 81: Data Object Notation 81

Figure 82: Technology Layer Metamodel 83

Figure 83: Technology Active Structure Elements 84

Figure 84: Node Notation 85

Figure 85: Device Notation 85

Figure 86: System Software Notation 86

Figure 87: Technology Collaboration Notation 86

Figure 88: Technology Interface Notation 87

Figure 89: Path Notation 87

Figure 90: Communication Network Notation 88

Figure 91: Technology Internal Behavior Elements 89

Figure 92: Technology Function Notation 90

Figure 93: Technology Process Notation 90

Figure 94: Technology Interaction Notation 91

Figure 95: Technology Event Notation 91

Figure 96: Technology Service Notation 92

Figure 97: Technology Passive Structure Elements	93
Figure 98: Artifact Notation	94
Figure 99: Physical Elements Metamodel	97
Figure 100: Equipment Notation	98
Figure 101: Facility Notation	98
Figure 102: Distribution Network Notation	99
Figure 103: Material Notation	99
Figure 104: Relationships Between Business Layer and Application and Technology Layer Elements	102
Figure 105: Relationships Between Application Layer and Technology Layer Elements	103
Figure 106: Implementation and Migration Metamodel	105
Figure 107: Work Package Notation	106
Figure 108: Deliverable Notation	106
Figure 109: Implementation Event Notation	107
Figure 110: Plateau Notation	107
Figure 111: Gap Notation	108
Figure 112: Relationships of Implementation and Migration Elements with Core Elements	109
Figure 113: Relationships of Implementation and Migration Elements with Motivation Elements	110
Figure 114: Conceptual Model of an Architecture Description (from [14])	112
Figure 115: Framing Stakeholder Concerns using the Viewpoint Mechanism	114
Figure 116: Correspondence Between the ArchiMate Language and the TOGAF ADM	173

List of Examples

Example 1: Grouping	21
Example 2: Composition	24
Example 3: Aggregation	25
Example 4: Assignment	26
Example 5: Realization	27
Example 6: Semantics of Structural Relationships	27
Example 7: Serving	29
Example 8: Access	29
Example 9: Influence	31
Example 10: Association	32
Example 11: Semantics of Dependency Relationships	33
Example 12: Triggering	33
Example 13: Flow	34
Example 14: Specialization	35
Example 15: (And) Junction	36
Example 16: Or Junction	36
Example 17: Derivation from a Chain of Relationships	38
Example 18: Stakeholder, Driver, and Assessment	43
Example 19: Goal, Outcome, Principle, Requirement, and Constraint	46
Example 20: Meaning and Value	48
Example 21: Capability, Resource, and Course of Action	55
Example 22: Value Stream with Capability Cross-Mapping	56
Example 23: Business Active Structure Elements	62
Example 24: Business Behavior Elements	67
Example 25: Business Passive Structure Elements	69
Example 26: Business Composite Element: Product	71
Example 27: Application Active Structure Elements	76
Example 28: Application Behavior Elements	80
Example 29: Application Passive Structure Elements	81
Example 30: Technology Active Structure Elements	89
Example 31: Technology Behavior Elements	93
Example 32: Technology Passive Structure Element: Artifact	94
Example 33: Physical Elements	100
Example 34: Cross-Layer Relationships	104
Example 35: Implementation and Migration Elements	108
Example 36: Specializations of Business Layer and Motivation Elements	121
Example 37: Transitivity of Specialization	129
Example 38: Derivation of Structural Relationships	129
Example 39: Derivation from a Chain of Structural Relationships	130
Example 40: Derivation from a Dependency and a Structural Relationship in Line	130
Example 41: Derivation from a Dependency and a Structural Relationship in the Opposite Direction	130
Example 42: Derivation from a Dynamic and a Structural Relationship in Line	131
Example 43: Derivation from a Flow and a Structural Relationship in the Opposite Direction	131
Example 44: Derivation from a Triggering and a Structural Relationship in Line	132

Example 45: Derivation from Dynamic Relationships.....	132
Example 46: Derivation from Triggering Relationships	132
Example 47: Derivation from Triggering and Structural Relationships.....	133
Example 48: Examples of Potential Derivation	134
Example 49: Potential Derivation from a Specialization and Another Relationship in Line	134
Example 50: Potential Derivation from a Specialization and Another Relationship in the Opposite Direction.....	135
Example 51: Potential Derivation from Another Relationship and a Specialization in Line	135
Example 52: Potential Derivation from Another Relationship and a Specialization in Line	136
Example 53: Specializations Used in Potential Derivations	136
Example 54: Potential Derivation from a Dependency and a Structural Relationship in Line ..	137
Example 55: Potential Derivation from a Dependency and a Structural Relationship in the Opposite Direction.....	137
Example 56: Potential Derivation from Two Dependency Relationships.....	138
Example 57: Potential Derivation from a Dynamic and a Structural Relationship in Line	138
Example 58: Potential Derivation from a Dynamic and a Structural Relationship in the Opposite Direction.....	138
Example 59: Potential Derivation from Two Flow Relationships	139
Example 60: Potential Derivation from a Triggering and Structural Relationships.....	139

List of Tables

Table 1: Core Elements	18
Table 2: Motivation Element.....	19
Table 3: Relationships	37
Table 4: Motivation Elements	49
Table 5: Strategy Elements.....	56
Table 6: Business Layer Elements	71
Table 7: Application Layer Elements.....	81
Table 8: Technology Layer Elements.....	95
Table 9: Physical Elements	100
Table 10: Implementation and Migration Elements	108
Table 11: Profile Example.....	117
Table 12: Example Specializations of Business Layer Elements	118
Table 13: Example Specializations of Application Layer Elements	119
Table 14: Example Specializations of Technology Layer Elements	119
Table 15: Example Specializations of Physical Elements.....	120
Table 16: Example Specializations of Motivation Elements.....	120
Table 17: Example Specializations of Strategy Elements	122
Table 18: Example Specializations of Implementation and Migration Elements	122
Table 19: Example Specializations of Composite Elements	122
Table 20: Example Specializations of Relationships	123
Table 21: Basic Viewpoints	153
Table 22: Organization Viewpoint Description	154
Table 23: Application Structure Viewpoint Description	155
Table 24: Information Structure Viewpoint Description.....	156
Table 25: Technology Viewpoint Description	156
Table 26: Layered Viewpoint Description	157
Table 27: Physical Viewpoint Description.....	157
Table 28: Product Viewpoint Description.....	158
Table 29: Application Usage Viewpoint Description.....	159
Table 30: Technology Usage Viewpoint Description	160
Table 31: Business Process Cooperation Viewpoint Description	161
Table 32: Application Cooperation Viewpoint Description.....	161
Table 33: Service Realization Viewpoint Description	162
Table 34: Implementation and Deployment Viewpoint Description.....	163
Table 35: Stakeholder Viewpoint Description	164
Table 36: Goal Realization Viewpoint Description	165
Table 37: Requirements Realization Viewpoint Description	165
Table 38: Motivation Viewpoint Description.....	166
Table 39: Strategy Viewpoint Description.....	167
Table 40: Capability Map Viewpoint Description	168
Table 41: Value Stream Viewpoint Description	168
Table 42: Outcome Realization Viewpoint Description.....	169
Table 43: Resource Map Viewpoint Description	169
Table 44: Project Viewpoint Description	170

Table 45: Migration Viewpoint Description 171
Table 46: Implementation and Migration Viewpoint Description 172

Preface

The Open Group

The Open Group is a global consortium that enables the achievement of business objectives through technology standards. Our diverse membership of more than 700 organizations includes customers, systems and solutions suppliers, tools vendors, integrators, academics, and consultants across multiple industries.

The mission of The Open Group is to drive the creation of Boundaryless Information Flow™ achieved by:

- Working with customers to capture, understand, and address current and emerging requirements, establish policies, and share best practices
- Working with suppliers, consortia, and standards bodies to develop consensus and facilitate interoperability, to evolve and integrate specifications and open source technologies
- Offering a comprehensive set of services to enhance the operational efficiency of consortia
- Developing and operating the industry's premier certification service and encouraging procurement of certified products

Further information on The Open Group is available at www.opengroup.org.

The Open Group publishes a wide range of technical documentation, most of which is focused on development of Open Group Standards and Guides, but which also includes white papers, technical studies, certification and testing documentation, and business titles. Full details and a catalog are available at www.opengroup.org/library.

This Document

This document is the ArchiMate® 3.1 Specification, a standard of The Open Group. It has been developed and approved by The Open Group.

This edition of the standard includes a number of corrections, clarifications, and improvements to the previous edition, as well as several additions.

Intended Audience

The intended audience of this standard is threefold:

- All those working to shape and implement complex organization change
Typical job titles include Enterprise Architecture practitioners, Business Architects, IT architects, application architects, data architects, information architects, process architects, infrastructure architects, software architects, systems architects, solutions architects,

product/service managers, senior and operational management, project leaders, and anyone working within the reference framework defined by an Enterprise Architecture.

- Those who intend to implement the ArchiMate language in a software tool
They will find a complete and detailed description of the language in this document.
- The academic community, on which we rely for amending and improving the language based on state-of-the-art research in the architecture field

Structure

The structure of this standard is as follows:

- Chapter 1, Introduction, provides the introduction to this standard, including the objectives, a brief overview, conformance requirements, and terminology
- Chapter 2, Definitions, defines the general terms used in this standard
- Chapter 3, Language Structure, describes the structure of the ArchiMate modeling language, including the top-level structure, layering, the ArchiMate Core Framework, and the ArchiMate Full Framework
- Chapter 4, Generic Metamodel, describes the structure and elements of the ArchiMate generic metamodel
- Chapter 5, Relationships, describes the relationships in the language
- Chapter 6, Motivation Elements, describes the concepts for expressing the motivation for an architecture, together with examples
- Chapter 7, Strategy Elements, provides elements for modeling the enterprise at a strategic level, together with examples
- Chapter 8, Business Layer, covers the definition and usage of the Business Layer elements, together with examples
- Chapter 9, Application Layer, covers the definition and usage of the Application Layer elements, together with examples
- Chapter 10, Technology Layer, covers the definition and usage of the Technology Layer elements, together with examples
- Chapter 11, Physical Elements, describes the language elements for modeling the physical world, together with examples
- Chapter 12, Relationships Between Core Layers, covers the relationships between different layers of the language
- Chapter 13, Implementation and Migration Elements, describes the language elements for expressing the implementation and migration aspects of an architecture (e.g., projects, programs, plateaus, and gaps)
- Chapter 14, Stakeholders, Architecture Views, and Viewpoints, describes the ArchiMate viewpoint mechanism

- Chapter 15, Language Customization Mechanisms, describes how to customize the ArchiMate language for specialized or domain-specific purposes
- Appendix A, Summary of Language Notation, is an informative appendix
- Appendix B, Relationships, is a normative appendix detailing the required relationships between elements of the language and the rules to derive these
- Appendix C, Example Viewpoints, presents a set of architecture viewpoints, developed in ArchiMate notation based on practical experience
All viewpoints are described in detail. The appendix specifies the elements, relationships, usage guidelines, goals, and target groups for each viewpoint.
- Appendix D, Relationship to Other Standards, Specifications, and Guidance Documents, describes the relationships of the ArchiMate language to other standards and specifications, including the TOGAF® framework, the BIZBOK® Guide, BPMN™, UML®, and BMM™
- Appendix E, Changes from Version 2.1 to Version 3.1, is an informative appendix outlining the changes in the standard between Version 2.1 and Version 3.1

Trademarks

ArchiMate[®], DirecNet[®], Making Standards Work[®], Open O[®] logo, Open O and Check[®] Certification logo, OpenPegasus[®], Platform 3.0[®], The Open Group[®], TOGAF[®], UNIX[®], UNIXWARE[®], and the Open Brand X[®] logo are registered trademarks and Agile Architecture Framework[™], Boundaryless Information Flow[™], Build with Integrity Buy with Confidence[™], Dependability Through Assuredness[™], Digital Practitioner Body of Knowledge[™], DPBoK[™], EMMM[™], FACE[™], the FACE[™] logo, IT4IT[™], the IT4IT[™] logo, O-AAF[™], O-DEF[™], O-HERA[™], O-PAS[™], Open FAIR[™], Open Platform 3.0[™], Open Process Automation[™], Open Subsurface Data Universe[™], Open Trusted Technology Provider[™], O-SDU[™], Sensor Integration Simplified[™], SOSA[™], and the SOSA[™] logo are trademarks of The Open Group.

A Guide to the Business Architecture Body of Knowledge[®] and BIZBOK[®] are registered trademarks of the Business Architecture Guild.

Java[®] is a registered trademark of Oracle and/or its affiliates.

UML[®] and Unified Modeling Language[®] are registered trademarks and BMM[™], BPMN[™], Business Motivation Model[™], and Business Process Modeling Notation[™] are trademarks of the Object Management Group.

All other brands, company, and product names are used for identification purposes only and may be trademarks that are the sole property of their respective owners.

Acknowledgements

The Open Group gratefully acknowledges The Open Group ArchiMate Forum for developing this standard.

The Open Group gratefully acknowledges the contribution of the following people in the development of this and earlier versions of this standard:

- Iver Band, EA Principals & Cambia Health Solutions
- Thorbjørn Ellefsen, Capgemini
- William Estrem, Metaplexity Associates
- Maria-Eugenia Iacob, University of Twente
- Henk Jonkers, BiZZdesign
- Marc M. Lankhorst, BiZZdesign
- Dag Nilsen, Biner
- Carlo Poli, Macaw
- Erik (H.A.) Proper, Luxembourg Institute for Science and Technology & Radboud University Nijmegen
- Dick A.C. Quartel, BiZZdesign
- G. Edward Roberts, Elparazim
- Jean-Baptiste Sarrodie, Accenture
- Serge Thorn, Metaplexity Fellow

The Open Group and ArchiMate project team would like to thank in particular the following individuals for their support and review of this and earlier versions of this standard:

- Adina Aldea
- Mary Beijleveld
- Alexander Bielowski
- Remco de Boer
- Adrian Campbell
- John Coleshaw
- Jörgen Dahlberg
- Garry Doherty

- Ingvar Elmér
- Wilco Engelsman
- Roland Ettema
- Henry M. Franken
- Mats Gejnevall
- Sonia González
- Kirk Hansen
- Jos van Hillegersberg
- Andrew Josey
- Ryan Kennedy
- Louw Labuschagne
- Antoine Lonjon
- Veer Muchandi
- Michelle Nieuwoudt
- Erwin Oord
- Antonio Plais, Centus
- Daniel Simon
- Gerben Wierda
- Egon Willemsz

The first version of this standard was largely produced by the ArchiMate project. The Open Group gratefully acknowledges the contribution of the many people – former members of the project team – who have contributed to it.

The ArchiMate project comprised the following organizations:

- ABN AMRO
- Centrum voor Wiskunde en Informatica
- Dutch Tax and Customs Administration
- Leiden Institute of Advanced Computer Science
- Novay
- Ordina
- Radboud Universiteit Nijmegen
- Stichting Pensioenfonds ABP

Referenced Documents

The following documents are referenced in this standard. These references are informative.

(Please note that the links below are good at the time of writing but cannot be guaranteed for the future.)

- [1] Enterprise Architecture at Work: Modeling, Communication, and Analysis, Third Edition, M.M. Lankhorst et al., Springer, 2013.
- [2] The Anatomy of the ArchiMate[®] Language, M.M. Lankhorst, H.A. Proper, H. Jonkers, International Journal of Information Systems Modeling and Design (IJISMD), 1(1):1-32, January-March 2010.
- [3] Extending Enterprise Architecture Modeling with Business Goals and Requirements, W. Engelsman, D.A.C. Quartel, H. Jonkers, M.J. van Sinderen, Enterprise Information Systems, 5(1):9-36, 2011.
- [4] TOGAF[®] Version 9.2, The Open Group Standard (C182), April 2018, published by The Open Group; refer to: www.opengroup.org/library/c182.
- [5] Extending and Formalizing the Framework for Information Systems Architecture, J.F. Sowa, J.A. Zachman, IBM Systems Journal, Volume 31, No. 3, pp.590-616, 1992.
- [6] TOGAF[®] Framework and ArchiMate[®] Modeling Language Harmonization: A Practitioner's Guide to Using the TOGAF[®] Framework and the ArchiMate[®] Language, White Paper (W14C), December 2014, published by The Open Group; refer to: www.opengroup.org/library/w14c.
- [7] Unified Modeling Language[®]: Superstructure, Version 2.0 (formal/05-07-04), Object Management Group, August 2005.
- [8] Unified Modeling Language[®]: Infrastructure, Version 2.4.1 (formal/201-08-05), Object Management Group, August 2011.
- [9] A Business Process Design Language, H. Eertink, W. Janssen, P. Oude Luttighuis, W. Teeuw, C. Vissers, in Proceedings of the First World Congress on Formal Methods, Toulouse, France, September 1999.
- [10] Enterprise Business Architecture: The Formal Link Between Strategy and Results, R. Whittle, C.B. Myrick, CRC Press, 2004.
- [11] Composition of Relations in Enterprise Architecture, R. van Buuren, H. Jonkers, M.E. Iacob, P. Strating, in Proceedings of the Second International Conference on Graph Transformation, pp.39-53, edited by H. Ehrig et al., Rome, Italy, 2004.
- [12] Business Process Modeling Notation[™] (BPMN[™]), Version 2.0 (formal/2011-01-03), Object Management Group, 2011.

- [13] Performance and Cost Analysis of Service-Oriented Enterprise Architectures, H. Jonkers, M.E. Iacob, in *Global Implications of Modern Enterprise Information Systems: Technologies and Applications*, edited by A. Gunasekaran, IGI Global, 2009.
- [14] ISO/IEC 42010:2011, *Systems and Software Engineering – Recommended Practice for Architectural Description of Software-Intensive Systems*, Edition 1.
- [15] Business Motivation Model™ (BMM™), Version 1.1 (formal/2010-05-01), Object Management Group, 2010.
- [16] Using the ArchiMate® Language with UML®, White Paper (W134), September 2013, published by The Open Group; refer to: www.opengroup.org/library/w134.
- [17] TOGAF® Series Guide: Value Streams (G178), October 2017, published by The Open Group; refer to: www.opengroup.org/library/g178.
- [18] Business Architecture Guild. A Guide to the Business Architecture Body of Knowledge® (BIZBOK® Guide), Version 7.0, 2018; refer to: www.businessarchitectureguild.org.
- [19] TOGAF® Series Guide: The TOGAF® Technical Reference Model (TRM) (G175), September 2017, published by The Open Group; refer to: www.opengroup.org/library/g175.
- [20] ArchiMate® Model Exchange File Format for the ArchiMate Modeling Language, Version 3.0, The Open Group Standard (C174), May 2017, published by The Open Group; refer to: www.opengroup.org/library/c174.

1 Introduction

1.1 Objective

This standard is the specification of the ArchiMate Enterprise Architecture modeling language, a visual language with a set of default iconography for describing, analyzing, and communicating many concerns of Enterprise Architectures as they change over time. The standard provides a set of entities and relationships with their corresponding iconography for the representation of Architecture Descriptions. The ArchiMate ecosystem also supports an exchange format in XML which allows model and diagram exchange between tools [20].

1.2 Overview

An Enterprise Architecture is typically developed because key people have concerns that need to be addressed by the business and IT systems within an organization. Such people are commonly referred to as the “stakeholders” of the Enterprise Architecture. The role of the architect is to address these concerns by identifying and refining the motivation and strategy expressed by stakeholders, developing an architecture, and creating views of the architecture that show how it addresses and balances stakeholder concerns. Without an Enterprise Architecture, it is unlikely that all concerns and requirements are considered and addressed.

The ArchiMate Enterprise Architecture modeling language provides a uniform representation for diagrams that describe Enterprise Architectures. It includes concepts for specifying inter-related architectures, specific viewpoints for selected stakeholders, and language customization mechanisms. It offers an integrated architectural approach that describes and visualizes different architecture domains and their underlying relations and dependencies. Its language framework provides a structuring mechanism for architecture domains, layers, and aspects. It distinguishes between the model elements and their notation, to allow for varied, stakeholder-oriented depictions of architecture information. The language uses service-orientation to distinguish and relate the Business, Application, and Technology Layers of Enterprise Architectures, and uses realization relationships to relate concrete elements to more abstract elements across these layers.

1.3 Conformance

The ArchiMate language may be implemented in software used for Enterprise Architecture modeling. For the purposes of this standard, the conformance requirements for implementations of the language given in this section apply. A conforming implementation:

1. Shall support the language structure, generic metamodel, relationships, layers, cross-layer dependencies, and other elements as specified in Chapters 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, and 13
2. Shall support the standard iconography as specified in Chapters 4, 5, 6, 7, 8, 9, 10, 11, and 13, and summarized in Appendix A

3. Shall support the viewpoint mechanism as specified in Chapter 14
4. Shall support the language customization mechanisms as specified in Chapter 15 in an implementation-defined manner
5. Shall support the relationships between elements as specified in Appendix B
6. May support the example viewpoints described in Appendix C

Readers are advised to check The Open Group website for additional conformance and certification requirements referencing this standard.

1.4 Normative References

None.

1.5 Terminology

For the purposes of this standard, the following terminology definitions apply:

Can Describes a possible feature or behavior available to the user.

Deprecated Items identified as deprecated may be removed in the next version of this standard.

Implementation-defined

Describes a value or behavior that is not defined by this standard but is selected by an implementor of a software tool. The value or behavior may vary among implementations that conform to this standard. A user should not rely on the existence of the value or behavior. The implementor shall document such a value or behavior so that it can be used correctly by a user.

May Describes a feature or behavior that is optional. To avoid ambiguity, the opposite of “may” is expressed as “need not”, instead of “may not”.

Obsolescent Certain features are obsolescent, which means that they may be considered for withdrawal in future versions of this standard. They are retained because of their widespread use, but their use is discouraged.

Shall Describes a feature or behavior that is a requirement. To avoid ambiguity, do not use “must” as an alternative to “shall”.

Shall not Describes a feature or behavior that is an absolute prohibition.

Should Describes a feature or behavior that is recommended but not required.

Will Same meaning as “shall”; “shall” is the preferred term.

1.6 Future Directions

None.

2 Definitions

For the purposes of this standard, the following terms and definitions apply. The TOGAF® framework [4] should be referenced for Enterprise Architecture-related terms not defined in this chapter. Merriam-Webster’s Collegiate Dictionary (11th Edition) should be referenced for all other terms not defined in this chapter.

Any conflict between definitions described here and the TOGAF framework is unintentional. If the definition of a term is specific to the ArchiMate modeling language, and a general definition is defined by the TOGAF framework, then this is noted in the definition.

2.1 ArchiMate Core Framework

A reference structure used to classify elements of the ArchiMate core language. It consists of three layers and three aspects.

Note: The ArchiMate Core Framework is defined in detail in Section 3.4.

2.2 ArchiMate Core Language

The central part of the ArchiMate language that defines the concepts to model Enterprise Architectures. It includes concepts from three layers: Business, Application, and Technology (including Physical).

2.3 Architecture View

A representation of a system from the perspective of a related set of concerns.

Note: In some sections of this standard, the term “view” is used as a synonym for “architecture view”.

2.4 Architecture Viewpoint

A specification of the conventions for a particular kind of architecture view.

Note: In some sections of this standard, the term “viewpoint” is used as a synonym for “architecture viewpoint”.

2.5 Aspect

Classification of elements based on layer-independent characteristics related to the concerns of different stakeholders. Used for positioning elements in the ArchiMate metamodel. See also Section 2.9.