

An **OMG**<sup>®</sup> Unified Architecture Framework<sup>™</sup> Publication



# Unified Architecture Framework Profile (UAFP)

Version 1.1

OMG Document Number: [dtc/2019-06-13](#)

Publication Date: [May 2019](#)

Normative reference: <http://www.omg.org/spec/UAF/1.1/>

Normative Machine readable file(s):

<http://www.omg.org/spec/UAF/20190619/UAF.xmi>

<http://www.omg.org/spec/UAF/20190620/Measurements->

[Library.xmi](#)

Style Definition: OMG Normal Paragraph

Deleted: 0

Deleted: formal

Deleted: 7

Deleted: 12-

Deleted: 01

Deleted: November

Deleted: 7

Deleted: [0](#)

Field Code Changed

Deleted: [7](#)

Deleted: [501](#)

Deleted: [P\\_Profile](#)

Field Code Changed

Deleted: [7](#)

Deleted: [516](#)

Deleted: [Class](#)

Deleted: [-UAF](#)

Copyright © 2019, IBM  
Copyright © 2019, KDM Analytics  
Copyright © 2019, Mega  
Copyright © 2019, Object Management Group, Inc.  
Copyright © 2019, No Magic Inc. a Dassault Systèmes Company  
Copyright © 2019, PTC  
Copyright © 2019, Sparx Systems

- Deleted: 7
- Deleted: 7
- Deleted: 7
- Formatted: Indent: First line: 0.18 cm
- Deleted: 7
- Formatted: Right: 0 cm, Line spacing: single
- Deleted: 7
- Deleted: 7
- Formatted: Font: 12 pt, English (UK)
- Formatted: Indent: Left: 0 cm, First line: 0.18 cm
- Deleted: 7
- Deleted: 7

### USE OF SPECIFICATION - TERMS, CONDITIONS & NOTICES

The material in this document details an Object Management Group specification in accordance with the terms, conditions and notices set forth below. This document does not represent a commitment to implement any portion of this specification in any company's products. The information contained in this document is subject to change without notice.

### LICENSES

The companies listed above have granted to the Object Management Group, Inc. (OMG) a nonexclusive, royalty-free, paid up, worldwide license to copy and distribute this document and to modify this document and distribute copies of the modified version. Each of the copyright holders listed above has agreed that no person shall be deemed to have infringed the copyright in the included material of any such copyright holder by reason of having used the specification set forth herein or having conformed any computer software to the specification.

Subject to all of the terms and conditions below, the owners of the copyright in this specification hereby grant you a fully-paid up, non-exclusive, nontransferable, perpetual, worldwide license (without the right to sublicense), to use this specification to create and distribute software and special purpose specifications that are based upon this specification, and to use, copy, and distribute this specification as provided under the Copyright Act; provided that: (1) both the copyright notice identified above and this permission notice appear on any copies of this specification; (2) the use of the specifications is for informational purposes and will not be copied or posted on any network computer or broadcast in any media and will not be otherwise resold or transferred for commercial purposes; and (3) no modifications are made to this specification. This limited permission automatically terminates without notice if you breach any of these terms or conditions. Upon termination, you will destroy immediately any copies of the specifications in your possession or control.

### PATENTS

The attention of adopters is directed to the possibility that compliance with or adoption of OMG specifications may require use of an invention covered by patent rights. OMG shall not be responsible for identifying patents for which a license may be required by any OMG specification, or for conducting legal inquiries into the legal validity or scope of those patents that are brought to its attention. OMG specifications are prospective and advisory only. Prospective users are responsible for protecting themselves against liability for infringement of patents.

### GENERAL USE RESTRICTIONS

Any unauthorized use of this specification may violate copyright laws, trademark laws, and communications regulations and statutes. This document contains information which is protected by copyright. All Rights Reserved. No part of this work covered by copyright herein may be reproduced or used in any form or by any means--graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems--without permission of the copyright owner.

## DISCLAIMER OF WARRANTY

WHILE THIS PUBLICATION IS BELIEVED TO BE ACCURATE, IT IS PROVIDED "AS IS" AND MAY CONTAIN ERRORS OR MISPRINTS. THE OBJECT MANAGEMENT GROUP AND THE COMPANIES LISTED ABOVE MAKE NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS PUBLICATION, INCLUDING BUT NOT LIMITED TO ANY WARRANTY OF TITLE OR OWNERSHIP, IMPLIED WARRANTY OF MERCHANTABILITY OR WARRANTY OF FITNESS FOR A PARTICULAR PURPOSE OR USE. IN NO EVENT SHALL THE OBJECT MANAGEMENT GROUP OR ANY OF THE COMPANIES LISTED ABOVE BE LIABLE FOR ERRORS CONTAINED HEREIN OR FOR DIRECT, INDIRECT, INCIDENTAL, SPECIAL, CONSEQUENTIAL, RELIANCE OR COVER DAMAGES, INCLUDING LOSS OF PROFITS, REVENUE, DATA OR USE, INCURRED BY ANY USER OR ANY THIRD PARTY IN CONNECTION WITH THE FURNISHING, PERFORMANCE, OR USE OF THIS MATERIAL, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

The entire risk as to the quality and performance of software developed using this specification is borne by you. This disclaimer of warranty constitutes an essential part of the license granted to you to use this specification.

## RESTRICTED RIGHTS LEGEND

Use, duplication or disclosure by the U.S. Government is subject to the restrictions set forth in subparagraph (c) (1) (ii) of The Rights in Technical Data and Computer Software Clause at DFARS 252.227-7013 or in subparagraph (c) (1) and (2) of the Commercial Computer Software - Restricted Rights clauses at 48 C.F.R. 52.227-19 or as specified in 48 C.F.R. 227-7202-2 of the DoD F.A.R. Supplement and its successors, or as specified in 48 C.F.R. 12.212 of the Federal Acquisition Regulations and its successors, as applicable. The specification copyright owners are as indicated above and may be contacted through the Object Management Group, 109 Highland Avenue, Needham, MA 02494, U.S.A.

## TRADEMARK S

CORBA®, CORBA logos®, FIBO®, Financial Industry Business Ontology®, FINANCIAL INSTRUMENT GLOBAL IDENTIFIER®, IIOP®, IMM®, Model Driven Architecture®, MDA®, Object Management Group®, OMG®, OMG Logo®, SoaML®, SOAML®, SysML®, UAF®, Unified Modeling Language®, UML®, UML Cube Logo®, VSIPL®, and XMI® are registered trademarks of the Object Management Group, Inc.

For a complete list of trademarks, see: [http://www.omg.org/legal/tm\\_list.htm](http://www.omg.org/legal/tm_list.htm). All other products or company names mentioned are used for identification purposes only, and may be trademarks of their respective owners.

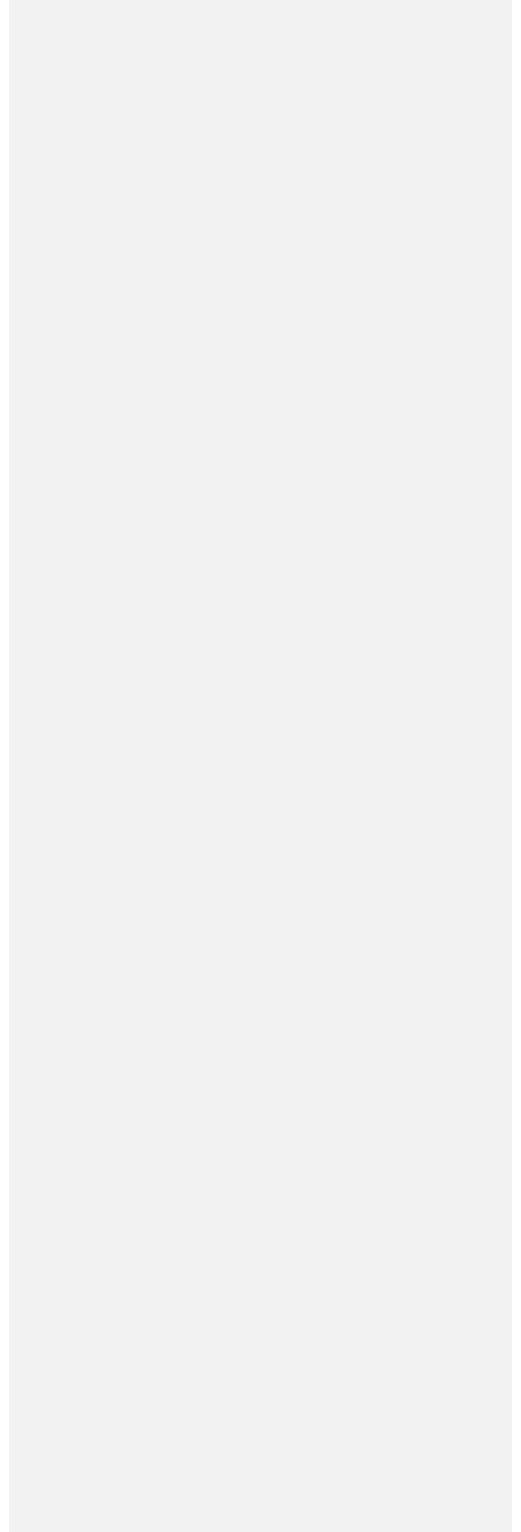
## COMPLIANC E

The copyright holders listed above acknowledge that the Object Management Group (acting itself or through its designees) is and shall at all times be the sole entity that may authorize developers, suppliers and sellers of computer software to use certification marks, trademarks or other special designations to indicate compliance with these materials.

Software developed under the terms of this license may claim compliance or conformance with this specification if and only if the software compliance is of a nature fully matching the applicable compliance points as stated in the specification. Software developed only partially matching the applicable compliance points may claim only that the software was based on this specification, but may not claim compliance or conformance with this specification. In the event that testing suites are implemented or approved by Object Management Group, Inc., software developed using this specification may claim compliance or conformance with the specification only if the software satisfactorily completes the testing suites.

## **OMG's Issue Reporting Procedure**

All OMG specifications are subject to continuous review and improvement. As part of this process we encourage readers to report any ambiguities, inconsistencies, or inaccuracies they may find by completing the Issue Reporting Form listed on the main web page <https://www.omg.org>, under OMG Specifications, Report an Issue.



## Table of Contents

<b>1 Scope</b>	<b>1</b>
1.1 UA FP Background	1
1.2 Intended Users	1
1.3 Related Documents	2
<b>2 Conformance</b>	<b>3</b>
<b>3 References</b>	<b>4</b>
3.1 Normative	4
3.2 OMG Documents (Normative References)	4
3.3 Other Normative References	4
3.4 Informative References	5
<b>4 Terms and Definitions</b>	<b>5</b>
<b>5 Symbols</b>	<b>6</b>
<b>6 Additional Information</b>	<b>8</b>
6.1 Changes to Adopted OMG Specifications	8
6.2 Language Architecture	8
6.3 Philosophy	8
6.4 Core Principals	9
6.5 Representing Stereotype Constraints	9
6.5.1 Metaconstraint dependency	10
6.5.2 Metarelationship dependency	11
6.5.3 Stereotyped relationship dependency	12
<b>7 UA F Profile</b>	<b>15</b>
7.1 UA F	15
7.1.1 UA F::Dictionary	15
7.1.2 UA F::Parameters	17
7.1.3 UA F::Metadata	35
7.1.3.1 UA F::Metadata:Taxonomy	35
7.1.3.2 UA F::Metadata:Connectivity	37
7.1.3.3 UA F::Metadata:Processes	38
7.1.3.4 UA F::Metadata:Information	43
7.1.3.5 UA F::Metadata:Constraints	46
7.1.3.6 UA F::Metadata:Traceability	48
7.1.4 UA F::Strategic	50
7.1.4.1 UA F::Strategic:Taxonomy	50
7.1.4.2 UA F::Strategic:Structure	58
7.1.4.3 UA F::Strategic:Processes	60
7.1.4.4 UA F::Strategic:States	63
7.1.4.5 UA F::Strategic:Traceability	66
7.1.5 UA F::Operational	69
7.1.5.1 UA F::Operational:Taxonomy	69
7.1.5.2 UA F::Operational:Structure	73
7.1.5.3 UA F::Operational:Connectivity	81
7.1.5.4 UA F::Operational:Processes	88
7.1.5.5 UA F::Operational:States	92
7.1.5.6 UA F::Operational:Interaction Scenarios	93
7.1.5.7 UA F::Operational:Information	94
7.1.5.8 UA F::Operational:Constraints	95
7.1.6 UA F::Services	97
7.1.6.1 UA F::Services:Taxonomy	97
7.1.6.2 UA F::Services:Structure	98
7.1.6.3 UA F::Services:Connectivity	102

7.1.6.4UAF::Services::Processes.....	104
7.1.6.5UAF::Services::States.....	106
7.1.6.6UAF::Services::InteractionScenarios.....	107
7.1.6.7UAF::Services::Constraints.....	108
7.1.6.8UAF::Services::Traceability.....	108
7.1.7UAF::Personnel.....	109
7.1.7.1UAF::Personnel::Taxonomy.....	109
7.1.7.2UAF::Personnel::Connectivity.....	113
7.1.7.3UAF::Personnel::Processes.....	114
7.1.7.4UAF::Personnel::Constraints.....	115
7.1.7.5UAF::Personnel::Traceability.....	118
7.1.8UAF::Resources.....	119
7.1.8.1UAF::Resources::Taxonomy.....	119
7.1.8.2UAF::Resources::Structure.....	125
7.1.8.3UAF::Resources::Connectivity.....	130
7.1.8.4UAF::Resources::Processes.....	136
7.1.8.5UAF::Resources::States.....	141
7.1.8.6UAF::Resources::InteractionScenarios.....	142
7.1.8.7UAF::Resources::Information.....	143
7.1.8.8UAF::Resources::Constraints.....	144
7.1.8.9UAF::Resources::Roadmap.....	146
7.1.8.10UAF::Resources::Traceability.....	152
7.1.9UAF::Security.....	153
7.1.9.1UAF::Security::Taxonomy.....	153
7.1.9.2UAF::Security::Structure.....	156
7.1.9.3UAF::Security::Processes.....	158
7.1.9.4UAF::Security::Constraints.....	163
7.1.9.5UAF::Security::Traceability.....	168
7.1.10UAF::Project.....	173
7.1.10.1UAF::Project::Taxonomy.....	173
7.1.10.2UAF::Project::Structure.....	176
7.1.10.3UAF::Project::Connectivity.....	180
7.1.10.4UAF::Project::Processes.....	182
7.1.10.5UAF::Project::Roadmap.....	183
7.1.11UAF::Standards.....	187
7.1.11.1UAF::Standards::Taxonomy.....	187
7.1.11.2UAF::Standards::Structure.....	190
7.1.12UAF::ActualResources.....	191
7.1.12.1UAF::ActualResources::Taxonomy.....	191
7.1.12.2UAF::ActualResources::Structure.....	199
7.1.12.3UAF::ActualResources::Connectivity.....	200
7.1.12.4UAF::ActualResources::Constraints.....	202
7.1.12.5UAF::ActualResources::Traceability.....	205
7.1.13UAF::SummaryandOverview.....	206
<b>AnnexA:UAFViews(Profile).....</b>	<b>215</b>
A.1General.....	215
A.2ViewSpecifications.....	215
A.2.1ViewSpecifications::Strategic.....	215
A.2.2ViewSpecifications::Operational.....	223
A.2.2ViewSpecifications::Services.....	233
A.2.3ViewSpecifications::Personnel.....	246
A.2.3ViewSpecifications::Resources.....	258
A.2.4ViewSpecifications::Security.....	276
A.2.4ViewSpecifications::Projects.....	279
A.2.5ViewSpecifications::Standards.....	287
A.2.6ViewSpecifications::ActualResources.....	291

A2.7 View Specifications :Dictionary .....	294
A.2.8 View Specifications :Requirements .....	295
A2.9 View Specifications::Summary & Overview .....	295
A2.10 View Specifications :Information .....	297
A2.11 View Specifications::Parameters .....	297
Annex B: Class Librarv.....	301
B.1 Class Libra .....	301

## List of Figures

	Figure 6.1 - MapsToCapability	
Stereotype.....		10
	Figure 6.2 - Connector	
Extension.....		11
	Figure 6.3 - Capabilities	
Generalization.....		11
	Figure 6.4 - Visualizing	
<<metarelationshps>>.....		12
	Figure 6.5 - Use of the AchievedEffect <<stereotyped relationship>> dependency.....	13
	Figure 7.1 -	
Alias.....		15
	Figure 7.2 -	
Definition.....		16
	Figure 7.3 -	
SameAs.....		17
	Figure 7.4 -	
ActualCondition.....		18
	Figure 7.5 -	
ActualEnvironment.....		18
	Figure 7.6 -	
ActualLocation.....		19
	Figure 7.7 -	
ActualMeasurement.....		20
	Figure 7.8 -	
ActualMeasurementSet.....		22
	Figure 7.9 -	
ActualPropertySet.....		23
	Figure 7.10 -	
Condition.....		24
	Figure 7.11 -	
Environment.....		24
	Figure 7.12 -	
EnvironmentProperty.....		26
	Figure 7.13 -	
GeoPoliticalExtentType.....		27
	Figure 7.14 -	
Location.....		29
	Figure 7.15 -	
LocationHolder.....		30
	Figure 7.16 -	
MeasurableElement.....		32
	Figure 7.17 -	
Measurement.....		33
	Figure 7.18 -	
MeasurementSet.....		34



PropertySet.....	Figure 7.19 -	35
ActualState.....	Figure 7.20 -	36
ISO8601DateTime.....	Figure 7.21 -	36
Exchange.....	Figure 7.22 -	37
Resource.....	Figure 7.23 -	38
Activity.....	Figure 7.24 -	39
CapableElement.....	Figure 7.25 -	40
IsCapableToPerform.....	Figure 7.26 -	41
PerformsInContext.....	Figure 7.27 -	42
ArchitectureMetadata.....	Figure 7.28 -	43
Information.....	Figure 7.29 -	44
Metadata.....	Figure 7.30 -	45
Rule.....	Figure 7.31 -	47
ArchitectureReference.....	Figure 7.32 -	48
Implements.....	Figure 7.33 -	49
ActualEnterprisePhase.....	Figure 7.34 -	51
Capability.....	Figure 7.35 -	53
EnterpriseGoal.....	Figure 7.36 -	54
EnterprisePhase.....	Figure 7.37 -	55
EnterpriseVision.....	Figure 7.38 -	56
VisionStatement.....	Figure 7.39 -	57
WholeLifeEnterprise.....	Figure 7.40 -	57
CapabilityProperty.....	Figure 7.41 -	58
StructuralPart.....	Figure 7.42 -	59
TemporalPart.....	Figure 7.43 -	60



Figure 7.44 - ActualEnduringTask.....	61
Figure 7.45 - CapabilityForTask.....	62
Figure 7.46 - EnduringTask.....	63
Figure 7.47 - AchievedEffect.....	63
Figure 7.48 - Achiever.....	64
Figure 7.49 - DesiredEffect.....	65
Figure 7.50 - Desirer.....	66
Figure 7.51 - Exhibits.....	67
Figure 7.52 - MapsToCapability.....	68
Figure 7.53 - OrganizationInEnterprise.....	69
Figure 7.54 - ArbitraryConnector.....	70
Figure 7.55 - ConceptItem.....	71
Figure 7.56 - ConceptRole.....	72
Figure 7.57 - HighLevelOperationalConcept.....	73
Figure 7.58 - KnownResource.....	73
Figure 7.59 - OperationalAgent.....	74
Figure 7.60 - OperationalArchitecture.....	75
Figure 7.61 - OperationalMethod.....	76
Figure 7.62 - OperationalParameter.....	77
Figure 7.63 - OperationalPerformer.....	78
Figure 7.64 - OperationalPort.....	79
Figure 7.65 - OperationalRole.....	80
Figure 7.66 - ProblemDomain.....	81
Figure 7.67 - OperationalConnector.....	82
Figure 7.68 - OperationalExchange.....	83

Figure 7.69 - OperationalExchangeItem.....	84
Figure 7.70 - OperationalInterface.....	86
Figure 7.71 - OperationalSignal.....	87
Figure 7.72 - OperationalSignalProperty.....	87
Figure 7.73 - OperationalActivity.....	88
Figure 7.74 - OperationalActivityAction.....	89
Figure 7.75 - OperationalEdge.....	90
Figure 7.76 - OperationalControlFlow.....	91
Figure 7.77 - OperationalObjectFlow.....	92
Figure 7.78 - StandardOperationalActivity.....	92
Figure 7.79 - OperationalStateDescription.....	93
Figure 7.80 - OperationalMessage.....	94
Figure 7.81 - InformationElement.....	95
Figure 7.82 - OperationalConstraint.....	96
Figure 7.83 - SubjectOfOperationalConstraint.....	97
Figure 7.84 - ServiceSpecification.....	98
Figure 7.85 - ServiceMethod.....	99
Figure 7.86 - ServiceParameter.....	100
Figure 7.87 - ServicePort.....	101
Figure 7.88 - ServiceSpecificationRole.....	102
Figure 7.89 - ServiceConnector.....	103
Figure 7.90 - ServiceInterface.....	104
Figure 7.91 - ServiceFunction.....	105
Figure 7.92 - ServiceFunctionAction.....	106
Figure 7.93 - ServiceStateDescription.....	107

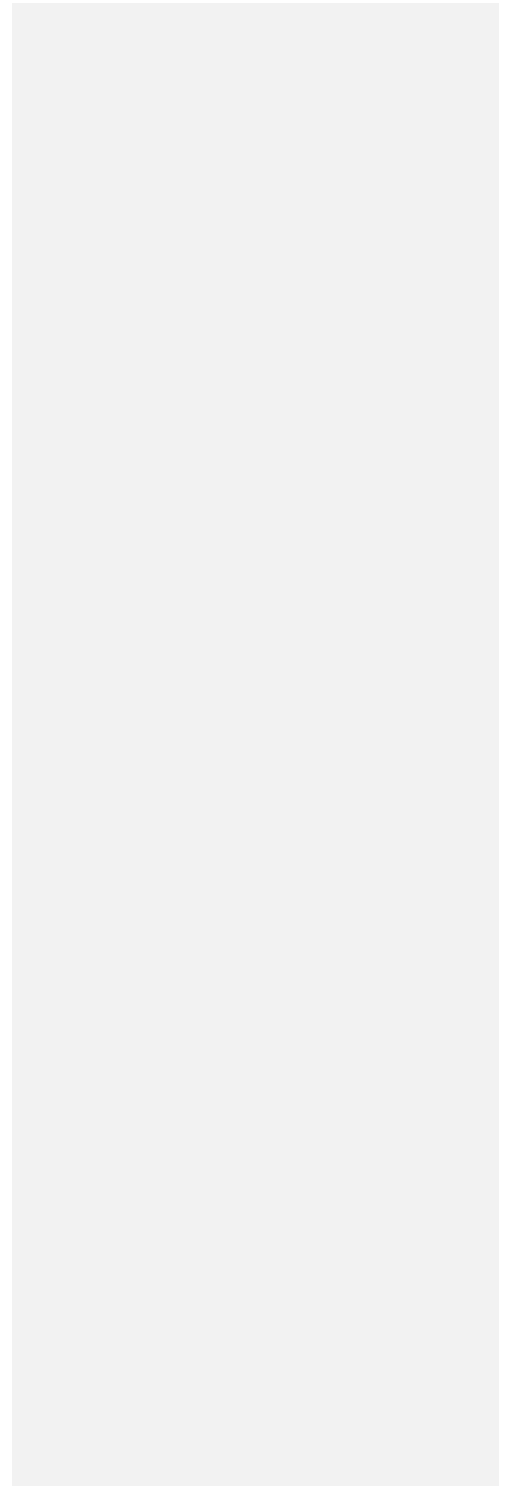


Figure 7.94 -  
ServiceMessage..... 107

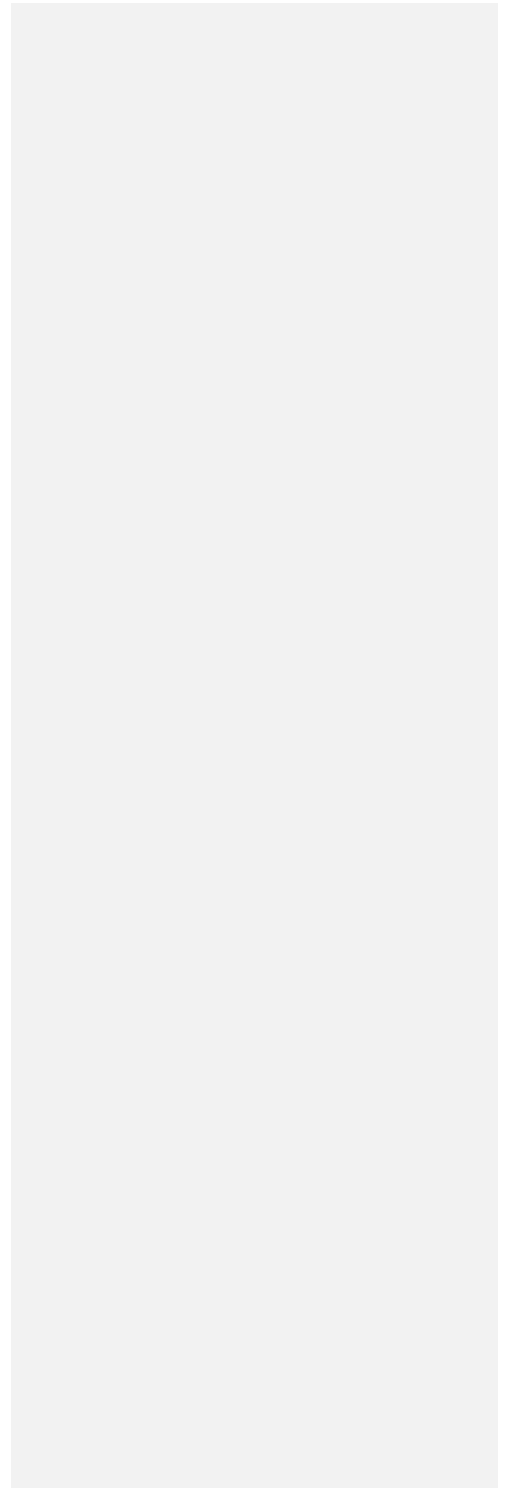


Figure 7.95 - ServicePolicy.....	108
Figure 7.96 - Consumes.....	109
Figure 7.97 - Organization.....	110
Figure 7.98 - OrganizationResource.....	111
Figure 7.99 - Person.....	111
Figure 7.100 - Post.....	112
Figure 7.101 - Responsibility.....	112
Figure 7.102 - Command.....	113
Figure 7.103 - Control.....	114
Figure 7.104 - CompetenceToConduct.....	115
Figure 7.105 - Competence.....	116
Figure 7.106 - CompetenceForRole.....	116
Figure 7.107 - RequiresCompetence.....	117
Figure 7.108 - ResponsibleFor.....	118
Figure 7.109 - CapabilityConfiguration.....	120
Figure 7.110 - NaturalResource.....	120
Figure 7.111 - PhysicalResource.....	121
Figure 7.112 - ResourceArchitecture.....	122
Figure 7.113 - ResourceArtifact.....	122
Figure 7.114 - ResourcePerformer.....	123
Figure 7.115 - Software.....	124
Figure 7.116 - System.....	125
Figure 7.117 - ResourceMethod.....	126
Figure 7.118 - ResourceParameter.....	127
Figure 7.119 - ResourcePort.....	128

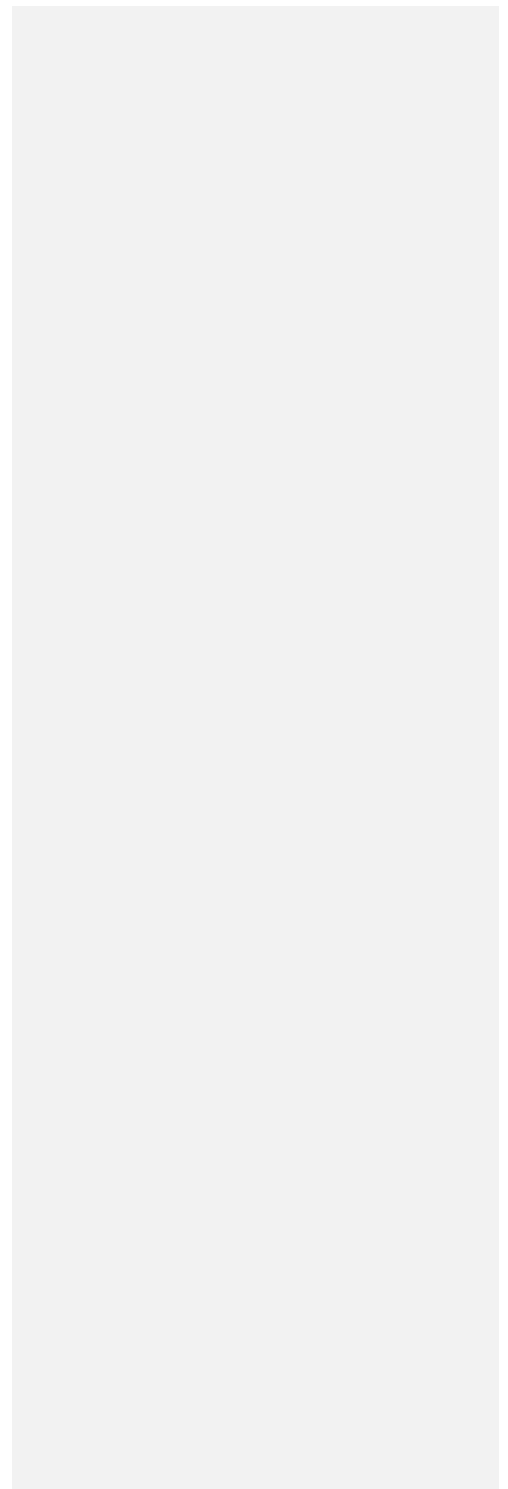


Figure 7.120 - ResourceRole.....	129
Figure 7.121 - ResourceConnector.....	131
Figure 7.122 - ResourceExchange.....	132
Figure 7.123 - ResourceExchangeItem.....	133
Figure 7.124 - ResourceInterface.....	134
Figure 7.125 - ResourceSignal.....	135
Figure 7.126 - ResourceSignalProperty.....	136
Figure 7.127 - Function.....	137
Figure 7.128 - FunctionAction.....	138
Figure 7.129 - FunctionControlFlow.....	139
Figure 7.130 - FunctionEdge.....	140
Figure 7.131 - FunctionObjectFlow.....	141
Figure 7.132 - ResourceStateDescription.....	141
Figure 7.133 - ResourceMessage.....	142
Figure 7.134 - DataElement.....	143
Figure 7.135 - DataModel.....	144
Figure 7.136 - ResourceConstraint.....	145
Figure 7.137 - SubjectOfResourceConstraint.....	146
Figure 7.138 - Forecast.....	146
Figure 7.139 - SubjectOfForecast.....	147
Figure 7.140 - Technology.....	148
Figure 7.141 - VersionedElement.....	149
Figure 7.142 - VersionOfConfiguration.....	150
Figure 7.143 - VersionSuccession.....	151
Figure 7.144 - WholeLifeConfiguration.....	152

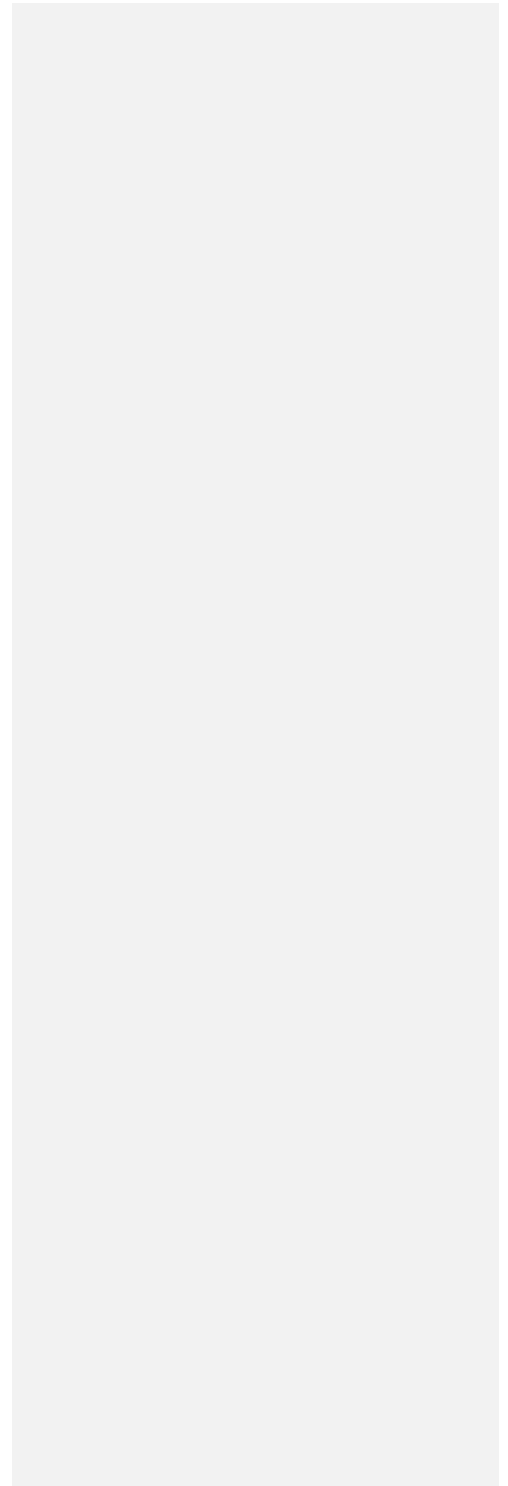


Figure 7.145 -  
ProtocolImplementation..... 153

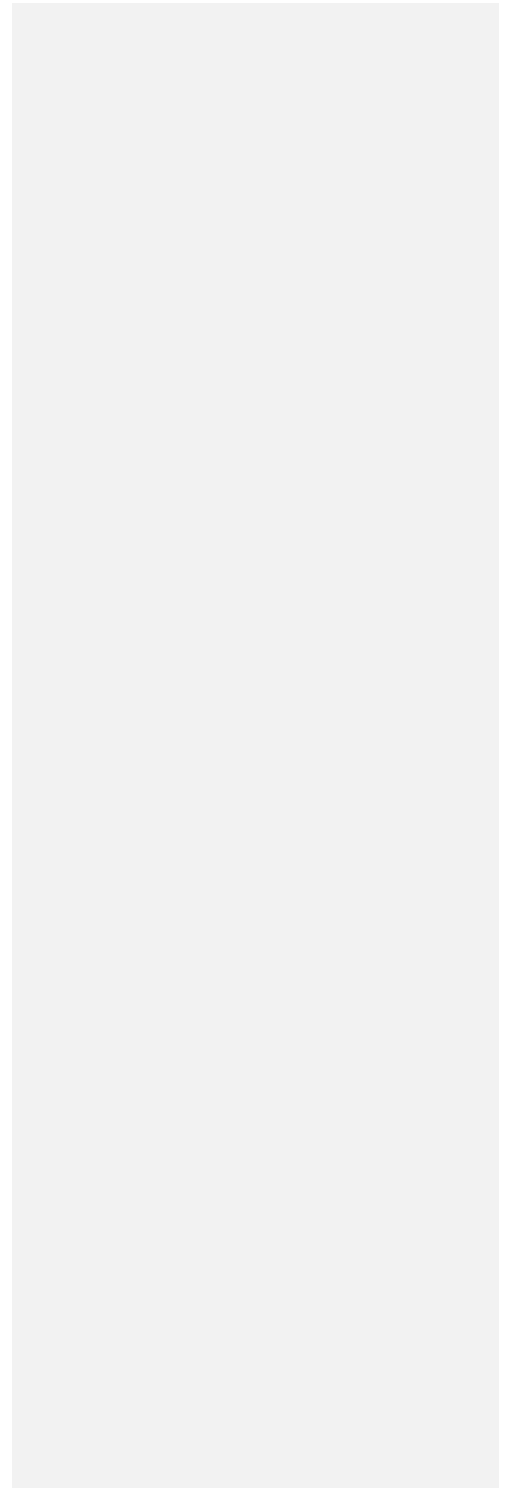




Figure 7.146 - Asset.....	154
Figure 7.147 - OperationalMitigation.....	155
Figure 7.148 - ResourceMitigation.....	155
Figure 7.149 - SecurityEnclave.....	156
Figure 7.150 - AssetRole.....	157
Figure 7.151 - SecurityProperty.....	158
Figure 7.152 - EnhancedSecurityControl.....	159
Figure 7.153 - Enhances.....	159
Figure 7.154 - Protects.....	160
Figure 7.155 - ProtectsContext.....	161
Figure 7.156 - SecurityProcess.....	162
Figure 7.157 - SecurityProcessAction.....	163
Figure 7.158 - ActualRisk.....	164
Figure 7.159 - Risk.....	165
Figure 7.160 - SecurityConstraint.....	166
Figure 7.161 - SecurityControl.....	167
Figure 7.162 - SecurityControlFamily.....	168
Figure 7.163 - SubjectOfSecurityConstraint.....	169
Figure 7.164 - Affects.....	169
Figure 7.165 - AffectsInContext.....	170
Figure 7.166 - Mitigates.....	171
Figure 7.167 - OwnsRisk.....	172
Figure 7.168 - OwnsRiskInContext.....	173
Figure 7.169 - Project.....	174
Figure 7.170 - ProjectMilestone.....	175

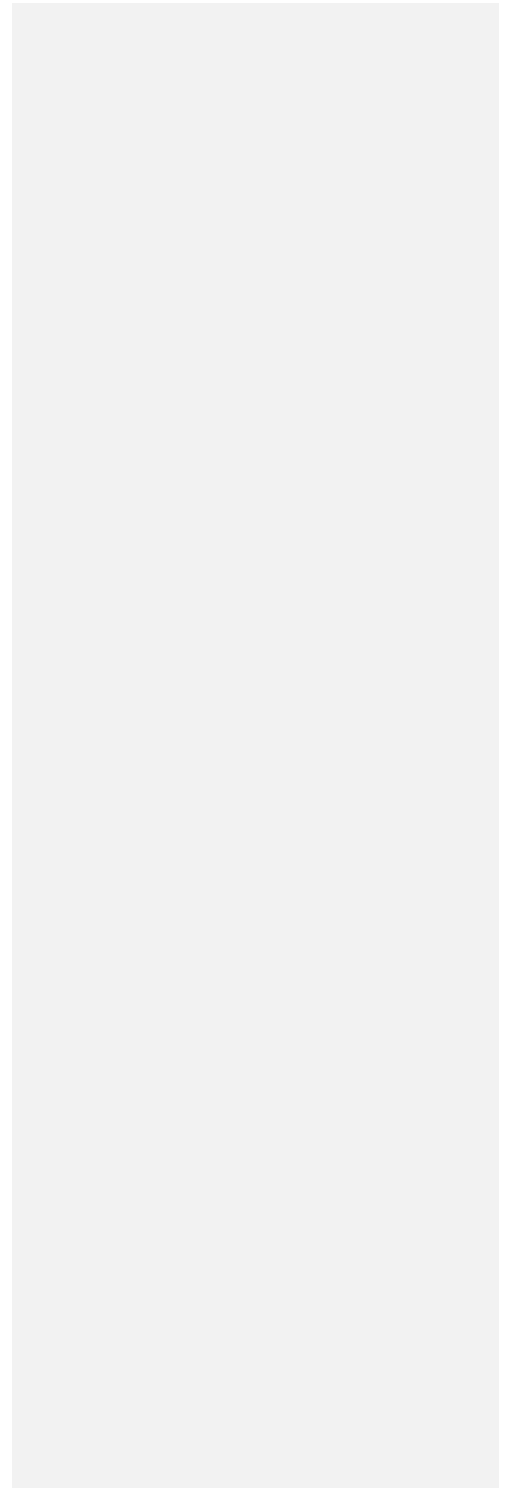


Figure 7.171 - ProjectMilestoneRole.....	176
Figure 7.172 - ProjectRole.....	177
Figure 7.173 - ProjectStatus.....	178
Figure 7.174 - ProjectTheme.....	179
Figure 7.175 - StatusIndicators.....	180
Figure 7.176 - MilestoneDependency.....	180
Figure 7.177 - ProjectSequence.....	181
Figure 7.178 - ProjectActivity.....	182
Figure 7.179 - ProjectActivityAction.....	183
Figure 7.180 - ActualProject.....	184
Figure 7.181 - ActualProjectMilestone.....	185
Figure 7.182 - ActualProjectMilestone.....	186
Figure 7.183 - ActualProjectRole.....	187
Figure 7.184 - Protocol.....	188
Figure 7.185 - ProtocolStack.....	189
Figure 7.186 - Standard.....	189
Figure 7.187 - ProtocolLayer.....	190
Figure 7.188 - ActualOrganization.....	192
Figure 7.189 - ActualOrganizationalResource.....	193
Figure 7.190 - ActualPerson.....	194
Figure 7.191 - ActualPost.....	195

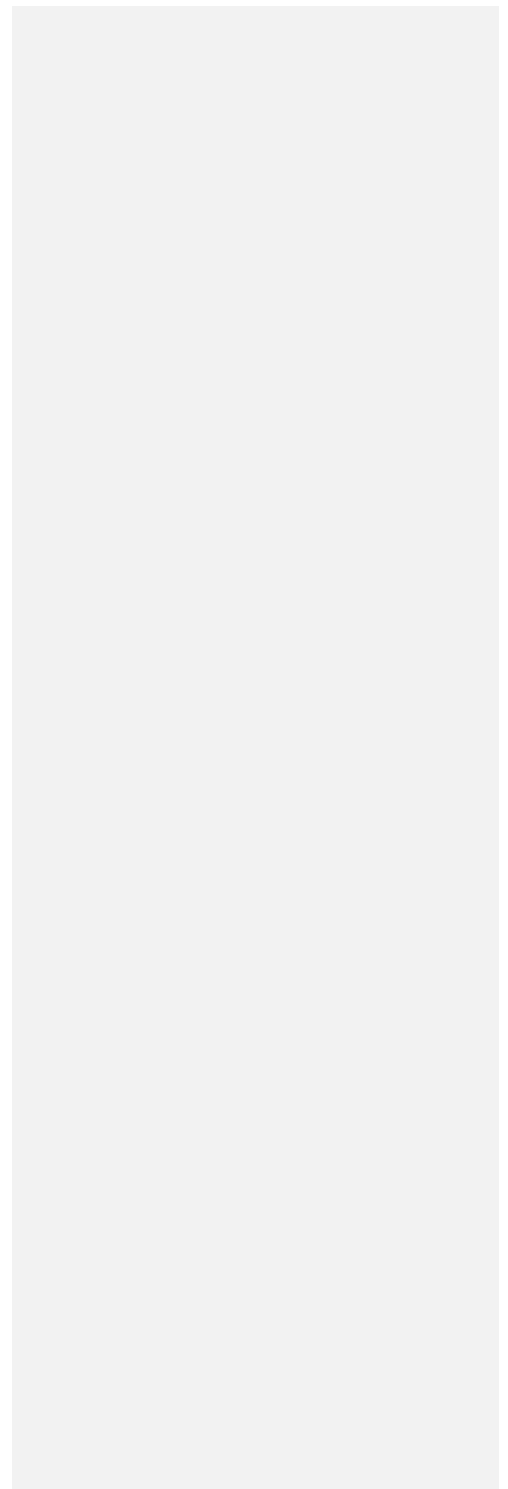


Figure 7.192 - ActualResource.....	196
Figure 7.193 - ActualResponsibility.....	197
Figure 7.194 - ActualResponsibleResource.....	198
Figure 7.195 - FieldedCapability.....	198
Figure 7.196 - ActualOrganizationRole.....	199
Figure 7.197 - ActualResourceRole.....	200
Figure 7.198 - ActualResourceRelationship.....	201
Figure 7.199 - FillsPost.....	202
Figure 7.200 - ActualService.....	203
Figure 7.201 - ProvidedServiceLevel.....	204
Figure 7.202 - ProvidesCompetence.....	204
Figure 7.203 - RequiredServiceLevel.....	205
Figure 7.204 - OwnsProcess.....	206
Figure 7.205 - ArchitecturalDescription.....	207
Figure 7.206 - Architecture.....	209
Figure 7.207 - Concern.....	210
Figure 7.208 - Stakeholder.....	211
Figure 7.209 - UAFElement.....	212
Figure 7.210 - View.....	213
Figure 7.211 - Viewpoint.....	214
Figure A.1 - Strategic Taxonomy.....	215
Figure A.2 - Strategic Structure.....	216
Figure A.3 - Strategic Connectivity.....	217
Figure A.4 - Strategic States.....	218
Figure A.5 - Strategic Constraints.....	219

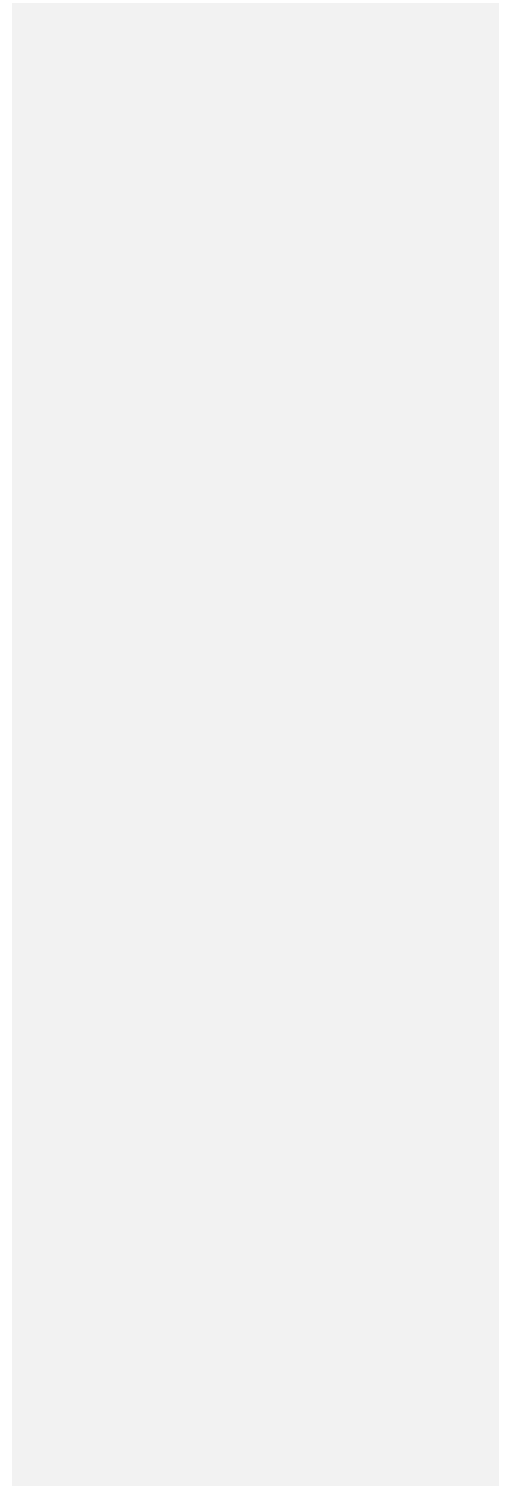


Figure A.6 - Strategic Roadmap: Deployment.....	220
Figure A.7 - Strategic Roadmap: Phasing.....	221
Figure A.8 - Strategic Traceability.....	222
Figure A.9 - Operational Taxonomy.....	223
Figure A.10 - Operational Structure.....	225
Figure A.11 - Operational Connectivity.....	227
Figure A.12 - Operational Processes.....	229
Figure A.13 - Operational States.....	230
Figure A.14 - Operational Interaction Scenarios.....	231
Figure A.15 - Operational Constraints.....	232
Figure A.16 - Operational Traceability.....	233
Figure A.17 - Services Taxonomy.....	234
Figure A.18 - Services Structure.....	235
Figure A.19 - Services Connectivity.....	236
Figure A.20 - Services Processes.....	237
Figure A.21 - Services States.....	238
Figure A.22 - Services Interaction Scenarios.....	238
Figure A.23 - Services Constraints.....	239
Figure A.24 - Services Roadmap.....	240
Figure A.25 - Services Traceability.....	241
Figure A.26 - Personnel Taxonomy.....	242
Figure A.27 - Personnel Structure.....	243
Figure A.28 - Personnel Connectivity.....	244
Figure A.29 - Personnel Processes.....	246
Figure A.30 - Personnel States.....	247

Figure A.31 - Personnel Interaction Scenarios.....	248
Figure A.32 - Personnel Constraints: Competence.....	249
Figure A.33 - Personnel Constraints: Drivers.....	250
Figure A.34 - Personnel Constraints: Performance.....	252
Figure A.35 - Personnel Roadmap: Evolution.....	254
Figure A.36 - Personnel Roadmap: Forecast.....	255
Figure A.37 - Personnel Roadmap: Forecast.....	256
Figure A.38 - Personnel Traceability.....	257
Figure A.39 - Resources Taxonomy.....	258
Figure A.40 - Resources Structure.....	260
Figure A.41 - Resources Connectivity.....	262
Figure A.42 - Resources Processes.....	264
Figure A.43 - Resources States.....	265
Figure A.44 - Resources Interaction Scenarios.....	266
Figure A.45 - Resources Constraints.....	267
Figure A.46 - Resources Roadmap: Evolution.....	268
Figure A.47 - Resources Roadmap: Forecast.....	269
Figure A.48 - Resources Roadmap: Traceability.....	270
Figure A.49 - Security Taxonomy.....	271
Figure A.50 - Security Structure.....	273
Figure A.51 - Security Connectivity.....	274
Figure A.52 - Security Processes.....	276
Figure A.53 - Security Constraints.....	277
Figure A.54 - Security Traceability.....	279

Figure A.55 - Project Taxonomy.....	280
Figure A.56 - Project Structure.....	281
Figure A.57 - Project Connectivity.....	282
Figure A.58 - Project Processes.....	283
Figure A.59 - Project Roadmap.....	285
Figure A.60 - Project Traceability.....	287
Figure A.61 - Standards Taxonomy.....	288
Figure A.62 - Standards Structure.....	289
Figure A.63 - Standards Roadmap.....	290
Figure A.64 - Standards Traceability.....	291
Figure A.65 - Actual Resources Structure.....	292
Figure A.66 - Actual Resources Connectivity.....	293
Figure A.67 - Dictionary.....	294
Figure A.68 - Requirements.....	295
Figure A.69 - Summary & Overview.....	296
Figure A.70 - Information Model.....	297
Figure A.71 - Parameters: Environment.....	298
Figure A.72 - Parameters: Measurements.....	299

## Preface

### OMG

Founded in 1989, the Object Management Group, Inc. (OMG) is an open membership, not-for-profit computer industry standards consortium that produces and maintains computer industry specifications for interoperable, portable, and reusable enterprise applications in distributed, heterogeneous environments. Membership includes Information Technology vendors, end users, government agencies, and academia.

OMG member companies write, adopt, and maintain its specifications following a mature, open process. OMG's specifications implement the Model Driven Architecture® (MDA®), maximizing ROI through a full-lifecycle approach to enterprise integration that covers multiple operating systems, programming languages, middleware and networking infrastructures, and software development environments. OMG's specifications include: UML® (Unified Modeling Language™); CORBA® (Common Object Request Broker Architecture); CWM™ (Common Warehouse Metamodel); and industry-specific standards for dozens of vertical markets.

More information on the OMG is available at <http://www.omg.org/>.

### OMG Specifications

As noted, OMG specifications address middleware, modeling and vertical domain frameworks. Adopted specifications are available from this URL:

<http://www.omg.org/spec>

All of OMG's formal specifications may be downloaded without charge from our website. (Products implementing OMG specifications are available from individual suppliers.) Copies of specifications, available in PostScript and PDF format, may be obtained from the Specifications Catalog cited above or by contacting the Object Management Group, Inc. at:

OMG Headquarters  
109 Highland Ave  
Needham, MA 02494  
USA  
Tel: +1-781-444-0404  
Fax: +1-781-444-0320  
Email: [pubs@omg.org](mailto:pubs@omg.org)

Certain OMG specifications are also available as ISO standards. Please consult <http://www.iso.org>

### Issues

The reader is encouraged to report any technical or editing issues/problems with this document by completing the Issue Reporting Form listed on the main web page <https://www.omg.org>, under OMG Specifications, Report an Issue.

**Commented [GB1]:** UAF11-298 move and update sections 1-6.5 to DMM  
Add contents of UAFP introduction,.docx to beginning of document

#### **Deleted: 1 Scope**

### **1.1 UAFP Background**

The scope of Unified Architecture Framework Profile (UAFP) includes the language extensions to enable the extraction of specified and custom models from an integrated architecture description (AD). The models describe a system<sup>1</sup> from a set of stakeholders' concerns such as security or information through a set of predefined viewpoints and associated views<sup>2</sup>. Developed models can also reflect custom viewpoints or to develop more formal extensions for new viewpoints. The UAFP specification supports the Department of Defense Architecture Framework (DoDAF) 2.02, the Ministry of Defence Architecture Framework (MODAF), Security Views from Canada's Department of National Defense Architecture Framework (DNDAF) and the North Atlantic Treaty Organization (NATO) Architecture Framework (NAF) v 3.1. The core concepts in<sup>3</sup> the UAF domain metamodel specify the UAFP based upon the DoDAF 2.0.2 Domain Metamodel (DM2) and the MODAF ontological data exchange mechanism (MODEM). MODEM is intended to provide the basis for the next version of NAF. The intent is to provide a standard representation for AD support for Defense Organizations. The intention of UAFP is also to support a standard representation for non-defense organizations' ADs as part of their Systems Engineering (SE) technical processes. The associated UAF metamodel (see c4i-2016-02-03) intent is to improve the ability to exchange architecture data between related tools that are UML/SysML based and tools that are based on other standards.<sup>4</sup>

UAFP 1.0 supports the capability to:<sup>5</sup>

- model architectures for a broad range of complex systems, which may include hardware, software, data, personnel, and facility elements;<sup>6</sup>
- model consistent architectures for system-of-systems (SoS) down to lower levels of design and implementation;<sup>7</sup>
- support the analysis, specification, design, and verification of complex systems; and<sup>8</sup>
- improve the ability to exchange architecture information among related tools that are SysML based and tools that are based on other standards.<sup>9</sup>

### **1.2 Intended Users**

The profile enables the modeling of strategic capabilities; business/operational activities, OperationalPerformers and their interfaces, measures of effectiveness; services and their interfaces, levels of agreement and measures of performance; system resources and their functions, ports, protocols, interfaces, measures of performance; security including cyb<sup>10</sup>



# 1. Introduction

## 1.1 Overview

This document is a normative supplement to the UAF DMM document (dtr/19-06-16).

This document specifies a UAF profile to enable practitioners to express architectural model elements and organize them in a set of domains, model kinds, and view specification (specified in the UAF DMM) that support the specific needs of end users in defense and commercial industry.

UAFP 1.1 defines a set of stereotypes and model elements and relationships to satisfy the requirements of the UPDM 3.0 RFP and the UAF DMM. The profile specification documents the language architecture in terms of UML profiling mechanism.

A number of UAFP stereotypes inherit from SysML stereotypes where reuse of SysML semantics is necessary. The reusable portions of the SysML specification are not included directly in the specification but are made explicit through the stereotype inheritance.

Formatted: Not Highlight

Formatted: Not Highlight

Formatted: Not Highlight

## 2. Additional Information

### 2.1 Language Architecture

The UAFP specification reuses a subset of UML 2.5.1 and SysML 1.5 and provides additional extensions needed to address requirements in the UPDM 3.0 RFP Mandatory Requirements. Those requirements form the basis for this document. This document describes the language architecture in terms of the UML 2.5.1 and SysML 1.5 parts that are reused and the defined UML 2.5.1 extensions; and specifies how to implement UAFP. This clause explains design principles and how they are applied to define the UAFP language architecture.

### 2.2 Core Principles

The fundamental design principles for UAFP are:

- **Requirements-driven:** UAFP is intended to satisfy the requirements of the UPDM 3.0 RFP Mandatory Requirements.
- **UAF Domain Metamodel (DMM) driven:** The DMM served as a foundation for profile development.
- **Reuse of existing specifications:** UAFP reuses UML/SysML wherever practical to satisfy the requirements of the UAFP 3.0 RFP and leverage features from both UML and SysML to provide a robust modeling capability. Consequently, UAFP is intended to be relatively easy to implement for vendors who support UML 2.x and SysML 1.x.
- **Compliance levels:** UAFP has a single compliance level based upon a combination of the reuse of UML and SysML elements. It is expected that the views that are created as result of this profile have frames that reflect the underlying SysML diagram type that is used as the basis for the view. It also expected that the graphical notation used to display elements within those views correspond to the standard SysML graphical notation of the SysML/UML metaclass that the stereotype extends.
- **Interoperability:** UAFP inherits the XMI interchange capability from UML. The UAFP specification reuses a subset of UML 2.5.1 and provides additional extensions needed to address requirements in the UPDM 3.0 RFP Mandatory Requirements.

### 2.3 Representing Stereotype Constraints

The UAF Profile uses an enhanced standard notation to represent metaconstraints graphically in the UAF profile diagrams to improve readability of the UAF Profile specification and overcome limitations of being unable to visualize constraints diagrammatically in UML.

The metaconstraints appears in the UAFP specification diagrams for visualization purposes only, however the representation in the XMI is as a UML constraint, specified in structured English. These constraints are implementable in a tool, by OCL for example.

A simple UML profile defines these metaconstraints.

The following subsections detail the metaconstraint profile definition within the UAF profile.

#### **Deleted: 6.5 Representing Stereotype Constraints**

The UAF Profile uses an enhanced standard notation to represent metaconstraints graphically in the UAF profile diagrams to improve readability of the UAF Profile specification and overcome limitations of being unable to visualize constraints diagrammatically in UML.

The metaconstraints appear in the UAFP specification diagrams for visualization purposes only, however the representation in the XMI is as a UML constraint, specified in structured English. These constraints are implementable in a tool, by OCL for example.

A simple UML profile defines these metaconstraints.

The following sub clauses detail the metaconstraint profile definition within the UAF profile.



### 2.3.1 Metaconstraint dependency

«metaconstraint» is a stereotype that extends the Dependency metaclass. It is used to specify constrained elements within the profile.

A sample of the «metaconstraint» dependency is a diagram for stereotype extending the Dependency metaclass.

MapsToCapability is a UAFP stereotype that extends Abstraction (a type of Dependency in UML). The constraint on this stereotype is that its client end must be stereotyped by an Activity (which is abstract) and its supplier end must be stereotyped by a Capability. But as it is not possible to show this constraint graphically the diagram does not communicate the needed information. We then use the "metaconstraint" dependency to visualize the constraint.

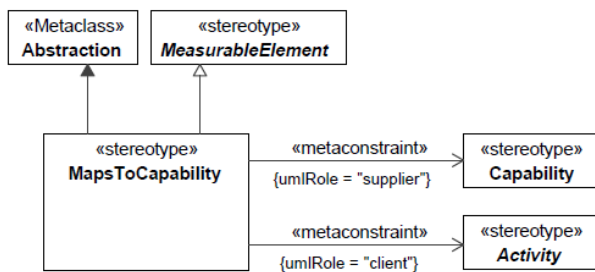


Figure 2.1 - MapsToCapability Stereotype

With the metaconstraint dependency added to the diagram (see Figure 6.1) which shows that MapsToCapability is a stereotype extending the Abstraction metaclass, that inherits the properties of a MeasurableElement and is used for modeling a relationship between an Activity (or its specializations) and a Capability (or its specializations). A Dependency stereotyped MapsToCapability must have its values for the client property stereotyped as an Activity, and its values for the supplier property must be stereotyped Capability.

**Note** – When stereotype extends Connector, the stereotype property umlRole has values "end[0].role" and "end[1].role."

For example:

This is done because Connector has no direct "linkage" to the connected element; it links to the Connector Ends, which references the linked element. So, end[n] gives the reference to the ConnectorEnd, and role gives the reference to the linked element.

Deleted: 6

Deleted: 5

Deleted: 6

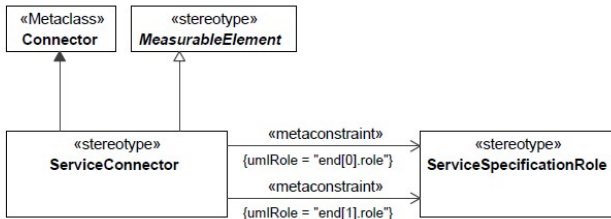


Figure 2.2 - Connector Extension

Deleted: 6

### 2.3.2 Metarelationship dependency

«metarelationship» is a stereotype for dependency, showing that certain domain concepts will be implemented using regular UML relationships.

For example: A Capability may depend on other Capabilities or be subtype of a Capability, but this concept cannot be visualized on the diagram.

Deleted: 6

Deleted: 5

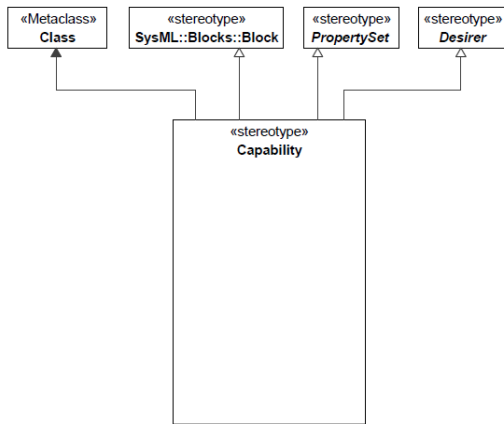


Figure 2.3 - Capabilities Generalization

Deleted: 6

We are using the «metarelationship» dependency to visualize the dependency and the generalization concept.

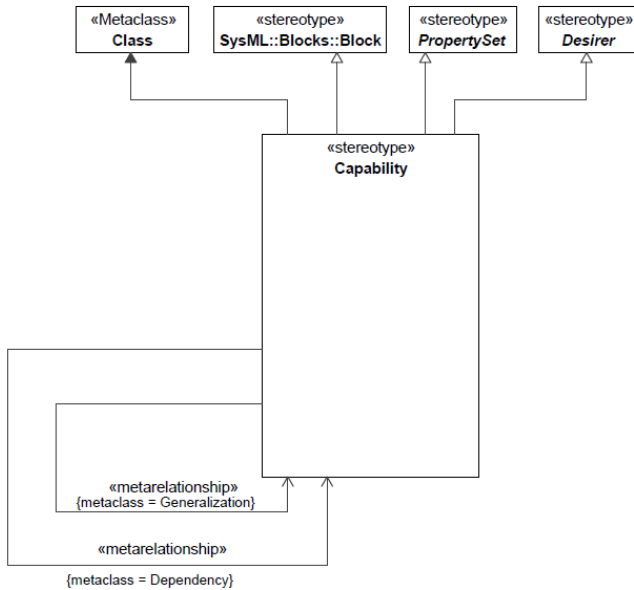


Figure 2.4 - Visualizing «metarelsationship»

Deleted: 6

This diagram should be read as follows:

Capability may have other Capabilities related to it, using the UML Dependency metaclass and it may have sub types of Capabilities related to it, using the the UML Generalization metaclass.

The «metarelsationship» dependency will appear only in the specification diagrams, but not the profile XMI.

### 2.3.3 Stereotyped relationship dependency

Although the «metarelsationship» dependency creates a good way to show the constrained ends of the stereotyped relationship, it also creates some overhead when showing the relationship between two stereotypes.

For example, Figure 6.5 shows that elements of subtype Achiever have a stereotyped relationship called AchievedEffect with elements of type ActualState.

Deleted: 6

Deleted: 5

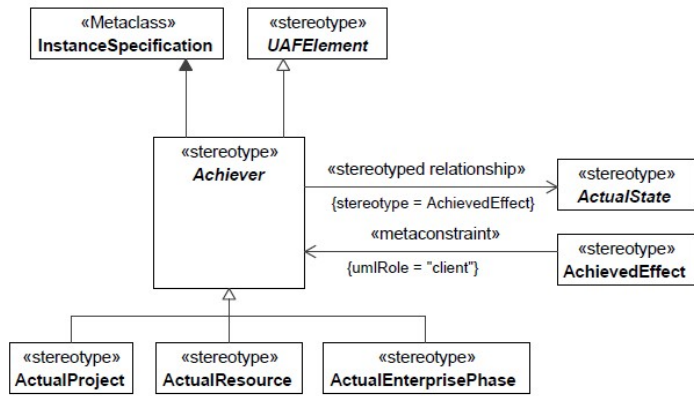


Figure 2.5 - Use of the AchievedEffect «stereotyped relationship» dependency

Deleted: 6

This page intentionally left blank.



## 3 UAF Profile

### 3.1 UAF

UAFP imports the entire SysML profile and a number of UAFP stereotypes inherit from SysML stereotypes. This is intended to provide more seamless integration with system modeling using SysML and to be able to fully leverage the capabilities of SysML in UAFP. An example of this is the integration of Requirements into the UAFP and also the use of Parametric Diagrams and integration of elements based upon instance specifications to allow the assessment of measures within an architecture developed using UAFP.

UAF is the top level profile root.

#### 3.1.1 UAF::Dictionary

Stakeholders: Architects, users of the architecture, Capability Owners, Systems Engineers, Solution Providers.

Concerns: Definitions for all the elements in the architecture, libraries of environments and measurements.

Definition: Presents all the elements used in an architecture. Can be used specifically to capture:

- a. elements and relationships that are involved in defining the environments applicable to capability, operational concept or set of systems.
- b. measurable properties that can be used to support analysis such as KPIs, MoEs, TPIs, etc.

#### Alias

**Package:** Dictionary

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Comment

Description

A metamodel Artifact used to define an alternative name for an element.

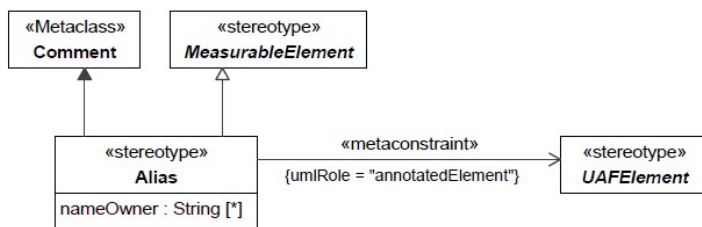


Figure 7.1 - Alias

**Formatted:** OMG Heading 1, Indent: Left: 0 cm, Right: 0 cm, Space Before: 0 pt

**Deleted:** 7

**Deleted:** 7

**Formatted:** Line spacing: single

**Commented [GB2]:** UAF11-77

**Deleted:** UAFP imports the entire SysML profile and contains a set of constraints that specify which SysML stereotypes are applied to the UAFP elements. This is intended to provide more seamless integration with system modeling using SysML and to be able to fully leverage the capabilities of SysML in UAFP. An example of this is the integration of Requirements into the UAFP and also the use of Parametric Diagrams and integration of elements based upon instance specifications to allow the assessment of measures within an architecture developed using UAFP.

**Formatted:** Line spacing: single

**Deleted:** 7

Attributes

nameOwner : String[\*]      Someone or something that uses this alternative name.

Constraints

[1] Alias.annotatedElement      Value for the annotatedElement metaproperty must be stereotyped by the specialization of «UAFElement».

**Definition**

**Package:** Dictionary

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Comment

Description

A comment containing a description of an element in the architecture.

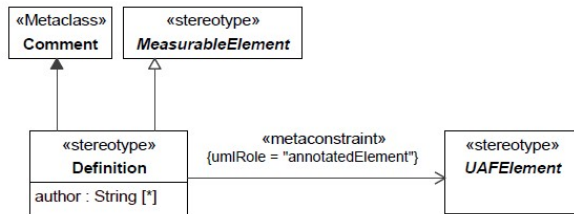


Figure 7.2 - Definition

Attributes

author : String[\*]      The original or current person (architect) responsible for the Definition.

Constraints

[1] Definition.annotatedElement      Value for the annotatedElement metaproperty must be stereotyped by the specialization of «UAFElement».

## SameAs

**Package:** Dictionary

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Dependency

**Description**

A dependency relationship that asserts that two elements refer to the same real-world thing.

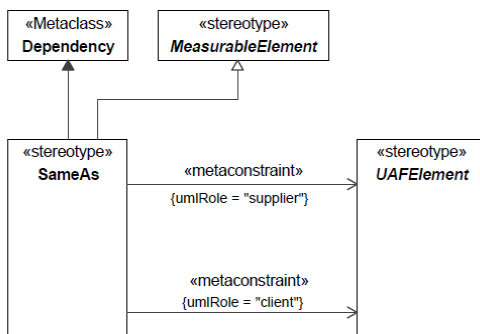


Figure 7.3 - SameAs

**Constraints**

- [1] SameAs.client Values for the client metaproperty must be stereotyped by the specialization of «UAFElement».
- [2] SameAs.supplier Values for the supplier metaproperty must be stereotyped by the specialization of «UAFElement».

## 7.1.2 UAF::Parameters

**ActualCondition**

**Package:** Parameters

**isAbstract:** No

**Generalization:** [ActualPropertySet](#)

**Extension:** InstanceSpecification

**Description**

The actual state of an environment or location describing its situation.

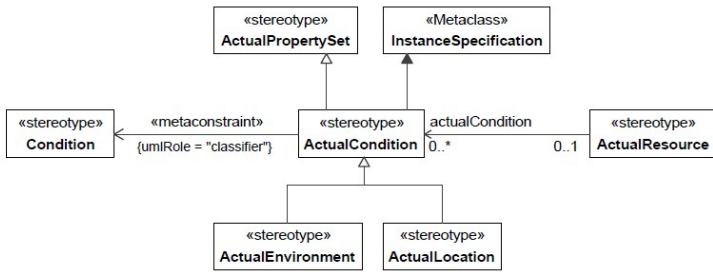


Figure 7.4 - ActualCondition

Constraints

- [1] ActualCondition.classifier Value for the classifier metaproperty has to be stereotyped «Condition» or its specializations.

**ActualEnvironment**

Package: Parameters

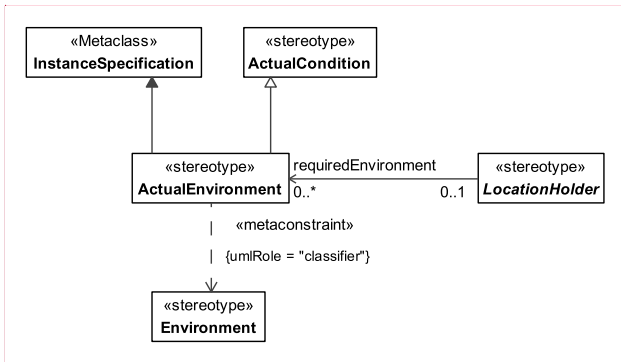
isAbstract: No

Generalization: [ActualCondition](#)

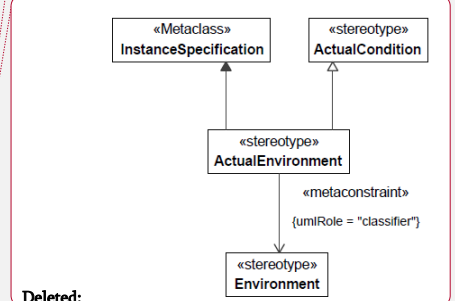
Extension: InstanceSpecification

Description

The ActualState that describes the circumstances of an Environment.

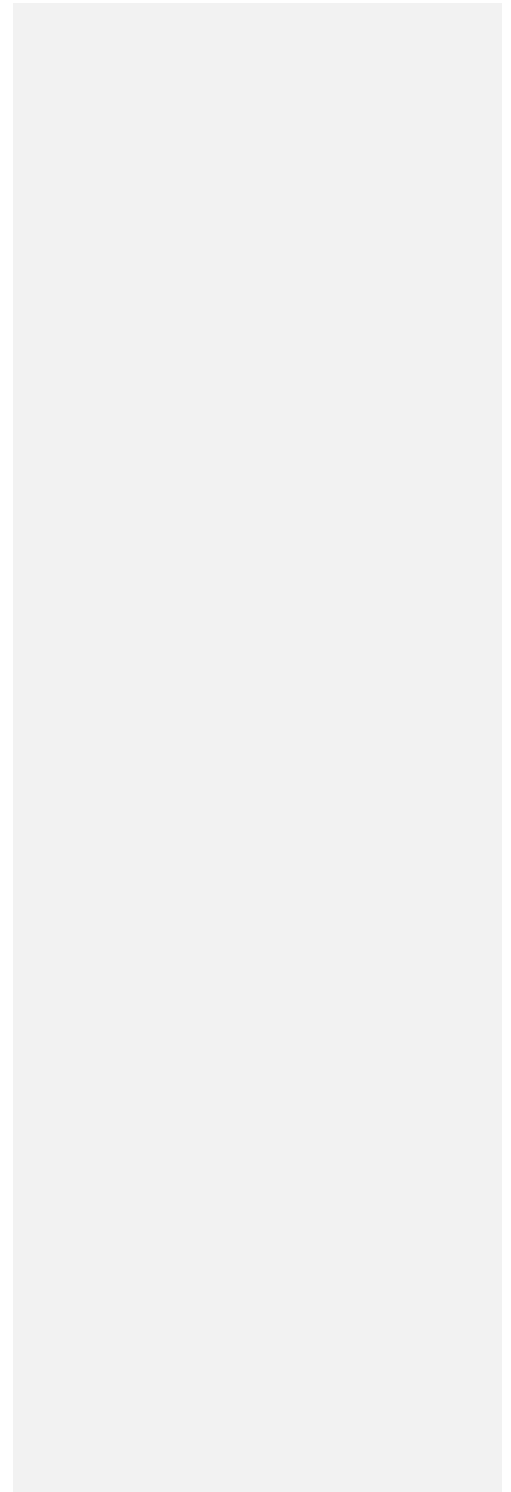


Commented [AM3]: [UAF11-15](#) Figure 7.5 – ActualEnvironment replaced by ActualEnvironment.svg



Deleted:

Figure 7.5 - ActualEnvironment



Constraints

[1] ActualEnvironment.classifier Value for the classifier metaproperty has to be stereotyped «Environment» or its specializations.

**ActualLocation**

**Package:** Parameters

**isAbstract:** No

**Generalization:** [ActualCondition](#)

**Extension:** InstanceSpecification

Description

An ActualState that describes a physical location, for example using text to provide an address, Geo-coordinates, etc.

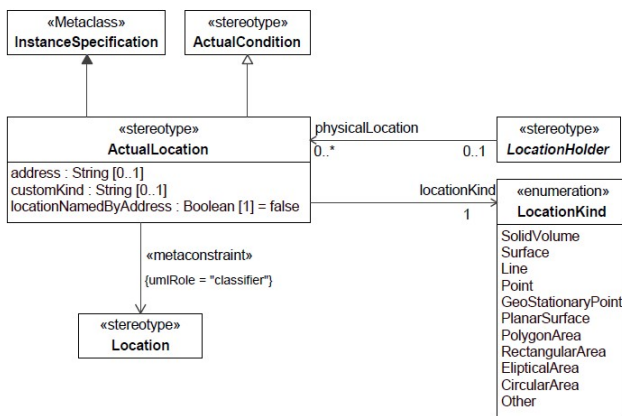


Figure 7.6 - ActualLocation

Attributes

- address : String[0..1] String describing the address of the ActualLocation, i.e., "1600 Pennsylvania avenue," "The White House"
- customKind : String[0..1] String describing a location kind that is not in the LocationKind enumerated list
- locationNamedByAddress : Boolean[1] Boolean that indicates if the ActualLocation address is embedded in the ActualLocation name. By default = false.



Associations

locationKind : LocationKind[1] Enumerated value describing the kind of ActualLocation.

Constraints

[1] ActualLocation.classifier Classifier metaproperty value must be stereotyped «Location» or its specializations.

**ActualMeasurement**

Package: Parameters

isAbstract: No

Generalization: [ActualState](#)

Extension: Slot

Description

An actual value that is applied to a Measurement.

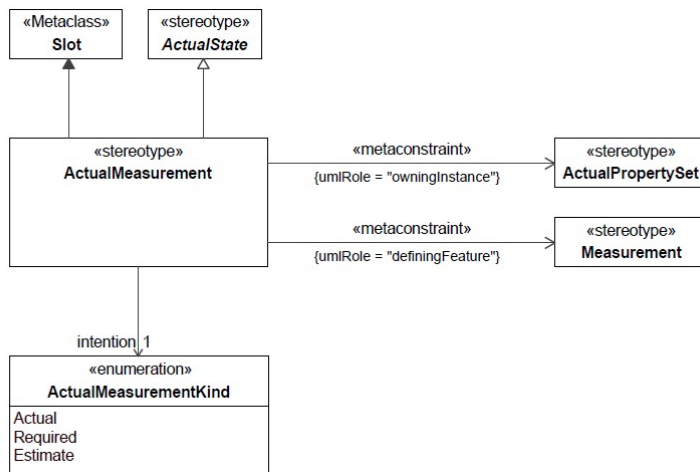


Figure 7.7 - ActualMeasurement



#### Associations

intention : ActualMeasurementKind[1] Enumerated value describing the intent of the ActualMeasurement.

#### Constraints

[1] ActualMeasurement.definingFeature Value for the definingFeature metaproperty must be stereotyped «Measurement» or its specializations.

[2] ActualMeasurement.owningInstance Value for the owningInstance metaproperty must be stereotyped «ActualPropertySet» or its specializations.

### **ActualMeasurementKind**

**Package:** Parameters

**isAbstract:** No

#### Description

Enumeration of the possible kinds of ActualMeasurement. Its enumeration literals are:

- Actual - Indicates that the ActualMeasurement associated with the ActualMeasurementKind is a realworld value.
- Required - Indicates that the ActualMeasurement associated with the ActualMeasurementKind is a value that is expected to be achieved.
- Estimate - Indicates that the ActualMeasurement associated with the ActualMeasurementKind is an estimate of a realworld value.

### **ActualMeasurementSet**

**Package:** Parameters

**isAbstract:** No

**Generalization:** [ActualPropertySet](#)

**Extension:** InstanceSpecification

#### Description

A set of ActualMeasurements.

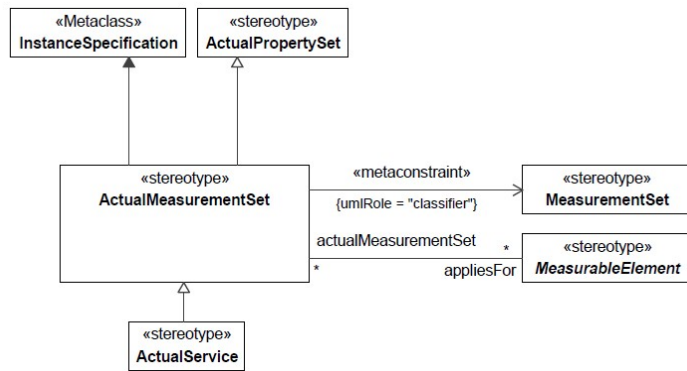


Figure 7.8 - ActualMeasurementSet

Associations

appliesFor : MeasurableElement[\*] Relates the ActualMeasurementSet to the elements that are being measured.

Constraints

- [1] ActualMeasurementSet.classifier Classifier metaproperty value must be stereotyped «MeasurementSet» or its specializations.
- [2] ActualMeasurementSet.slot Value for the slot metaproperty must be stereotyped «ActualMeasurement» or its specializations.

**ActualPropertySet**

**Package:** Parameters

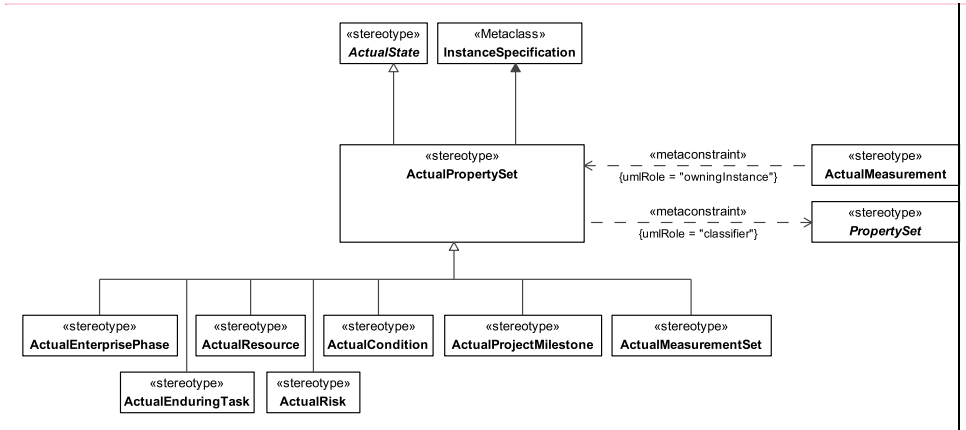
**isAbstract:** No

**Generalization:** [ActualState](#)

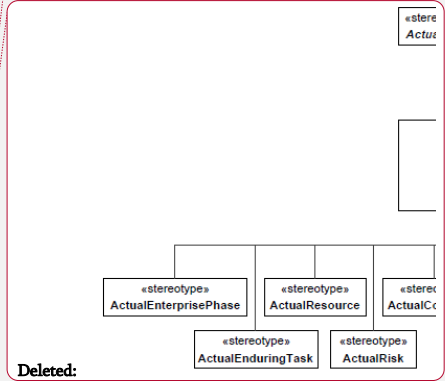
**Extension:** InstanceSpecification

Description

A set or collection of Actual properties.



Commented [AM4]: UAF11-125 Figure 7.9 – ActualPropertySet replaced by ActualPropertySet.svg.



Deleted:

Figure 7.9 - ActualPropertySet

Constraints

[1] ActualPropertySet.classifier Value for the classifier metaproperty must be stereotyped by the specialization of «PropertySet».

Condition

Package: Parameters

isAbstract: No

Generalization: [PropertySet](#), ValueType

Extension: DataType

Description

Defines the Location, Environment and/or GeoPoliticalExtent under which an OperationalActivity, Function or ServiceFunction can be performed.

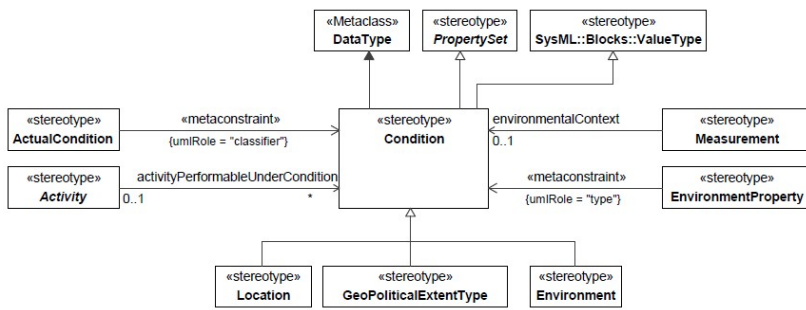


Figure 7.10 - Condition

**Environment Package:**

Parameters isAbstract: No

Generalization: [Condition](#)

Extension: DataType

Description

A definition of the environmental factors in which something exists or functions. The definition of an Environment element can be further defined using EnvironmentKind.

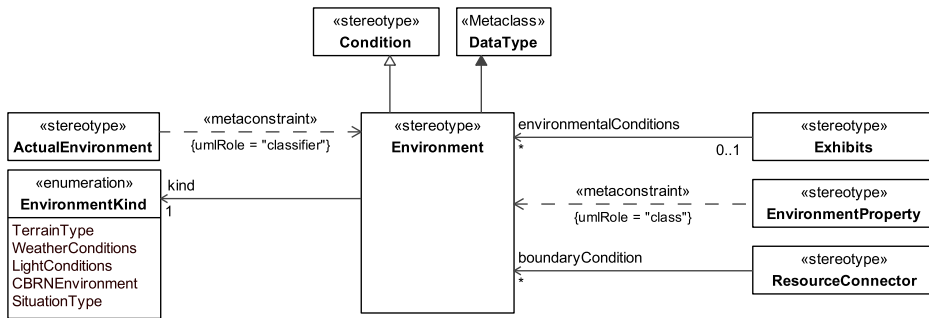
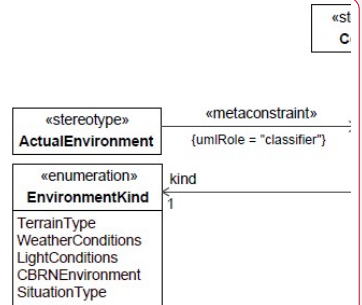


Figure 7.11 - Environment

Commented [AM5]: [UAF11-157](#) diagram replaced by Environment.svg



Deleted:

Associations

kind : EnvironmentKind[1]      Captures the kind of Environment.

### **EnvironmentKind**

**Package:** Parameters

**isAbstract:** No

Description

Enumeration of the possible kinds of Environment. Its enumeration literals are:

- TerrainType - Indicates that the Environment associated with EnvironmentKind captures a kind of terrain used to describe the terrain state of an environment at a particular time (e.g., muddy, frozen ground, deep snow, etc.).
- WeatherConditions - Indicates that the Environment associated with EnvironmentKind captures a kind of weather condition (e.g., Typhoon, Hurricane, Very Hot, Humid, etc.).
- LightConditions - Indicates that the Environment associated with EnvironmentKind captures a kind of light condition (e.g., broad daylight, dusk, moonlit, etc.).
- CBRNEnvironment - Indicates that the Environment associated with EnvironmentKind is of a Chemical, Biological, Radiological, or Nuclear (CBRN) kind.
- SituationType - Indicates that the Environment associated with EnvironmentKind captures a kind of situation used to describe the types and levels of threat (e.g., Corrosive, Fire, Smoke, Peaceful, etc.).

### **EnvironmentProperty**

**Package:** Parameters

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Property

Description

A property of an Environment that is typed by a Condition. The kinds of Condition that can be represented are Location, GeoPoliticalExtentType, and Environment.

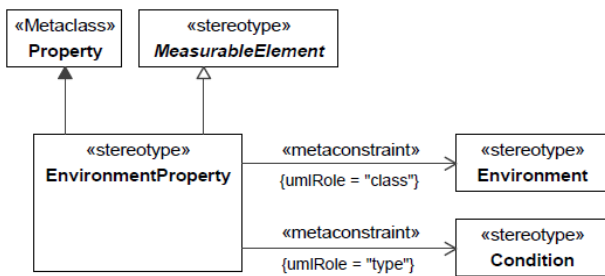


Figure 7.12 - EnvironmentProperty

Constraints

- [1] EnvironmentalProperty.class Value for the class metaproperty must be stereotyped «Environment» or its specializations.
- [2] EnvironmentalProperty.type Value for the type property must be stereotyped «Condition» or its specializations.

**GeoPoliticalExtentType**

**Package:** Parameters

**isAbstract:** No

**Generalization:** [ResourceExchangeItem](#), [OperationalExchangeItem](#), [Condition](#)

**Extension:** DataType

Description

A geospatial extent whose boundaries are defined by declaration or agreement by political parties.

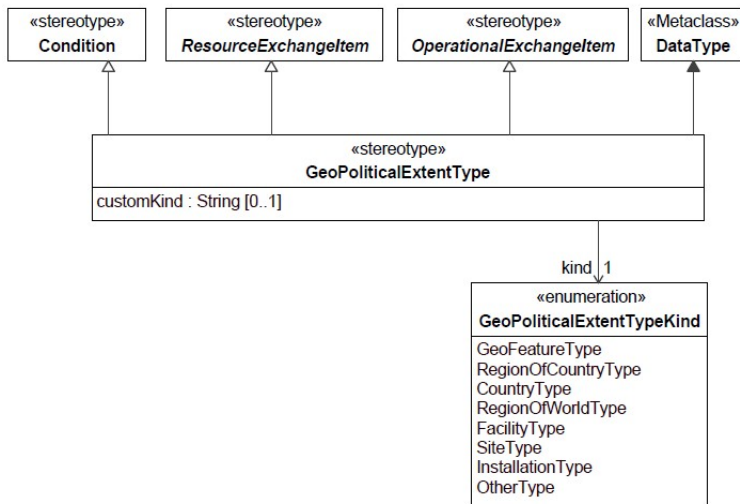


Figure 7.13 - GeoPoliticalExtentType

Attributes

customKind : String[0..1]      Captures the kind of GeopoliticalExtentType if the GeoPoliticalExtentTypeKind has been set to "OtherType."

Associations

kind : GeoPoliticalExtentTypeKind[1]      Captures the kind of GeopoliticalExtentType.

**GeoPoliticalExtentTypeKind**

**Package:** Parameters

**isAbstract:** No

Description

Enumeration of the possible kinds of GeoPoliticalExtentType. Its enumeration literals are:

- GeoFeatureType - Indicates that the GeoPoliticalExtentType associated with the GeoPoliticalExtentTypeKind is a type of object that encompasses meteorological, geographic, and control features mission significance.
- RegionOfCountryType - Indicates that the GeoPoliticalExtentType associated with the GeoPoliticalExtentTypeKind is a type of large, usually continuous segment of a political state, nation, or its territory.

- **CountryType** - Indicates that the `GeoPoliticalExtentType` associated with the `GeoPoliticalExtentTypeKind` is a type of political state, nation, or its territory.
- **RegionOfWorldType** - Indicates that the `GeoPoliticalExtentType` associated with the `GeoPoliticalExtentTypeKind` is a type of large, usually continuous segment of a surface or space; area.
- **FacilityType** - Indicates that the `GeoPoliticalExtentType` associated with the `GeoPoliticalExtentTypeKind` is a type of a real property entity consisting of underlying land and one or more of the following: a building, a structure (including linear structures), a utility system, or pavement.
- **SiteType** - Indicates that the `GeoPoliticalExtentType` associated with the `GeoPoliticalExtentTypeKind` is a type of Physical (geographic) location that is or was owned by, leased to, or otherwise possessed. Each site is assigned to a single installation. A site may exist in one of three forms: (1) Land only, where there are no facilities present and where the land consists of either a single land parcel or two or more contiguous land parcels. (2) Facility or facilities only, where the underlying land is neither owned nor controlled by the government. A stand-alone facility can be a site. If a facility is not a stand-alone facility, it must be assigned to a site. (3). Land and all the facilities thereon, where the land consists of either a single land parcel or two or more contiguous land parcels.
- **InstallationType** - Indicates that the `GeoPoliticalExtentType` associated with the `GeoPoliticalExtentTypeKind` is a type of base, camp, post, station, yard, center, or other activity, including leased facilities, without regard to the duration of operational control. An installation may include one or more sites.
- **OtherType** - Indicates that the `GeoPoliticalExtentType` associated with the `GeoPoliticalExtentTypeKind` is a type not covered by the standard `GeoPoliticalExtentTypeKinds`.

## **Location**

**Package:** Parameters

**isAbstract:** No

**Generalization:** [ConceptItem](#), [Condition](#)

**Extension:** `DataType`

Description

A specification of the generic area in which a `LocationHolder` is required to be located.



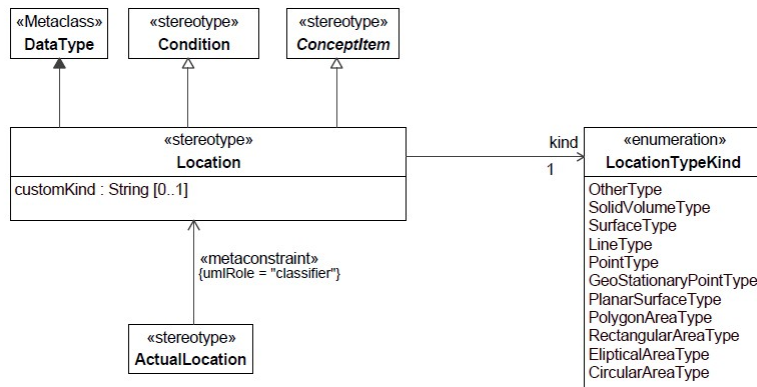


Figure 7.14 - Location

Attributes

customKind : String[0..1] Captures the kind of Location if the LocationTypeKind has been set to "OtherType."

Associations

kind : LocationTypeKind[1] Captures the kind of Location.

**LocationHolder Package:**

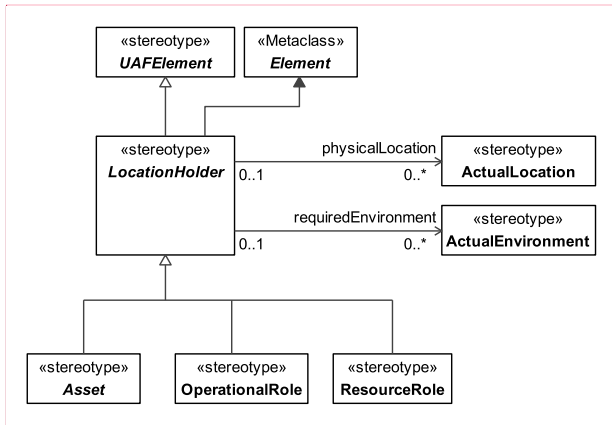
Parameters **isAbstract**: Yes

**Generalization**: [UAFElement](#)

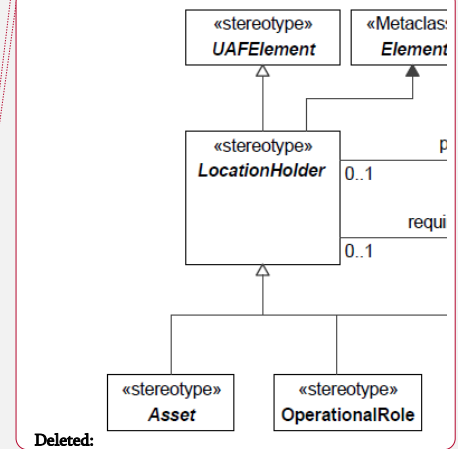
**Extension**: Element

Description

Abstract grouping used to define elements that are allowed to be associated with a Location.



Commented [AM6]: UAF11-15 Figure 7.15 – LocationHolder replaced by LocationHolder.svg



Deleted:

Figure 7.15 - LocationHolder

Associations

- physicalLocation : ActualLocation[0..\*] Relates a LocationHolder (i.e., OperationalPerformer, OperationalRole, ResourceRole, etc.) to its ActualLocation.
- requiredEnvironment : Environment[0..\*] Relates a LocationHolder (i.e., OperationalPerformer, OperationalRole, ResourceRole etc.) to the Environment in which it is required to perform/be used.

LocationKind

Package: Parameters

isAbstract: No

Description

Enumeration of the possible kinds of location applicable to an ActualLocation. Its enumeration literals are:

- SolidVolume - Indicates that the ActualLocation associated with the LocationKind is the amount of space occupied by a three-dimensional object of definite shape; not liquid or gaseous.
- Surface - Indicates that the ActualLocation associated with the LocationKind is a portion of space having length and breadth but no thickness or regards to time.
- Line - Indicates that the ActualLocation associated with the LocationKind is a geometric figure formed by a point moving along a fixed direction and the reverse direction.
- Point - Indicates that the ActualLocation associated with the LocationKind is a unidimensional Individual.

- GeoStationaryPoint - Indicates that the ActualLocation associated with the LocationKind is a unidimensional Individual.
- PlanarSurface - Indicates that the ActualLocation associated with the LocationKind is a two-dimensional portion of space.
- PolygonArea - Indicates that the ActualLocation associated with the LocationKind is a space enclosed by a polygon.
- RectangularArea - Indicates that the ActualLocation associated with the LocationKind is a space enclosed by a rectangle.
- EllipticalArea - Indicates that the ActualLocation associated with the LocationKind is a space enclosed by an ellipse.
- CircularArea - Indicates that the ActualLocation associated with the LocationKind is a space enclosed by a circle.
- Other - Indicates that the ActualLocation associated with the LocationKind is a LocationKind that is not on the enumerated list.

### LocationTypeKind

**Package:** Parameters

**isAbstract:** No

Description

Enumeration of the possible kinds of location type that are applicable to a Location. Its enumeration literals are:

- OtherType - Indicates that the Location associated with the LocationTypeKind describes a type of is a LocationKindType that is not on the enumerated list.
- SolidVolumeType - Indicates that the Location associated with the LocationTypeKind describes a type of amount of space occupied by a three-dimensional object of definite shape; not liquid or gaseous.
- SurfaceType - Indicates that the Location associated with the LocationTypeKind describes a type of portion of space having length and breadth but no thickness or regards to time.
- LineType - Indicates that the Location associated with the LocationTypeKind describes a type of geometric figure formed by a point moving along a fixed direction and the reverse direction.
- PointType - Indicates that the Location associated with the LocationTypeKind describes a type of unidimensional Individual.
- GeoStationaryPointType - Indicates that the Location associated with the LocationTypeKind describes a type of unidimensional Individual.
- PlanarSurfaceType - Indicates that the Location associated with the LocationTypeKind describes a type of is a two-dimensional portion of space.
- PolygonAreaType - Indicates that the Location associated with the LocationTypeKind describes a type of space enclosed by a polygon.
- RectangularAreaType - Indicates that the Location associated with the LocationTypeKind describes a type of space enclosed by a rectangle.
- EllipticalAreaType - Indicates that the Location associated with the LocationTypeKind describes a type of space enclosed by an ellipse.
- CircularAreaType - Indicates that the Location associated with the LocationTypeKind describes a type of space enclosed by a circle.

### MeasurableElement

**Package:** Parameters

**isAbstract:** Yes

**Generalization:** [UAFElement](#)

**Extension:** Element

Description

Abstract grouping for elements that can be measured by applying MeasurementSets to them.

Deleted: <object>

Commented [AM7]: UAF11-117, UAF11-56 Figure 7.16  
 – MeasurableElement replaced by MeasurableElement.svg

Formatted: Line spacing: single

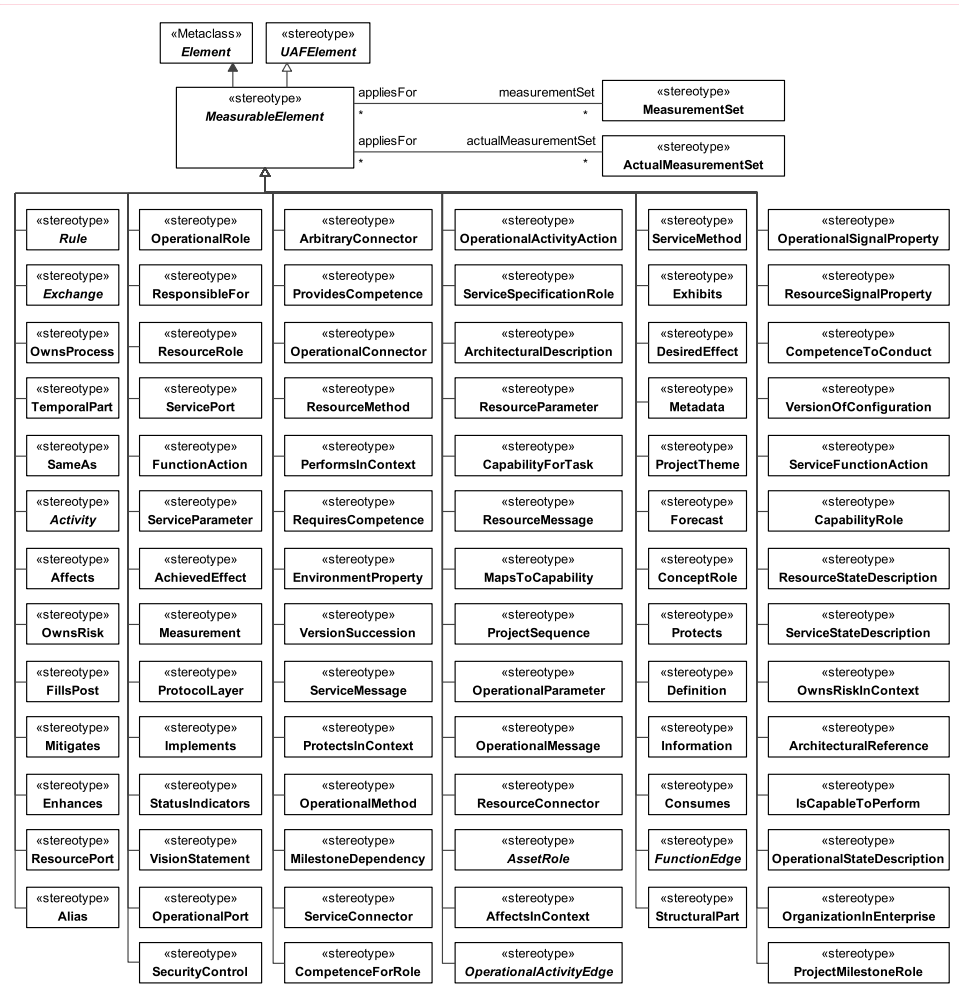


Figure 7.16 - MeasurableElement

Associations

actualMeasurementSet : ActualMeasurementSet[*]	Relates the MeasurableElement to the ActualMeasurementSet that provides its ActualMeasurements.
measurementSet : MeasurementSet[*]	Relates the MeasurableElement to the MeasurementSet that provides its Measurements by which it can be measured.

**Measurement**

**Package:** Parameters

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Property

Description

A property of an element representing something in the physical world, expressed in amounts of a unit of measure.

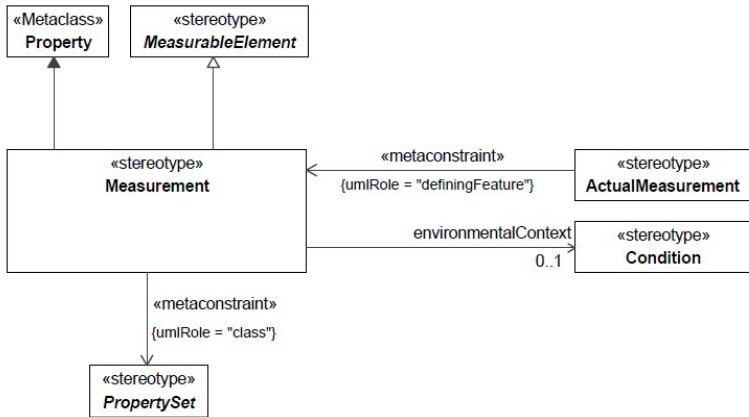


Figure 7.17 - Measurement

Associations

environmentalContext : Condition[0..1] Relates the Measurement to the Condition (which provides the environmentalContext) under which the Measurement is expected to be taken.

Constraints

[1] Measurement.class Value for the class metaproperty must be stereotyped by the specialization of «PropertySet».

**MeasurementSet**

Package: Parameters

isAbstract: No

Generalization: [PropertySet](#), ValueType

Extension: DataType

Description

A collection of Measurements.

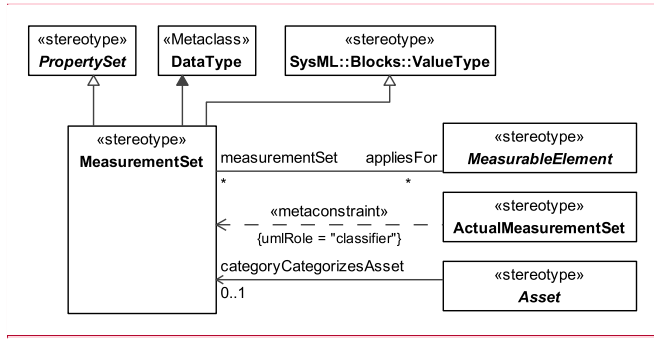


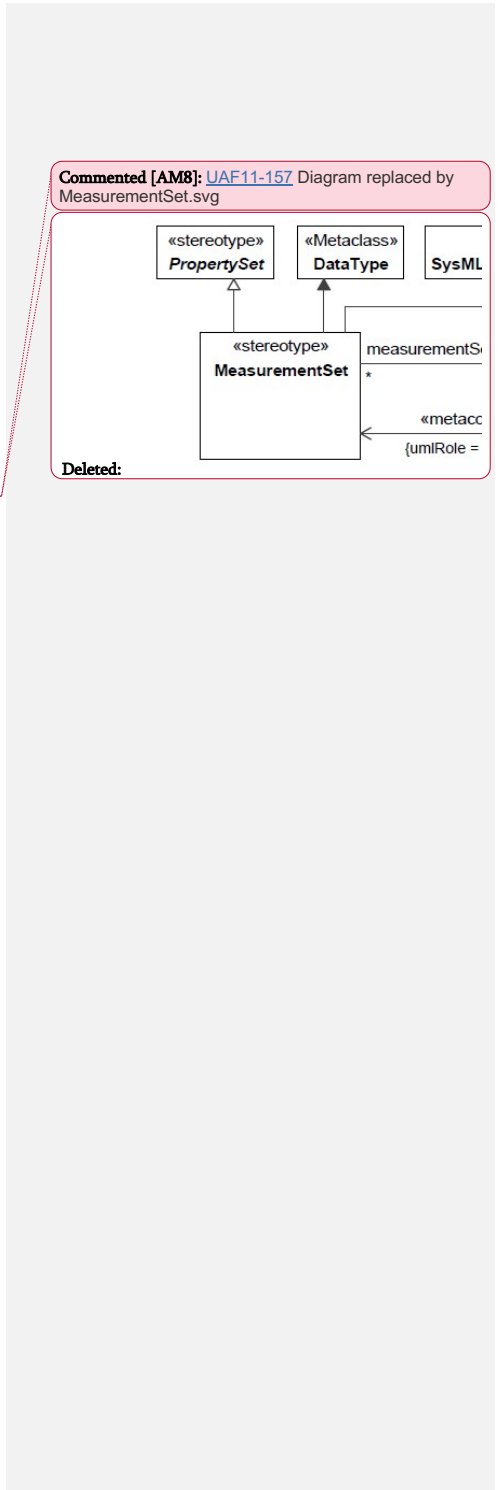
Figure 7.18 - MeasurementSet

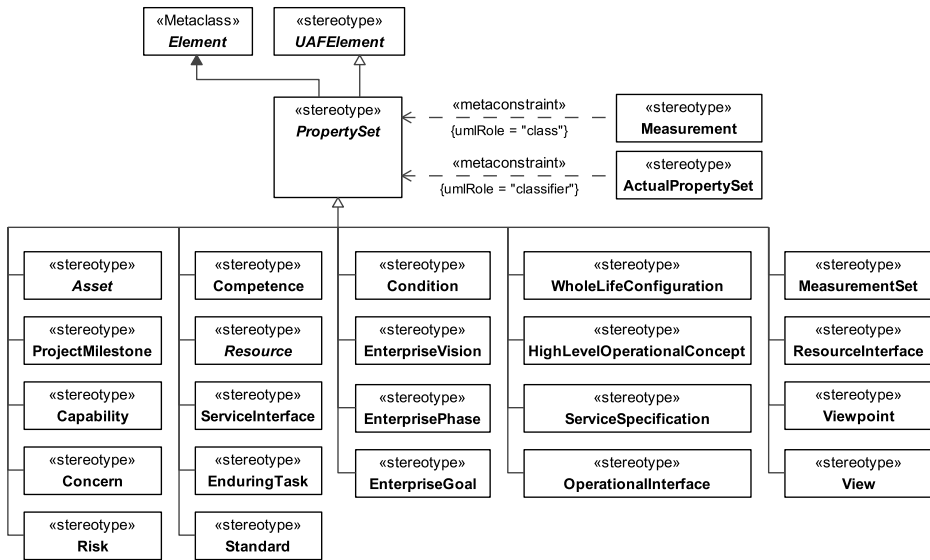
Associations

appliesFor : MeasurableElement[\*]      Relates the MeasurementSet to the MeasurableElement that it is applicable to.

**PropertySet Package:**  
 Parameters isAbstract: Yes  
 Generalization: [UAFElement](#)  
 Extension: Element

Description  
 An abstract grouping of architectural elements that can own Measurements.





**Commented [AM9]:** UAF11-116, UAF11-56 Figure 7.19  
 – PropertySet replaced by PropertySet.svg.

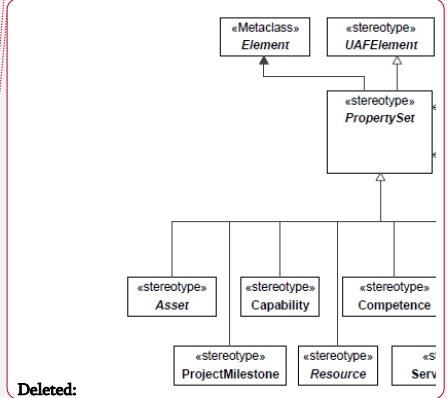


Figure 7.19 - PropertySet

### 7.1.3 UAF::Metadata

Stakeholders: Enterprise Architects, people who want to discover the architecture, Technical Managers.

Concerns: Captures meta-data relevant to the entire architecture

Definition: Provide information pertinent to the entire architecture. Present supporting information rather than architectural models.

#### 7.1.3.1 UAF::Metadata::Taxonomy

Contains the elements that contribute to the Metadata Taxonomy Viewpoint.

##### ActualState Package:

Taxonomy isAbstract: Yes

Generalization: [UAFElement](#)

Extension: Element

Description

Abstract element that applies temporal extent to a set of elements realized as Instance Specifications.



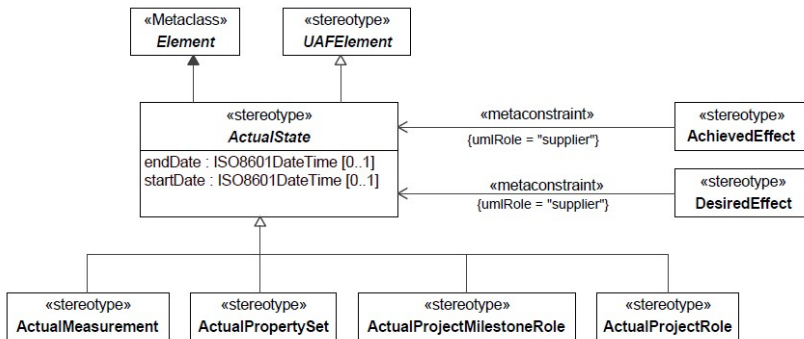


Figure 7.20 - ActualState

Attributes

- endDate : ISO8601DateTime[0..1]      End time for all "actual" elements.
- startDate : ISO8601DateTime[0..1]      Start time for all "actual" elements.

**ISO8601DateTime**

**Package:** Taxonomy

**isAbstract:** No

**Generalization:** [UAFElement](#)

**Extension:** LiteralString

Description

A date and time specified in the ISO8601 date-time format including timezone designator (TZD): YYYY-MM-DDThh:mm:ssTZD.

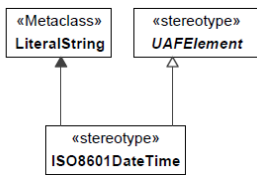


Figure 7.21 - ISO8601DateTime

### 7.1.3.2 UAF::Metadata::Connectivity

Contains the elements that contribute to the Metadata Connectivity Viewpoint.

#### Exchange

**Package:** Connectivity

**isAbstract:** Yes

**Generalization:** [MeasurableElement](#), [ItemFlow](#)

**Extension:** [InformationFlow](#)

Description

Abstract grouping for OperationalExchanges and ResourceExchanges that exchange Resources.

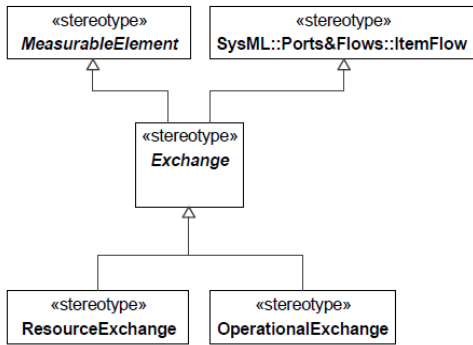


Figure 7.22 - Exchange

#### Resource

**Package:** Connectivity

**isAbstract:** Yes

**Generalization:** [PropertySet](#)

**Extension:** [Element](#)

Description

Abstract element grouping for all elements that can be conveyed by an Exchange.

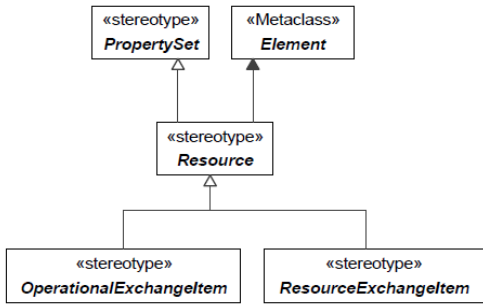


Figure 7.23 - Resource

### 7.1.3.3 UAF::Metadata::Processes

Contains the elements that contribute to the Metadata Processes Viewpoint.

Activity

**Package:** Processes

**isAbstract:** Yes

**Generalization:** [MeasurableElement](#)

**Extension:** Activity

Description

An abstract element that represents a behavior or process (i.e., a Function or OperationalActivity) that can be performed by a Performer.

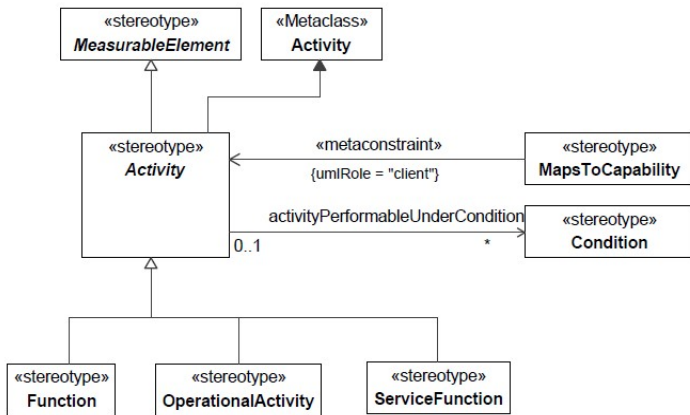


Figure 7.24 - Activity

Associations

activityPerformableUnderCondition : Condition[\*]      The environment under which an activity is performed.

**CapableElement Package:**

Processes isAbstract: Yes

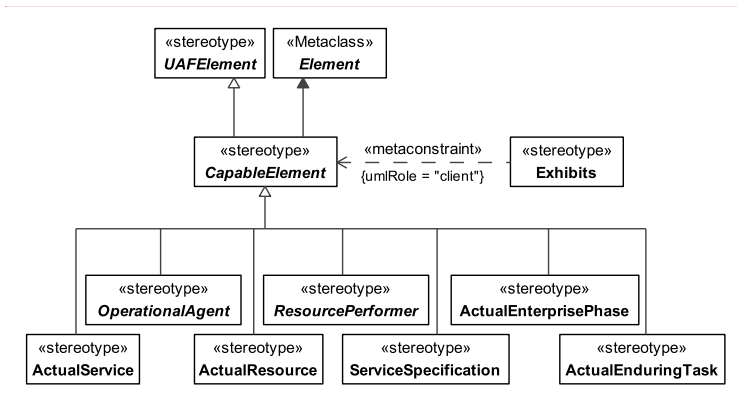
Generalization: [UAFElement](#)

Extension: Element

Description

An abstract element that represents a structural element that can perform behaviors (i.e., OperationalActivity).





Commented [AM10]: UAF11-128 Figure 7.25 – CapableElement replaced by CapableElement.svg

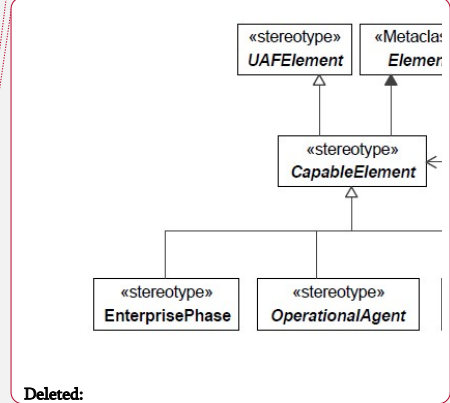


Figure 7.25 - CapableElement

**IsCapableToPerform**

**Package:** Processes

**isAbstract:** No

**Generalization:** [MeasurableElement](#), Allocate

**Extension:** Abstraction

Description

An Abstraction relationship defining the traceability between the CapableElements to the Activities that they can perform.

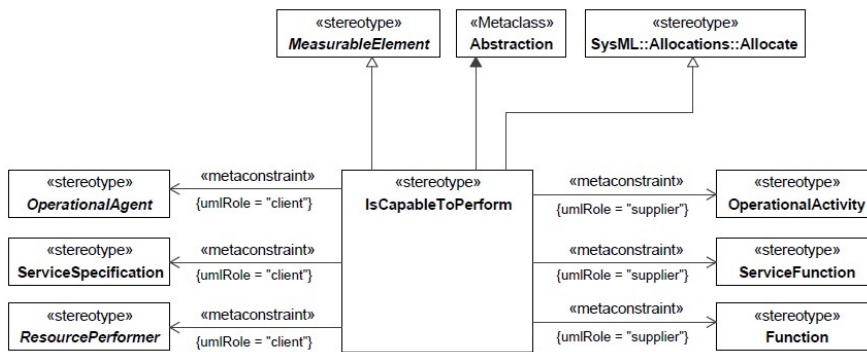


Figure 7.26 - IsCapableToPerform

Constraints

- [1] IsCapableOfPerforming.client      In case of value for IsCapableToPerform.supplier is stereotyped:
- a. «OperationalActivity» or its specializations, values for the client metaproperty must be stereotyped by any of specializations of «OperationalAgent».
  - b. «ServiceFunction» or its specializations, values for the client metaproperty must be stereotyped «ServiceSpecification» or its specializations.
  - c. «Function» or its specializations, except for «ProjectActivity», values for the client metaproperty must be stereotyped by any of specializations of «ResourcePerformer».
  - d. «ProjectActivity» or its specializations, values for the client metaproperty must be stereotyped by any of specializations of «Project».
- [2] IsCapableOfPerforming.supplier      In case of value for IsCapableToPerform.client is stereotyped:
- a. by a specialization of «OperationalAgent», values for the supplier metaproperty must be stereotyped «OperationalActivity» or its specializations.
  - b. «ServiceSpecification» or its specializations, values for the supplier metaproperty must be stereotyped «ServiceFunction» or its specializations.
  - c. by a specialization of «ResourcePerformer», values for the supplier metaproperty must be stereotyped «Function» or its specializations, except for «ProjectActivity».
  - d. by a specialization of «Project», values for the supplier metaproperty must be stereotyped «ProjectActivity» or its specializations.

## PerformsInContext

**Package:** Processes

**isAbstract:** No

**Generalization:** [MeasurableElement](#), [Allocate](#)

**Extension:** Abstraction

Description

An abstraction relationship that relates an OperationalAction to a OperationalRole, or a FunctionAction to a ResourceRole. It indicates that the action can be carried out by the role when used in a specific context or configuration.

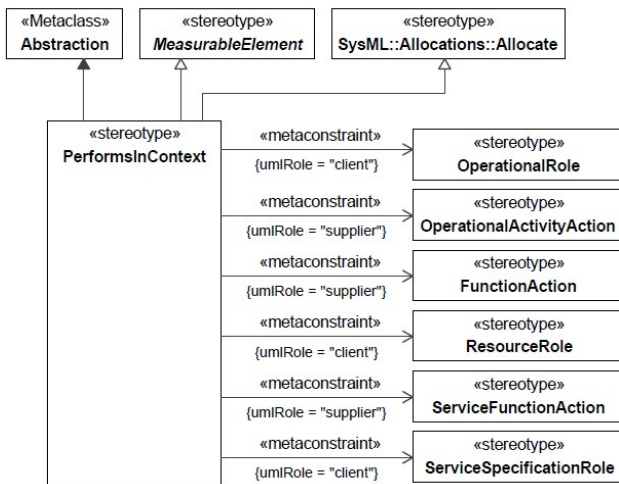


Figure 7.27 - PerformsInContext

Constraints

[1] PerformsInContext.client

In case of value for PerformsInContext.supplier is stereotyped:

- «OperationalActivityAction» or its specializations, values for the client metaproperty must be stereotyped «OperationalRole» or its specializations.
- «ServiceFunctionAction» or its specializations, values for the client metaproperty must be stereotyped «ServiceSpecificationRole» or its specializations.
- «FunctionAction» or its specializations, except for «ProjectActivityAction», values for the client metaproperty must be stereotyped «ResourceRole» or its specializations.
- «ProjectActivityAction» or its specializations, values for the client metaproperty must be stereotyped «ProjectRole» or its specializations.



[2] PerformsInContext.supplier In case of value for PerformsInContext.client is stereotyped:

- a. «OperationalRole» or its specializations, values for the supplier metaproperty must be stereotyped «OperationalActivityAction» or its specializations.
- b. «ServiceSpecificationRole» or its specializations, values for the supplier metaproperty must be stereotyped «ServiceFunctionAction» or its specializations.
- c. «ResourceRole» or its specializations, values for the supplier metaproperty must be stereotyped «FunctionAction» or its specializations.

### 7.1.3.4 UAF::Metadata::Information

Contains the elements that contribute to the Metadata Information Viewpoint.

ArchitectureMetadata

**Package:** Information

**isAbstract:** No

**Generalization:** [Metadata](#)

**Extension:** Comment

Description

Information associated with an ArchitecturalDescription, that supplements the standard set of tags used to summarize the Architecture. It states things like what methodology was used, notation, etc.

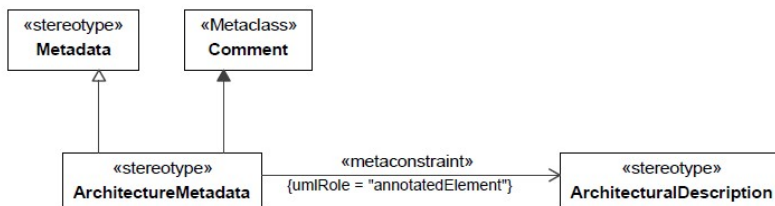


Figure 7.28 - ArchitectureMetadata

Constraints

- [1] ArchitectureMetadata.annotatedElement Value for the annotatedElement metaproperty must be stereotyped «ArchitecturalDescription» or its specializations.

## Information

**Package:** Information

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Comment

Description

A comment that describes the state of an item of interest in any medium or form -- and is communicated or received.

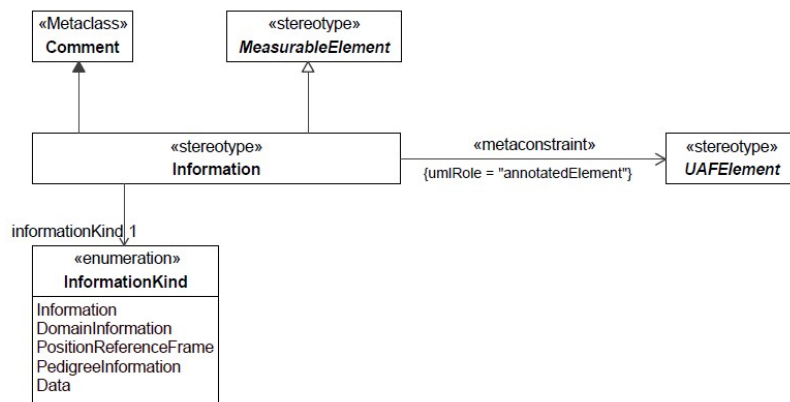


Figure 7.29 - Information

Associations

informationKind : InformationKind[1]      Captures the kind of information.

Constraints

[1] Information.annotatedElement      Value for the annotatedElement metaproperty must be stereotyped by a specialization of **«UAFElement»**.

**Commented [BY(11)]:** UAF11-32 changed «ConceptItem» to «UAFElement»

**Deleted:** «ConceptItem»

## InformationKind

**Package:** Information

**isAbstract:** No

Description

Enumeration of the possible kinds of Information. Its enumeration literals are:

- Information - Indicates that the Information associated with the InformationKind describes the state of a something of interest that is materialized -- in any medium or form -- and communicated or received.
- DomainInformation - Indicates that the Information associated with the InformationKind describes information within the scope or domain of the architecture.
- PositionReferenceFrame - Indicates that the Information associated with the InformationKind describes an arbitrary set of axes with reference to which the position or motion of something is described or physical laws are formulated.
- PedigreeInformation - Indicates that the Information associated with the InformationKind describes information pedigree.
- Data - Indicates that the Information associated with the InformationKind describes the representation of information in a formalized manner suitable for communication, interpretation, or processing by humans or by automatic means. Examples could be whole models, packages, entities, attributes, classes, domain values, enumeration values, records, tables, rows, columns, and fields.

**Metadata**

**Package:** Information

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Comment

Description

A comment that can be applied to any element in the architecture. The attributes associated with this element details the relationship between the element and its related dublinCoreElement, metaDataScheme, category, and name. This allows the element to be referenced using the Semantic Web.

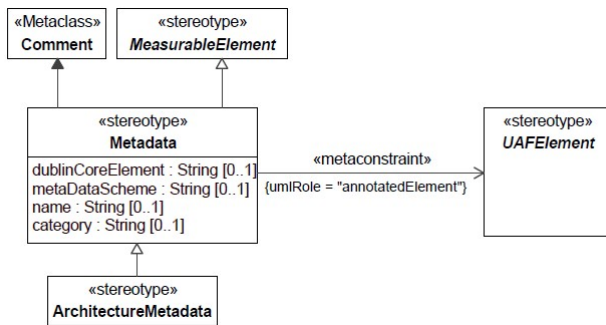


Figure 7.30 - Metadata

#### Attributes

category : String[0..1]	Defines the category of a Metadata element example: <a href="http://purl.org/dc/terms/abstract.dublinCoreElement">http://purl.org/dc/terms/abstract.dublinCoreElement</a> :
String[0..1]	A metadata category that is a DublinCore tag.
metaDataScheme : String[0..1]	A representation scheme that defines a set of Metadata.
name : String[0..1]	The name of the Metadata.

Constraints	
[1] Metadata.annotatedElement	Value for the annotatedElement metaproperty must be stereotyped by a specialization of «UAFElement».

#### 7.1.3.5 UAF::Metadata::Constraints

Contains the elements that contribute to the Metadata Constraints Viewpoint.

##### Rule

**Package:** Constraints

**isAbstract:** Yes

**Generalization:** [MeasurableElement](#)

**Extension:** Constraint

##### Description

An abstract grouping for all types of constraint (i.e. an OperationalConstraint could detail the rules of accountancy best practice).

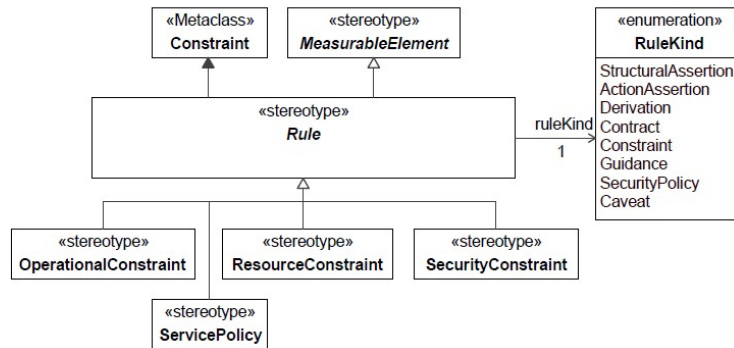


Figure 7.31 - Rule

Associations

ruleKind : RuleKind[1]      Captures the kind of Rule that is being described.

**RuleKind**

**Package:** Constraints

**isAbstract:** No

Description

Enumeration of the possible kinds of Rules applicable to constraints. Its enumeration literals are:

- StructuralAssertion - Indicates that the Rule associated with the RuleKind is a statement that details that something of importance either exists as a concept of interest or exists in relationship to another thing of interest.
- ActionAssertion - Indicates that the Rule associated with the RuleKind is a statement that concerns some dynamic aspect.
- Derivation - Indicates that the Rule associated with the RuleKind is a statement that details a Rule derived from another Rule.
- Contract - Indicates that the Rule associated with the RuleKind is a statement that details a consent among parties regarding the terms and conditions of activities that said parties participate in.
- Constraint - Indicates that the Rule associated with the RuleKind is a statement that details a limitation, e.g., business rule, restraint, operational limitation.
- Guidance - Indicates that the Rule associated with the RuleKind is a statement that details an authoritative statement intended to lead or steer the execution of actions.
- SecurityPolicy - Indicates that the Rule associated with the RuleKind is a statement that details a constraint that specifies policy for information handling, physical security, encryption, etc.

- Caveat - Indicates that the Rule associated with the RuleKind is a statement that details alternate conditions under which the rule is not valid.

### 7.1.3.6 UAF::Metadata::Traceability

Contains the elements that contribute to the Metadata Traceability Viewpoint.

#### ArchitecturalReference

**Package:** Traceability

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Dependency

Description

A dependency relationship that specifies that one architectural description refers to another.

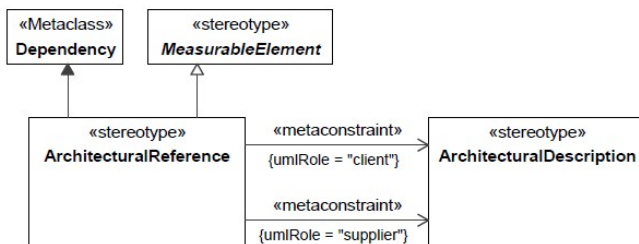


Figure 7.32 - ArchitecturalReference

#### Constraints

- |                                     |  |
|-------------------------------------|--|
| [1] ArchitecturalReference.client   | Value for the client metaproperty must be stereotyped «ArchitecturalDescription» or its specializations.   |
| [2] ArchitecturalReference.supplier | Value for the supplier metaproperty must be stereotyped «ArchitecturalDescription» or its specializations. |

#### Implements

**Package:** Traceability

**isAbstract:** No

**Generalization:** [MeasurableElement](#), Allocate

**Extension:** Abstraction

Description

An Abstraction relationship that defines how an element in the upper layer of abstraction is implemented by a semantically equivalent element (i.e., tracing the OperationalActivities to the Functions that implement them) in the lower level of abstraction.

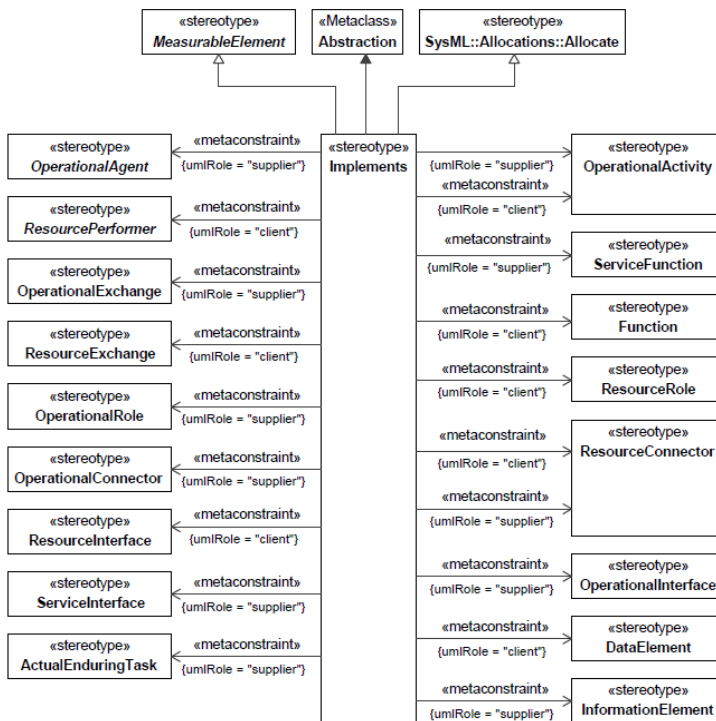


Figure 7.33 - Implements

Constraints

- [1] Implements.client In case of value for Implements.supplier is stereotyped:
- a. by any of specializations of «OperationalAgent», values for the client metaproperty must be stereotyped by any of specializations of «ResourcePerformer».
  - b. «OperationalActivity» or its specializations, values for the client metaproperty must be stereotyped «Function» or its specializations.

- c. «ServiceFunction» or its specializations, values for the client metaproperty must be stereotyped «Function» or its specializations.
- d. «ServiceInterface» or its specializations, values for the client metaproperty must be stereotyped «ResourceInterface» or its specializations.
- e. «OperationalInterface» or its specializations, values for the client metaproperty must be stereotyped «ResourceInterface» or its specializations.
- f. «OperationalConnector» or its specializations, values for the client metaproperty must be stereotyped «ResourceConnector» or its specializations.
- g. «OperationalExchange» or its specializations, values for the client metaproperty must be stereotyped «ResourceExchange» or its specializations.
- g. «OperationalRole» or its specializations, values for the client metaproperty must be stereotyped «ResourceRole» or its specializations.
- h. «ResourceConnector» or its specializations, values for the client metaproperty must be stereotyped «ResourceConnector» or its specializations.
- i. «ActualEnduringTask» or its specializations, values for the client metaproperty must be stereotyped «OperationalActivity» or its specializations.
- j. «InformationElement» or its specializations, values for the client metaproperty must be stereotyped «DataElement» or its specializations.

[2] Implements.supplier In case of value for Implements.client is stereotyped:

- a. by any of specializations of «ResourcePerformer», values for the supplier metaproperty must be stereotyped by any of specializations of «OperationalAgent».
- b. «Function» or its specializations, values for the supplier metaproperty must be stereotyped «OperationalActivity», «ServiceFunction» or their specializations.
- c. «ResourceInterface» or its specializations, values for the supplier metaproperty must be stereotyped «ServiceInterface», «OperationalInterface», or their specializations.**
- e. «ResourceConnector» or its specializations, values for the supplier metaproperty must be stereotyped «OperationalConnector», «ResourceConnector» or their specializations.
- f. «ResourceExchange» or its specializations, values for the supplier metaproperty must be stereotyped «OperationalExchange» or its specializations.
- g. «ResourceRole» or its specializations, values for the supplier metaproperty must be stereotyped «OperationalRole» or its specializations.
- h. «OperationalActivity» or its specializations, values for the supplier metaproperty must be stereotyped «ActualEnduringTask» or its specializations.
- i. «DataElement» or its specializations, values for the supplier metaproperty must be stereotyped «InformationElement» or its specializations.

## 7.1.4 UAF::Strategic

Stakeholders: Capability Portfolio Managers.

Concerns: capability management process.

Definition: describe capability taxonomy, composition, dependencies and evolution.

### 7.1.4.1 UAF::Strategic::Taxonomy

Contains the elements that contribute to the Strategic Taxonomy Viewpoint.

#### ActualEnterprisePhase

Package: Taxonomy

isAbstract: No

Formatted: Not Expanded by / Condensed by

Formatted: Indent: Left: 4 cm, Hanging: 0.25 cm, Space Before: 0 pt, Line spacing: Exactly 11.4 pt, Tab stops: 4.25 cm, Left

Commented [AM12]: [UAF11-33](#) Constraints [c] and [d] merged into one. It resulted in the text change from "c. «ResourceInterface» or its specializations, values for the supplier metaproperty must be stereotyped «ServiceInterface» or its specializations. d. «ResourceInterface» or its specializations, values for the supplier metaproperty must be stereotyped «OperationalInterface» or its specializations." to "c. «ResourceInterface» or its specializations, values for the supplier metaproperty must be stereotyped «ServiceInterface», «OperationalInterface», or their specializations,".

Deleted: c. «ResourceInterface» or its specializations, values for the supplier metaproperty must be stereotyped «ServiceInterface» or its specializations.

Formatted ... [2]

Deleted: d. «ResourceInterface» or its specializations, values for the supplier metaproperty must be stereotyped «OperationalInterface» or its specializations.

Formatted ... [3]

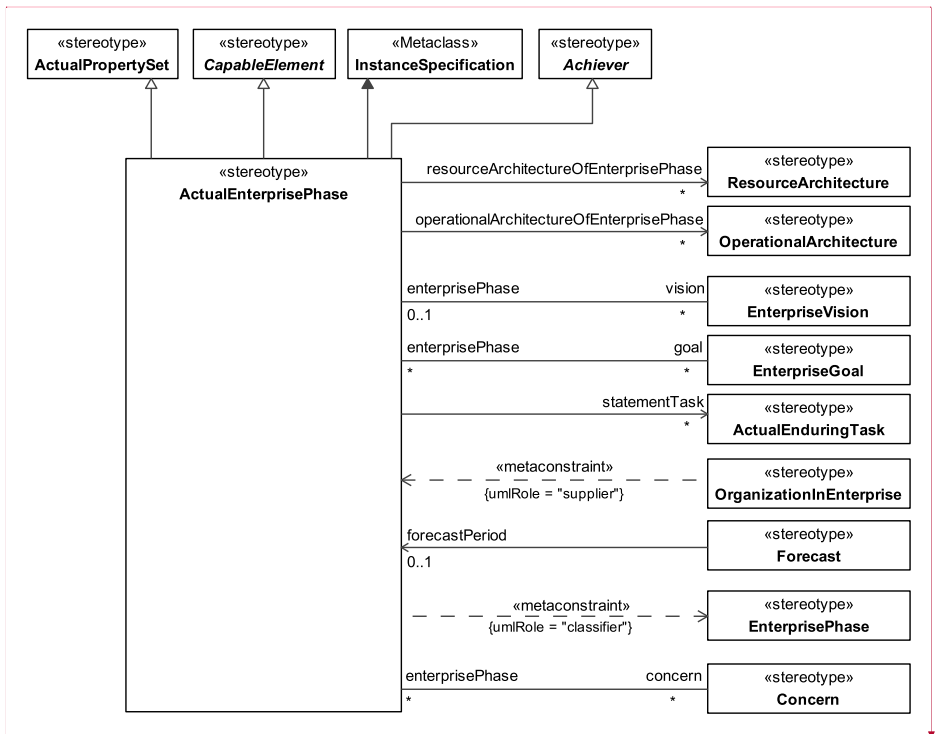


**Generalization:** [ActualPropertySet](#), [CapableElement](#), [Achiever](#)

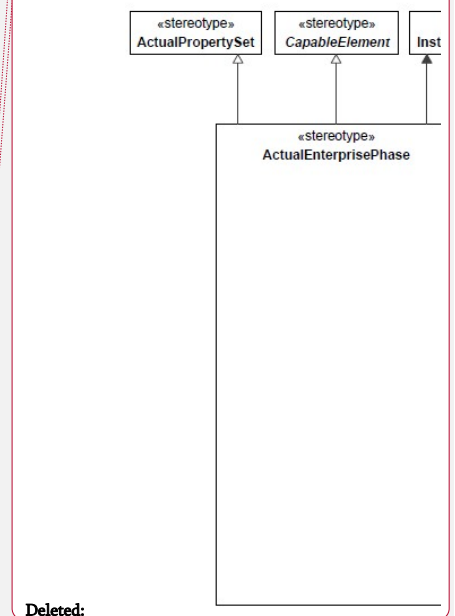
**Extension:** InstanceSpecification

Description

An ActualState that describes the phase of an Enterprise endeavor.



**Commented [AM13]:** UAF11-36 Figure 7.34 – ActualEnterprisePhase replaced by ActualEnterprisePhase.svg



**Deleted:**

Figure 7.34 - ActualEnterprisePhase

**Associations**

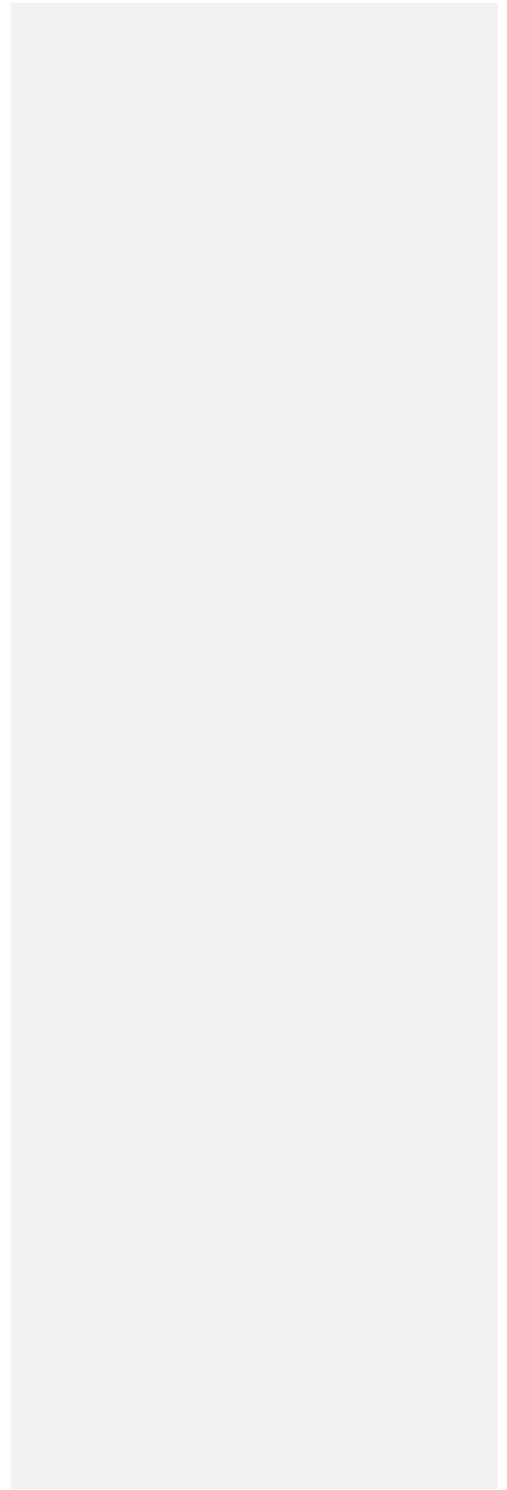
goal : EnterpriseGoal[\*]                      The Goal towards which this Phase is directed and is in support of.

operationalArchitectureOfEnterprisePhase :                      Relates an ActualEnterprisePhase to its relevant  
OperationalArchitecture[\*]                      OperationalArchitecture.

resourceArchitectureOfEnterprisePhase :                      Relates an ActualEnterprisePhase to its relevant ResourceArchitecture.  
ResourceArchitecture[\*]

statementTask : ActualEnduringTask[\*]

Relates the ActualEnterprisePhase to the ActualEnduringTasks that are intended to be implemented during that phase.



vision : EnterpriseVision[\*]                      The Vision towards which this Phase is directed and is in support of.

Constraints

- [1] ActualEnterprisePhase.classifier            Value for the classifier metaproperty must be stereotyped by «EnterprisePhase» or its specializations.
- [2] ActualEnterprisePhase.start/endDate      Must fall within the start and end dates of the enclosing ActualEnterprisePhase having this ActualEnterprisePhase set as a value for a slot.

**Capability**

**Package:** Taxonomy

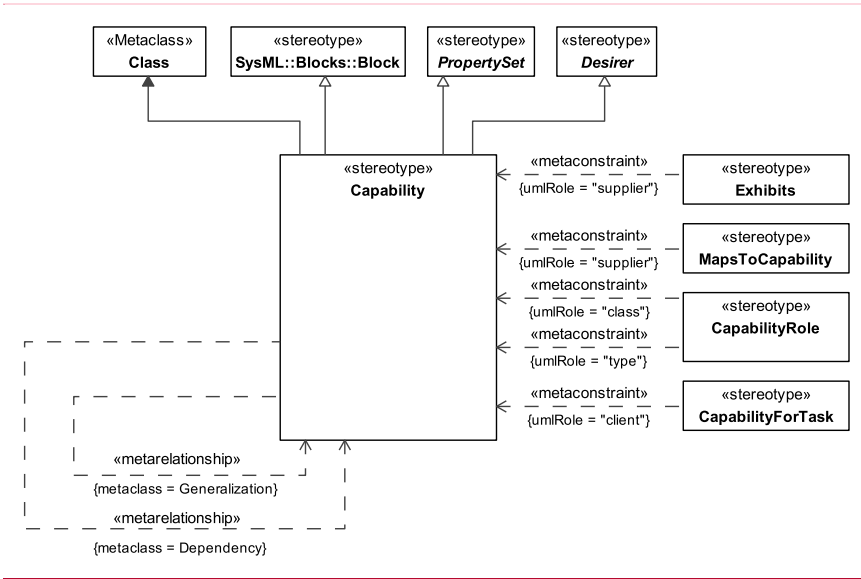
**isAbstract:** No

**Generalization:** [PropertySet](#), [Desirer](#), Block

**Extension:** Class

Description

An enterprise's ability to Achieve a DesiredEffect realized through a combination of ways and means (e.g., CapabilityConfigurations) along with specified measures.



Commented [AM14]: UAF11-150 Figure 7.35 – Capability replaced by Capability.svg

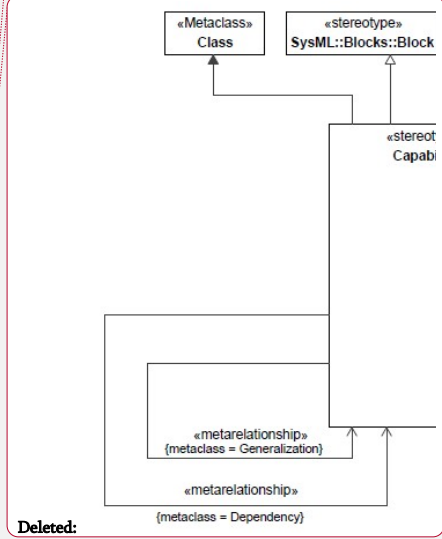


Figure 7.35 - Capability

**EnterpriseGoal**

**Package:** Taxonomy

**isAbstract:** No

**Generalization:** [PropertySet](#), Requirement

**Extension:** Class

**Description**

A statement about a state or condition of the enterprise to be brought about or sustained through appropriate Means. An EnterpriseGoal amplifies an EnterpriseVision that is, it indicates what must be satisfied on a continuing basis to effectively attain the EnterpriseVision. <http://www.omg.org/spec/BMM/1.3/>

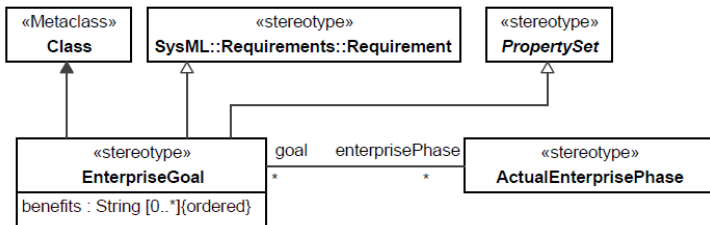


Figure 7.36 - EnterpriseGoal

Attributes

benefits : String[0..\*]      A description of the usefulness of the Goal in terms of why the state or condition of the Enterprise is worth attaining.

Associations

enterprisePhase : ActualEnterprisePhase[\*]      Relates the EnterpriseGoal to the ActualEnterprisePhase in which the EnterpriseGoal is attained.

**EnterprisePhase**

**Package:** Taxonomy

**isAbstract:** No

**Generalization:** [PropertySet](#), [Block](#)

**Extension:** Class

Description

A type of a current or future state of the enterprise.

**Commented [AM15]:** [UAF11-28](#) CapableElement removed from Generalization list.  
**Deleted:** [CapableElement](#)

**Commented [AM16]:** [UAF11-28](#) Description of the EnterprisePhase changed from “A current or future state of the wholeLifeEnterprise or another EnterprisePhase.” to “A type of a current or future state of the enterprise.”.  
**Deleted:** A current or future state of the wholeLifeEnterprise or another EnterprisePhase.

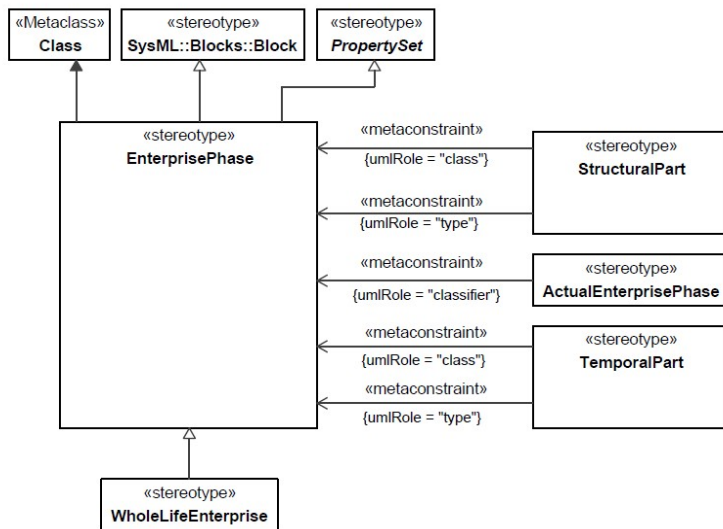


Figure 7.37 - EnterprisePhase

**EnterpriseVision**

**Package:** Taxonomy

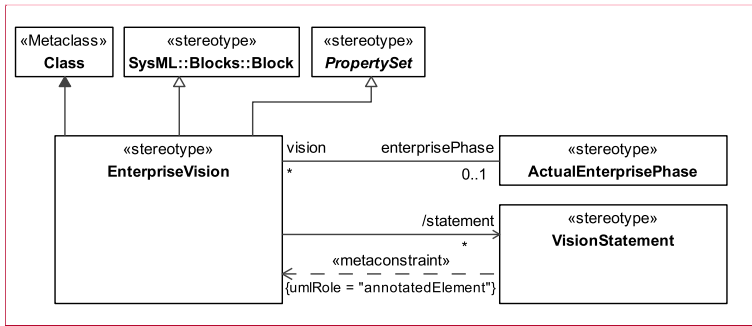
**isAbstract:** No

**Generalization:** [PropertySet](#), Block

**Extension:** Class

**Description**

A Vision describes the future state of the enterprise, without regard to how it is to be achieved.  
<http://www.omg.org/spec/BMM/1.3/>



Commented [AM17]: UAF11-39 - Figure 7.38 – EnterpriseVision replaced by EnterpriseVision.svg

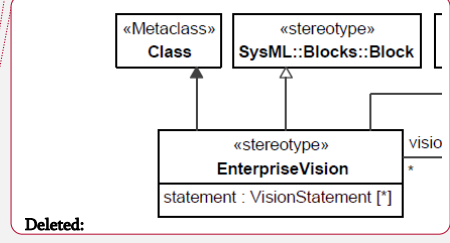


Figure 7.38 - EnterpriseVision

**Associations**

**enterprisePhase : ActualEnterprisePhase[0..1]** *Relates the EnterpriseVision to the ActualEnterprisePhase in which the EnterpriseVision is expected to be realized.*

**statement : VisionStatement[\*]** *A description of the Vision.*

**VisionStatement**

**Package:** Taxonomy  
**isAbstract:** No  
**Generalization:** [MeasurableElement](#)  
**Extension:** Comment

**Description**

A type of comment that describes the future state of the enterprise, without regard to how it is to be achieved.  
<http://www.omg.org/spec/BMM/1.3/>

Deleted:

Deleted:  
 Attributes:  
 statement : VisionStatement[\*] → A description of the Vision.

Commented [AM18]: UAF11-39 statement attribute is removed from the attributes section and added to associations section. Attributes section removed.

Formatted: Font: 10 pt

Formatted: Normal, Indent: Left: 0.18 cm, Right: -0.04 cm

Deleted: Associations:  
 enterprisePhase : ActualEnterprisePhase[0..1]  
 ..... Column Break .....  
 Relates the EnterpriseVision to the ActualEnterprisePhase in which the EnterpriseVision is expected to be realized.

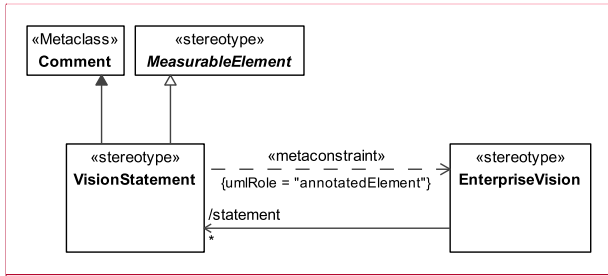


Figure 7.39 - VisionStatement

Constraints

[1] VisionStatement.ownedAttribute Values for annotatedElement metaproperty must be stereotyped «EnterpriseVision» or its specializations.

**WholeLifeEnterprise**

**Package:** Taxonomy

**isAbstract:** No

**Generalization:** [EnterprisePhase](#)

**Extension:** Class

Description

A WholeLifeEnterprise is a purposeful endeavor of any size involving people, organizations and supporting systems. It is made up of TemporalParts and StructuralParts.

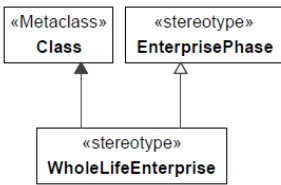
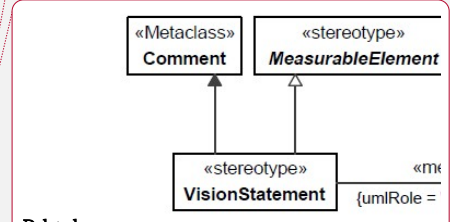


Figure 7.40 - WholeLifeEnterprise

Commented [AM19]: UAF11-39 - Figure 7.39 – Vision Statement replaced by VisionStatement.svg



Deleted:



#### 7.1.4.2 UAF::Strategic::Structure

Contains the elements that contribute to the Strategic Structure Viewpoint.

##### CapabilityProperty

**Package:** Structure

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Property

Description

Property of a Capability typed by another Capability, enabling whole-part relationships and structures.

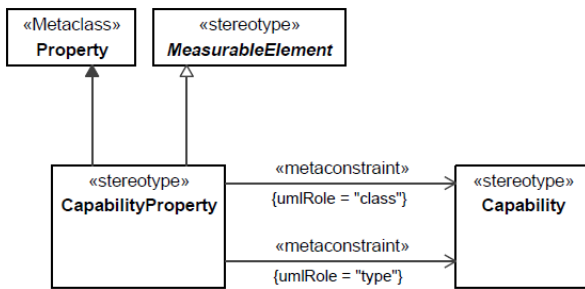


Figure 7.41 - CapabilityProperty

Constraints

- [1] CapabilityProperty.class Value for class metaproperty must be stereotyped «Capability» or its specializations.
- [2] CapabilityProperty.type Value for type metaproperty must be stereotyped «Capability» or its specializations.

##### StructuralPart

**Package:** Structure

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Property

Description

Usage of an EnterprisePhase in the context of another EnterprisePhase. It asserts that one EnterprisePhase is a spatial part of another. Creates a whole-part relationship that represents the structure of the EnterprisePhase.

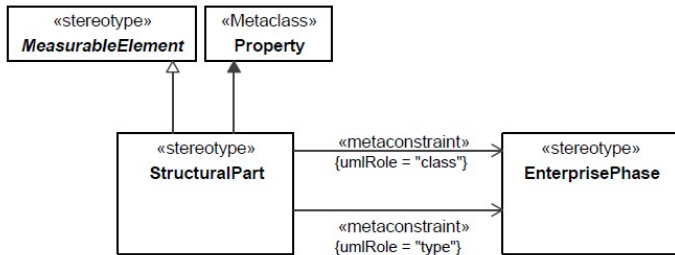


Figure 7.42 - StructuralPart

Constraints

- [1] StructuralPart.class Value for class metaproperty must be stereotyped «EnterprisePhase» or its specializations.
- [2] StructuralPart.type Value for type metaproperty must be stereotyped «EnterprisePhase» or its specializations.

**TemporalPart**

**Package:** Structure

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Property

Description

Usage of an EnterprisePhase in the context of another EnterprisePhase. It asserts that one EnterprisePhase is a spatial part of another. Creates a whole-part relationship that represents the temporal structure of the EnterprisePhase.



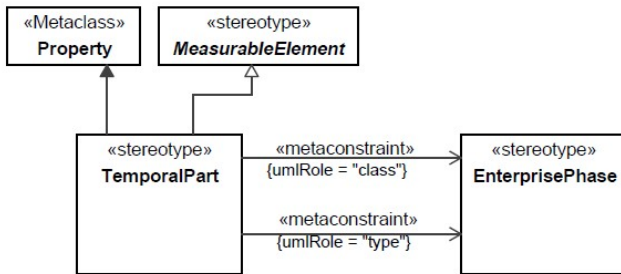


Figure 7.43 - TemporalPart

Constraints

- [1] TemporalPart.class Value for class metaproperty must be stereotyped «EnterprisePhase» or its specializations.
- [2] TemporalPart.type Value for type metaproperty must be stereotyped «EnterprisePhase» or its specializations.

**7.1.4.3 UAF::Strategic::Processes**

Contains the elements that contribute to the Strategic Processes Viewpoint.

**ActualEnduringTask**

**Package:** Processes

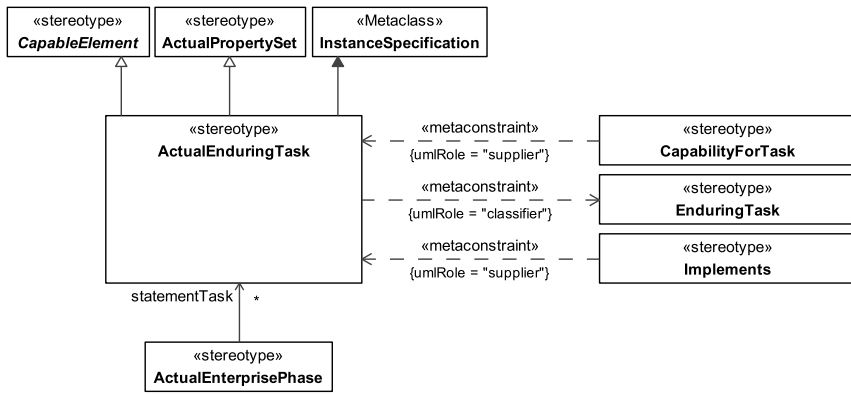
**isAbstract:** No

**Generalization:** [CapableElement](#), [ActualPropertySet](#)

**Extension:** InstanceSpecification

Description

An actual undertaking recognized by an enterprise as being essential to achieving its goals - i.e., a strategic specification of what the enterprise does.



7.44 - ActualEnduringTask

Constraints

- [1] ActualEnduringTask.classifier Value for the classifier metaproperty must be stereotyped by «EnduringTask» or its specializations.

**CapabilityForTask**

Package: Processes

isAbstract: No

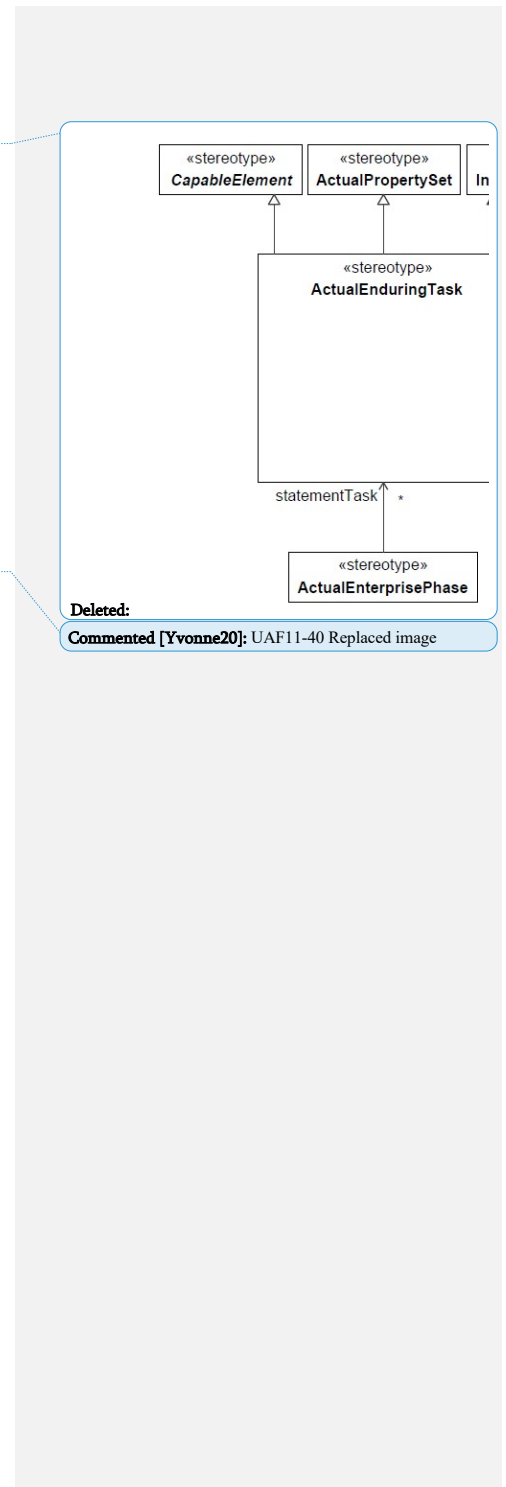
Generalization: [MeasurableElement](#), Allocate

Extension: Abstraction

Description

An abstraction relationship that asserts that a Capability is required in order for an Enterprise to conduct a phase of an EnduringTask.

Figure



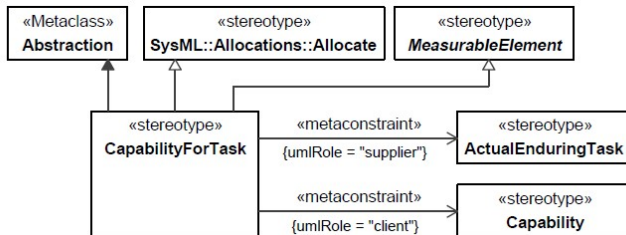


Figure 7.45 - CapabilityForTask

Constraints

- [1] CapabilityForTask.client Value for the client metaproperty must be stereotyped «Capability» or its specializations.
- [2] CapabilityForTask.supplier Value for the supplier metaproperty must be stereotyped «ActualEnduringTask» or its specializations.

**EnduringTask**

**Package:** Processes

**isAbstract:** No

**Generalization:** [PropertySet](#), Block

**Extension:** Class

Description

A type of template behavior recognized by an enterprise as being essential to achieving its goals - i.e., a template for a strategic specification of what the enterprise does.

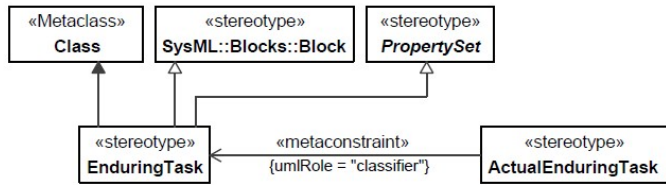


Figure 7.46 - EnduringTask

#### 7.1.4.4 UAF::Strategic::States

Contains the elements that contribute to the Strategic States Viewpoint.

#### AchievedEffect

**Package:** States

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Dependency

Description

A dependency relationship that exists between an ActualState (e.g., observed/measured during testing) of an element that attempts to achieve a DesiredEffect and an Achiever.

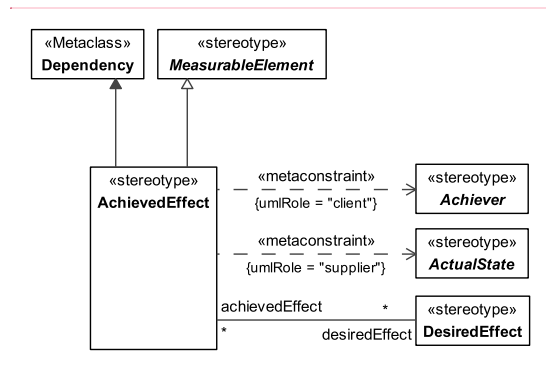
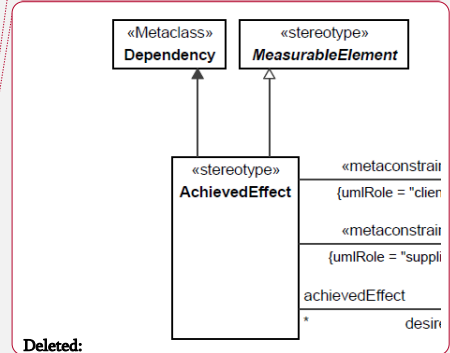


Figure 7.47 - AchievedEffect

Commented [AM21]: [UAF11-43](#) Figure 7.47 – AchievedEffect replaced by AchievedEffect.svg



Associations

desiredEffect : DesiredEffect[\*]      Relates the effect that is achieved with the originally expected DesiredEffect. Providing a means of comparison, between the expectation of the desirer and the actual result.

Constraints

- [1] AchievedEffect.client      Value for the client metaproperty must be stereotyped by the specialization of «Achiever».
- [2] AchievedEffect.supplier      Value for the supplier metaproperty must be stereotyped by the specialization of «ActualState».

**Achiever**

**Package:** States

**isAbstract:** Yes

**Generalization:** [UAFElement](#)

**Extension:** InstanceSpecification

Description

An ActualResource, ActualProject, or ActualEnterprisePhase that can deliver a DesiredEffect.

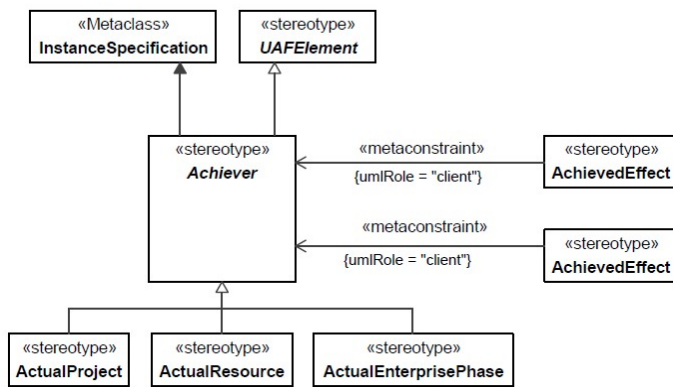


Figure 7.48 - Achiever



**DesiredEffect**

**Package:** States

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Dependency

Description

A dependency relationship relating the Desirer (a Capability or OrganizationalResource) to an ActualState.

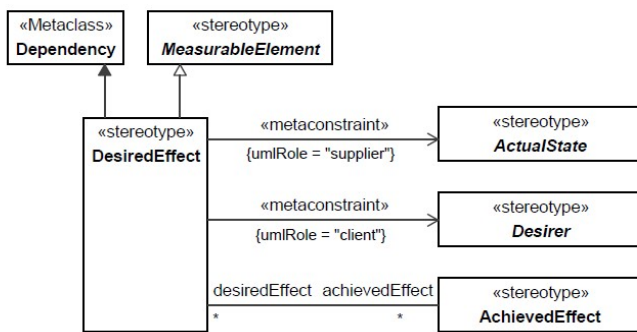


Figure 7.49 - DesiredEffect

Associations

achievedEffect : AchievedEffect[\*]

Constraints

- [1] DesiredEffect.client Value for the client metaproperty must be stereotyped a specialization of «Desirer».
- [2] DesiredEffect.supplier Value for the supplier metaproperty must be stereotyped a specialization of «ActualState».

**Desirer**

**Package:** States

**isAbstract:** Yes

**Generalization:** [UAFElement](#)

**Extension:** Class

Description

Abstract element used to group architecture elements that might desire a particular effect.

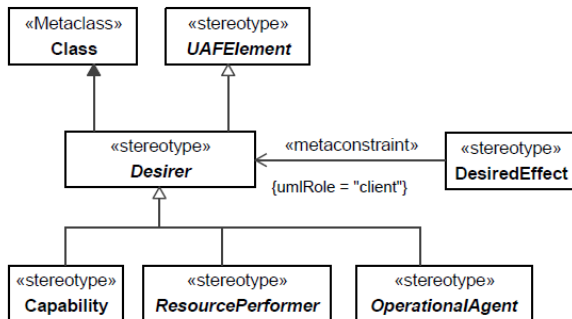


Figure 7.50 - Desirer

#### 7.1.4.5 UAF::Strategic::Traceability

Contains the elements that contribute to the Strategic Traceability Viewpoint.

**Exhibits**

**Package:** Traceability

**isAbstract:** No

**Generalization:** [MeasurableElement](#), Allocate

**Extension:** Abstraction

Description

An abstraction relationship that exists between a CapableElement and a Capability that it meets under specific environmental conditions.

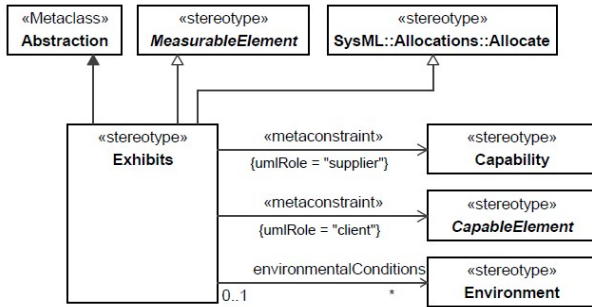


Figure 7.51 - Exhibits

Associations

environmentalConditions : Environment[\*] Defines the environmental conditions constraining the way that a Capability is exhibited.

Constraints

- [1] Exhibits.client Value for the client metaproperty must be stereotyped a specialization of «CapableElement».
- [2] Exhibits.supplier Value for the supplier metaproperty must be stereotyped «Capability».

**MapsToCapability**

**Package:** Traceability

**isAbstract:** No

**Generalization:** [MeasurableElement](#), [Allocate](#)

**Extension:** Abstraction

Description

An Abstraction relationship denoting that an Activity contributes to providing a Capability.

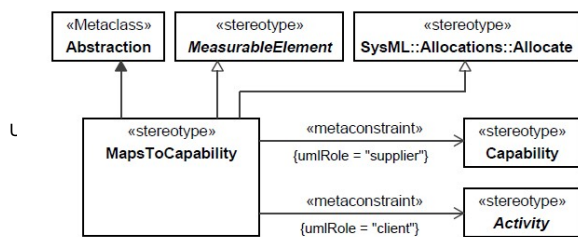


Figure 7.52 - MapsToCapability

Constraints

- [1] MapsToCapability.client Value for the client metaproperty must be stereotyped a specialization of «Activity».
- [2] MapsToCapability.supplier Value for the supplier metaproperty must be stereotyped «Capability».

**OrganizationInEnterprise**

**Package:** Traceability

**isAbstract:** No

**Generalization:** [MeasurableElement](#), Allocate

**Extension:** Abstraction

Description

An abstraction relationship relating an ActualOrganization to an ActualEnterprisePhase to denote that the ActualOrganization plays a role or is a stakeholder in an ActualEnterprisePhase.

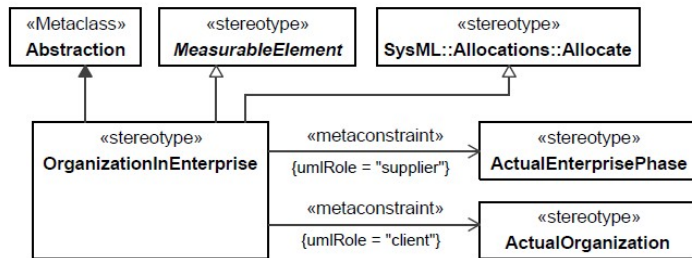


Figure 7.53 - OrganizationInEnterprise

Constraints

- [1] OrganizationInEnterprise.client Value for the client metaproperty must be stereotyped «ActualOrganization» or its specializations.
- [2] OrganizationInEnterprise.supplier Value for the supplier metaproperty must be stereotyped «ActualEnterprisePhase» or its specializations.

**7.1.5 UAF::Operational**

Stakeholders: Business Architects, Executives.

Concerns: illustrate the Logical Architecture of the enterprise.

Definition: describe the requirements, operational behavior, structure, and exchanges required to support (exhibit) capabilities. Defines all operational elements in an implementation/solution independent manner.

**7.1.5.1 UAF::Operational::Taxonomy**

Contains the elements that contribute to the Operational Taxonomy Viewpoint.

**ArbitraryConnector**

**Package:** Taxonomy

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Dependency

Description

Represents a visual indication of a connection used in high level operational concept diagrams.

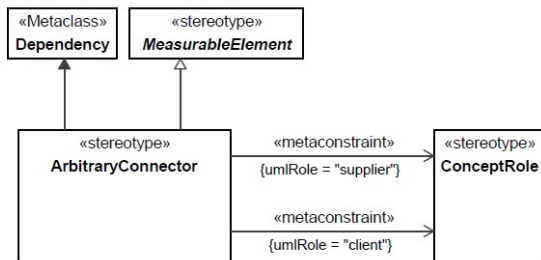


Figure 7.54 - ArbitraryConnector

Constraints

- [1] ArbitraryConnector.client      The value for client metaproperty has to be stereotyped «ConceptRole» or its specializations.
- [2] ArbitraryConnector.supplier      The value for supplier metaproperty has to be stereotyped «ConceptRole» or its specializations.

**ConceptItem Package:**

Taxonomy **isAbstract:** Yes

**Generalization:** [UAFElement](#)

**Extension:** Element

Description

Abstract, an item which may feature in a HighLevelOperationalConcept.

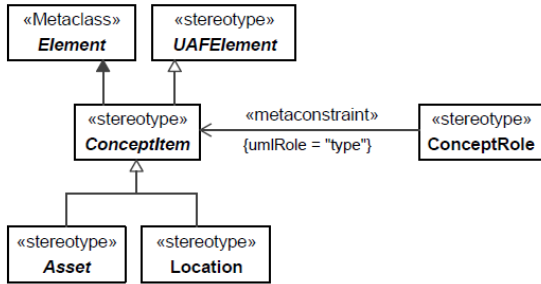


Figure 7.55 - ConceptItem

**ConceptRole**

**Package:** Taxonomy

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Property

Description

Usage of a ConceptItem in the context of a HighLevelOperationalConcept.

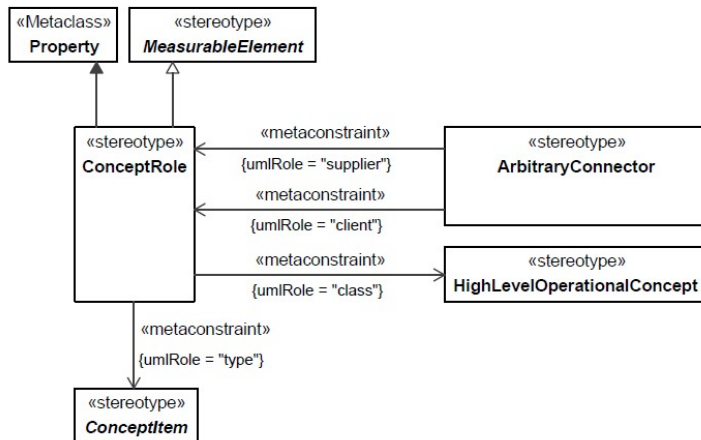


Figure 7.56 - ConceptRole

Constraints

- [1] ConceptRole.class Value for the class metaproperty must be stereotyped «HighLevelOperationalConcept» or its specializations.
- [2] ConceptRole.type Value for the type metaproperty must be stereotyped by a specialization of «ConceptItem».

**HighLevelOperationalConcept**

**Package:** Taxonomy

**isAbstract:** No

**Generalization:** [PropertySet](#), Block

**Extension:** Class

Description

Describes the Resources and Locations required to meet an operational scenario from an integrated systems point of view. It is used to communicate overall quantitative and qualitative system characteristics to stakeholders.



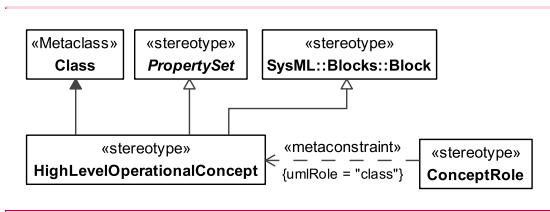
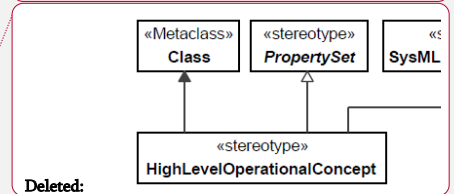


Figure 7.57 - HighLevelOperationalConcept

**Commented [AM22]:** [UAF11-157](#) Diagram replaced by HighLevelOperationalConcept.svg



### 7.1.5.2 UAF::Operational::Structure

Contains the elements that contribute to the Operational Structure Viewpoint.

#### KnownResource

**Package:** Structure

**isAbstract:** No

**Generalization:** [OperationalPerformer](#), [ResourcePerformer](#)

**Extension:** Class

**Description**

Asserts that a known ResourcePerformer constrains the implementation of the OperationalPerformer that plays the role in the LogicalArchitecture.

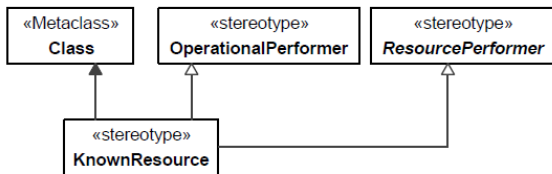


Figure 7.58 - KnownResource

**Commented [AM23]:** [UAF11-110](#) Text changed from: "Asserts that a known ResourcePerformer plays a part in the LogicalArchitecture." to "Asserts that a known ResourcePerformer constrains the implementation of the OperationalPerformer that plays the role in the LogicalArchitecture."

**Deleted:** Asserts that a known ResourcePerformer plays a part in the LogicalArchitecture...

**OperationalAgent**

Package: Structure

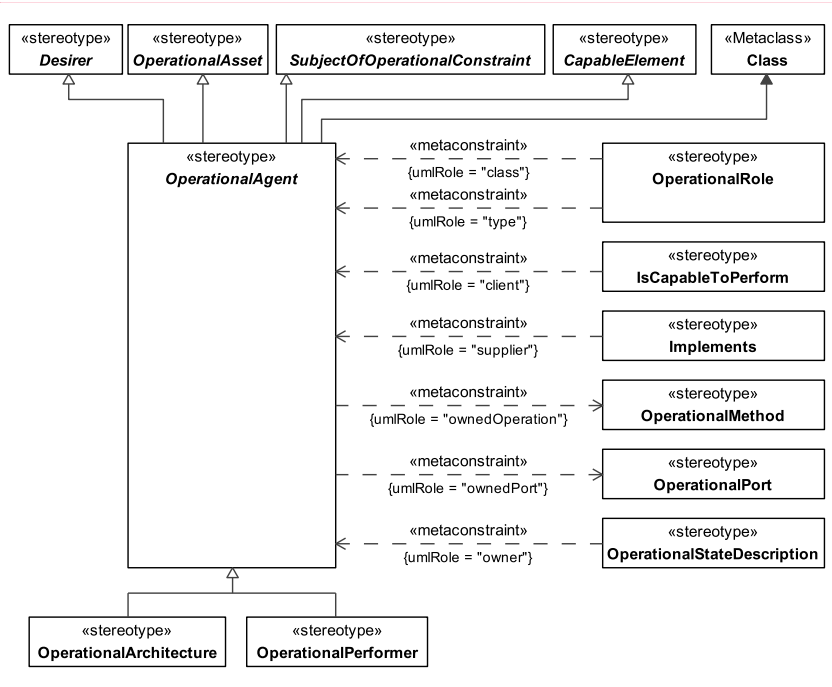
isAbstract: Yes

Generalization: [OperationalAsset](#), [SubjectOfOperationalConstraint](#), [CapableElement](#), [Desirer](#)

Extension: Class

Description

[An abstract type grouping OperationalArchitecture and OperationalPerformer.](#)



Commented [AM24]: UAF11-204  
Deleted: [Asset](#)

Commented [AM25]: UAF11-204  
Deleted: An abstract element grouping LogicalArchitecture and OperationalPerformer.

Commented [AM26]: UAF11-204. Remove original, insert OperationalAgent.svg

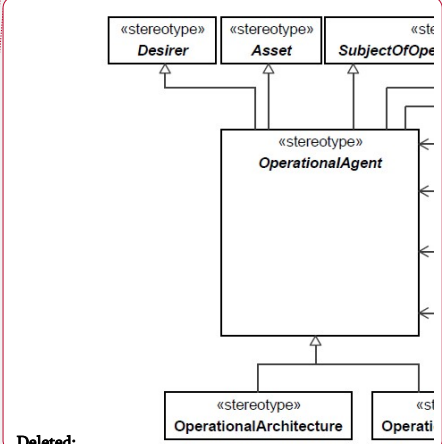


Figure 7.59 - OperationalAgent

**Constraints**

[1] **OperationalAgent.isCapableOfPerforming** Is capable of performing only «OperationalActivity» elements or its specializations.

[2] **OperationalAgent.ownedOperation** Values for the ownedOperation metaproperty must be stereotyped «OperationalMethod» or its specializations.

[3] `OperationalAgent.ownedPort` Values for the `ownedPort` metaproperty must be stereotyped `«OperationalPort»` or its specializations.

Commented [AM27]: UAF11-204. Constraints added

### **OperationalArchitecture**

**Package:** Structure

**isAbstract:** No

**Generalization:** [OperationalAgent](#), [Architecture](#)

**Extension:** Class

Description

An element used to denote a model of the Architecture, described from the Operational perspective.

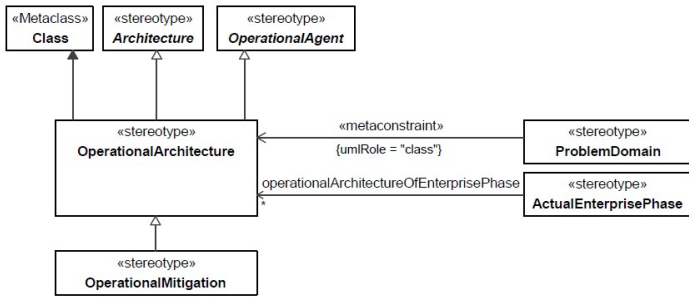


Figure 7.60 - OperationalArchitecture

**OperationalMethod**

Package: Structure

isAbstract: No

Generalization: [MeasurableElement](#)

Extension: Operation

Description

A behavioral feature of an [OperationalAgent](#) whose behavior is specified in an [OperationalActivity](#).

Commented [AM28]: UAF11-204  
 Deleted: OperationalPerformer

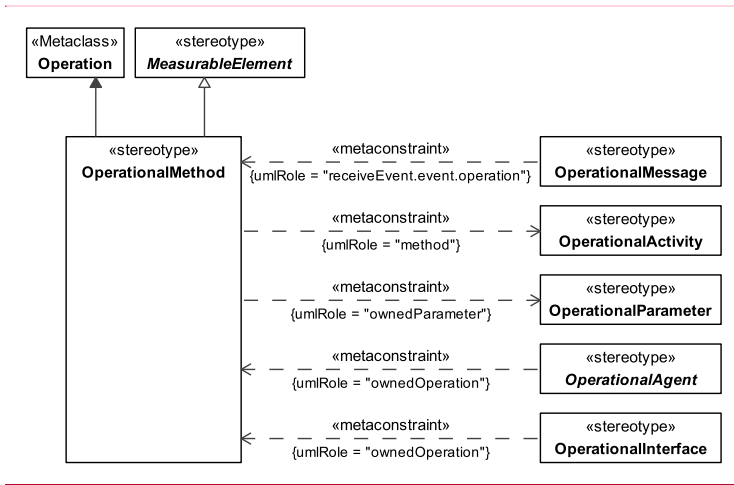


Figure 7.61 - OperationalMethod

Constraints

- [1] OperationalMethod.method Value for the method metaproperty must be stereotyped «OperationalActivity» or its specializations.
- [2] OperationalMethod.ownedParameter The values for the ownedParameter metaproperty must be stereotyped «OperationalParameter» or its specializations.

**OperationalParameter**

**Package:** Structure

**isAbstract:** No

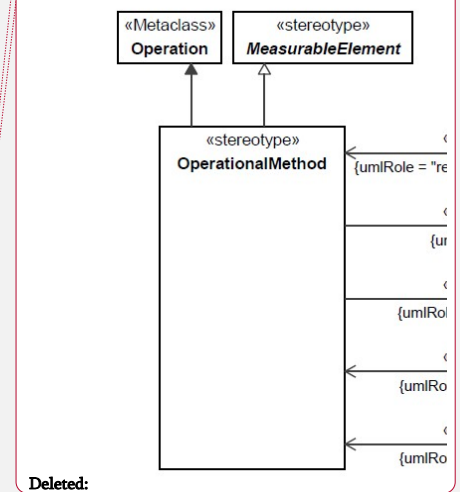
**Generalization:** [MeasurableElement](#)

**Extension:** Parameter

Description

An element that represents inputs and outputs of an OperationalActivity. It is typed by an OperationalExchangeItem.

Commented [AM29]: UAF11-204. Image removed. Image OperationalMethod.svg added.



Deleted:

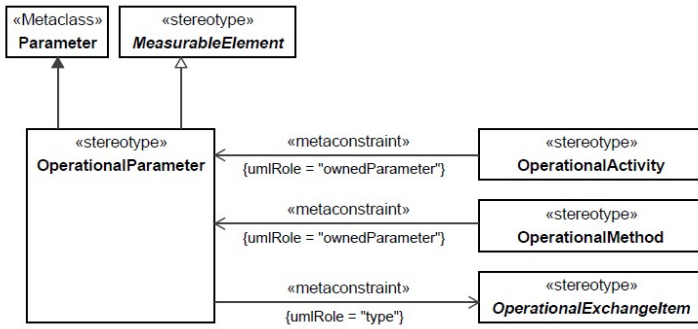


Figure 7.62 - OperationalParameter

Constraints

- [1] OperationalParameter.type Value for the type metaproperty must be stereotyped by specialization of «OperationalExchangeItem».

**OperationalPerformer**

Package: Structure

isAbstract: No

Generalization: [OperationalAgent](#)

Extension: Class

Description

A logical agent that IsCapableToPerform OperationalActivities which produce, consume, and process Resources.

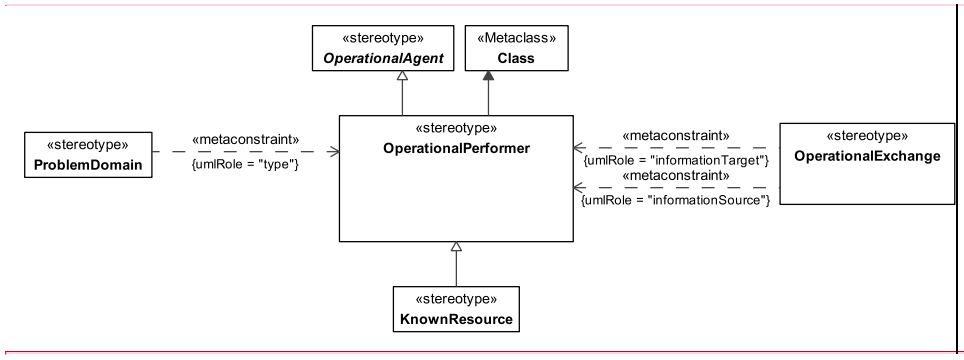


Figure 7.63 - OperationalPerformer

**OperationalPort**

**Package:** Structure

**isAbstract:** No

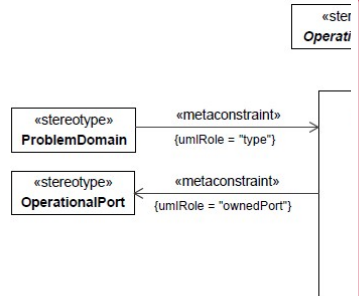
**Generalization:** [MeasurableElement](#), ProxyPort

**Extension:** Port

Description

An interaction point for an OperationalAgent through which it can interact with the outside environment and which is defined by an OperationalInterface.

**Commented [AM30]:** UAF11-204. Removed image. Added OperationalPerformer.svg



**Deleted:**

**Deleted:** Constraints

[1] OperationalPerformer.isCapableOfPerforming - Is capable of performing only «OperationalActivity» elements or its specializations.

[2] OperationalPerformer.ownedOperation Values for the ownedOperation metaproperty must be stereotyped «OperationalMethod» or its specializations.

**Commented [AM31]:** UAF11-204. OwnedPort constraint removed

**Deleted:** [3] OperationalPerformer.ownedPort Values for the ownedPort metaproperty must be stereotyped «OperationalPort» or its specializations.

**Commented [AM32]:** UAF11-204

**Deleted:** Usage of a OperationalPerformer or LogicalArchitecture in the context of another OperationalPerformer or LogicalArchitecture. Creates a whole-part relationship.

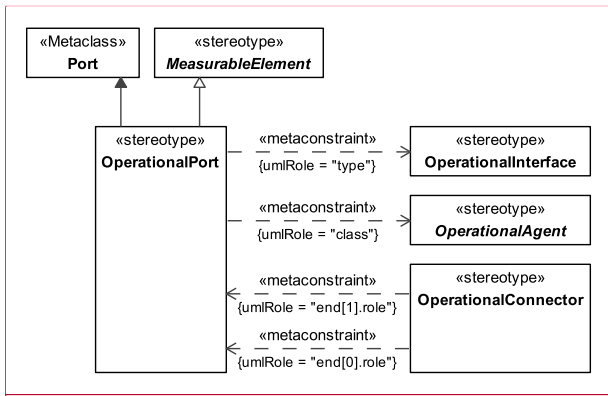


Figure 7.64 - OperationalPort

Constraints

- [1] OperationalPort.class Value for class metaproperty must be stereotyped «OperationalAgent» or its specializations.
- [2] OperationalPort.type Value for type metaproperty must be stereotyped «OperationalInterface» or its specializations.

**OperationalRole**

**Package:** Structure

**isAbstract:** No

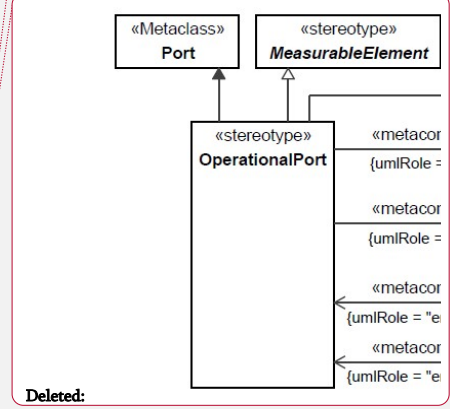
**Generalization:** [MeasurableElement](#), [LocationHolder](#), [SubjectOfSecurityConstraint](#), [AssetRole](#)

**Extension:** Property

Description

Usage of a OperationalPerformer or OperationalArchitecture in the context of another OperationalPerformer or OperationalArchitecture. Creates a whole-part relationship.

Commented [AM33]: UAF11-204. Removed image. Added image OperationalPort.svg



Deleted:

Commented [AM34]: UAF11-204

Deleted: Performer





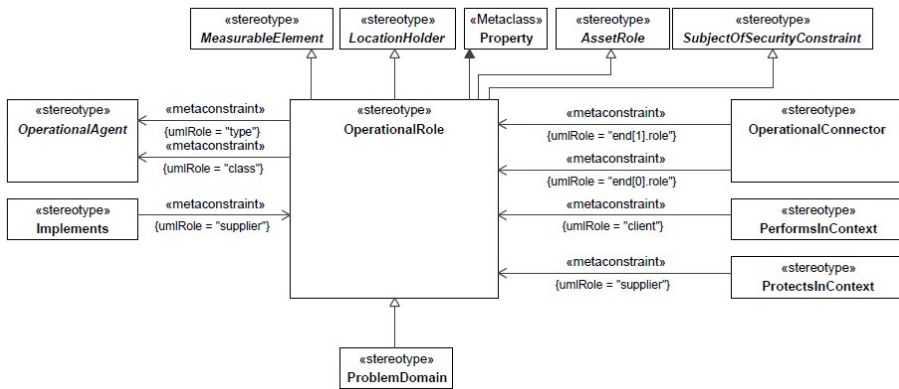


Figure 7.65 - OperationalRole

Constraints

- [1] OperationalRole.class Value for class metaproperty must be stereotyped by a specialization of «OperationalAgent».
- [2] OperationalRole.type Value for type metaproperty must be stereotyped by a specialization of «OperationalAgent».

**ProblemDomain**

**Package:** Structure

**isAbstract:** No

**Generalization:** [OperationalRole](#)

**Extension:** Property

Description

A property associated with a logical architecture, used to specify the scope of the problem.

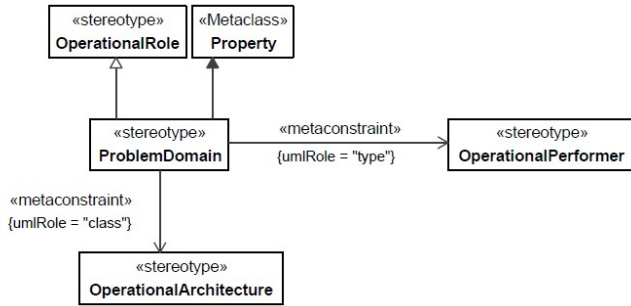


Figure 7.66 - ProblemDomain

Constraints

- [1] ProblemDomain.class Value for the class metaproperty must be stereotyped «OperationalArchitecture» or its specializations.
- [2] ProblemDomain.type Value for the type metaproperty must be stereotyped «OperationalPerformer» or its specializations.

7.1.5.3 UAF::Operational::Connectivity

Contains the elements that contribute to the Operational Connectivity Viewpoint.

**OperationalConnector**

**Package:** Connectivity

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Connector

Description

A Connector that goes between OperationalRoles representing a need to exchange Resources. It can carry a number of OperationalExchanges.

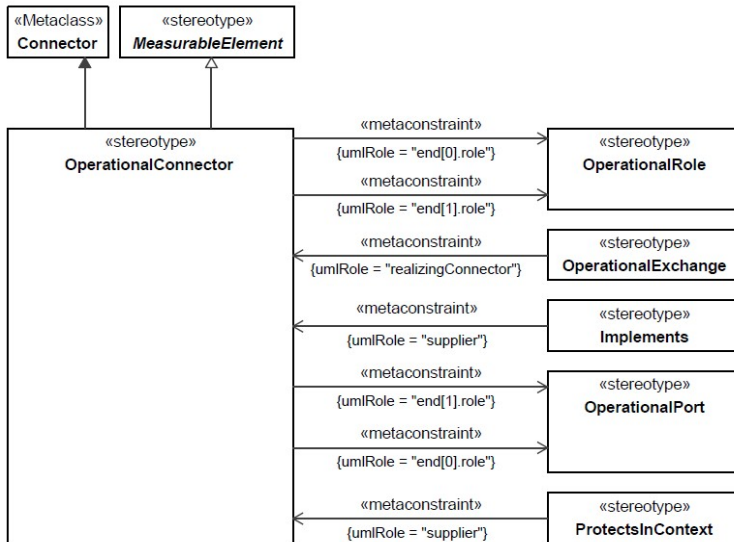


Figure 7.67 - OperationalConnector

Constraints

[1] OperationalConnector.end The value for the role metaproperty for the owned ConnectorEnd must be stereotype «OperationalRole»/«OperationalPort» or its specializations.

**OperationalExchange**

**Package:** Connectivity

**isAbstract:** No

**Generalization:** [Exchange](#), [SubjectOfOperationalConstraint](#)

**Extension:** InformationFlow

Description

Asserts that a flow can exist between OperationalPerformers (i.e., flows of information, people, materiel, or energy).

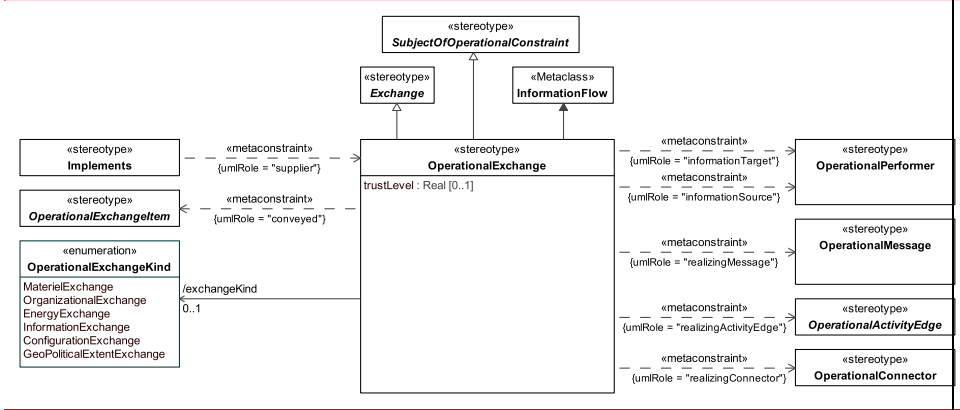


Figure 7.68 - OperationalExchange

Attributes

trustlevel : Real[0..1] Captures the directional arbitrary level of trust related to an OperationalExchange between two OperationalPerformers.

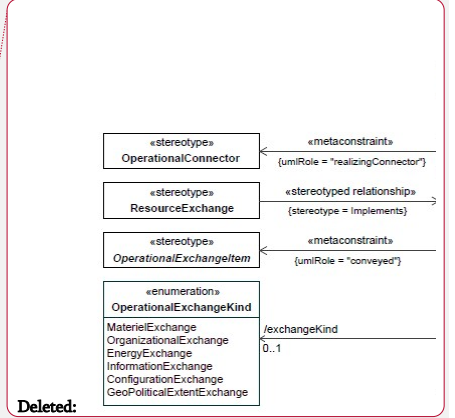
Associations

exchangeKind : OperationalExchangeKind[0..1] Captures the kind of Resource being exchanged.

Constraints

[1] OperationalExchange.conveyed In case of OperationalExchange.operationalExchangeKind:  
 = InformationExchange, the conveyed element must be stereotyped «InformationElement» or its specializations.  
 = MaterielExchange, the conveyed element must be stereotyped «ResourceArtifact» or its specializations.  
 = EnergyExchange, the conveyed element must be stereotyped «NaturalResource» or its specializations.  
 = OrganizationalExchange, the conveyed element must be stereotyped «OrganizationalResource» or its specializations.  
 = ConfigurationExchange, the conveyed element must be stereotyped «CapabilityConfiguration» or its specializations, or

Commented [AM35]: UAF11-149, UAF11-49 Figure 7.68 – OperationalExchange replaced by OperationalExchange.svg



Deleted:

- [2] OperationalExchange.informationSource = GeoPoliticalExtentExchange, the conveyed element must be stereotyped «GeoPoliticalExtentType» or its specializations. Value for informationSource metaproperty has to be stereotyped «OperationalPerformer» or its specializations.
- [3] OperationalExchange.informationTarget Value for informationTarget metaproperty has to be stereotyped «OperationalPerformer» or its specializations.
- [4] OperationalExchange.realizingActivityEdge Value for realizingActivityEdge metaproperty has to be stereotyped by any specialization of «OperationalActivityEdge».
- [5] OperationalExchange.realizingConnector Value for realizingConnector metaproperty has to be stereotyped «OperationalConnector» or its specializations.
- [6] OperationalExchange.realizingMessage Value for realizingMessage metaproperty has to be stereotyped «OperationalMessage» or its specializations.

**OperationalExchangeItem**

**Package:** Connectivity

**isAbstract:** Yes

**Generalization:** [Resource](#)

Description

An abstract grouping for elements that defines the types of elements that can be exchanged between OperationalPerformers and conveyed by an OperationalExchange.

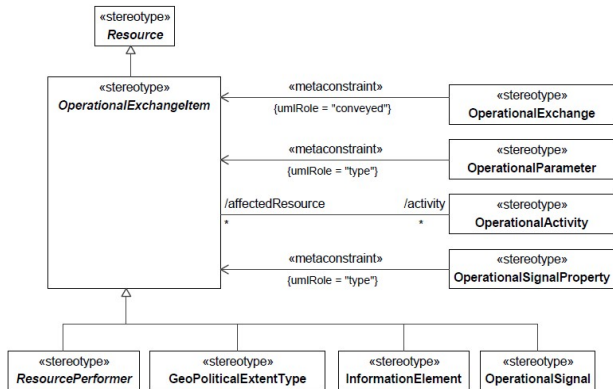


Figure 7.69 - OperationalExchangeItem

## Associations

activity : OperationalActivity[\*]      A collection of OperationalActivities that consume and/or produce the OperationalExchangeItem internally.

### **OperationalExchangeKind**

**Package:** Connectivity

**isAbstract:** No

#### Description

Enumeration of the possible kinds of operational exchange applicable to an OperationalExchange. Its enumeration literals are:

- **MatérielExchange** - Indicates that the OperationalExchange associated with the OperationalExchangeKind is a logical flow of materiel (artifacts) between Functions.
- **OrganizationalExchange** - Indicates that the OperationalExchange associated with the OperationalExchangeKind is a logical flow where human resources (PostTypes, RoleTypes) flow between OperationalPerformers.
- **EnergyExchange** - Indicates that the OperationalExchange associated with the OperationalExchangeKind is a logical flow where energy is flowed from one OperationalPerformer to another.
- **InformationExchange** - Indicates that the OperationalExchange associated with the OperationalExchangeKind is a logical flow where information is flowed from one OperationalPerformer to another.
- **ConfigurationExchange** - Indicates that the OperationalExchange associated with the OperationalExchangeKind is a logical flow where CapabilityConfigurations flow from one OperationalPerformer to another.
- **GeoPoliticalExtentExchange** - Indicates that the OperationalExchange associated with the OperationalExchangeKind is a logical flow where GeoPoliticalExtentTypes (i.e. Borders) flow from one place to another.

### **OperationalInterface**

**Package:** Connectivity

**isAbstract:** No

**Generalization:** [PropertySet](#), InterfaceBlock

**Extension:** Class

#### Description

A declaration that specifies a contract between the OperationalPerformer it is related to, and any other OperationalPerformers it can interact with.

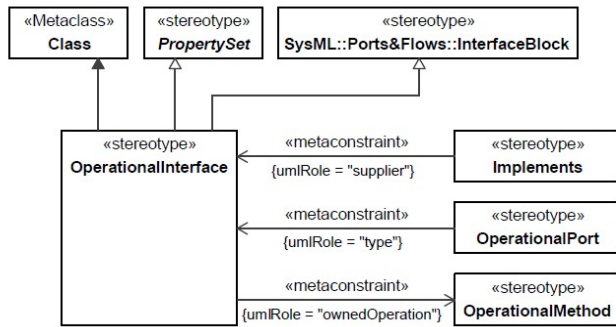


Figure 7.70 - OperationalInterface

Constraints

- [1] OperationalInterface.ownedOperation Values for the ownedOperation metaproperty must be stereotyped «OperationalMethod» or its specializations.

**OperationalSignal**

**Package:** Connectivity

**isAbstract:** No

**Generalization:** [OperationalExchangeItem](#)

**Extension:** Signal

Description

An OperationalSignal is a specification of a kind of communication between operational performers in which a reaction is asynchronously triggered in the receiver without a reply.



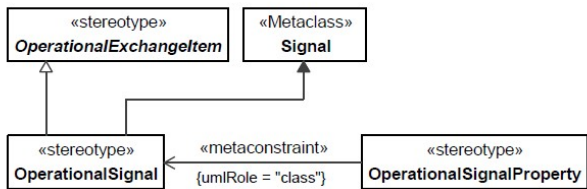


Figure 7.71 - OperationalSignal

### OperationalSignalProperty

**Package:** Connectivity

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Property

**Description**

A property of an OperationalSignal typed by OperationalExchangeItem. It enables OperationalExchangeItem e.g., InformationElement to be passed as arguments of the OperationalSignal.

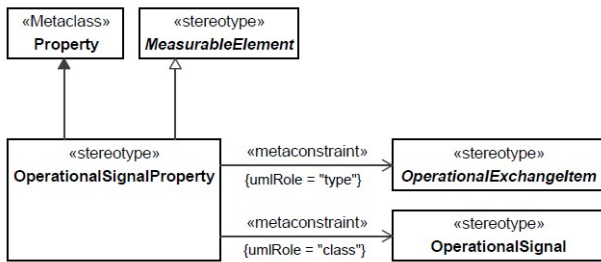


Figure 7.72 - OperationalSignalProperty

Constraints

- [1] OperationalSignalProperty.class Value for class metaproperty must be stereotyped «OperationalSignal» or its specializations.
- [2] OperationalSignalProperty.type Value for type metaproperty must be stereotyped by a specialization of «OperationalExchangeItem».

7.1.5.4 UAF::Operational::Processes

Contains the elements that contribute to the Operational Processes Viewpoint.

**OperationalActivity**

Package: Processes

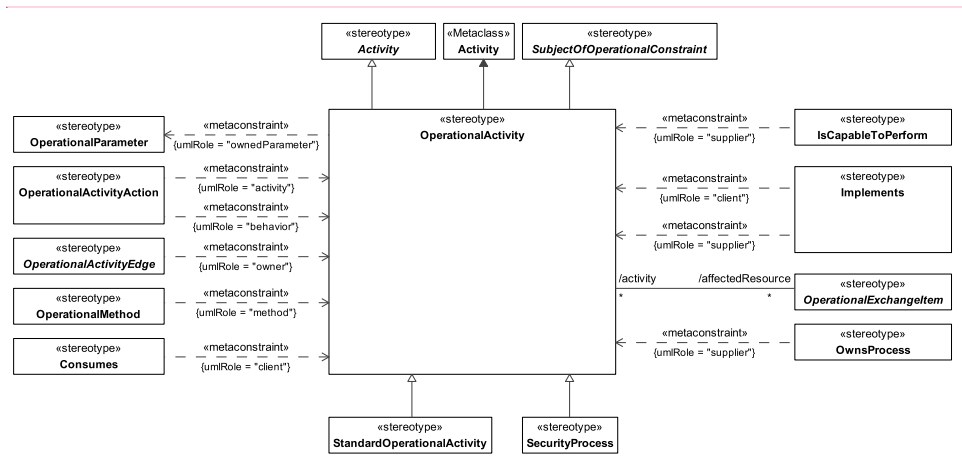
isAbstract: No

Generalization: [Activity](#), [SubjectOfOperationalConstraint](#)

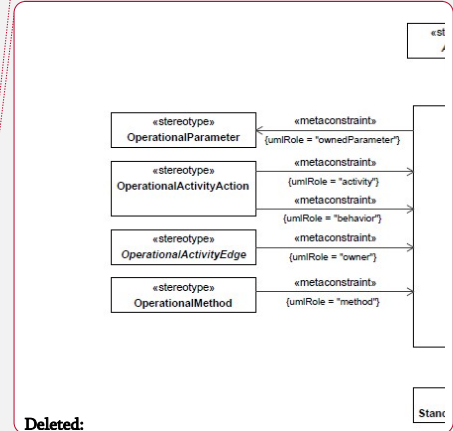
Extension: Activity

Description

An Activity that captures a logical process, specified independently of how the process is carried out.



Commented [AM36]: UAF11-157, UAF11-111 Diagram replaced by OperationalActivity.svg



Deleted:

Figure 7.73 - OperationalActivity

Associations

affectedResource :  
OperationalExchangeItem[\*]      A collection of OperationalExchangeItems consumed and produced internally within the OperationalActivity.

Constraints

[1] OperationalActivity.ownedParameter      The values for the ownedParameter metaproperty must be stereotyped «OperationalParameter» or its specializations.

**OperationalActivityAction**

**Package:** Processes

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** CallBehaviorAction

Description

A call of an OperationalActivity in the context of another OperationalActivity.

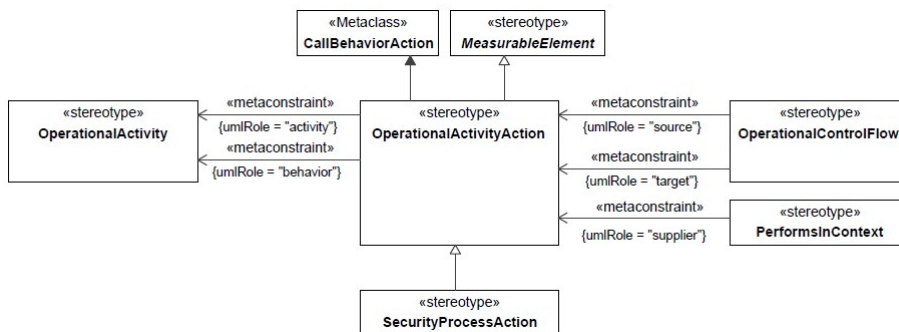


Figure 7.74 - OperationalActivityAction

Constraints

- [1] OperationalActivityAction.activity Value for the activity metaproperty must be stereotyped «OperationalActivity» or its specializations.
- [2] OperationalActivityAction.behavior Value for activity metaproperty must be stereotyped «OperationalActivity» or its specializations.

**OperationalActivityEdge**

Package: Processes

isAbstract: Yes

Generalization: [MeasurableElement](#)

Extension: ActivityEdge

Description

Abstract grouping for OperationalControlFlow and OperationalObjectFlow.

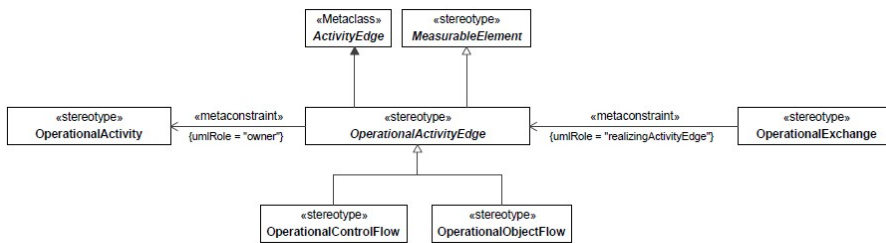


Figure 7.75 - OperationalActivityEdge

Constraints

- [1] OperationalActivityEdge.owner «OperationalActivityEdge» must be owned directly or indirectly by «OperationalActivity» or its specializations.

**OperationalControlFlow**

Package: Processes

isAbstract: No

Generalization: [OperationalActivityEdge](#)

**Extension:** ControlFlow

**Description**

An ActivityEdge that shows the flow of control between OperationalActivityActions.

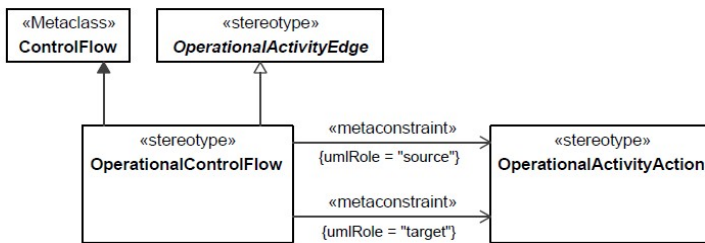


Figure 7.76 - OperationalControlFlow

**Constraints**

- [1] OperationalControlFlow.source Value for the source metaproperty must be stereotyped «OperationalActivityAction» or its specializations.
- [2] OperationalControlFlow.target Value for the target metaproperty must be stereotyped «OperationalActivityAction» or its specializations.

**OperationalObjectFlow**

**Package:** Processes

**isAbstract:** No

**Generalization:** [OperationalActivityEdge](#)

**Extension:** ObjectFlow

**Description**

An ActivityEdge that shows the flow of Resources (objects/information) between OperationalActivityActions.

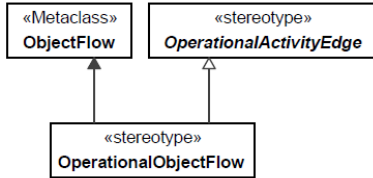


Figure 7.77 - OperationalObjectFlow

**StandardOperationalActivity**

Package: Processes

isAbstract: No

Generalization: [OperationalActivity](#)

Extension: Activity

Description

A sub-type of OperationalActivity that is a standard operating procedure.

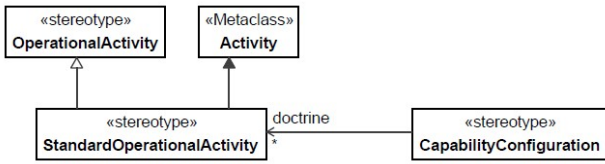


Figure 7.78 - StandardOperationalActivity

**7.1.5.5 UAF::Operational::States**

Contains the elements that contribute to the Operational States Viewpoint.

**OperationalStateDescription**

Package: States

isAbstract: No

Generalization: [MeasurableElement](#)

Extension: StateMachine

#### Description

A state machine describing the behavior of a `OperationalPerformer`, depicting how the `OperationalPerformer` responds to various events and the actions.

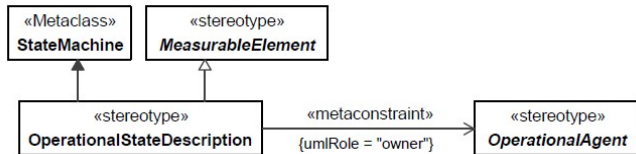


Figure 7.79 - `OperationalStateDescription`

#### Constraints

[1] `OperationalStateDescription.owner` Values for the owner metaproperty must be stereotyped with specializations of `«OperationalAgent»`.

#### 7.1.5.6 UAF::Operational::Interaction Scenarios

Contains the elements that contribute to the Operational Interaction Scenarios Viewpoint.

##### **OperationalMessage**

**Package:** Interaction Scenarios

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Message

#### Description

Message for use in an Operational Event-Trace which carries any of the subtypes of `OperationalExchange`.

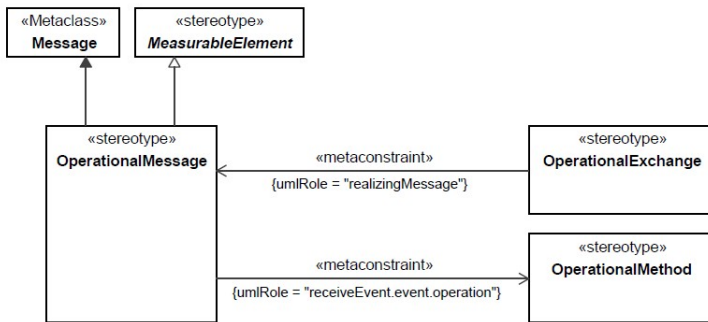


Figure 7.80 - OperationalMessage

Constraints

- [1] OperationalMessage.receiveEvent.event.operation Values for the receiveEvent.event.operation metaproperty must be stereotyped with «OperationalMethod» or its specializations.

7.1.5.7 UAF::Operational::Information

Contains the elements that contribute to the Operational Information Viewpoint.

InformationElement

Package: Information

isAbstract: No

Generalization: [OperationalAsset](#), [OperationalExchangeItem](#), [SubjectOfOperationalConstraint](#)

Extension: Class

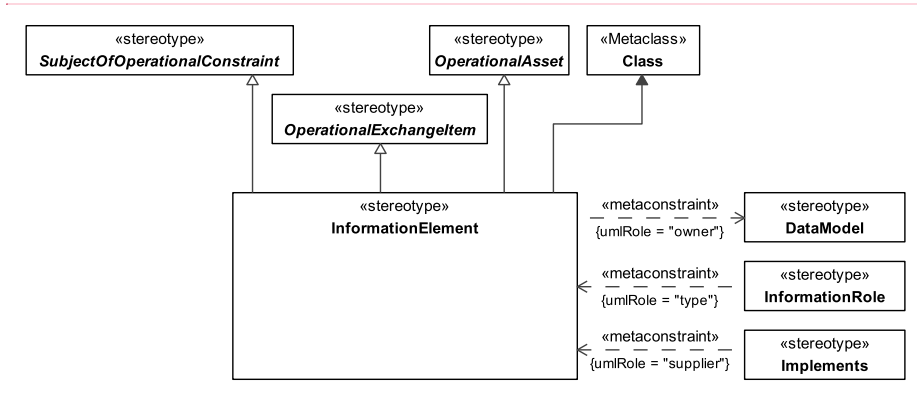
Description

An item of information that flows between OperationalPerformers and is produced and consumed by the OperationalActivities that the OperationalPerformers are capable to perform (see IsCapableToPerform).

Commented [AM37]: UAF11-16 In the generalizations list "Asset" replaced by "ResourceAsset".

Deleted: [Asset](#)





Commented [AM38]: UAF11-16 Figure 7.81 – InformationElement replaced by InformationElement.svg

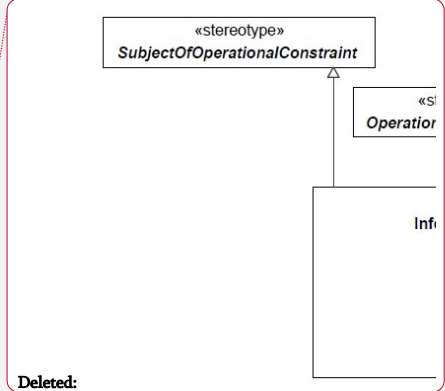


Figure 7.81 - InformationElement

Constraints

- [1] InformationElement.owner Values for the owner metaproperty must be stereotyped «DataModel» or its specializations.

7.1.5.8 UAF::Operational::Constraints

Contains the elements that contribute to the Operational Constraints Viewpoint.

**OperationalConstraint**

**Package:** Constraints

**isAbstract:** No

**Generalization:** [Rule](#)

**Extension:** Constraint

Description

A Rule governing a logical architectural element i.e., OperationalPerformer, OperationalActivity, InformationElement, etc.

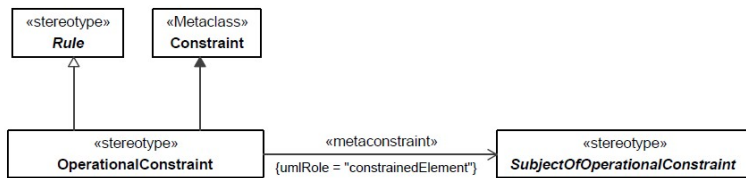


Figure 7.82 - OperationalConstraint

Constraints

[1] OperationalConstraint.constrainedElement Value for the constrainedElement metaproperty must be stereotyped by any specialization of «SubjectOfOperationalConstraint».

**SubjectOfOperationalConstraint**

**Package:** Constraints

**isAbstract:** Yes

**Generalization:** [UAFElement](#)

**Extension:** Element

Description

An abstract grouping of elements that can be the subject of an OperationalConstraint.

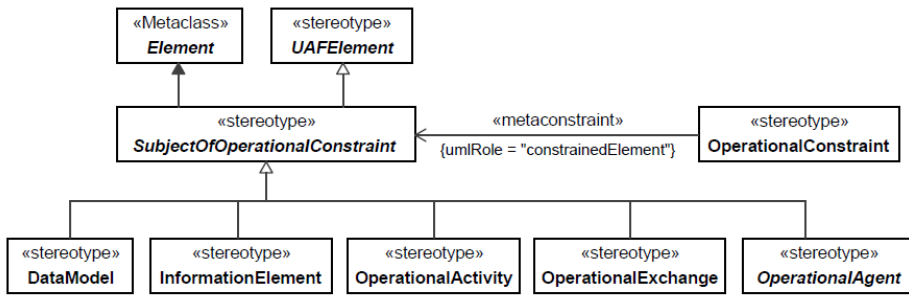


Figure 7.83 - SubjectOfOperationalConstraint

### 7.1.6 UAF::Services

Stakeholders: Enterprise Architects, Solution Providers, Systems Engineers, Software Architects, Business Architects..

Concerns: specifications of services required to exhibit a Capability.

Definition: shows Service Specifications and required and provided service levels of these specifications required to exhibit a Capability or to support an Operational Activity.

#### 7.1.6.1 UAF::Services::Taxonomy

Contains the elements that contribute to the Services Taxonomy Viewpoint.

#### ServiceSpecification

**Package:** Taxonomy

**isAbstract:** No

**Generalization:** [PropertySet](#), [VersionedElement](#), [CapableElement](#), Block

**Extension:** Class

Description

The specification of a set of functionality provided by one element for the use of others.

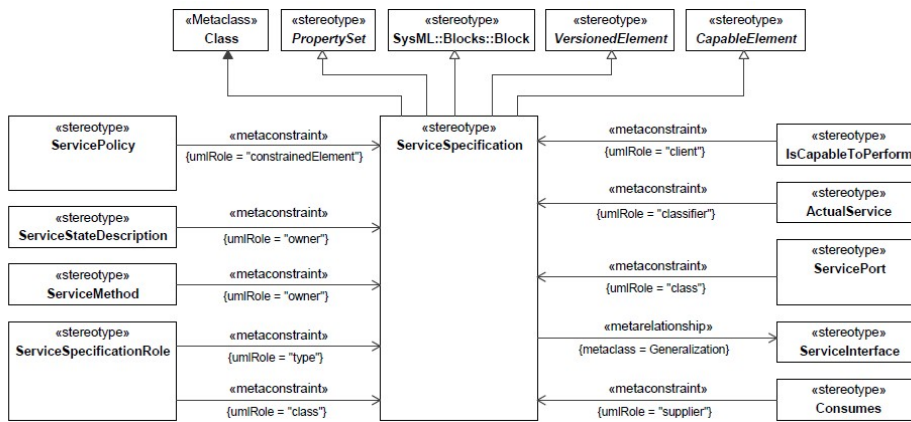


Figure 7.84 - ServiceSpecification

### 7.1.6.2 UAF::Services::Structure

Contains the elements that contribute to the Services Structure Viewpoint.

#### ServiceMethod

**Package:** Structure

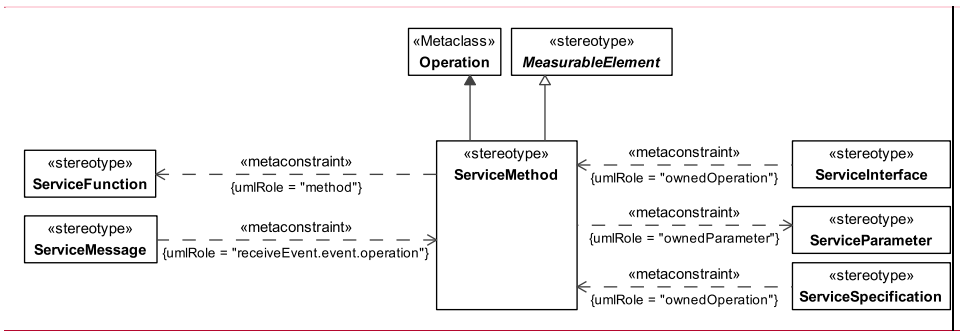
**isAbstract:** No

**Generalization:** [MeasurableElement](#)

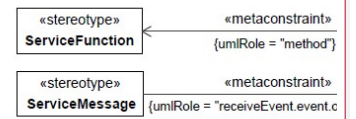
**Extension:** Operation

**Description**

A behavioral feature of a ServiceSpecification whose behavior is specified in a ServiceFunction.



Commented [AM39]: UAF11-44 Figure 7.85 – ServiceMethod replaced by ServiceMethod.svg



Deleted:

Figure 7.85 - ServiceMethod

Constraints

- [1] ServiceMethod.method Value for the method metaproperty must be stereotyped «ServiceFunction» or its specializations.
- [2] ServiceMethod.ownedParameter The values for the ownedParameter metaproperty must be stereotyped «ServiceParameter» or its specializations.
- [3] ServiceMethod.owner The values for the owner metaproperty must be stereotyped «ServiceSpecification» or its specializations.

**ServiceParameter**

Package: Structure

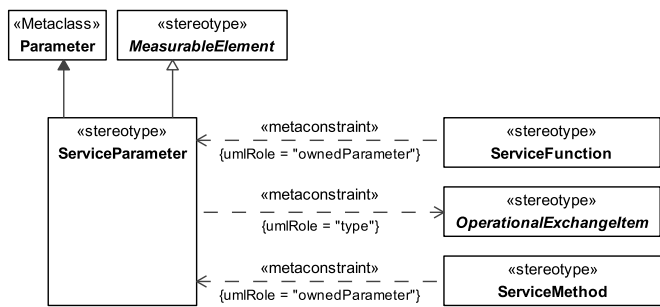
isAbstract: No

Generalization: [MeasurableElement](#)

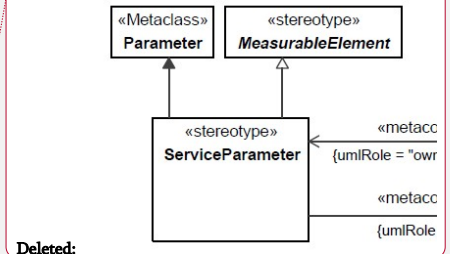
Extension: Parameter

Description

An element that represents inputs and outputs of a ServiceFunction, represents inputs and outputs of a ServiceSpecification.



Commented [AM40]: UAF11-114 Figure 7.86 – ServiceParameter replaced by ServiceParameter.svg.



Deleted:

Figure 7.86 - ServiceParameter

Constraints

[1] ServiceParameter.type The values for the type metaproperty must be stereotyped a specialization of «OperationalExchangeItem».

**ServicePort**

**Package:** Structure

**isAbstract:** No

**Generalization:** ProxyPort, [MeasurableElement](#)

**Extension:** Port

Description

An interaction point for a ServiceSpecification through which it can interact with the outside environment and which is defined by a ServiceInterface.

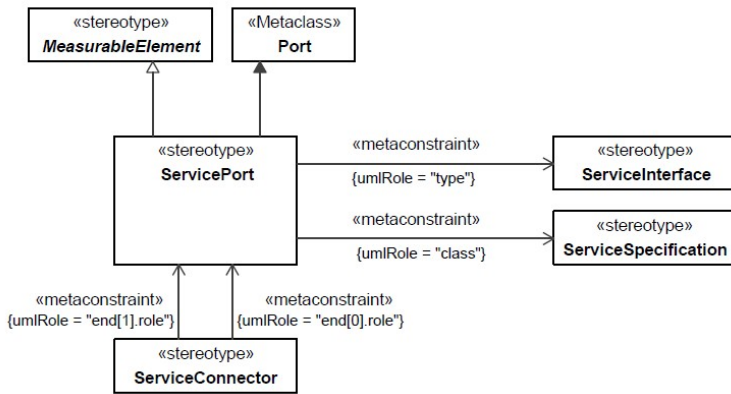


Figure 7.87 - ServicePort

Constraints

- [1] ServicePort.class Value for the class metaproperty must be stereotyped «ServiceSpecification» or its specializations.
- [2] ServicePort.type Value for the type metaproperty must be stereotyped «ServiceInterface» or its specializations.

**ServiceSpecificationRole**

**Package:** Structure

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Property

Description

Usage of a ServiceSpecification in the context of another ServiceSpecification. Creates a whole-part relationship.

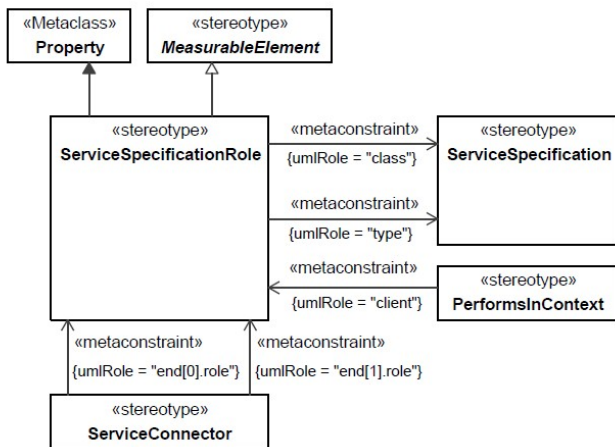


Figure 7.88 - ServiceSpecificationRole

Constraints

- [1] ServiceSpecificationRole.class Value for the class metaproperty must be stereotyped «ServiceSpecification» or its specializations.
- [2] ServiceSpecificationRole.type Value for the type metaproperty must be stereotyped «ServiceSpecification» or its specializations.

**7.1.6.3 UAF::Services::Connectivity**

Contains the elements that contribute to the Services Connectivity Viewpoint.

**ServiceConnector**

**Package:** Connectivity

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Connector

Description

A channel for exchange between two ServiceSpecifications. Where one acts as the consumer of the other.



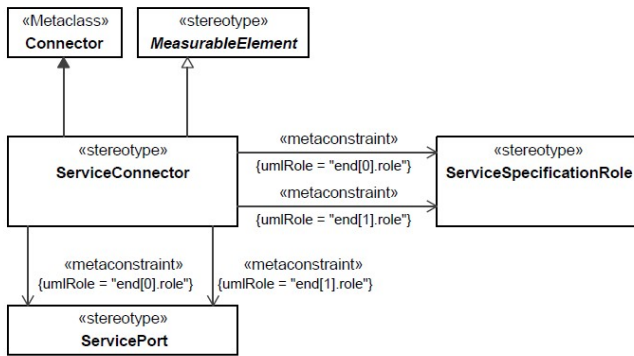


Figure 7.89 - ServiceConnector

Constraints

[1] ServiceConnector.end The value for the role metaproperty for the owned ConnectorEnd must be stereotyped «ServicePort», «ServiceSpecificationRole», or their specializations.

**ServiceInterface**

**Package:** Connectivity

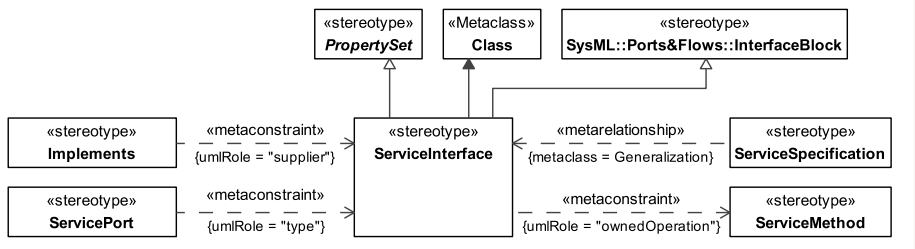
**isAbstract:** No

**Generalization:** [PropertySet](#), InterfaceBlock

**Extension:** Class

Description

A contract that defines the ServiceMethods and ServiceMessageHandlers that the ServiceSpecification realizes.



Commented [AM41]: UAF11-157 replaced by ServiceInterface.svg

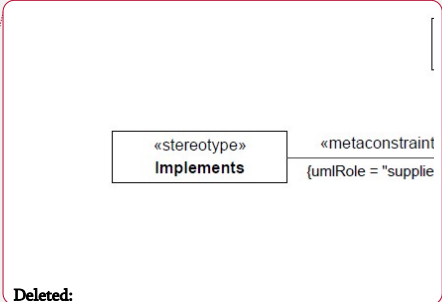


Figure 7.90 - ServiceInterface

Constraints

[1] ServiceInterface.ownedOperation Values for the ownedOperation metaproperty must be stereotyped «ServiceMethod» or its specializations.

7.1.6.4 UAF::Services::Processes

Contains the elements that contribute to the Services Processes Viewpoint.

ServiceFunction

Package: Processes

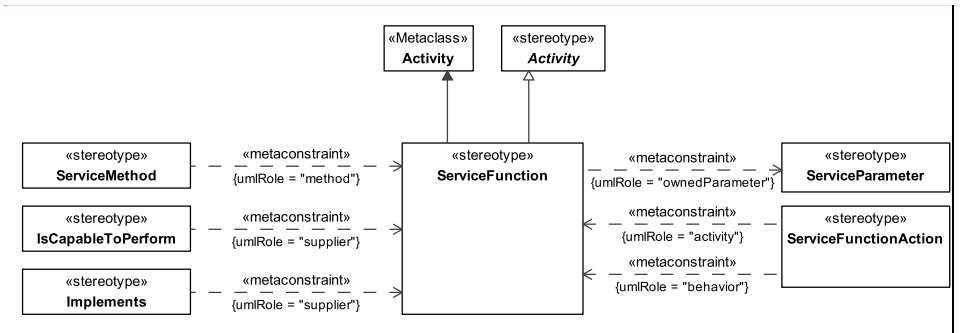
isAbstract: No

Generalization: [Activity](#)

Extension: Activity

Description

An Activity that describes the abstract behavior of ServiceSpecifications, regardless of the actual implementation.



Commented [AM42]: UAF11-148 Figure 7.91 – ServiceFunction is replaced by ServiceFunction.svg

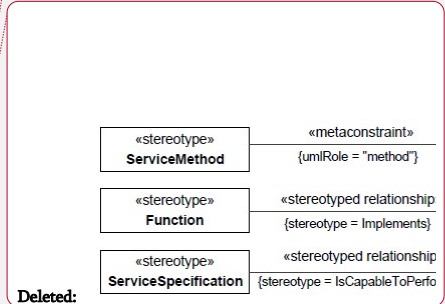


Figure 7.91 - ServiceFunction

Constraints

[1] ServiceFunction.ownedParameter The values for the ownedParameter metaproperty must be stereotyped «ServiceParameter».

**ServiceFunctionAction**

**Package:** Processes

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** CallBehaviorAction

Description

A call of a ServiceFunction in the context of another ServiceFunction.

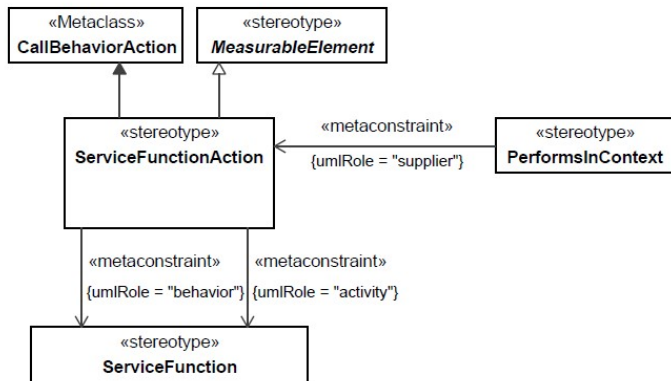


Figure 7.92 - ServiceFunctionAction

Constraints

- [1] ServiceFunctionAction.activity Value for the behavior metaproperty must be stereotyped «ServiceFunction» or its specializations.
- [2] ServiceFunctionAction.behavior Value for the activity metaproperty must be stereotyped «ServiceFunction» or its specializations.

**7.1.6.5 UAF::Services::States**

Contains the elements that contribute to the Services States Viewpoint.

**ServiceStateDescription**

**Package:** States

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** StateMachine

Description

A state machine describing the behavior of a ServiceSpecification, depicting how the ServiceSpecification responds to various events and the actions.

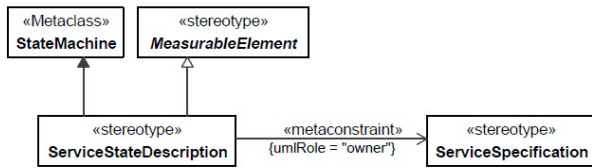


Figure 7.93 - ServiceStateDescription

Constraints

- [1] ServiceStateMachine.owner Values for the owner metaproperty must be stereotyped «ServiceSpecification» or its specializations.

7.1.6.6 UAF::Services::Interaction Scenarios

Contains the elements that contribute to the Services Interaction Scenarios Viewpoint.

ServiceMessage

Package: Interaction Scenarios

isAbstract: No

Generalization: [MeasurableElement](#)

Extension: Message

Description

Message for use in a Service Event-Trace.

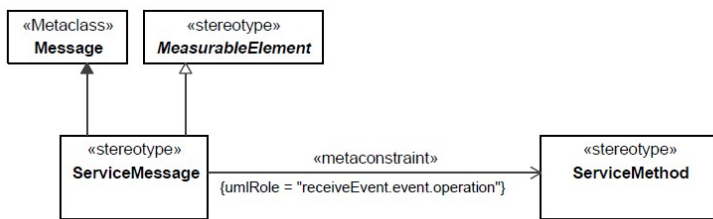


Figure 7.94 - ServiceMessage

## Constraints

[1] ServiceMessage.receiveEvent.event.operation Values for the receiveEvent.event.operation metaproperty must be stereotyped with «ServiceMethod» or its specializations.

### 7.1.6.7 UAF::Services::Constraints

Contains the elements that contribute to the Services Constraints Viewpoint.

#### ServicePolicy

**Package:** Constraints

**isAbstract:** No

**Generalization:** [Rule](#)

**Extension:** Constraint

#### Description

A constraint governing the use of one or more ServiceSpecifications.

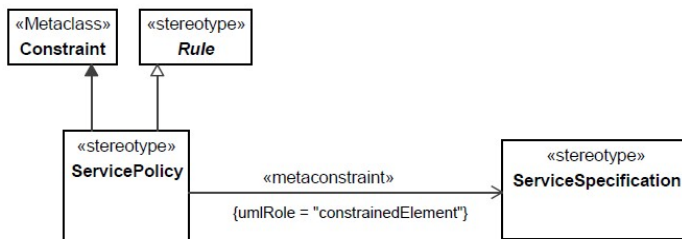


Figure 7.95 - ServicePolicy

## Constraints

[1] ServicePolicy.constrainedElement Values for constrainedElement metaproperty must be stereotyped «ServiceSpecification» or its specializations.

### 7.1.6.8 UAF::Services::Traceability

Contains the elements that contribute to the Services Traceability Viewpoint.

## Consumes

**Package:** Traceability

**isAbstract:** No

**Generalization:** Allocate, [MeasurableElement](#)

**Extension:** Abstraction

Description

An abstraction relationship that asserts that a service in some way contributes or assists in the execution of an OperationalActivity.

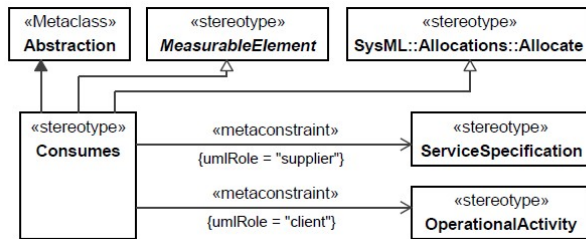


Figure 7.96 - Consumes

Constraints

- [1] Consumes.client Value for the client metaproperty must be stereotyped «OperationalActivity» or its specializations.
- [2] Consumes.supplier Value for the supplier metaproperty must be stereotyped «ServiceSpecification» or its specializations.

### 7.1.7 UAF::Personnel

Stakeholders: Human resources, Solution Providers, PMs.

Concerns: human factors.

Definition: aims to clarify the role of Human Factors (HF) when creating architectures in order to facilitate both Human Factors Integration (HFI) and systems engineering (SE).

#### 7.1.7.1 UAF::Personnel::Taxonomy

Contains the elements that contribute to the Personnel Taxonomy Viewpoint.

## Organization

**Package:** Taxonomy

**isAbstract:** No

**Generalization:** [OrganizationalResource](#)

**Extension:** Class

Description

A group of OrganizationalResources (Persons, Posts, Organizations, and Responsibilities) associated for a particular purpose.

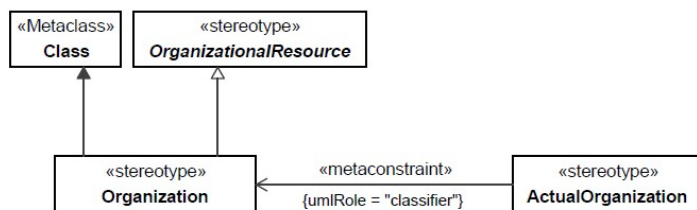


Figure 7.97 - Organization

## OrganizationalResource

**Package:** Taxonomy

**isAbstract:** Yes

**Generalization:** [PhysicalResource](#), [Stakeholder](#)

**Extension:** Class

Description

An abstract element grouping for Organization, Person Post, and Responsibility.



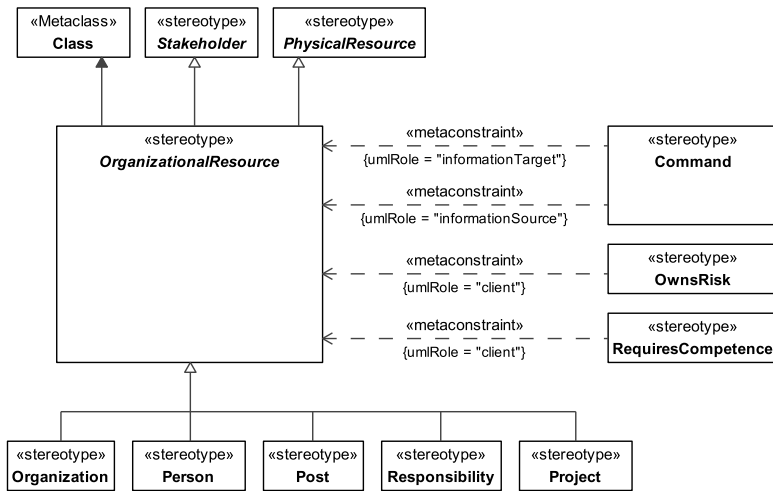
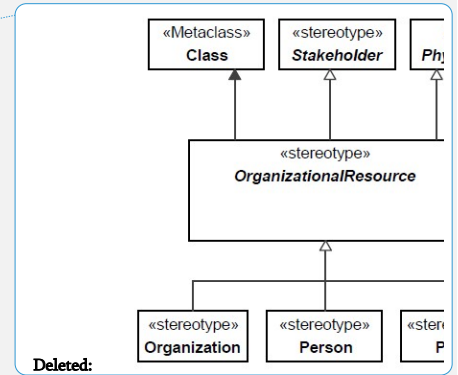


Figure 7.98 - OrganizationalResource



Commented [Yvonne43]: UAF11-45 Replaced image.

**Person**

Package: Taxonomy

isAbstract: No

Generalization: [OrganizationalResource](#)

Extension: Class

Description

A type of a human being used to define the characteristics that need to be described for ActualPersons (e.g., properties such as address, telephone number, nationality, etc.).

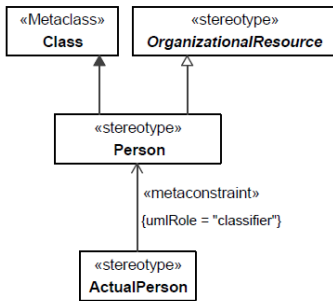
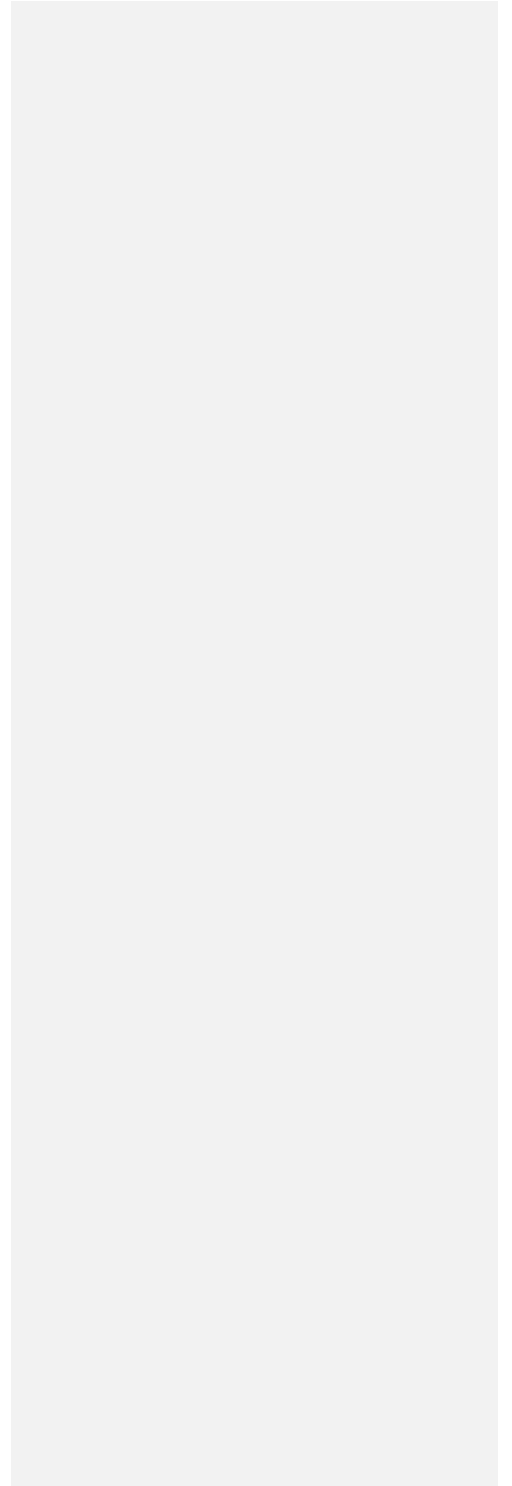


Figure 7.99 - Person



**Post**

**Package:** Taxonomy

**isAbstract:** No

**Generalization:** [OrganizationalResource](#)

**Extension:** Class

Description

A type of job title or position that a person can fill (e.g., Lawyer, Solution Architect, Machine Operator, or Chief Executive Officer).

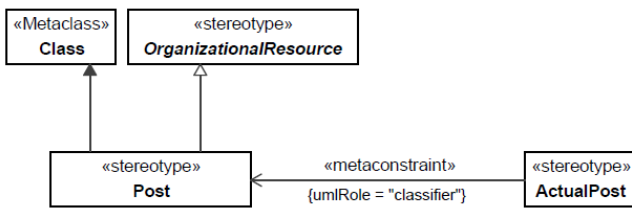


Figure 7.100 - Post

**Responsibility**

**Package:** Taxonomy

**isAbstract:** No

**Generalization:** [OrganizationalResource](#)

**Extension:** Class

Description

The type of duty required of a Person or Organization.

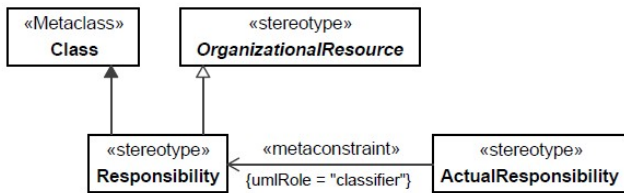


Figure 7.101 - Responsibility

### 7.1.7.2 UAF::Personnel::Connectivity

Contains the elements that contribute to the Personnel Connectivity Viewpoint.

#### Command

**Package:** Connectivity

**isAbstract:** No

**Generalization:** [ResourceExchange](#)

**Extension:** InformationFlow

Description

A type of ResourceExchange that asserts that one OrganizationalResource commands another.

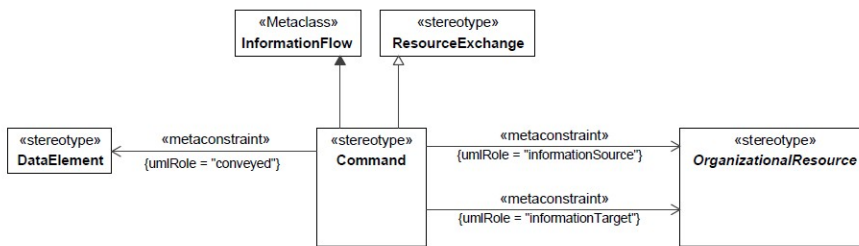


Figure 7.102 - Command

#### Constraints

- [1] Command.conveyed Value for the conveyed metaproperty must be stereotyped «DataElement» or its specializations.
- [2] Command.informationSource Value for the informationSource metaproperty must be stereotyped by the specialization of «OrganizationalResource».
- [3] Command.informationTarget Value for the informationTarget metaproperty must be stereotyped by the specialization of «OrganizationalResource».

#### Control

**Package:** Connectivity

**isAbstract:** No

**Generalization:** [ResourceExchange](#)

**Extension:** InformationFlow

Description

A type of ResourceExchange that asserts that one PhysicalResource controls another PhysicalResource (i.e., the driver of a vehicle controlling the vehicle speed or direction).

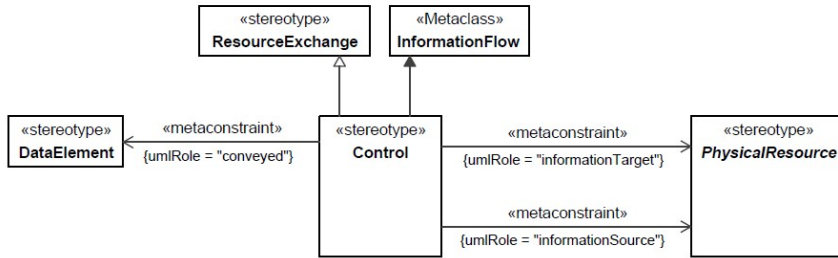


Figure 7.103 - Control

Constraints

- [1] Control.conveyed Value for the conveyed metaproperty must be stereotyped «DataElement» or its specializations.
- [2] Control.informationSource Value for the informationSource metaproperty must be stereotyped by the specialization of «PhysicalResource».
- [3] Control.informationTarget Value for the informationTarget metaproperty must be stereotyped by the specialization of «PhysicalResource» or its specializations.

7.1.7.3 UAF::Personnel::Processes

Contains the elements that contribute to the Personnel Processes Viewpoint.

CompetenceToConduct

Package: Processes

isAbstract: No

Generalization: [MeasurableElement](#), Allocate

Extension: Abstraction

Description

An abstraction relationship used to associate a Function with a specific set of Competencies needed to conduct the Function.

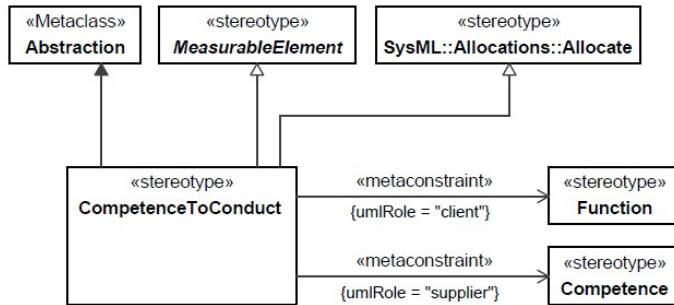


Figure 7.104 - CompetenceToConduct

Constraints

- [1] CompetenceToConduct.client Value for the client metaproperty must be stereotyped «Function» or its specializations.
- [2] CompetenceToConduct.supplier Value for the supplier metaproperty must be stereotyped «Competence» or its specializations.

**7.1.7.4 UAF::Personnel::Constraints**

Contains the elements that contribute to the Personnel Constraints Viewpoint.

**Competence**

**Package:** Constraints

**isAbstract:** No

**Generalization:** [SubjectOfForecast](#), [PropertySet](#), Block

**Extension:** Class

Description

A specific set of abilities defined by knowledge, skills, and aptitude.

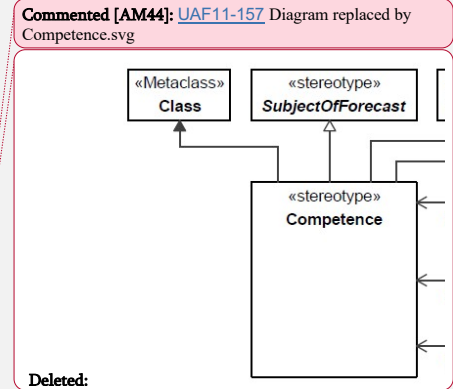
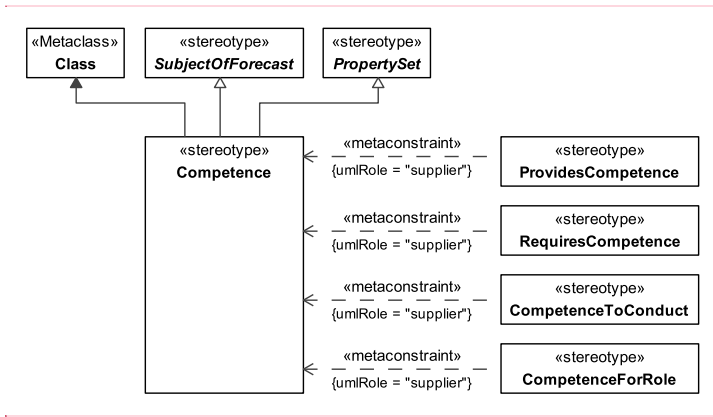


Figure 7.105 - Competence

**CompetenceForRole**

**Package:** Constraints

**isAbstract:** No

**Generalization:** [MeasurableElement](#), [Allocate](#)

**Extension:** Abstraction

Description

An abstraction relationship used to associate an organizational role with a specific set of required competencies.

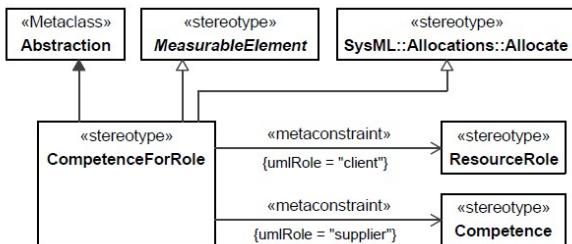


Figure 7.106 - CompetenceForRole

Constraints

- [1] `CompetenceForRole.client` Value for the client metaproperty must be stereotyped «ResourceRole» or its specializations.
- [2] `CompetenceForRole.supplier` Value for the supplier metaproperty must be stereotyped «Competence» or its specializations.

**RequiresCompetence**

**Package:** Constraints

**isAbstract:** No

**Generalization:** [MeasurableElement](#), `Allocate`

**Extension:** Abstraction

Description

An abstraction relationship that asserts that an `ActualOrganizationalResource` is required to have a specific set of Competencies.

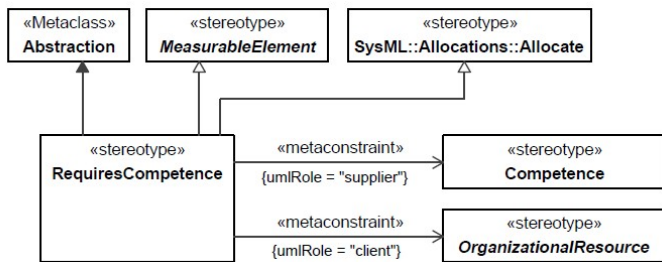


Figure 7.107 - RequiresCompetence

Constraints

- [1] `RequiresCompetence.client` Value for the client metaproperty must be stereotyped a specialization of «OrganizationalResource».
- [2] `RequiresCompetence.supplier` Value for the supplier metaproperty must be stereotyped «Competence» or its specializations.



### 7.1.7.5 UAF::Personnel::Traceability

Contains the elements that contribute to the Personnel Traceability Viewpoint.

#### ResponsibleFor

**Package:** Traceability

**isAbstract:** No

**Generalization:** [MeasurableElement](#), [Allocate](#)

**Extension:** Abstraction

Description

An abstraction relationship between an ActualResponsibleResource and an ActualResponsibility or ActualProject. It defines the duties that the ActualResponsibleResource is ResponsibleFor.

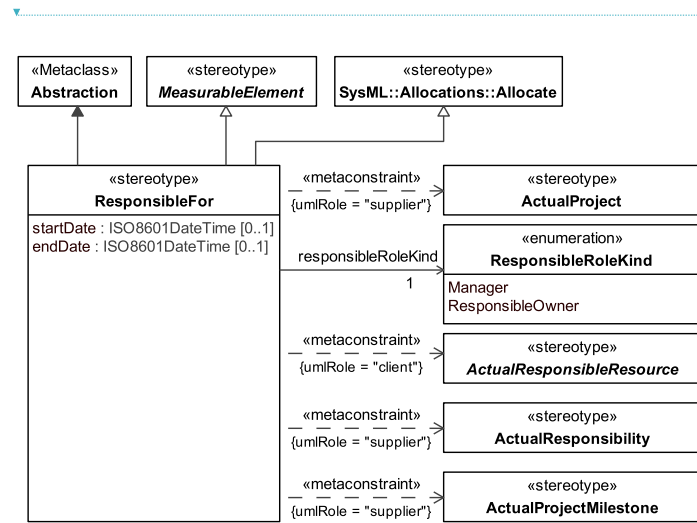


Figure 7.108 - ResponsibleFor

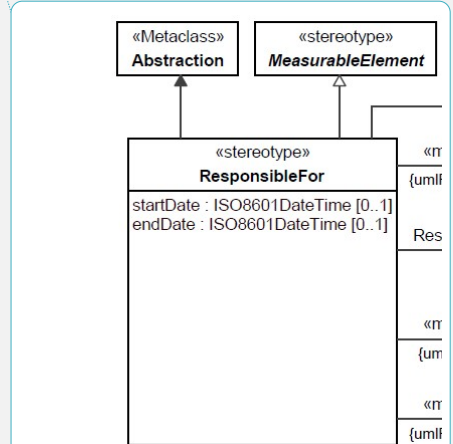
Attributes

endDate : ISO8601DateTime[0..1] End date of an ActualResponsibleResource being ResponsibleFor and ActualProject or ActualResponsibility.

startDate : ISO8601DateTime[0..1] Start date of an ActualResponsibleResource being ResponsibleFor and ActualProject or ActualResponsibility.

**Commented [AM45]: UAF11-121** Description under Responsible for changed from "Description An abstraction relationship between an ActualResponsibleResource and an ActualResponsibility or ActualProject. It defines the duties that the ActualResponsibleResource is ResponsibleFor." to "An abstraction relationship between an ActualResponsibleResource and an ActualResponsibility or ActualProject. It defines the duties that the ActualResponsibleResource is ResponsibleFor.".

**Deleted:** i



**Deleted:**

**Commented [Y46]: UAF11-46** Replace image with ResponsibleFor.svg

## Associations

**ResponsibleRoleKind :**  
**ResponsibleRoleKind[1]** Captures the kind of role (Manager or ResponsibleOwner) responsible for the ActualProject or ActualResponsibility.

## Constraints

[1] **ResponsibleFor.client** Value for the client metaproperty must be stereotyped by the specialization of «ActualResponsibleResource».

[2] **ResponsibleFor.supplier** Value for the supplier metaproperty must be stereotyped «ActualProject», «ActualResponsibility», or their specializations.

## ResponsibleRoleKind

**Package:** Traceability

**isAbstract:** No

### Description

**Enumeration of the possible kinds of ResponsibleFor relationship.** Its enumeration literals are:

- **Manager** - Indicates that the ResourceInteraction associated with the ResourceInteractionKind is an implementation of logical flow.
- **ResponsibleOwner** - Indicates that the ResourceInteraction associated with the ResourceInteractionKind is an implementation of logical flow.

**Commented [AM47]:** [UAF11-121](#) Description under ResponsibleRoleKind changed from "Enumeration of the possible kinds or ResponsibleRole." to "Enumeration of the possible kinds of ResponsibleFor relationship."

**Deleted:** Enumeration of the possible kinds or ResponsibleRole...

## 7.1.8 UAF::Resources

Stakeholders: Systems Engineers, Resource Owners, Implementers, Solution Providers, IT Architects.

Concerns: definition of solution architectures to implement operational requirements.

Definition: captures a solution architecture consisting of resources, e.g., organizational, software, artifacts, capability configurations, natural resources that implement the operational requirements. Further design of a resource is typically detailed in SysML or UML.

### 7.1.8.1 UAF::Resources::Taxonomy

Contains the elements that contribute to the Resources Taxonomy Viewpoint.

#### CapabilityConfiguration

**Package:** Taxonomy

**isAbstract:** No

**Generalization:** [ResourceArchitecture](#)

**Extension:** Class

#### Description

A composite structure representing the physical and human resources (and their interactions) in an enterprise, assembled to meet a capability.



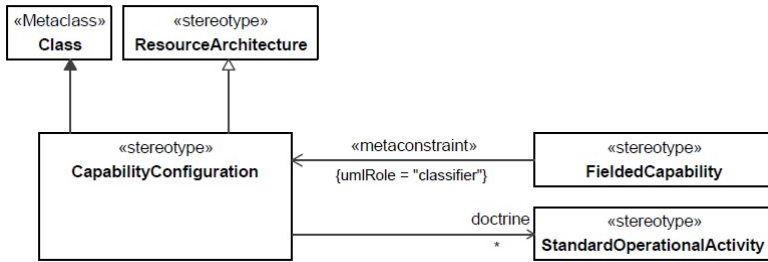


Figure 7.109 - CapabilityConfiguration

Associations

doctrine : StandardOperationalActivity[\*]      Represents the doctrinal line of development of the Capability.

**NaturalResource**

**Package:** Taxonomy

**isAbstract:** No

**Generalization:** [PhysicalResource](#)

**Extension:** Class

Description

Type of physical resource that occurs in nature such as oil, water, gas, or coal.

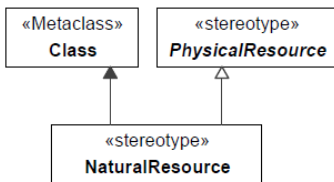


Figure 7.110 - NaturalResource

### PhysicalResource

**Package:** Taxonomy

**isAbstract:** Yes

**Generalization:** [ResourcePerformer](#)

**Extension:** Class

**Description**

An abstract grouping that defines physical resources (i.e., OrganizationalResource, ResourceArtifact, and NaturalResource).

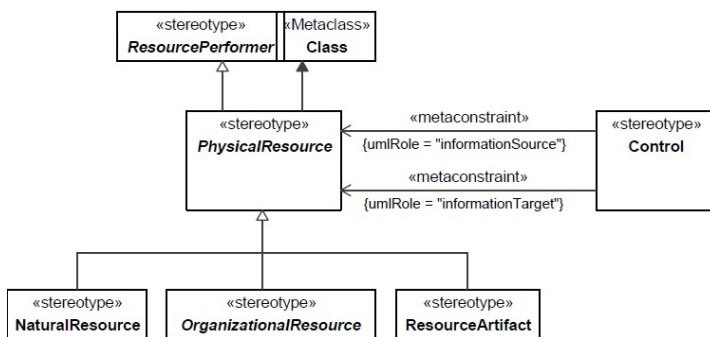


Figure 7.111 - PhysicalResource

### ResourceArchitecture

**Package:** Taxonomy

**isAbstract:** No

**Generalization:** [ResourcePerformer](#), [Architecture](#)

**Extension:** Class

**Description**

An element used to denote a model of the Architecture, described from the ResourcePerformer perspective.

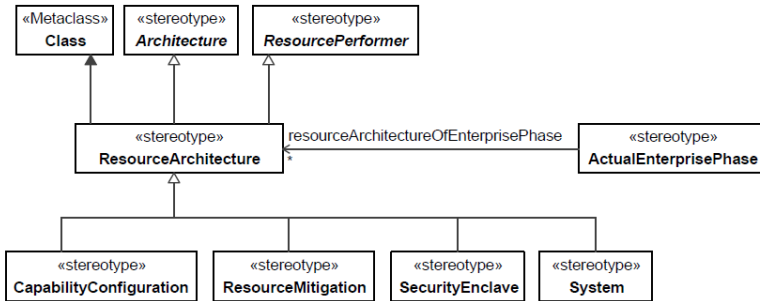


Figure 7.112 - ResourceArchitecture

**ResourceArtifact**

**Package:** Taxonomy

**isAbstract:** No

**Generalization:** [PhysicalResource](#)

**Extension:** Class

Description

A type of man-made object that contains no human beings (i.e., satellite, radio, petrol, gasoline, etc.).

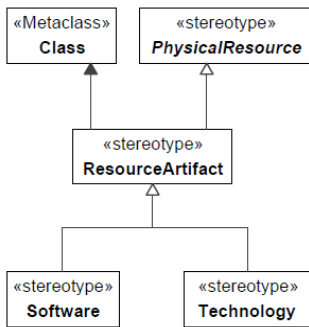


Figure 7.113 - ResourceArtifact

## ResourcePerformer

Package: Taxonomy

isAbstract: Yes

Generalization: [ResourceAsset](#), [ResourceExchangeItem](#), [SubjectOfResourceConstraint](#), [VersionedElement](#), [CapableElement](#), [SubjectOfForecast](#), [OperationalExchangeItem](#), [Desirer](#)

Extension: Class

Description

An abstract grouping of elements that can perform Functions.

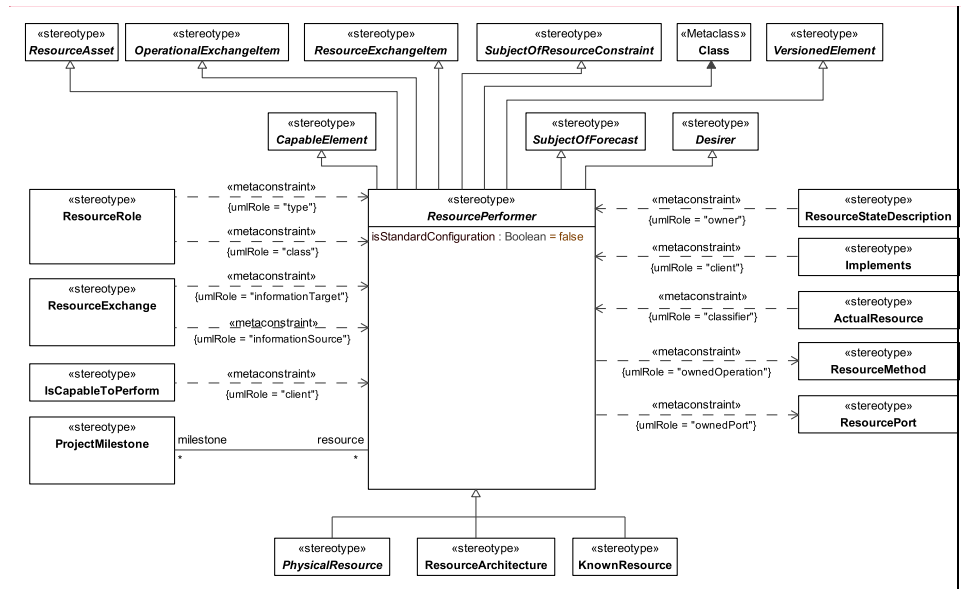


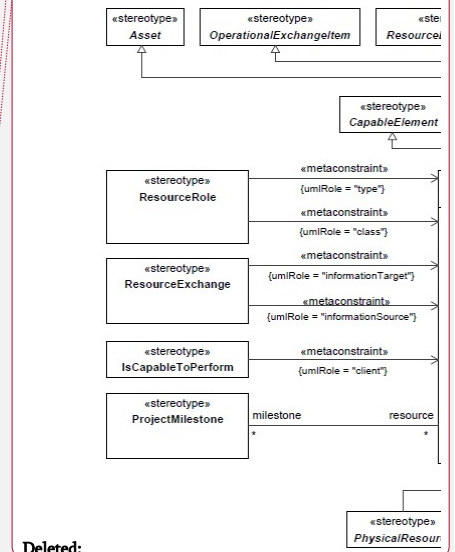
Figure 7.114 - ResourcePerformer

### Attributes

isStandardConfiguration : Boolean[] Indicates if the ResourcePerformer is StandardConfiguration, default=false.

Commented [AM48]: UAF11-16 Text "Asset" Replaced by "ResourceAsset"

Commented [AM49]: UAF11-16 Figure 7.114 – ResourcePerformer replaced by ResourcePerformer.svg



Deleted:

#### Associations

milestone : ProjectMilestone[\*]      Relates ResourcePerformer to ProjectMilestones that affect it.

#### Constraints

[1] ResourcePerformer.isCapableOfPerforming      Is capable of performing only «Function» elements or its specializations.

[2] ResourcePerformer.ownedOperation      Values for the ownedOperation metaproperty must be stereotyped «ResourceMethod» or its specializations.

[3] ResourcePerformer.ownedPort      Values for the ownedPort metaproperty must be stereotyped «ResourcePort» or its specializations.

### Software

**Package:** Taxonomy

**isAbstract:** No

**Generalization:** [ResourceArtifact](#)

**Extension:** Class

#### Description

A sub-type of ResourceArtifact that specifies an executable computer program.

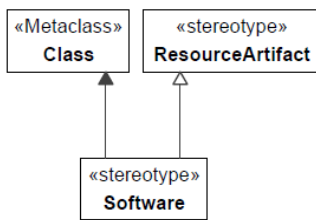


Figure 7.115 - Software

### System

**Package:** Taxonomy

**isAbstract:** No

**Generalization:** [ResourceArchitecture](#)

**Extension:** Class



Description

An integrated set of elements, subsystems, or assemblies that accomplish a defined objective. These elements include products (hardware, software, firmware), processes, people, information, techniques, facilities, services, and other support elements (INCOSE SE Handbook V4, 2015).

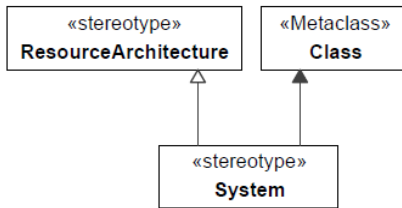


Figure 7.116 - System

7.1.8.2 UAF::Resources::Structure

Contains the elements that contribute to the Resources Structure Viewpoint.

ResourceMethod

Package: Structure

isAbstract: No

Generalization: [MeasurableElement](#)

Extension: Operation

Description

A behavioral feature of a ResourcePerformer whose behavior is specified in a Function.

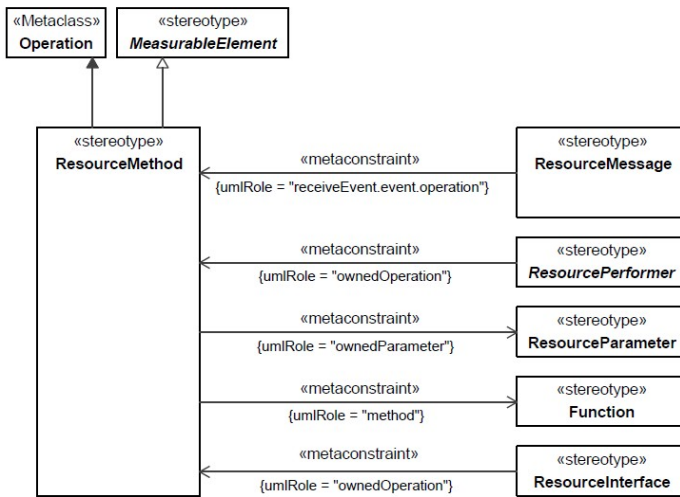


Figure 7.117 - ResourceMethod

Constraints

- [1] ResourceMethod.method Value for the method metaproperty must be stereotyped «Function» or its specializations.
- [2] ResourceMethod.ownedParameter The values for the ownedParameter metaproperty must be stereotyped «ResourceParameter».

**ResourceParameter**

**Package:** Structure

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Parameter

Description

An element that represents inputs and outputs of an Function. It is typed by a ResourceInteractionItem.

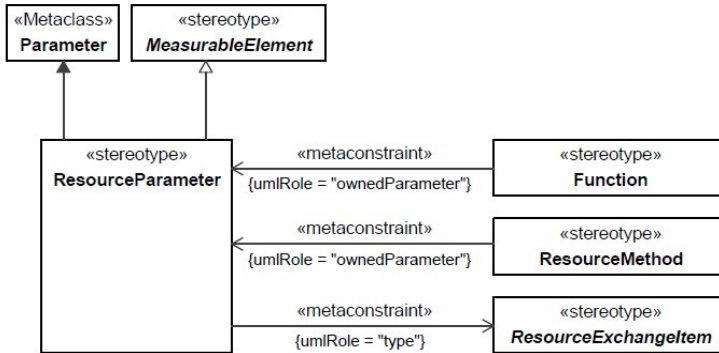


Figure 7.118 - ResourceParameter

Constraints

- [1] ResourceParameter.type Value for the type metaproperty must be stereotyped with a specialization of «ResourceInteractionItem».

**ResourcePort**

**Package:** Structure

**isAbstract:** No

**Generalization:** ProxyPort, [MeasurableElement](#), [ProtocolImplementation](#)

**Extension:** Port

Description

An interaction point for a ResourcePerformer through which it can interact with the outside environment and which is defined by a ResourceInterface.

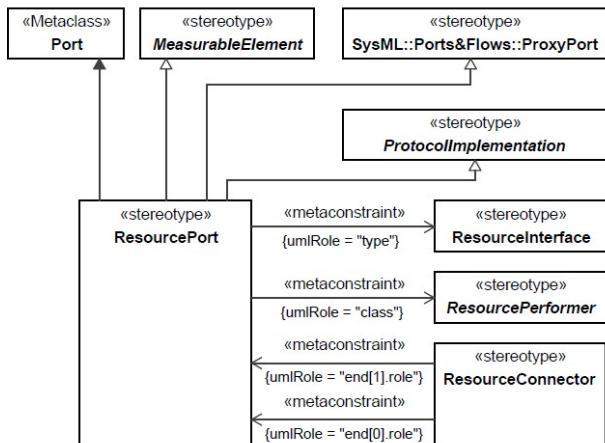


Figure 7.119 - ResourcePort

Constraints

- [1] ResourcePort.type Value for the type metaproperty must be stereotyped «ResourceInterface» or its specializations.
- [2] ResourcePort.class Value for the class metaproperty must be stereotyped by the specialization of «ResourcePerformer».

**ResourceRole**

**Package:** Structure

**isAbstract:** No

**Generalization:** [LocationHolder](#), [SubjectOfResourceConstraint](#), [MeasurableElement](#), [SubjectOfSecurityConstraint](#), [AssetRole](#)

**Extension:** Property

Description

Usage of a ResourcePerformer in the context of another ResourcePerformer. Creates a whole-part relationship.

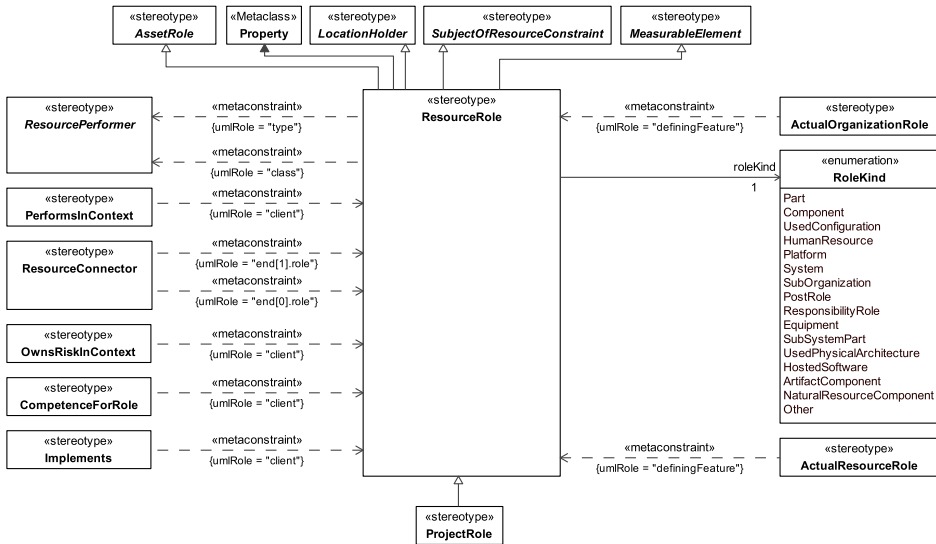


Figure 7.120 - ResourceRole

Associations

`roleKind : RoleKind[1]` Captures the kind of role a Resource can play.

Constraints

- [1] ResourceRole.type Value for the type metaproperty must be stereotyped by the specialization of «ResourcePerformer».
- [2] ResourceRole.class Value for the class metaproperty must be stereotyped by the specialization of «ResourcePerformer».

**RoleKind**

**Package:** Structure

**isAbstract:** No

Description

Unified Architecture Framework Profile (UAFP), v1.0

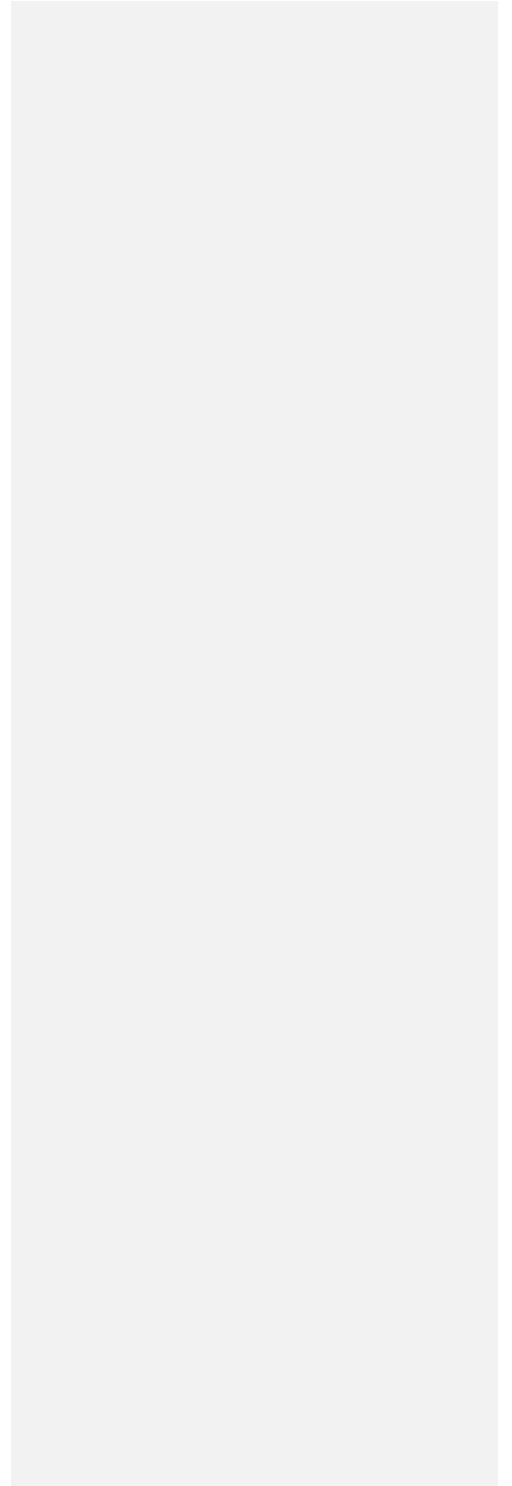
**Deleted:**

**Commented [Yvonne50]:** UAF11-47 also UAF11-153 replaced image. With ResourceRole.svg

**Commented [Yvonne51]:** UAF11-47 made roleKind start with lowercase

**Deleted:** R

Enumeration of the possible kinds of roles that a ResourceRole may play in the context of a ResourcePerformer. Its enumeration literals are:





- **Part** - Indicates that the ResourceRole associated with the ResourceRoleKind is a kind of a ResourcePerformer that is used as a part of another ResourcePerformer.
- **Component** - Indicates that the ResourceRole associated with the ResourceRoleKind is a kind of Software that is used in the context of a ResourcePerformer.
- **UsedConfiguration** - Indicates that the ResourceRole associated with the ResourceRoleKind is a kind of existing CapabilityConfiguration that is used in the context of a ResourcePerformer.
- **HumanResource** - Indicates that the ResourceRole associated with the ResourceRoleKind is a kind of human resource that is used in the context of a ResourcePerformer.
- **Platform** - Indicates that the ResourceRole associated with the ResourceRoleKind is a kind of a ResourcePerformer that represents a platform (e.g., vessel, aircraft, etc.) that is used in the context of a SystemsResource.
- **System** - Indicates that the ResourceRole associated with the ResourceRoleKind is a kind of assembly of ResourcePerformers that is used in the context of another ResourcePerformer.
- **SubOrganization** - Indicates that the ResourceRole associated with the ResourceRoleKind is a kind of Organization that is typically the parent of another - e.g., a squadron may be part of a battalion, that is used in the context of a ResourcePerformer.
- **PostRole** - Indicates that the ResourceRole associated with the ResourceRoleKind is a kind of Post that is used in the context of a ResourcePerformer.
- **ResponsibilityRole** - Indicates that the ResourceRole associated with the ResourceRoleKind is a kind of Responsibility associated with a role that is used in the context of a ResourcePerformer.
- **Equipment** - Indicates that the ResourceRole associated with the ResourceRoleKind is a kind of man made resource that is used to accomplish a task or function in the context of a ResourcePerformer.
- **SubSystemPart** - Indicates that the ResourceRole associated with the ResourceRoleKind is a kind of subsystem (represented as a ResourcePerformers) is part of another ResourcePerformer.
- **UsedPhysicalArchitecture** - Indicates that the ResourceRole associated with the ResourceRoleKind is a kind of existing PhysicalArchitecture that is used in the context of a ResourcePerformer.
- **HostedSoftware** - Indicates that the ResourceRole associated with the ResourceRoleKind is a kind of software that is used in the context of a ResourcePerformer.
- **ArtifactComponent** - Indicates that the ResourceRole associated with the ResourceRoleKind is a kind of non human resource that is used as a component in the context of a ResourcePerformer.
- **NaturalResourceComponent** - Indicates that the ResourceRole associated with the ResourceRoleKind is a kind of natural resource that is used as a component in the context of a ResourcePerformer.
- **Other** - Indicates that the ResourceRole associated with the ResourceRoleKind is another kind of RoleKind that is not on the enumerated list.

### 7.1.8.3 UAF::Resources::Connectivity

Contains the elements that contribute to the Resources Connectivity Viewpoint.

#### **ResourceConnector**

**Package:** Connectivity

**isAbstract:** No

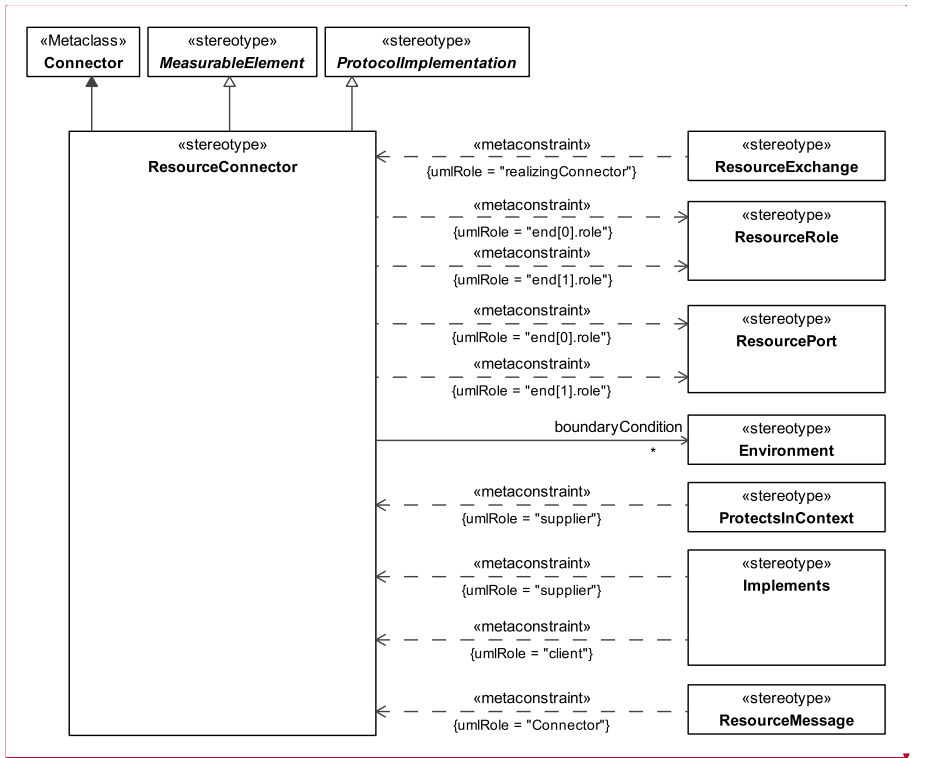
**Generalization:** [MeasurableElement](#), [ProtocolImplementation](#)

**Extension:** Connector

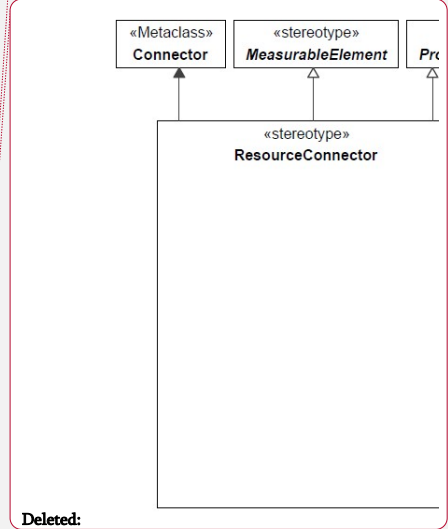


Description

A channel for exchange between two ResourceRoles.



Commented [AM52]: UAF11-157 Diagram replaced by the ResourceConnector.svg



Deleted:

Figure 7.121 - ResourceConnector

Associations

boundaryCondition : Environment[\*] Relates a ResourceConnector to the extremes of the Environment in which it is required to be made available.

Constraints

[1] ResourceConnector.end The value for the role metaproperty for the owned ConnectorEnd must be stereotype «ResourcePort», «ResourceRole», or their specializations.

**ResourceExchange**

**Package:** Connectivity

**isAbstract:** No

**Generalization:** [Exchange](#)

**Extension:** InformationFlow

Description

Asserts that a flow can exist between ResourcePerformers (i.e., flows of data, people, materiel, or energy).

Figure 7.122 - ResourceExchange

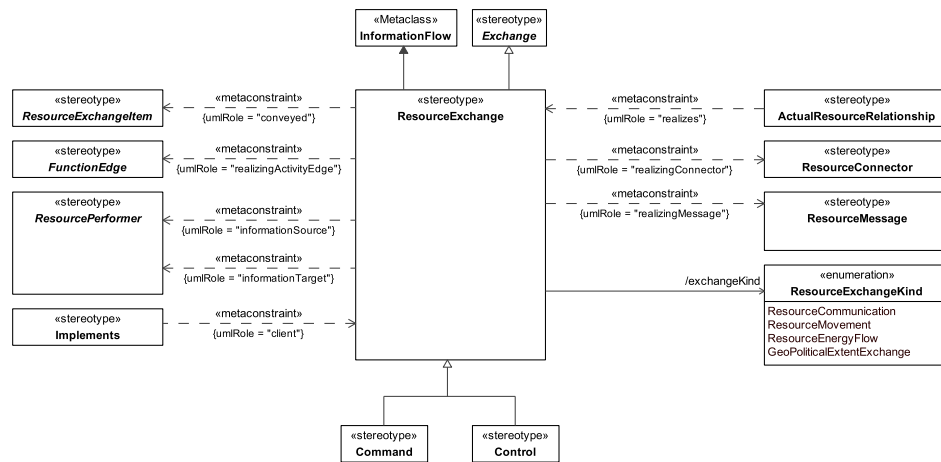
Associations

exchangeKind: ResourceExchangeKind [] Captures the kind of ResourceExchange.

Constraints

[1] ResourceExchange.conveyed

In case of ResourceExchange.exchangeKind:  
 = ResourceCommunication, the conveyed element must be stereotyped «DataElement» or its specializations.  
 = ResourceMovement, the conveyed element must be stereotyped by the specialization of «PhysicalResource».



= ResourceEnergyFlow, the conveyed element must be stereotyped «NaturalResource» or its specializations.  
 = GeoPoliticalExtentExchange, the conveyed element must be stereotyped «GeoPoliticalExtentType» or its specializations.

[2] ResourceInteraction.informationSource

Value for the informationSource metaproperty must be stereotyped by the specialization of «ResourcePerformers».

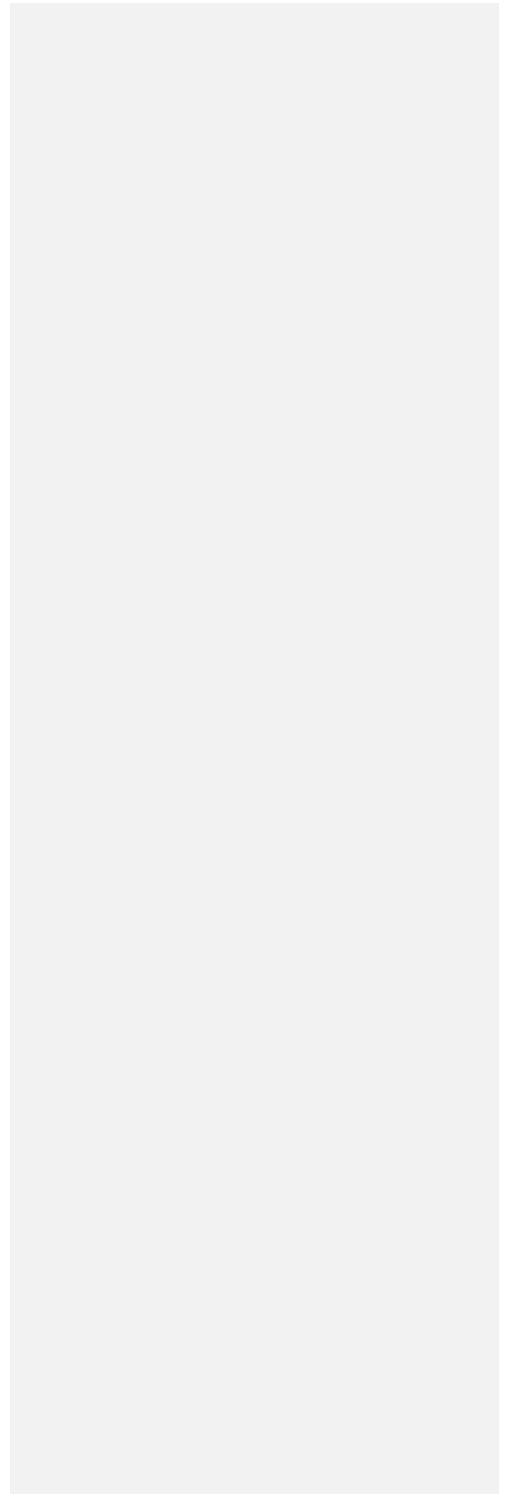
Deleted:

Commented [Y53]: UAF11-48 Replace image with ResourceExchange.svg

Commented [Y54]: UAF11-48 replaced ResourceExchangeKind with ResourceInteractionKind

Deleted: ResourceInteractionKind

[3] ResourceInteraction.informationTarget Value for the informationTarget metaproperty must be stereotyped by the



specialization of «ResourcePerformer».

- [4] ResourceInteraction.realizingActivityEdge Value for the realizingActivityEdge metaproperty must be stereotyped by the specialization of «FunctionEdge».
- [5] ResourceInteraction.realizingConnector Value for the realizingConnector metaproperty must be stereotyped «ResourceConnector» or its specializations.
- [6] ResourceInteraction.realizingMessage Value for the realizingMessage metaproperty must be stereotyped «ResourceMessage» or its specializations.

**ResourceExchangeItem**

**Package:** Connectivity

**isAbstract:** Yes

**Generalization:** [Resource](#)

Description

An abstract grouping for elements that defines the types of elements that can be exchanged between ResourcePerformers and conveyed by a ResourceExchange.

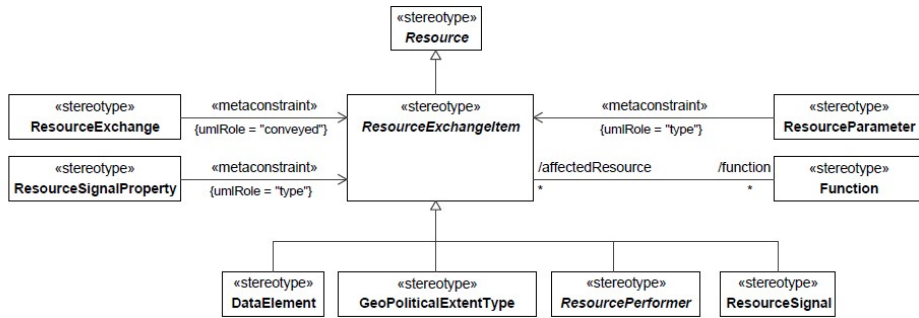


Figure 7.123 - ResourceExchangeItem

Associations

function : Function[\*] Function using the ResourceExchangeItem internally.

## ResourceExchangeKind

Package: Connectivity

isAbstract: No

### Description

Enumeration of the possible kinds of resource exchange applicable to a ResourceExchange. Its enumeration literals are:

- ResourceCommunication - Indicates that the ResourceInteraction associated with the ResourceExchangeKind is an implementation of logical flow of data between Resources.
- ResourceMovement - Indicates that the ResourceInteraction associated with the ResourceExchangeKind is an implementation of logical flow of Resources between Resources.
- ResourceEnergyFlow - Indicates that the ResourceInteraction associated with the ResourceExchangeKind is an implementation of logical flow of natural resources between Resources.
- GeoPoliticalExtentExchange - Indicates that the ResourceInteraction associated with the ResourceExchangeKind is an implementation of logical flow where GeoPoliticalExtents (i.e., Borders) flow from one place to another.

## ResourceInterface

Package: Connectivity

isAbstract: No

Generalization: PropertySet, InterfaceBlock

Extension: Class

### Description

A declaration that specifies a contract between the ResourcePerformers it is related to and any other ResourcePerformers it can interact with. It is also intended to be an implementation of a specification of an Interface in the Business and/or Service layer.

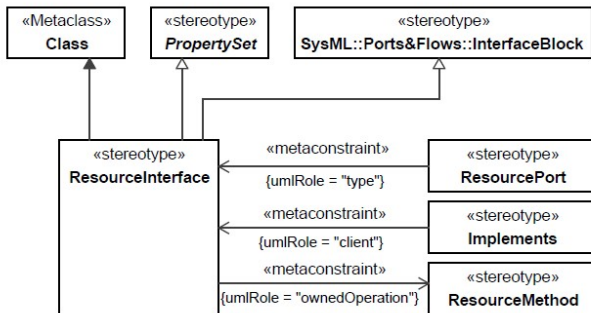


Figure 7.124 - ResourceInterface

Deleted: ResourceInteraction Kind ...

Deleted: ResourceInteractionKind

Deleted: ResourceInteractionKind

Commented [Y55]: UAF11-48 Replaced ResourceInteractionKind with ResourceExchangeKind

Constraints

- [1] ResourceInterface.ownedOperation Values for ownedOperation metaproperty must be stereotyped «ResourceMethod» or its specializations.

**ResourceSignal**

Package: Connectivity

isAbstract: No

Generalization: [ResourceExchangeItem](#)

Extension: Signal

Description

A ResourceSignal is a specification of a kind of communication between resources (ResourcePerformers) in which a reaction is asynchronously triggered in the receiver without a reply.

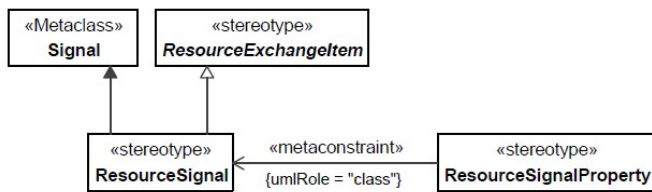


Figure 7.125 - ResourceSignal

**ResourceSignalProperty**

Package: Connectivity

isAbstract: No

Generalization: [MeasurableElement](#)

Extension: Property

Description

A property of an ResourceSignal typed by ResourceExchangeItem. It enables ResourceExchangeItem e.g., DataElement to be passed as arguments of the ResourceSignal.

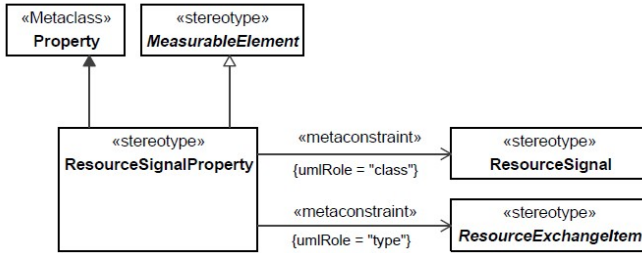


Figure 7.126 - ResourceSignalProperty

Constraints

- [1] ResourceSignalProperty.class Value for class metaproperty must be stereotyped «ResourceSignal» or its specializations.
- [2] ResourceSignalProperty.type Value for type metaproperty must be stereotyped by a specialization of «ResourceExchangeItem».

**7.1.8.4 UAF::Resources::Processes**

Contains the elements that contribute to the Resources Processes Viewpoint.

**Function**

**Package:** Processes

**isAbstract:** No

**Generalization:** [Activity](#), [SubjectOfResourceConstraint](#)

**Extension:** Activity

Description

An Activity which is specified in the context to the ResourcePerformer (human or machine) that IsCapableToPerform it.



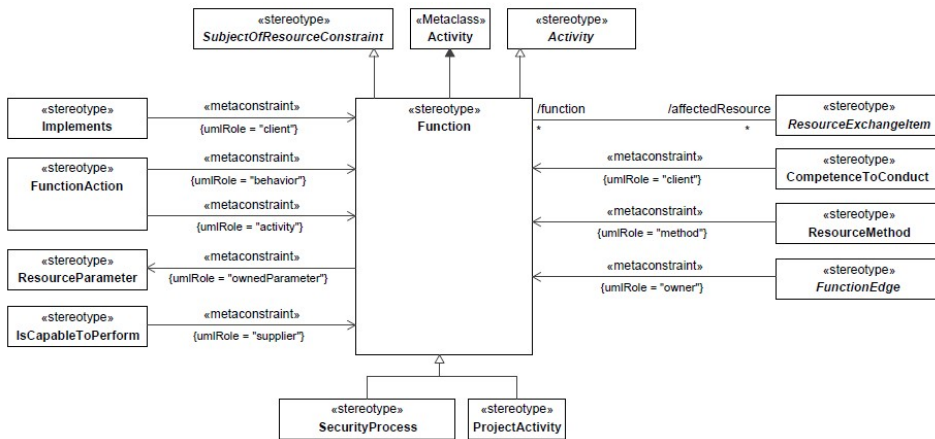


Figure 7.127 - Function

Associations

affectedResource : ResourceExchangeItem[\*]      ResourceExchangeItems consumed and produced internally within a Function.

Constraints

[1] Function.ownedParameter      The values for the ownedParameter metaproperty must be stereotyped «ResourceParameter» or its specializations.

**FunctionAction**

**Package:** Processes

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** CallBehaviorAction

Description

A call of a Function indicating that the Function is performed by a ResourceRole in a specific context.

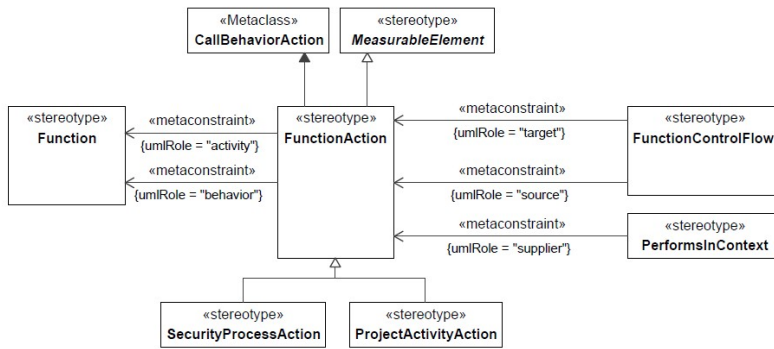


Figure 7.128 - FunctionAction

Constraints

- [1] FunctionAction.activity Value for the activity metaproperty must be stereotyped «Function» or its specializations.
- [2] FunctionAction.behavior Value for the behavior metaproperty must be stereotyped «Function» or its specializations.

**FunctionControlFlow**

**Package:** Processes

**isAbstract:** No

**Generalization:** [FunctionEdge](#)

**Extension:** ControlFlow

Description

An ActivityEdge that shows the flow of control between FunctionActions.

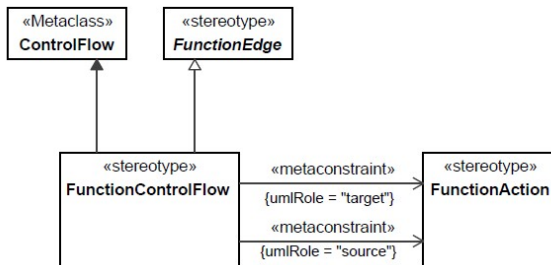


Figure 7.129 - FunctionControlFlow

Constraints

- [1] FunctionControlFlow.source Value for the source metaproperty must be stereotyped «FunctionAction» or its specializations.
- [2] FunctionControlFlow.target Value for the target metaproperty must be stereotyped «FunctionAction» or its specializations.

**FunctionEdge**

**Package:** Processes

**isAbstract:** Yes

**Generalization:** [MeasurableElement](#)

**Extension:** ActivityEdge

Description

Abstract grouping for FunctionControlFlow and FunctionObjectFlow.

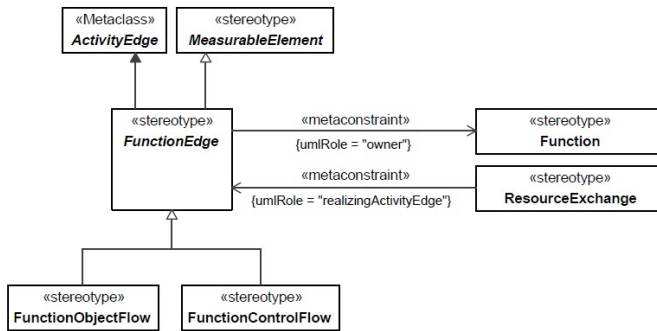


Figure 7.130 - FunctionEdge

Constraints

[1] FunctionEdge.owner «FunctionEdge» must be owned directly or indirectly by «Function» or its specializations.

**FunctionObjectFlow**

**Package:** Processes

**isAbstract:** No

**Generalization:** [FunctionEdge](#)

**Extension:** ObjectFlow

Description

An ActivityEdge that shows the flow of Resources (objects/data) between FunctionActions.

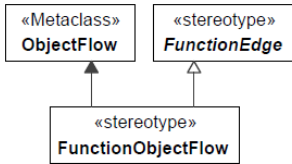


Figure 7.131 - FunctionObjectFlow

### 7.1.8.5 UAF::Resources::States

Contains the elements that contribute to the Resources States Viewpoint.

#### ResourceStateDescription

**Package:** States

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** StateMachine

**Description**

A state machine describing the behavior of a ResourcePerformer, depicting how the ResourcePerformer responds to various events and the actions.

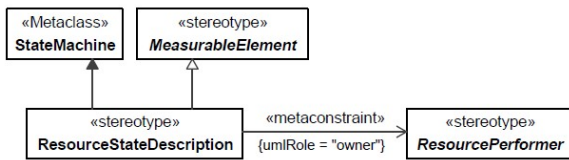


Figure 7.132 - ResourceStateDescription

Constraints

[1] ResourceStateDescription.owner Values for the owner metaproperty must be stereotyped with the specialization of «ResourcePerformer».

### 7.1.8.6 UAF::Resources::Interaction Scenarios

Contains the elements that contribute to the Resources Interaction Scenarios Viewpoint.

#### ResourceMessage

**Package:** Interaction Scenarios

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Message

Description

Message for use in a Resource Event-Trace which carries any of the subtypes of ResourceExchange.

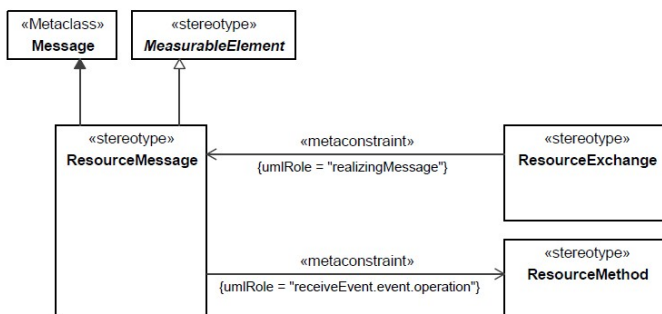


Figure 7.133 - ResourceMessage

Constraints

[1] ResourceMessage.receiveEvent.event.operation Values for the receiveEvent.event.operation metaproperty must be stereotyped with «ResourceMethod» or its specializations.

### 7.1.8.7 UAF::Resources::Information

Contains the elements that contribute to the Resources Information Viewpoint.

#### DataElement

**Package:** Information

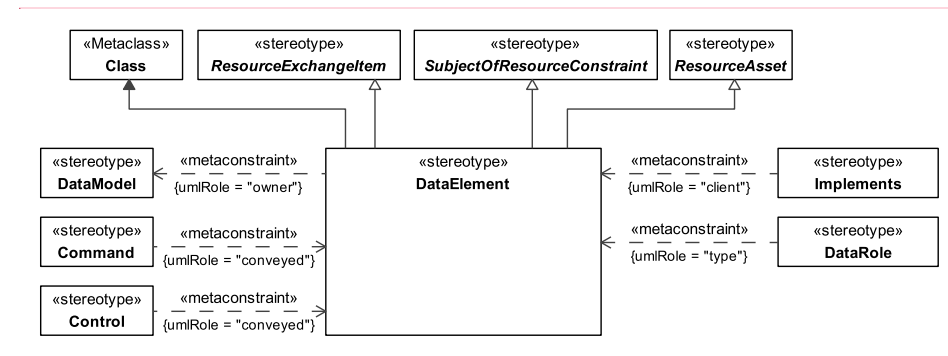
**isAbstract:** No

**Generalization:** [ResourceExchangeItem](#), [SubjectOfResourceConstraint](#), [ResourceAsset](#)

**Extension:** Class

**Description:**

A formalized representation of data that is managed by or exchanged between systems.



**Commented [AM56]:** [UAF11-16](#) In the generalizations list "Asset" replaced by "ResourceAsset".  
**Deleted:** [Asset](#)

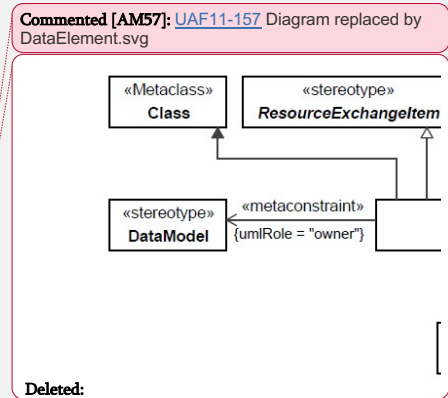


Figure 7.134 - DataElement

#### Constraints

[1] DataElement.owner Values for the owner metaproperty must be stereotyped «DataModel» or its specializations.

#### DataModel

**Package:** Information

**isAbstract:** No

**Generalization:** [SubjectOfOperationalConstraint](#), [SubjectOfResourceConstraint](#)

**Extension:** Package

**Commented [Y58]:** UAF11-53 Added SubjectOfResourceConstraint

Description

A structural specification of data types, showing relationships between them that is devoid of implementation detail. The type of data captured in the DataModel is described using the enumeration DataModelKind (Conceptual, Logical, and Physical).

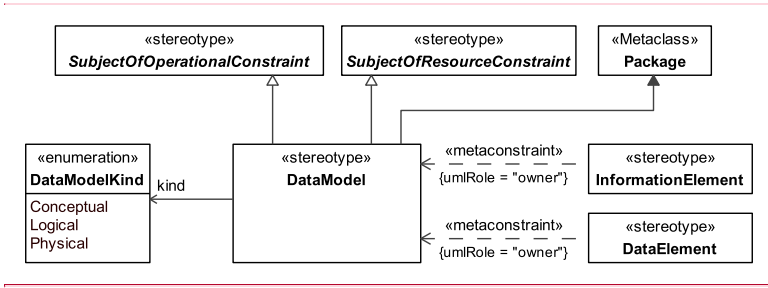


Figure 7.135 - DataModel

Associations

kind : DataModelKind[] Captures the kind of DataModel being represented, Conceptual, Logical, or Physical.

DataModelKind

Package: Information

isAbstract: No

Description

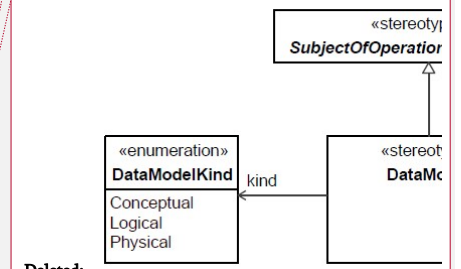
Enumeration of the possible kinds of DataModel. Its enumeration literals are:

- Conceptual - Indicates that the DataModel associated with the DataModelKind is a conceptual DataModel that defines the required high-level data concepts and their relationships.
- Logical - Indicates that the DataModel associated with the DataModelKind is a logical data model that allows analysis of an architecture’s data definition aspect, without consideration of implementation specific or product specific issues. It details the conceptual data model.
- Physical - Indicates that the DataModel associated with the DataModelKind is a physical data model that is an implementable specification of a data structure. A physical data model realizes a logical data model, taking into account implementation restrictions and performance issues while still enforcing the constraints, relationships and typing of the logical data model.

7.1.8.8 UAF::Resources::Constraints

Contains the elements that contribute to the Resources Constraints Viewpoint.

Commented [AM59]: UAF11-52, UAF11-53 Figure 7.135 – DataModel replaced by DataModel.svg



Deleted:



## ResourceConstraint

**Package:** Constraints

**isAbstract:** No

**Generalization:** [Rule](#)

**Extension:** Constraint

Description

A rule governing the structural or functional aspects of an implementation.

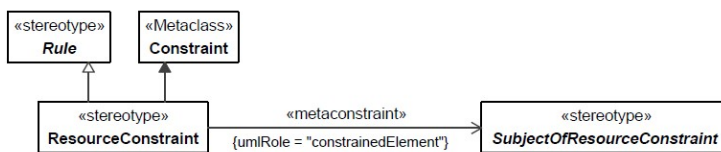


Figure 7.136 - ResourceConstraint

Constraints

[1] ResourceConstraint.constrainedElement Value for the constrainedElement metaproperty must be stereotyped by the specialization of «SubjectOfResourceConstraint».

## SubjectOfResourceConstraint

**Package:** Constraints

**isAbstract:** Yes

**Generalization:** [UAFElement](#)

**Extension:** Element

Description

An abstract grouping of elements that can be the subject of a ResourceConstraint.

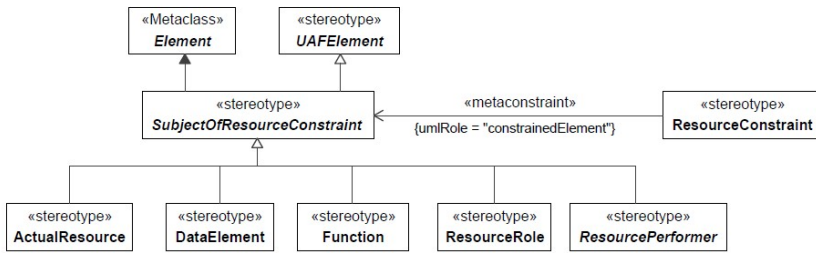


Figure 7.137 - SubjectOfResourceConstraint

### 7.1.8.9 UAF::Resources::Roadmap

Contains the elements that contribute to the Resources Roadmap Viewpoint.

#### Forecast

**Package:** Roadmap

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Dependency

**Description**

A dependency relationship that specifies a transition from one Asset, Standard, Competence to another future one. It is related to an ActualEnterprisePhase to give it a temporal context.

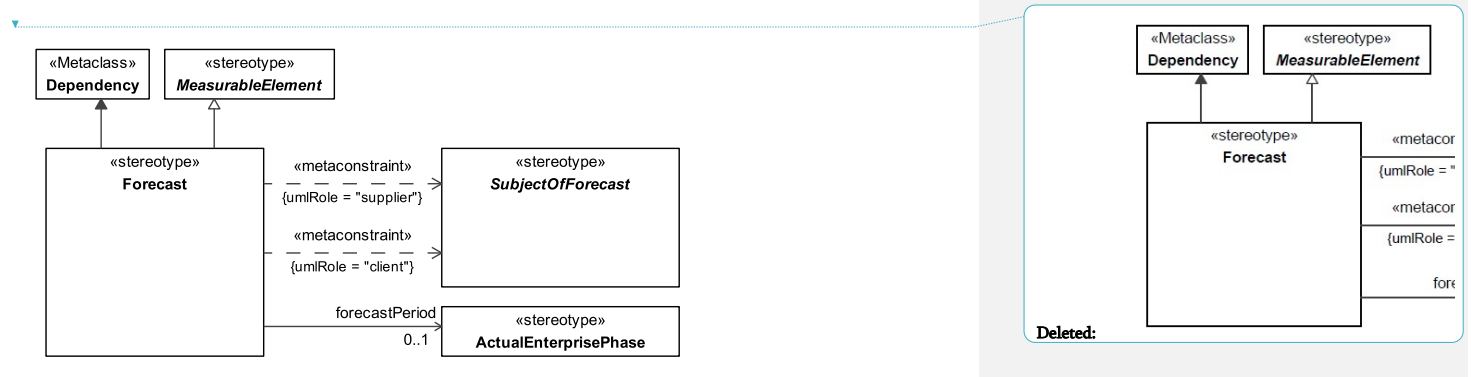


Figure 7.138 - Forecast

**Commented [Y60]:** UAF11-35 replaced image with forecast.svg

Associations

forecastPeriod : ActualEnterprisePhase[0..1] Relates the SubjectOfForecast to the ActualEnterprisePhase in which the SubjectOfForecast is expected to be provided.

Constraints

- [1] Forecast.client Value for the client metaproperty must be stereotyped by the specialization of «SubjectOfForecast».
- [2] Forecast.pair Values for the client and supplier metaproperties must be stereotyped by the same specialization of «SubjectOfForecast» (e.g., «Software» to «Software», «Standard» to «Standard», etc.).
- [3] Forecast.supplier Value for the supplier property must be stereotyped by the specialization of «SubjectOfForecast».

**Deleted:** \*

**Commented [Y61]:** UAF11-35 replaced forecastPeriod : ActualEnterprisePhase[\*] - Relates the SubjectOfForecast to the ActualEnterprisePhase in which the SubjectOfForecast is expected to be provided." with:  
forecastPeriod : ActualEnterprisePhase[\*0..1] - Relates the SubjectOfForecast to the ActualEnterprisePhase in which the SubjectOfForecast is expected to be provided.

**SubjectOfForecast**

Package: Roadmap

isAbstract: Yes

Generalization: [UAFElement](#)

Extension: Class

Description

An abstract grouping of elements that can be the subject of a Forecast.

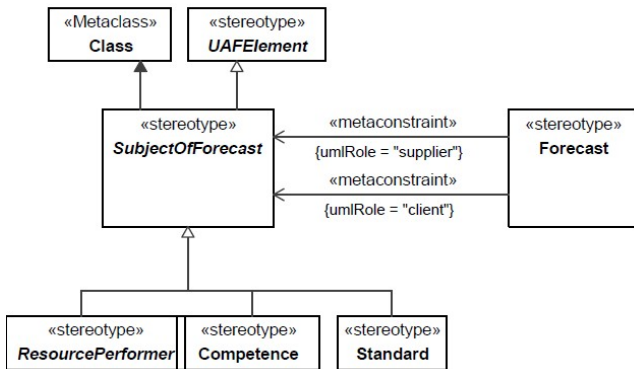


Figure 7.139 - SubjectOfForecast

## Technology

**Package:** Roadmap

**isAbstract:** No

**Generalization:** [ResourceArtifact](#)

**Extension:** Class

Description

A sub type of ResourceArtifact that indicates a technology domain, i.e., nuclear, mechanical, electronic, mobile telephony, etc.

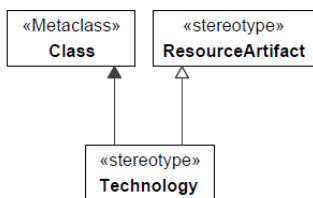


Figure 7.140 - Technology

## VersionedElement

**Package:** Roadmap

**isAbstract:** Yes

**Generalization:** [UAFElement](#)

**Extension:** Class

Description

An abstract grouping of ResourcePerformer and ServiceSpecification that allows VersionOfConfiguration to be related to ActualProjectMilestones.

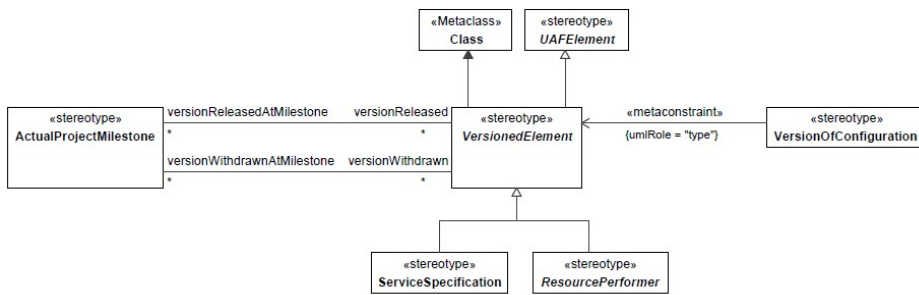


Figure 7.141 - VersionedElement

Associations

- versionReleasedAtMilestone : ActualProjectMilestone[\*] Relates a VersionedElement to the ActualProjectMilestone. It indicates the ActualProjectMilestone at which the VersionedElement is released.
- versionWithdrawnAtMilestone : ActualProjectMilestone[\*] Relates a VersionedElement to the ActualProjectMilestone. It indicates the ActualProjectMilestone at which the VersionedElement is withdrawn.

**VersionOfConfiguration**

**Package:** Roadmap

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Property

Description

A property of a WholeLifeConfiguration, used in version control of a VersionedElement. It asserts that a VersionedElement is a version of a WholeLifeConfiguration.



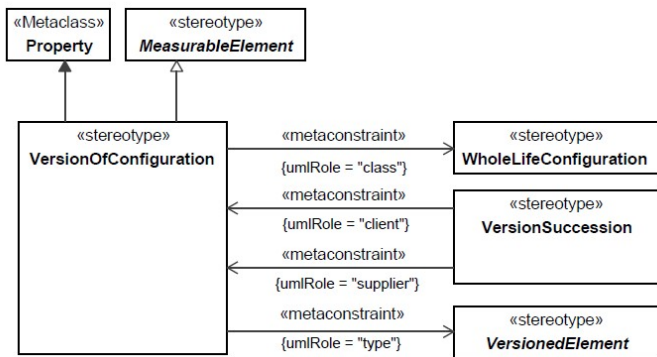


Figure 7.142 - VersionOfConfiguration

Constraints

- [1] VersionOfConfiguration.class Value for the class metaproperty must be stereotyped «WholeLifeConfiguration» or its specializations.
- [2] VersionOfConfiguration.type Value for the type metaproperty must be stereotyped by the specialization of «VersionedElement».

**VersionSuccession**

**Package:** Roadmap

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Dependency

Description

A dependency relationship between two VersionOfConfigurations that denotes that one VersionOfConfiguration follows from another.



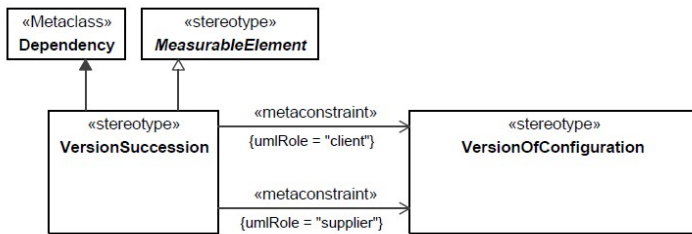


Figure 7.143 - VersionSuccession

Constraints

- [1] VersionSuccession.client Value for the client metaproperty must be stereotyped «VersionOfConfiguration» or its specializations.
- [2] VersionSuccession.supplier Value for the supplier metaproperty must be stereotyped «VersionOfConfiguration» or its specializations.

**WholeLifeConfiguration**

**Package:** Roadmap

**isAbstract:** No

**Generalization:** [PropertySet](#), Block

**Extension:** Class

Description

A set of VersionedElements.

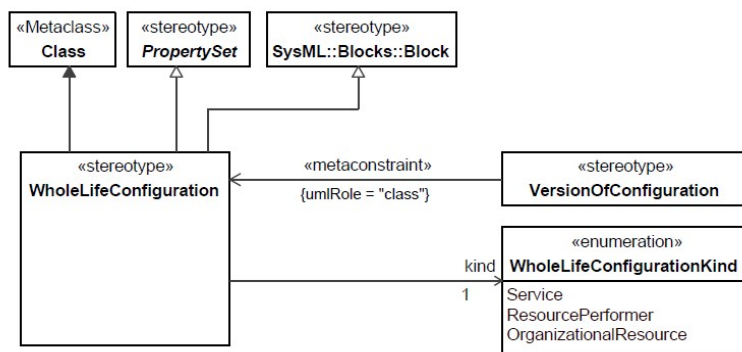


Figure 7.144 - WholeLifeConfiguration

Associations

kind : WholeLifeConfigurationKind[1]      Captures the kind of WholeLifeConfiguration.

**WholeLifeConfigurationKind**

Package: Roadmap

isAbstract: No

Description

Enumeration of the possible kinds of WholeLifeConfiguration. Its enumeration literals are:

- Service - Indicates that the WholeLifeConfiguration associated with the WholeLifeConfigurationKind is the master specification from which Services are versioned.
- ResourcePerformer - Indicates that the WholeLifeConfiguration associated with the WholeLifeConfigurationKind is the master specification from which ResourcePerformers are versioned.
- OrganizationalResource - Indicates that the WholeLifeConfiguration associated with the WholeLifeConfigurationKind is the master specification from which OrganizationalResources are versioned.

**7.1.8.10 UAF::Resources::Traceability**

Contains the elements that contribute to the Resources Traceability Viewpoint.

## ProtocollImplementation

**Package:** Traceability

**isAbstract:** Yes

**Generalization:** [UAFElement](#)

**Extension:** Element

**Description**

An abstract grouping of architectural elements that can implement Protocols.

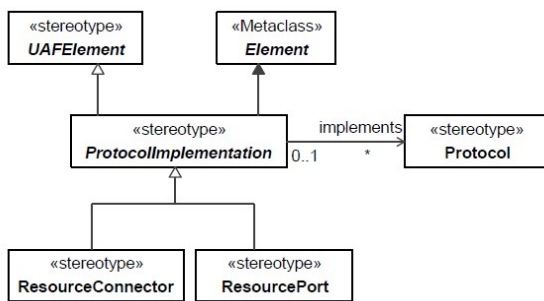


Figure 7.145 - ProtocollImplementation

**Associations**

implements : Protocol[\*] Relates the ResourceConnector and ResourcePort to the Protocols that they can implement.

### 7.1.9 UAF::Security

**Stakeholders:** Security Architects, Security Engineers, Systems Engineers, Operational Architects.

**Concerns:** addresses the security constraints and information assurance attributes that exist on exchanges between resources and OperationalPerformers

**Definition:** illustrates the security assets, security constraints, security controls, families, and measures required to address specific security concerns.

#### 7.1.9.1 UAF::Security::Taxonomy

Contains the elements that contribute to the Security Taxonomy Viewpoint.

**Asset**

**Package:** Taxonomy

isAbstract: Yes

Generalization: [ConceptItem](#), [PropertySet](#), [LocationHolder](#), [SubjectOfSecurityConstraint](#), [Block](#)

Extension: Class

Description

Asset as applied to Security views, an abstract element that indicates the types of elements that can be considered as a subject for security analysis.

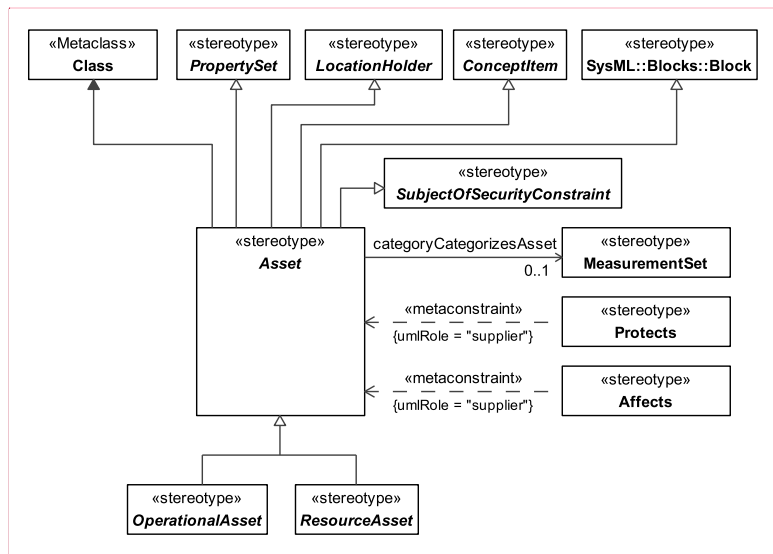


Figure 7.146 - Asset

Associations

categoryCategorizesAsset : MeasurementSet[0..1] Enables association of an Asset to the set of security related measurements (MeasurementSet).

### OperationalAsset

Package: [Taxonomy](#)

isAbstract: Yes

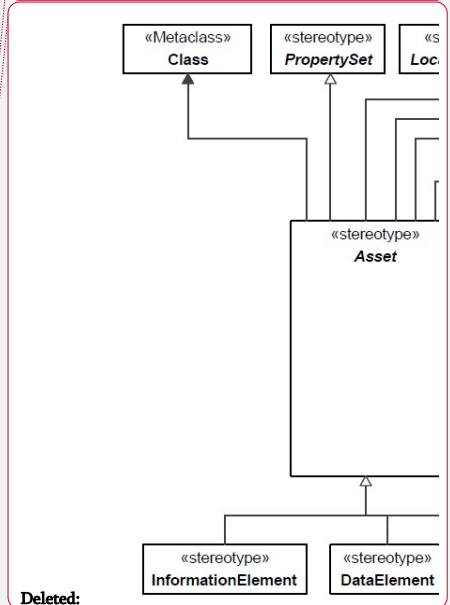
Generalization: [Asset](#)

Extension: [Class](#)

Description

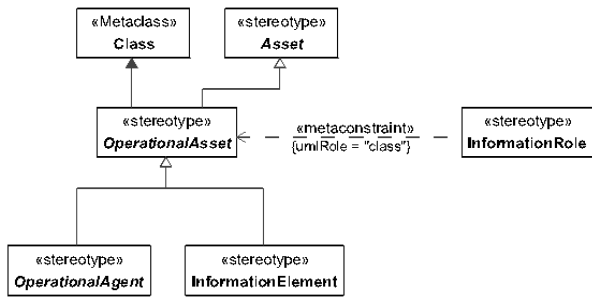
An abstract element used to group the elements of OperationalAgent and InformationElement allowing them to own InformationRoles.

Commented [AM62]: [UAF11-16](#) Figure 7.146 – Asset replaced by Asset.svg



Deleted:

Commented [AM63]: [UAF11-16](#) OperationalAsset section added.



**Figure 3:147 - OperationalAsset**

**OperationalMitigation**

**Package:** Taxonomy

**isAbstract:** No

**Generalization:** [OperationalArchitecture](#)

**Extension:** Class

**Description**

A set of security measures intended to address against specific cyber risks. Comprises a subset of SecurityControls that are required to protect the asset at OperationalPerformer (OperationalRole).

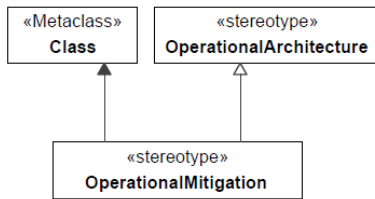


Figure 7.147 - OperationalMitigation

### **ResourceAsset**

**Package:** Taxonomy

**isAbstract:** Yes

**Generalization:** Asset

**Extension:** Class

**Description**

An abstract element used to group the elements of ResourcePerformer and DataElement allowing them to own DataRoles

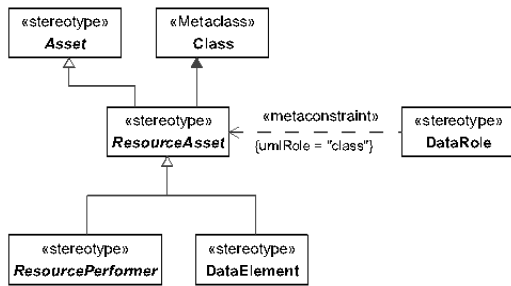


Figure 3:149 - ResourceAsset

### **ResourceMitigation**

**Package:** Taxonomy

**isAbstract:** No

**Generalization:** [ResourceArchitecture](#)

**Extension:** Class

156

Commented [AM64]: UAF11-16 ResourceAsset section added.

Description

A set of security measures intended to address specific cyber risks. Comprises a subset of TailoredSecurityControls that are used to protect the asset at resource (ResourceRole).

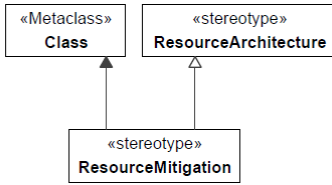


Figure 7.148 - ResourceMitigation

## SecurityEnclave

**Package:** Taxonomy

**isAbstract:** No

**Generalization:** [ResourceArchitecture](#)

**Extension:** Class

Description

Collection of information systems connected by one or more internal networks under the control of a single authority and security policy. The systems may be structured by physical proximity or by function, independent of location.

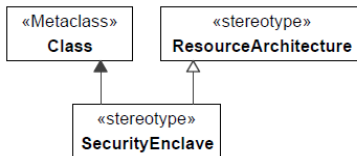


Figure 7.149 - SecurityEnclave

### 7.1.9.2 UAF::Security::Structure

Contains the elements that contribute to the Security Structure Viewpoint.

**AssetRole Package:**

Structure **isAbstract:** Yes

**Generalization:** [UAFElement](#)

**Extension:** Element

Description

AssetRole as applied to Security views, an abstract element that indicates the type of elements that can be considered as a subject for security analysis in the particular context.



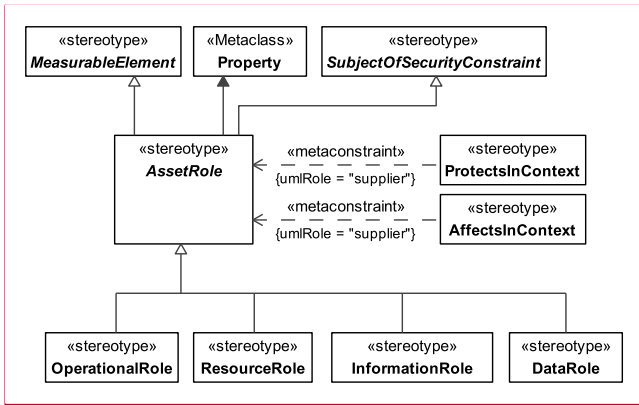
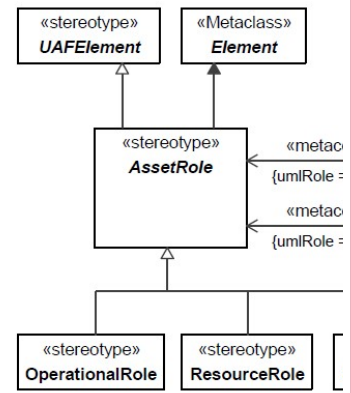


Figure 7.150 - AssetRole

Commented [AM65]: UAF11-16 Figure 7.150 – AssetRole replaced by AssetRole.svg



Deleted:

Commented [AM67]: UAF11-16 SecurityProperty section removed.

Deleted: SecurityProperty  
 Package: Structure isAbstract: No  
 Generalization: MeasurableElement, AssetRole  
 Extension: Property  
 Description:

SecurityProperty is used to assign an aggregated security marking (from the SecurityAttributes enumerated list: ClassificationType) to designate this "aggregated" security classification. The inter-connectivity of different data sets may allow more sensitive connections to be made by association. Aggregation, accumulation, and association of data (within ICT systems and on removable media) must be carefully considered as part of the risk management process as additional protective controls may or may not be appropriate. Aggregation does not ... [4]

Deleted:

Commented [AM68]: UAF11-16 DataRole section added.

Formatted: Body Text

### DataRole

Package: Structure

isAbstract: No

Generalization: AssetRole

Extension: Property

Description

A usage of DataElement that exists in the context of an ResourceAsset. It also allows the representation of the whole-part aggregation of DataElements.

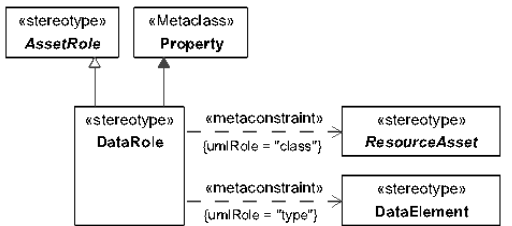


Figure 3:153 – DataRole Constraints

- [1] DataRole.class Value for the class metaproperty must be stereotyped by the specialization of «ResourceAsset».
- [2] DataRole.type Value for the type metaproperty must be stereotyped «DataElement» or its specializations.

## InformationRole

**Package:** Structure

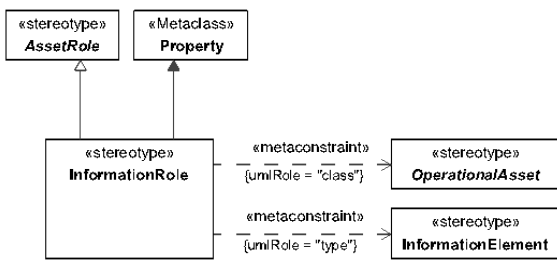
**isAbstract:** No

**Generalization:** AssetRole

**Extension:** Property

**Description**

A usage of InformationElement that exists in the context of an OperationalAsset. It also allows the representation of the whole-part aggregation of InformationElements.



**Figure 3:154 - InformationRole**

### Constraints

- [1] **InformationRole.class** Value for the class metaproperty must be stereotyped by the specialization of `«OperationalAsset»`.
- [2] **InformationRole.type** Value for the type metaproperty must be stereotyped `«InformationElement»` or its specializations.

### 7.1.9.3 UAF::Security::Processes

Contains the elements that contribute to the Security Processes Viewpoint.

#### EnhancedSecurityControl

**Package:** Processes

**isAbstract:** No

**Generalization:** [SecurityControl](#)

**Extension:** Class

Commented [AM69]: UAF11-16 InformationRole section added.

Description

Statement of security capability to: (i) build in additional but related, functionality to a basic control; and/or (ii) increase the strength of a basic control.

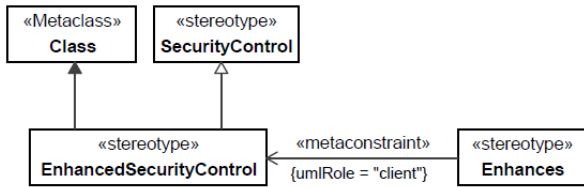


Figure 7.152 - EnhancedSecurityControl

**Enhances**

**Package:** Processes

**isAbstract:** No

**Generalization:** [MeasurableElement](#), [DeriveReq](#)

**Extension:** Abstraction

Description

A dependency relationship relating the EnhancedSecurityControl to a SecurityControl.

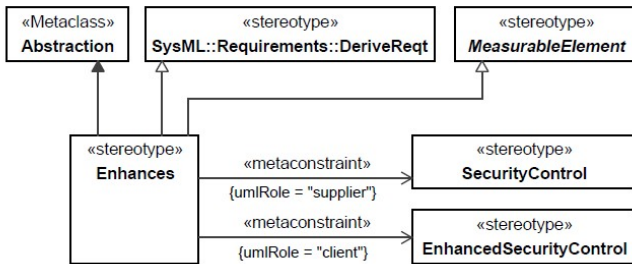


Figure 7.153 - Enhances

Constraints

- [1] Enhances.client Value for the client metaproperty must be stereotyped «EnhancedSecurityControl» or its specializations.
- [2] Enhances.supplier Value for the supplier metaproperty must be stereotyped «SecurityControl» or its specializations.

**Protects**

Package: Processes

isAbstract: No

Generalization: [MeasurableElement](#)

Extension: Dependency

Description

A dependency that asserts that a SecurityControl is required to protect an Asset.

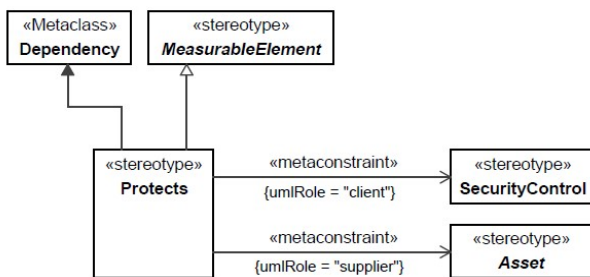


Figure 7.154 - Protects

Constraints

- [1] Protects.client Value for the client metaproperty must be stereotyped «SecurityControl» or its specializations.
- [2] Protects.supplier Value for the supplier metaproperty must be stereotyped by the specialization of «Asset».

### ProtectsInContext

**Package:** Processes

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Dependency

Description

A dependency relationship that relates a SecurityControlAction to an OperationalRole, or a ResourceRole. It indicates that SecurityControl is required to protect an Asset in a specific context or configuration.

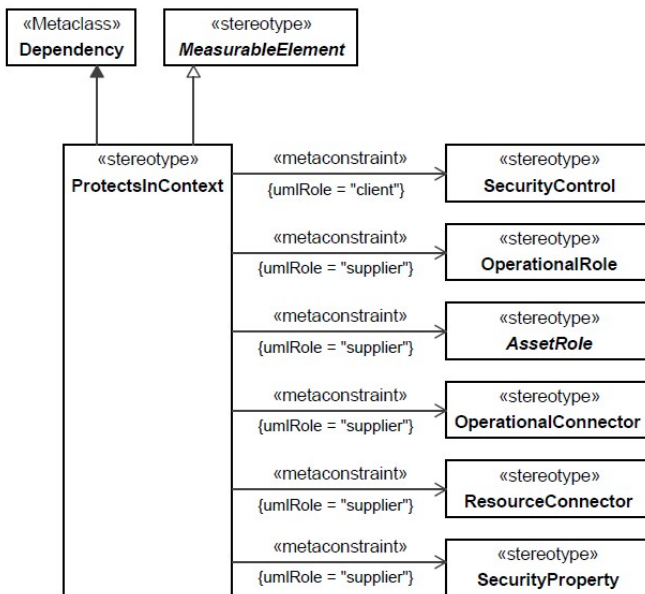


Figure 7.155 - ProtectsInContext

Constraints

- [1] ProtectsInContext.client Value for the client metaproperty must be stereotyped «SecurityControlAction» or its specializations.
- [2] ProtectsInContext.supplier Value for the supplier metaproperty must be stereotyped «OperationalRole», «ResourceRole», «OperationalConnector», «ResourceConnector», «SecurityProperty», or their specializations.

**SecurityProcess**

**Package:** Processes

**isAbstract:** No

**Generalization:** [OperationalActivity](#), [Function](#)

**Extension:** Activity

Description

The security-related procedure that satisfies the security control requirement.

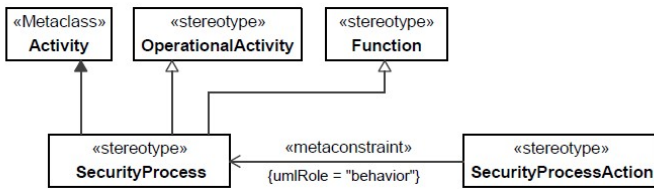


Figure 7.156 - SecurityProcess

**SecurityProcessAction**

**Package:** Processes

**isAbstract:** No

**Generalization:** [OperationalActivityAction](#), [FunctionAction](#)

**Extension:** CallBehaviorAction

Description

A call of a SecurityProcess in the context of another SecurityProcess.

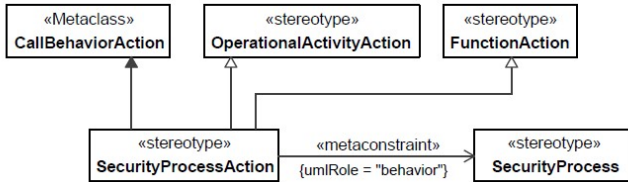


Figure 7.157 - SecurityProcessAction

Constraints

[1] SecurityControlAction.behavior Value for behavior metaproperty must be stereotyped «SecurityControl» or its specializations.

7.1.9.4 UAF::Security::Constraints

Contains the elements that contribute to the Security Constraints Viewpoint.

**ActualRisk**

**Package:** Constraints

**isAbstract:** No

**Generalization:** [ActualPropertySet](#)

**Extension:** InstanceSpecification

Description

An instance of a Risk. A value holder for Risk Measurements.

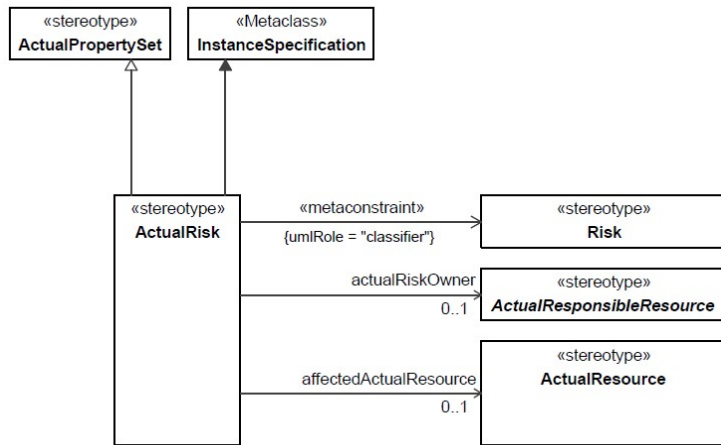


Figure 7.158 - ActualRisk

Associations

- actualRiskOwner : ActualResponsibleResource[0..1] Enables association of an ActualRisk to an actual organizational role that is responsible for executing the actual mitigation.
- affectedActualResource : ActualResource[0..1] Asserts that an ActualRisk is applicable to an ActualResource.

**Risk**

**Package:** Constraints

**isAbstract:** No

**Generalization:** [PropertySet](#), Block

**Extension:** Class

Description

A statement of the impact of an event on Assets. It represents a constraint on an Asset in terms of adverse effects, with an associated measure. The measure is used to capture the extent to which an entity is threatened by a potential circumstance or event. Risk is typically a function of: (i) the adverse impacts that would arise if the circumstance or event occurs; and (ii) the likelihood of occurrence.



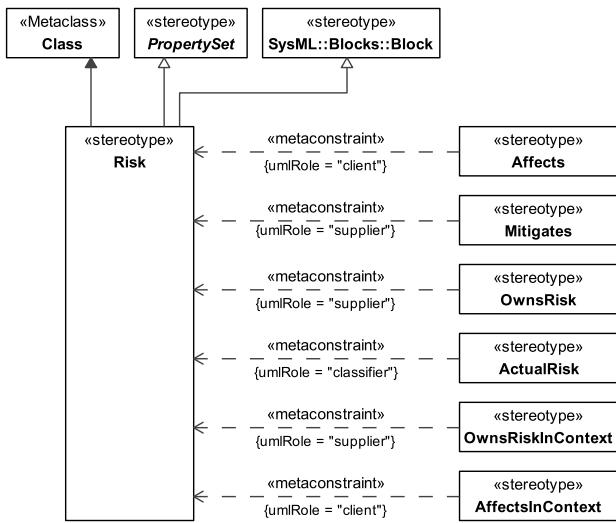


Figure 7.159 - Risk

**SecurityConstraint**

**Package:** Constraints

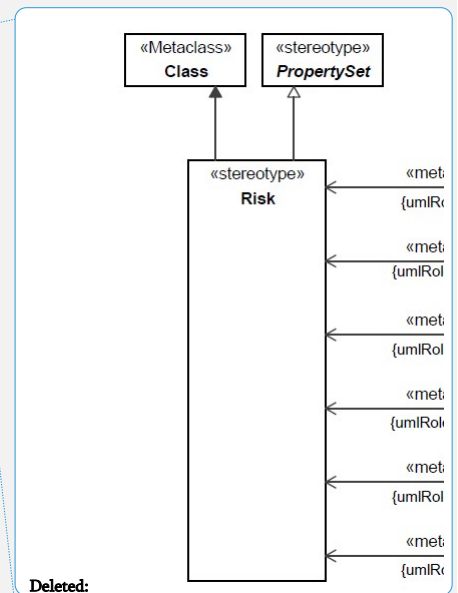
**isAbstract:** No

**Generalization:** [Rule](#)

**Extension:** Constraint

**Description**

A type of rule that captures a formal statement to define security laws, regulations, guidances, and policy.



Deleted:  
 Commented [Yvonne70]: UAF11-55 Replaced figure.

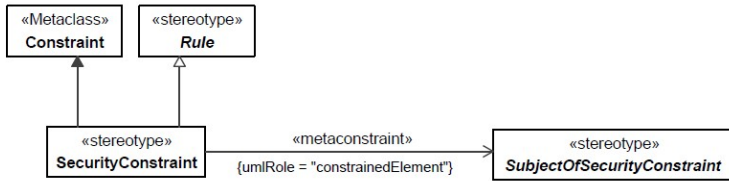


Figure 7.160 - SecurityConstraint

Constraints

[1] Security.constrainedElement Value for the constrainedElement metaproperty must be stereotyped by the specialization of «SubjectOfSecurityConstraint».

**SecurityControl**

**Package:** Constraints

**isAbstract:** No

**Generalization:** Requirement, MeasurableElement

**Extension:** Class

Description

The management, operational, and technical control (i.e., safeguard or countermeasure) prescribed for an information system to protect the confidentiality, integrity, and availability of the system and its information [NIST SP 800-53].

**Commented [AM71]:** [UAF11-56](#) PropertySet in the Generalization list replaced by MeasurableElement

**Deleted:** [PropertySet](#)

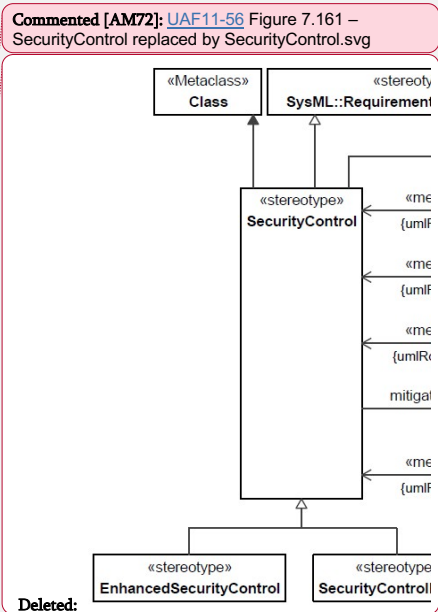
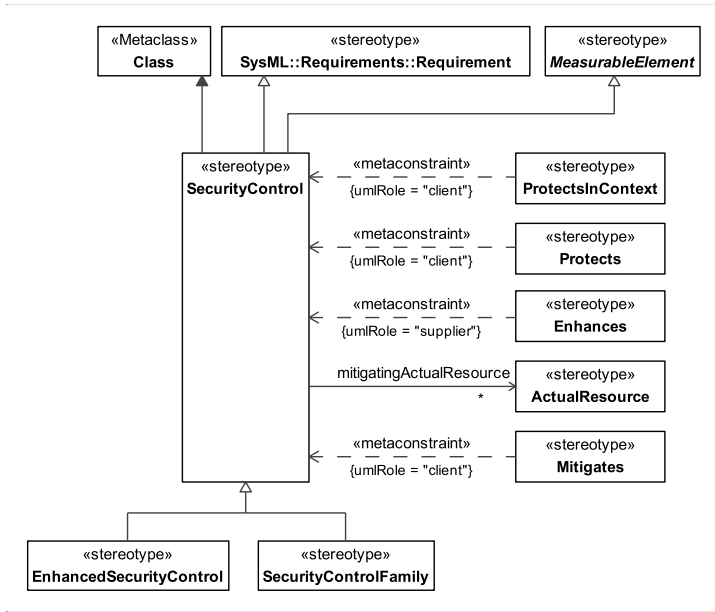


Figure 7.161 - SecurityControl

Associations

mitigatingActualResource : ActualResource[\*] Relates an actual mitigation (an ActualResource for mitigating a Risk) to an ActualRisk.

**SecurityControlFamily**

**Package:** Constraints

**isAbstract:** No

**Generalization:** [SecurityControl](#)

**Extension:** Class

Description

An element that organizes security controls into a family. Each Security Control Family contains security controls related to the general security topic of the family.

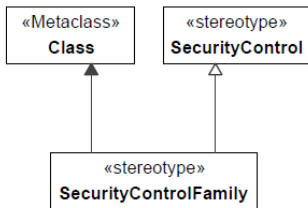


Figure 7.162 - SecurityControlFamily

**SubjectOfSecurityConstraint**

**Package:** Constraints

**isAbstract:** Yes

**Generalization:** [UAFElement](#)

**Extension:** Element

Description

An abstract grouping of elements that can be the subject of a SecurityConstraint.

**Deleted:** Constraints

**Commented [Y73]:** UAF11-57 removed constraint

**Deleted:** [1]  
SecurityControlFamily.annotatedElement → Value for the annotatedElement metaproperty must be stereotyped «SecurityControl» or its specializations.

**Deleted:**

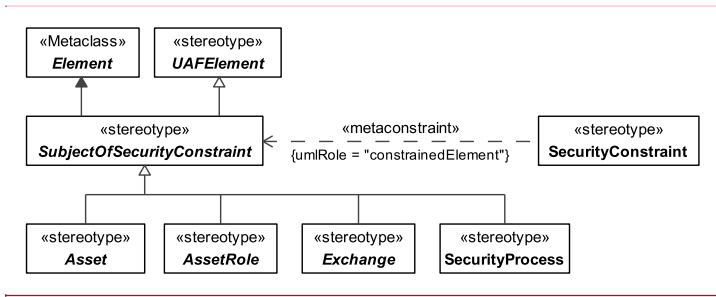
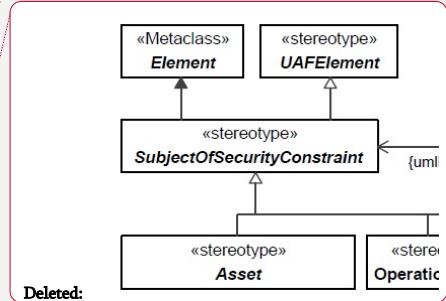


Figure 7.163 - SubjectOfSecurityConstraint

Commented [AM74]: UAF11-60 Figure 7.163 – SubjectOfSecurityConstraint replaced by SubjectOfSecurityConstraint.svg



### 7.1.9.5 UAF::Security::Traceability

Contains the elements that contribute to the Security Traceability Viewpoint.

#### Affects

Package: Traceability

isAbstract: No

Generalization: [MeasurableElement](#)

Extension: Dependency

Description

A dependency that asserts that a Risk is applicable to an Asset.

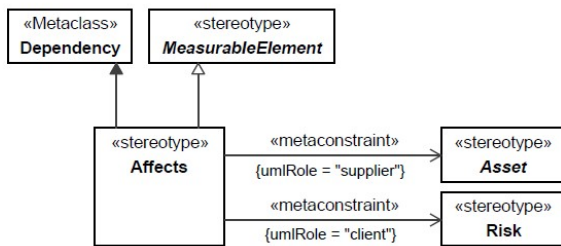


Figure 7.164 - Affects

Constraints

- [1] Affects.client Value for the client metaproperty must be stereotyped «Risk» or its specializations.
- [2] Affects.supplier Value for the supplier metaproperty must be stereotyped «Asset» or its specializations.

**AffectsInContext**

**Package:** Traceability

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Dependency

Description

A dependency that asserts that a Risk is applicable to an AssetRole in the specific context or configuration.

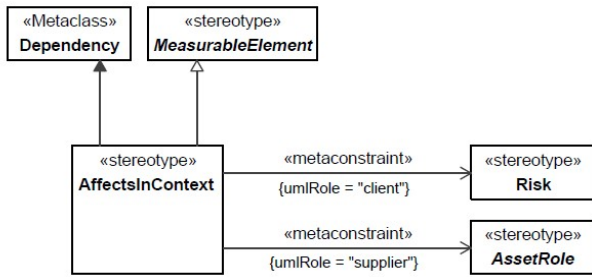


Figure 7.165 - AffectsInContext

Constraints

- [1] AffectsInContext.client Value for the client metaproperty must be stereotyped «Risk» or its specializations.
- [2] AffectsInContext.supplier Value for the supplier metaproperty must be stereotyped «AssetRole» or its specializations.

**Mitigates**

**Package:** Traceability

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Dependency

**Description**

A dependency relating a Security Control to a Risk. Mitigation is established to manage risk and could be represented as an overall strategy or through techniques (mitigation configurations) and procedures (SecurityProcesses).

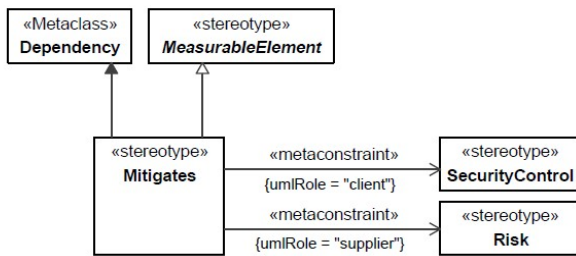


Figure 7.166 - Mitigates

**Constraints**

- [1] Mitigates.client Value for the client metaproperty must be stereotyped «SecurityControl» or its specializations.
- [2] Mitigates.supplier Value for the supplier metaproperty must be stereotyped «Risk» or its specializations.

**OwnsRisk**

**Package:** Traceability

**isAbstract:** No

**Generalization:** [MeasurableElement](#), Allocate

**Extension:** Abstraction

**Description**

An abstraction relating a Risk to an organizational resource that is responsible for executing the risk mitigation.

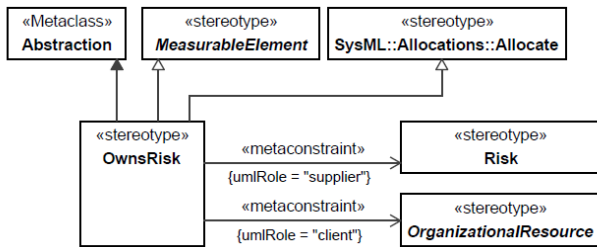


Figure 7.167 - OwnsRisk

Constraints

- [1] OwnsRisk.client Value for the client metaproperty must be stereotyped «OrganizationalResource» or its specializations.
- [2] OwnsRisk.supplier Value for the supplier metaproperty must be stereotyped «Risk» or its specializations.

**OwnsRiskInContext**

**Package:** Traceability

**isAbstract:** No

**Generalization:** [MeasurableElement](#), Allocate

**Extension:** Abstraction

Description

An abstraction relating a Risk to an organizational role that is responsible for executing the risk mitigation in the specific context or configuration.



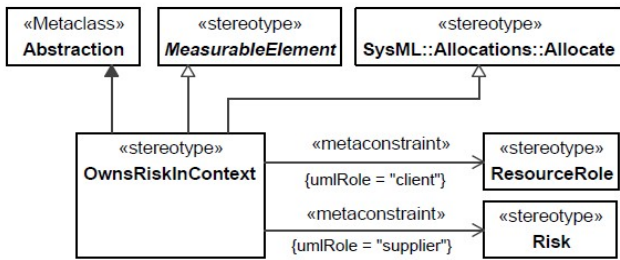


Figure 7.168 - OwnsRiskInContext

Constraints

- [1] **OwnsRiskInContext**.client Value for the client metaproperty must be stereotyped «ResourceRole» or its specializations.
- [2] **OwnsRiskInContext**.supplier Value for the supplier metaproperty must be stereotyped «Risk» or its specializations.

**Commented [AM75]: UAF11-108** Text changed from OwnsProcess to OwnsRiskInContext

**Deleted:** OwnsProcess

**Commented [AM76]: UAF11-108** Text changed from OwnsProcess to OwnsRiskInContext

**Deleted:** OwnsProcess

7.1.10 UAF::Project

Stakeholders: PMs, Project Portfolio Managers, Enterprise Architects.  
 Concerns: project portfolio, projects and project milestones.  
 Definition: describes projects and project milestones, how those projects deliver capabilities, the organizations contributing to the projects and dependencies between projects.

7.1.10.1 UAF::Project::Taxonomy

Contains the elements that contribute to the Project Taxonomy Viewpoint.

**ActualMilestoneKind**

**Package:** Taxonomy

**isAbstract:** No

Description

**Enumeration of the possible kinds of ActualProjectMilestone.** Its enumeration literals are:

- InService - Indicates that the ActualProjectMilestone associated with the ActualMilestoneKind is when the configuration goes into service.
- Deployed - Indicates that the ActualProjectMilestone associated with the ActualMilestoneKind is a configuration deployment milestone.

**Commented [AM77]: UAF11-155** Text changed from "Enumeration of the possible kinds of Actual Measurements" replaced by " Enumeration of the possible kinds of ActualProjectMilestone"

**Deleted:** Enumeration of the possible kinds of ActualMeasurement...

- NoLongerUsed - Indicates that the ActualProjectMilestone associated with the ActualMilestoneKind is when the deployed configuration is no longer used.
- OutOfService - Indicates that the ActualProjectMilestone associated with the ActualMilestoneKind is when the in service configuration goes out of service.
- Other - Indicates that the ActualProjectMilestone associated with the ActualMilestoneKind is not one of the standard ActualMilestoneKinds.

**Project**

**Package:** Taxonomy

**isAbstract:** No

**Generalization:** [OrganizationalResource](#)

**Extension:** Class

**Description**

An element that describes types of time-limited endeavours that are required to meet one or more Capability needs.

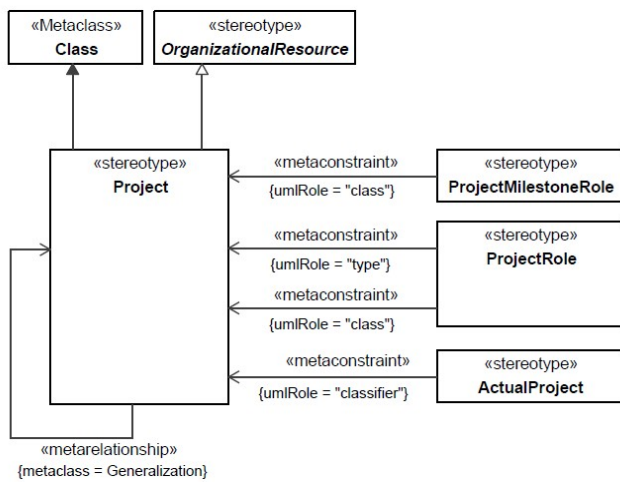


Figure 7.169 - Project

**ProjectKind**

**Package:** Taxonomy

**isAbstract:** No

Description

Enumeration of the possible kinds of project applicable to an ActualProject. Its enumeration literals are:

- Programme - Indicates that the ActualProject associated with the ProjectKind is an undertaking that is a temporary, flexible organization created to coordinate, direct, and oversee the implementation of a set of related Projects and Tasks in order to deliver outcomes and benefits related to the organization's strategic objectives. A programme is likely to have a lifespan of several years. During a programme lifecycle, projects are initiated, executed, and closed. Programmes provide an umbrella under which these projects can be coordinated. The programme integrates the projects so that it can deliver an outcome greater than the sum of its parts.
- Portfolio - Indicates that the ActualProject associated with the ProjectKind is an undertaking comprised of the Projects and Programmes that are the totality of an organization's investment (or segment thereof) in the changes required to achieve its strategic objectives.
- Project - Indicates that the ActualProject associated with the ProjectKind is an undertaking that is a time-limited endeavor to create a specific set of products or services.
- PersonnelDevelopment - Indicates that the ActualProject associated with the ProjectKind is an undertaking that relates to the training and enablement of personnel to enable them help achieve the organizations objectives.

**ProjectMilestone**

**Package:** Taxonomy

**isAbstract:** No

**Generalization:** [PropertySet](#), Block

**Extension:** Class

Description

A type of event in a Project by which progress is measured.

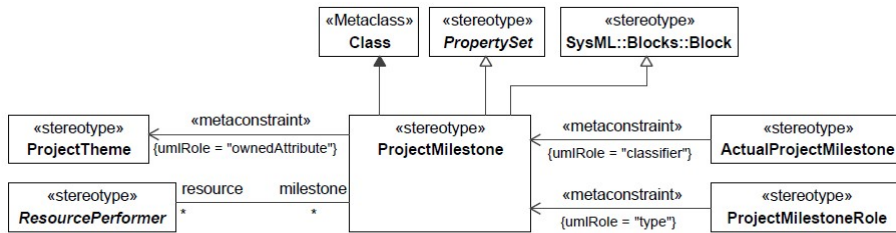


Figure 7.170 - ProjectMilestone

Associations

resource : ResourcePerformer[\*] Relates a ProjectMilestone to the Resources that can be affected by the milestone. It is used to describe aspects of the lifecycle of a Resource.

Constraints

- [1] ProjectMilestone.ownedAttribute All of the «ProjectThemes», owned by a «ProjectMilestone», must be typed by the same «StatusIndicators» or its specializations.

7.1.10.2 UAF::Project::Structure

Contains the elements that contribute to the Project Structure Viewpoint.

**ProjectMilestoneRole**

Package: Structure

isAbstract: No

Generalization: [MeasurableElement](#)

Extension: Property

Description

The role played by a ProjectMilestone in the context of a Project.

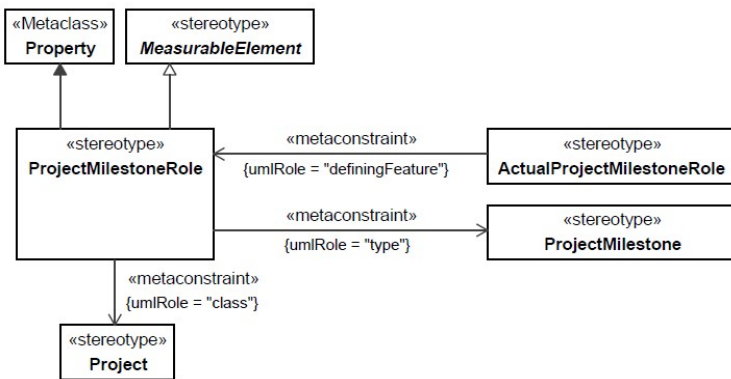


Figure 7.171 - ProjectMilestoneRole

Constraints

- [1] ProjectMilestoneRole.class Value for the class metaproperty must be stereotyped «Project» or its specializations.
- [2] ProjectMilestoneRole.type Value for the type metaproperty must be stereotyped «ProjectMilestone» or its specializations.

## ProjectRole

**Package:** Structure

**isAbstract:** No

**Generalization:** [ResourceRole](#)

**Extension:** Property

**Description**

Usage of a Project in the context of another Project. Creates a whole-part relationship.

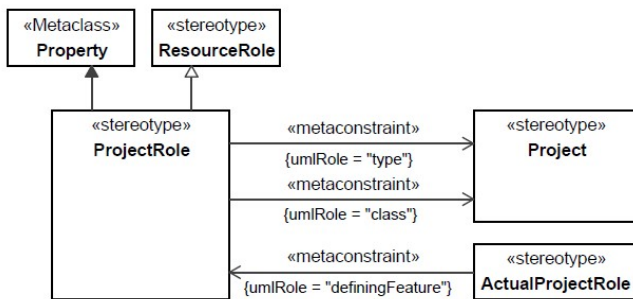


Figure 7.172 - ProjectRole

**Constraints**

- [1] ProjectRole.class Value for the class metaproperty must be stereotyped «Project» or its specializations.
- [2] ProjectRole.type Value for the type metaproperty must be stereotyped «Project» or its specializations.

## ProjectStatus Package:

Structure **isAbstract:** No

**Generalization:** [UAFElement](#)

**Extension:** Slot

**Description**

The status (i.e., level of progress) of a ProjectTheme for an ActualProject at the time of the ActualProjectMilestone.

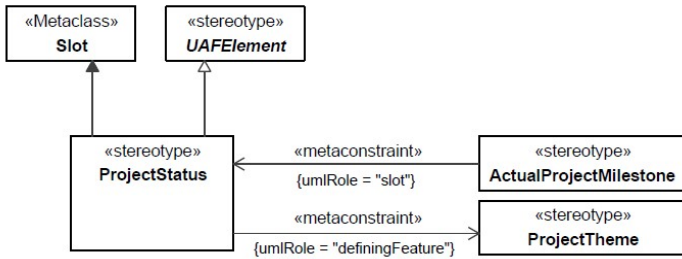


Figure 7.173 - ProjectStatus

Constraints

- [1] ProjectStatus.definingFeature Value for the DefiningFeature metaproperty must be stereotyped «ProjectTheme» or its specializations.

**ProjectTheme**

**Package:** Structure

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Property

Description

A property of a ProjectMilestone that captures an aspect by which the progress of ActualProjects may be measured.

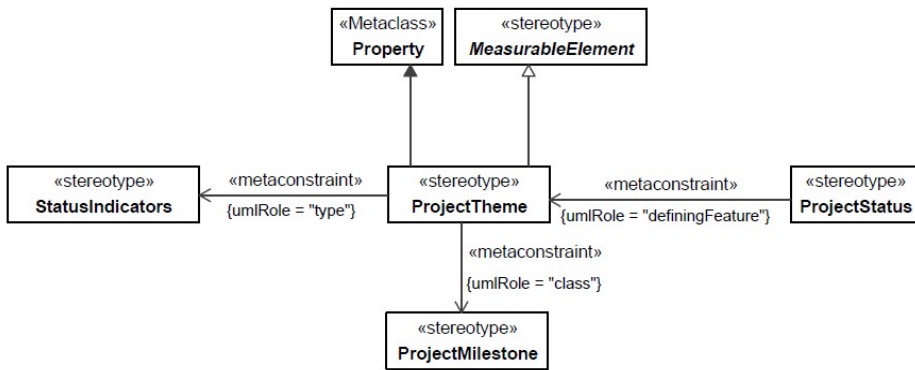


Figure 7.174 - ProjectTheme

Constraints

- [1] ProjectTheme.class Value for the class metaproperty must be stereotyped «ProjectMilestone» or its specializations.
- [2] ProjectTheme.type Value for the type metaproperty must be stereotyped «StatusIndicators» or its specializations.

**StatusIndicators**

**Package:** Structure

**isAbstract:** No

**Generalization:** [MeasurableElement](#), ValueType

**Extension:** Enumeration

Description

An enumerated type that specifies a status for a ProjectTheme.

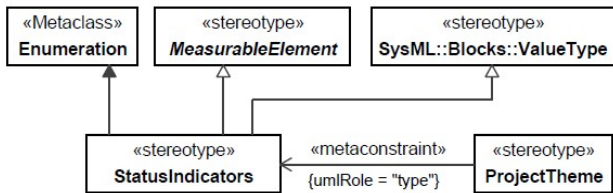


Figure 7.175 - StatusIndicators

### 7.1.10.3 UAF::Project::Connectivity

Contains the elements that contribute to the Project Connectivity Viewpoint.

#### MilestoneDependency

**Package:** Connectivity

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Dependency

**Description**

A dependency relationship between two ActualProjectMilestones that denotes one ActualProjectMilestone follows from another.

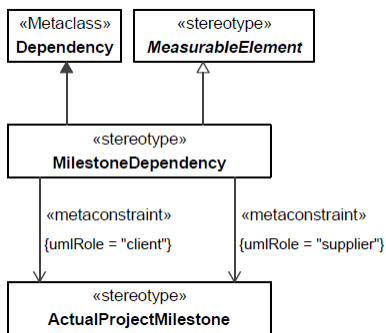


Figure 7.176 - MilestoneDependency



Constraints

- [1] MilestoneDependency.client Value for the client metaproperty must be stereotyped «ActualProjectMilestone» or its specializations.
- [2] MilestoneSequence.supplier Value for the supplier metaproperty must be stereotyped «ActualProjectMilestone» or its specializations.

**ProjectSequence**

**Package:** Connectivity

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Dependency

Description

A dependency relationship between two ActualProjects that denotes one ActualProject cannot start before the previous ActualProject is finished.

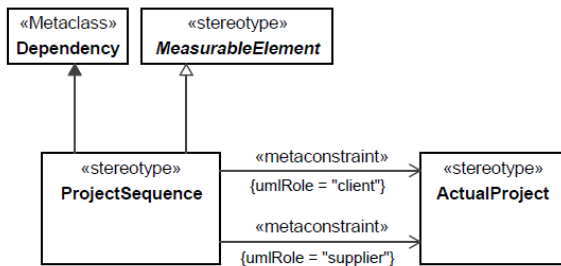


Figure 7.177 - ProjectSequence

Constraints

- [1] ProjectSequence.client Value for the client metaproperty must be stereotyped «ActualProject» or its specializations.
- [2] ProjectSequence.supplier Value for the supplier metaproperty must be stereotyped «ActualProject» or its specializations.

#### 7.1.10.4 UAF::Project::Processes

Contains the elements that contribute to the Project Processes Viewpoint.

##### ProjectActivity

**Package:** Processes

**isAbstract:** No

**Generalization:** [Function](#)

**Extension:** Activity

Description

An activity carried out during a project.

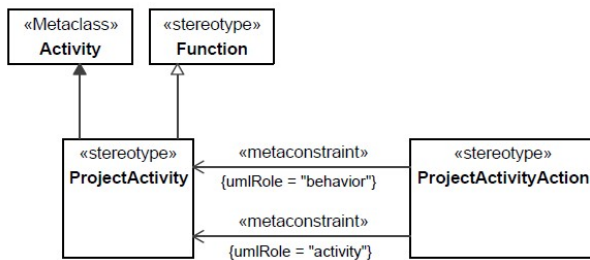


Figure 7.178 - ProjectActivity

##### ProjectActivityAction

**Package:** Processes

**isAbstract:** No

**Generalization:** [FunctionAction](#)

**Extension:** CallBehaviorAction, Activity

Description

The ProjectActivityAction is defined as a call behavior action that invokes the activity that needs to be preformed.

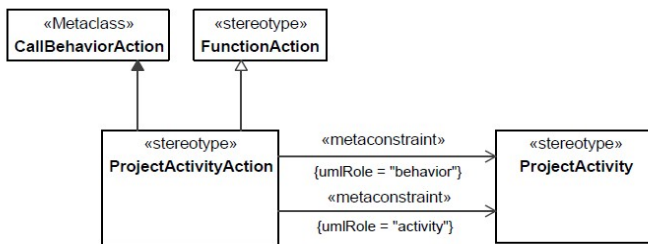


Figure 7.179 - ProjectActivityAction

Constraints

- [1] FunctionAction.behavior Value for the behavior metaproperty must be stereotyped «ProjectActivity» or its specializations.
- [2] ProjectActivityAction.activity Value for the activity metaproperty must be stereotyped «ProjectActivity» or its specializations.

7.1.10.5 UAF::Project::Roadmap

Contains the elements that contribute to the Project Roadmap Viewpoint.

**ActualProject**

**Package:** Roadmap

**isAbstract:** No

**Generalization:** [ActualOrganizationalResource](#), [Achiever](#)

**Extension:** InstanceSpecification

Description

A time-limited endeavor to provide a specific set of ActualResources that meet specific Capability needs.

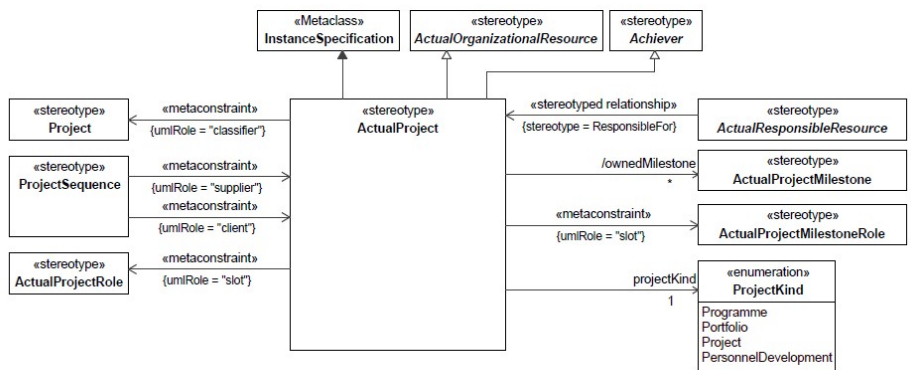


Figure 7.180 - ActualProject

Associations

ownedMilestone : ActualProjectMilestone[\*]      Relates the ActualProjectMilestones to the relevant ActualProject.

projectKind : ProjectKind[1]      Enumerated value describing the kind of ActualProject.

Constraints

[1] ActualProject.classifier      Value for the classifier metaproperty must be stereotyped «Project» or its specializations.

[2] ActualProject.slot      Value for the slot metaproperty must be stereotyped «ActualProjectRole», «ActualProjectMilestoneRole», or their specializations.

**ActualProjectMilestone**

**Package:** Roadmap

**isAbstract:** No

**Generalization:** [ActualPropertySet](#)

**Extension:** InstanceSpecification

Description

An event with a start date in a ActualProject from which progress is measured.

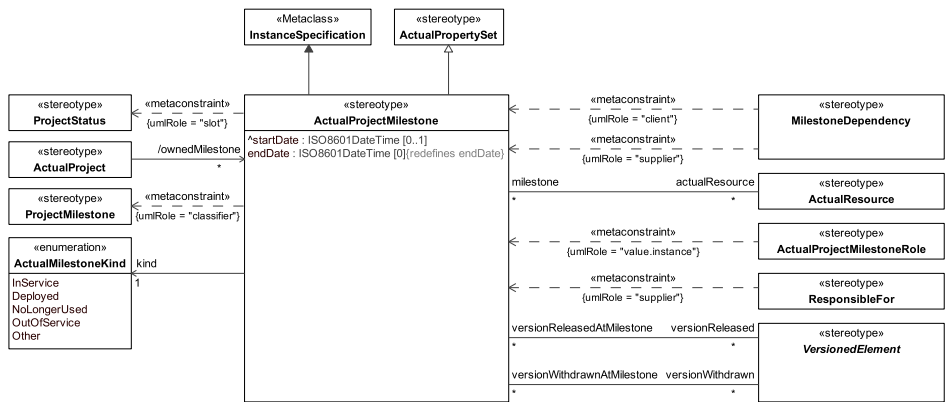


Figure 7.181 - ActualProjectMilestone

Attributes

endDate : ISO8601DateTime[0]      End time for this ActualProjectMilestone.

Associations

actualResource : ActualResource[\*]      Relates an ActualProjectMilestone to the ActualResources that are affected by the milestone. It is used to describe aspects of the lifecycle of an ActualResource.

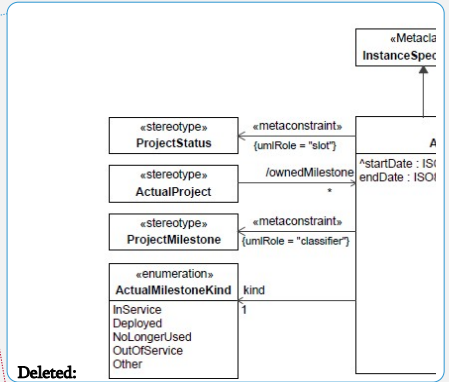
kind : ActualMilestoneKind[1]      Enumerated value describing the kind of ActualProjectMilestone.

versionReleased : VersionedElement[\*]

versionWithdrawn :  
VersionedElement[\*]

Constraints

[1] ActualProjectMilestone.classifier      Value for the classifier metaproperty must be stereotyped «ProjectMilestone» or its specializations.



**Deleted:**  
Commented [AM78]: UAF11-154, UAF11-63 Figure 7.181 – ActualProjectMilestone replaced by ActualProjectMilestone.svg

## ActualProjectMilestoneRole

**Package:** Roadmap

**isAbstract:** No

**Generalization:** [ActualState](#)

**Extension:** Slot

Description

An ActualProjectMilestone that is applied to a ProjectMilestoneRole.

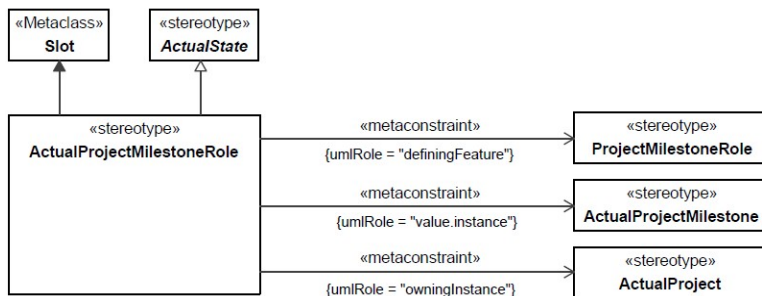


Figure 7.182 - ActualProjectMilestone

Constraints

- |  |  |
|--|--|
| [1] ActualProjectMilestoneRole.definingFeature | Value for the definingFeature metaproperty has to be stereotyped «ProjectMilestoneRole» or its specializations.  |
| [2] ActualProjectMilestoneRole.owningInstance  | Value for the owningInstance metaproperty has to be stereotyped «ActualProject» or its specializations.          |
| [3] ActualProjectMilestoneRole.value.instance  | Value for the value.instance metaproperty has to be stereotyped «ActualProjectMilestone» or its specializations. |

## ActualProjectRole

**Package:** Roadmap

**isAbstract:** No

**Generalization:** [ActualState](#)

**Extension:** Slot

#### Description

An ActualProject that is applied to a ProjectRole.

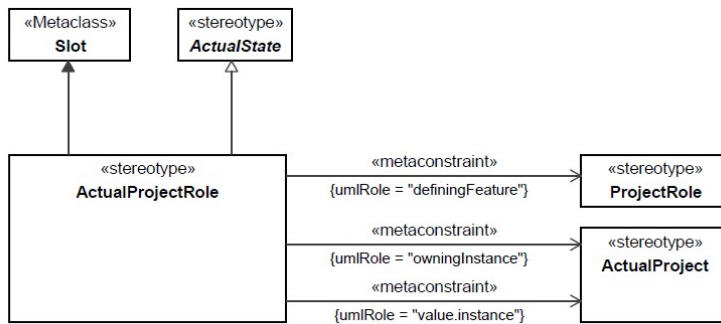


Figure 7.183 - ActualProjectRole

#### Constraints

- |                                       |   |
|---------------------------------------|---|
| [1] ActualProjectRole.definingFeature | Value for the definingFeature metaproperty has to be stereotyped «ProjectRole» or its specializations.  |
| [2] ActualProjectRole.owningInstance  | Value for the owningInstance metaproperty has to be stereotyped «ActualProject» or its specializations. |
| [3] ActualProjectRole.value.instance  | Value for the value.instance metaproperty has to be stereotyped «ActualProject» or its specializations. |

### 7.1.11 UAF::Standards

Stakeholders: Solution Providers, Systems Engineers, Software Engineers, Systems Architects, Business Architects.

Concerns: technical and non-technical Standards applicable to the architecture.

Definition: shows the technical, operational, and business Standards applicable to the architecture. Defines the underlying current and expected Standards.

#### 7.1.11.1 UAF::Standards::Taxonomy

Contains the elements that contribute to the Standards Taxonomy Viewpoint.

##### Protocol

**Package:** Taxonomy

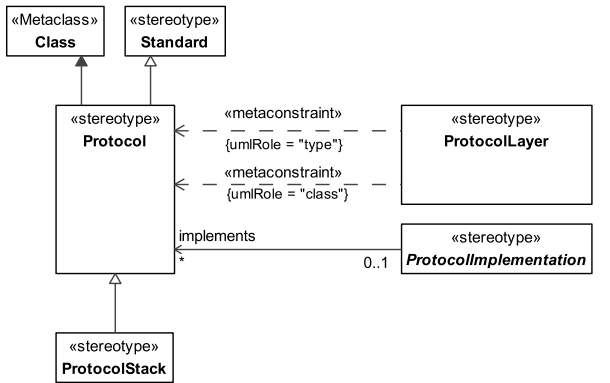
**isAbstract:** No

**Generalization:** [Standard](#)

**Extension:** Class

Description

A Standard for communication over a network. Protocols may be composite, represented as a ProtocolStack made up of ProtocolLayers.



Commented [AM79]: UAF11-112 Figure 7.184 – Protocol replaced by Protocol.svg

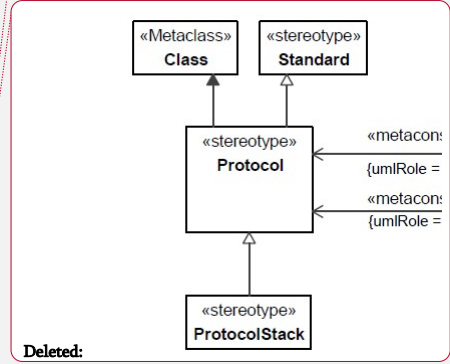


Figure 7.184 - Protocol

**ProtocolStack**

**Package:** Taxonomy

**isAbstract:** No

**Generalization:** [Protocol](#)

**Extension:** Class

Description

A sub-type of Protocol that contains the ProtocolLayers, defining a complete stack.



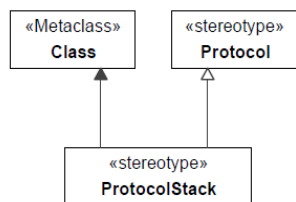


Figure 7.185 - ProtocolStack

**Standard**

**Package:** Taxonomy

**isAbstract:** No

**Generalization:** [SubjectOfForecast](#), [PropertySet](#), [Block](#)

**Extension:** Class

**Description**

A ratified and peer-reviewed specification that is used to guide or constrain the architecture. A Standard may be applied to any element in the architecture.

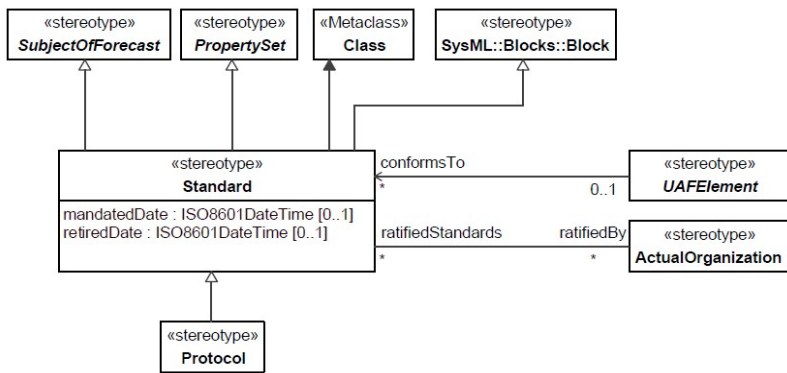


Figure 7.186 - Standard

Attributes

- mandatedDate : ISO8601DateTime[0..1]      The date when this version of the Standard was published.
- retiredDate : ISO8601DateTime[0..1]      The date when this version of the Standard was retired.

Associations

- ratifiedBy : ActualOrganization[\*]      Relates a Standard to the ActualOrganization that ratified the Standard.

### 7.1.11.2 UAF::Standards::Structure

Contains the elements that contribute to the Standards Structure Viewpoint.

#### ProtocolLayer

Package: Structure

isAbstract: No

Generalization: [MeasurableElement](#)

Extension: Property

#### Description

Usage of a Protocol in the context of another Protocol. Creates a whole-part relationship.

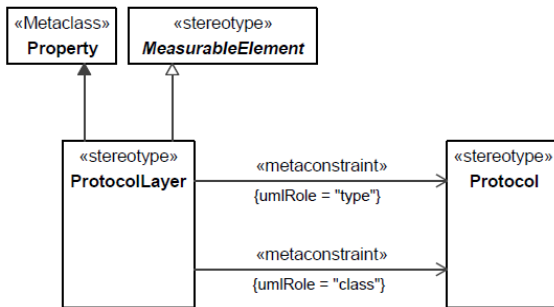


Figure 7.187 - ProtocolLayer

Constraints

[1] ProtocolLayer.class Value for the class metaproperty must be stereotyped «Protocol» or its specializations.

[2] ProtocolLayer.type Value for the type metaproperty must be stereotyped «Protocol» or its specializations.

### 7.1.12 UAF::Actual Resources

Stakeholders: Solution Providers, Systems Engineers, Business Architects, Human Resources.

Concerns: the analysis.- e.g., evaluation of different alternatives, what-if, trade-offs, V&V on the actual resource configurations.

Definition: illustrates the expected or achieved actual resource configurations and actual relationships between them.

#### 7.1.12.1 UAF::Actual Resources::Taxonomy

Contains the elements that contribute to the Actual Resources Taxonomy Viewpoint.

##### **ActualOrganization**

**Package:** Taxonomy

**isAbstract:** No

**Generalization:** [ActualResponsibleResource](#)

**Extension:** InstanceSpecification

Description

An actual formal or informal organizational unit, e.g., "Driving and Vehicle Licensing Agency," "UAF team Alpha."

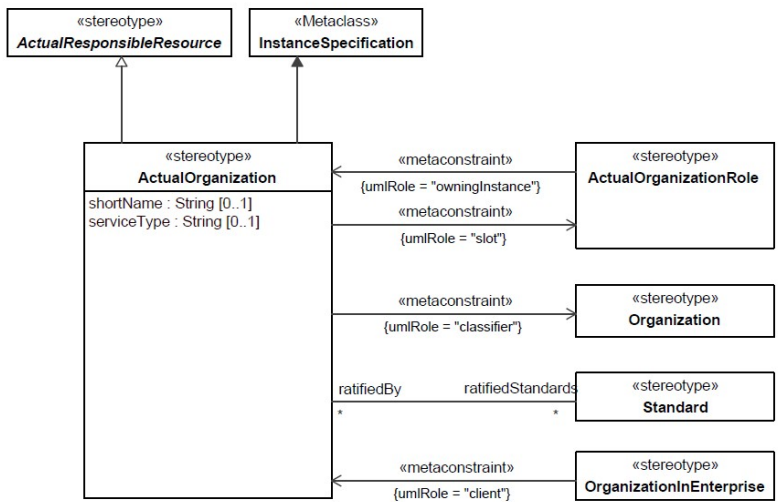


Figure 7.188 - ActualOrganization

Attributes

- serviceType : String[0..1]      Service office code or symbol
- shortName : String[0..1]      String providing a simplified means of identifying an ActualOrganization, i.e., SoftWareGroup could use SWG as the shortName.

Associations

- ratifiedStandards : Standard[\*]      Standards that were ratified by this ActualOrganization.

Constraints

- [1] ActualOrganization.classifier      Classifier metaproperty value must be stereotyped «Organization» or its specializations.
- [2] ActualOrganization.slot      Slot metaproperty value must be stereotyped «ActualOrganizationRole» or its specializations.

**ActualOrganizationalResource**

**Package:** Taxonomy

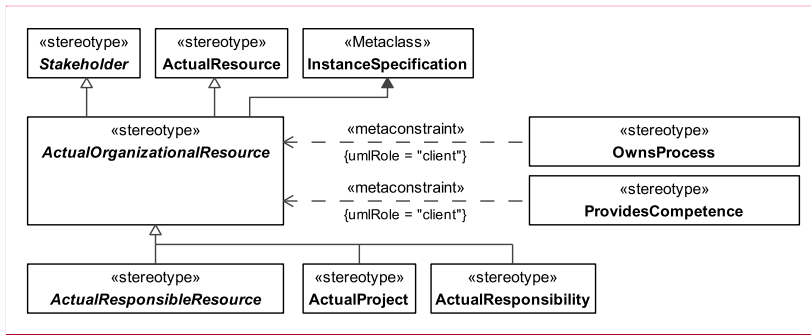
**isAbstract:** Yes

**Generalization:** [Stakeholder](#), [ActualResource](#)

**Extension:** InstanceSpecification

Description

Abstract element for an ActualOrganization, ActualPerson, or ActualPost.



**Commented [AM80]:** 1. [UAF11-156](#). Diagram replaced by ActualOrganizationalResource.svg

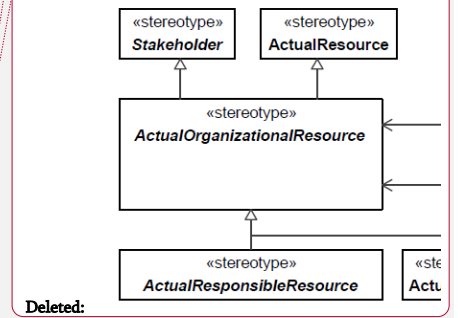


Figure 7.189 - ActualOrganizationalResource

### ActualPerson

**Package:** Taxonomy

**isAbstract:** No

**Generalization:** [ActualResponsibleResource](#)

**Extension:** InstanceSpecification

Description

An individual human being.

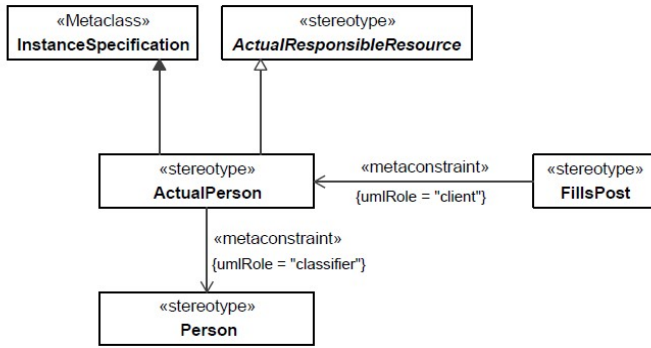


Figure 7.190 - ActualPerson

Constraints

[1] ActualPerson.classifier Value for the classifier metaproperty has to be stereotyped «Person» or its specializations.

**ActualPost**

**Package:** Taxonomy

**isAbstract:** No

**Generalization:** [ActualResponsibleResource](#)

**Extension:** InstanceSpecification

Description

An actual, specific post, an instance of a Post "type" - e.g., "President of the United States of America" where the Post would be president.

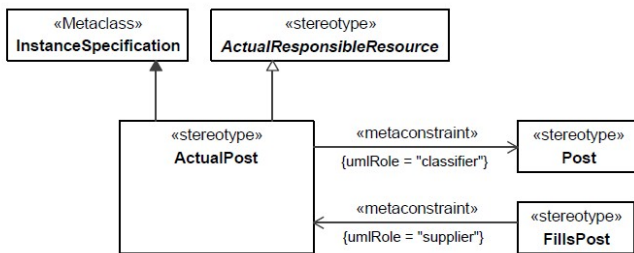


Figure 7.191 - ActualPost

Constraints

[1] ActualPost.classifier Classifier metaproperty value must be stereotyped «Post» or its specializations.

**ActualResource**

**Package:** Taxonomy

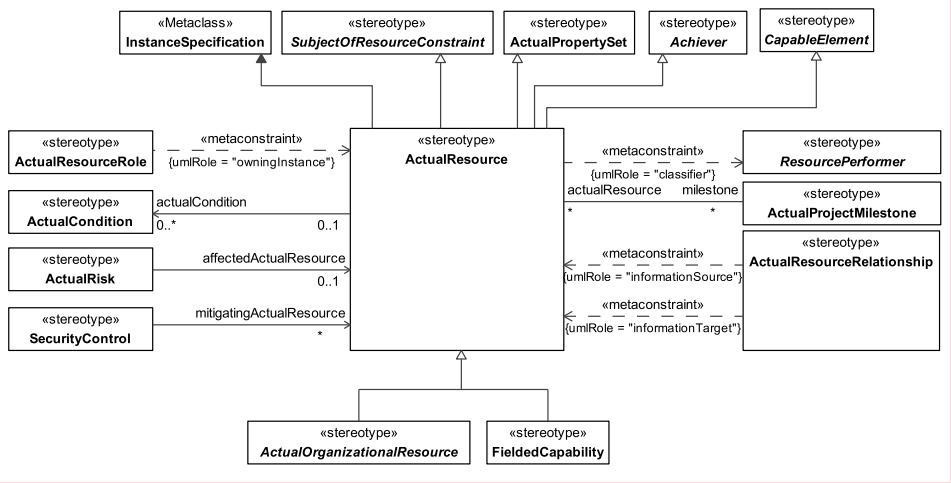
**isAbstract:** No

**Generalization:** [ActualPropertySet](#), [SubjectOfResourceConstraint](#), [Achiever](#)

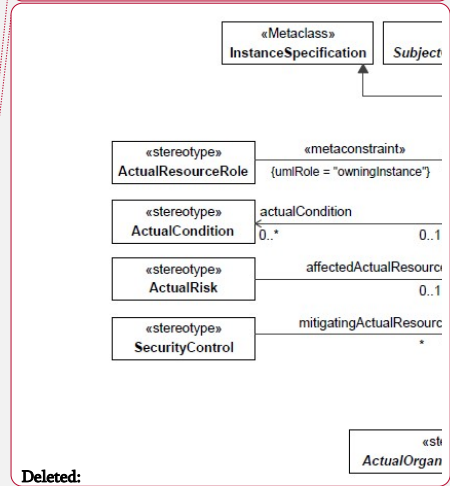
**Extension:** InstanceSpecification

Description

Role in an Organisation, where the role carries the authority to undertake a function - though the ActualOrganizationalResource given the role has the responsibility.



Commented [AM81]: UAF11-128 Figure 7.192 – ActualResource replaced by ActualResource.svg



Deleted:

Figure 7.192 - ActualResource

Associations

- actualCondition : ActualCondition[0..\*] Relates the ActualResource to the ActualStates of an environment or location describing its situation.
- milestone : ActualProjectMilestone[\*] Relates an ActualResource to the ActualProjectMilestones. It is used to describe aspects of the lifecycle of an ActualResource.

Constraints

- [1] ActualResource.classifier Classifier metaproperty value must be stereotyped by a specialization of «ResourcePerformer».

**ActualResponsibility**

Package: Taxonomy

isAbstract: No

Generalization: [ActualOrganizationalResource](#)

Extension: InstanceSpecification



Description

The duty required of a Person or Organization.

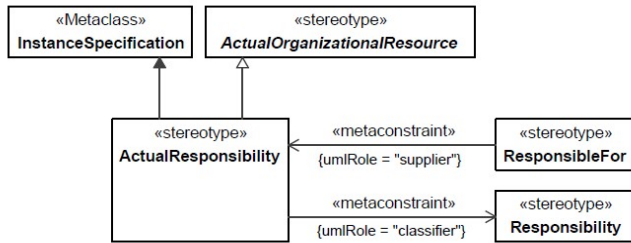


Figure 7.193 - ActualResponsibility

Constraints

[1] ActualResponsibility.classifier Classifier metaproperty value must be stereotyped «Responsibility» or its specializations.

**ActualResponsibleResource**

**Package:** Taxonomy

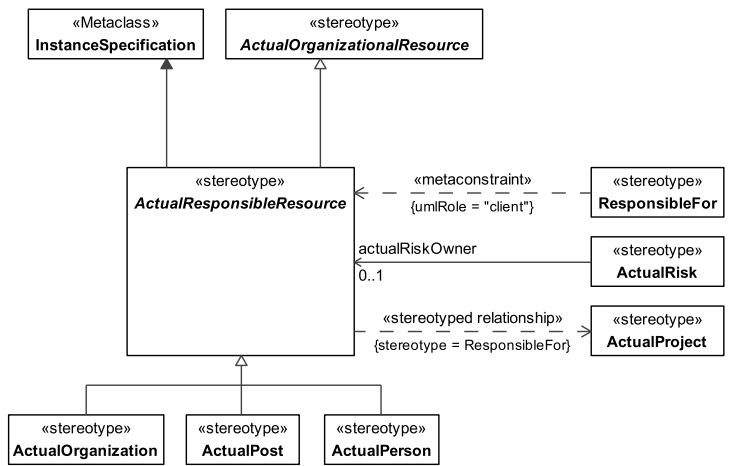
**isAbstract:** Yes

**Generalization:** [ActualOrganizationalResource](#)

**Extension:** InstanceSpecification

Description

An abstract grouping of responsible OrganizationalResources.



Commented [AM82]: UAF11-157 Diagram replaced by ActualResponsibleResource.svg →

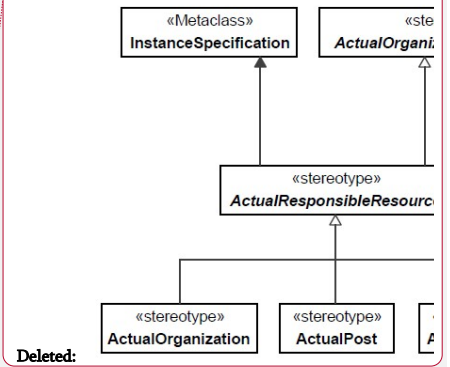


Figure 7.194 - ActualResponsibleResource

**FieldedCapability**

Package: Taxonomy

isAbstract: No

Generalization: [ActualResource](#)

Extension: InstanceSpecification

Description

An actual, fully-realized capability. A FieldedCapability is typed by a CapabilityConfiguration.

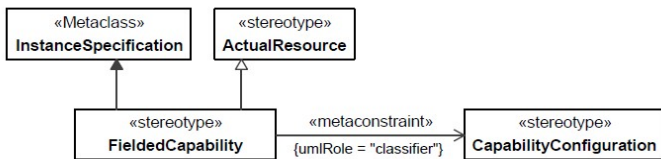


Figure 7.195 - FieldedCapability

Constraints

- [1] FieldedCapability.classifier Value for the classifier metaproperty must be stereotyped «CapabilityConfiguration» or its specializations.

7.1.12.2 UAF::Actual Resources::Structure

Contains the elements that contribute to the Actual Resources Structure Viewpoint.

**ActualOrganizationRole**

Package: Structure

isAbstract: No

Generalization: [ActualResourceRole](#)

Extension: Slot

Description

An ActualOrganizationalResource that is applied to a ResourceRole.

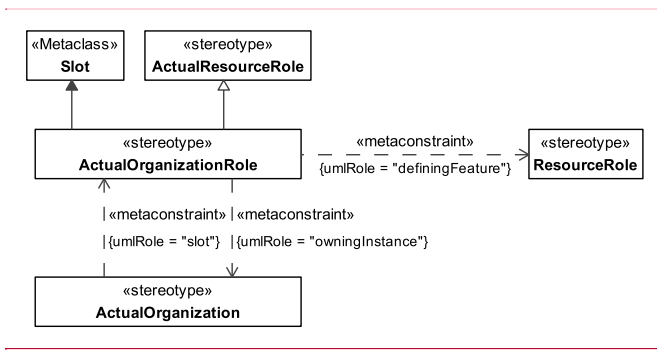
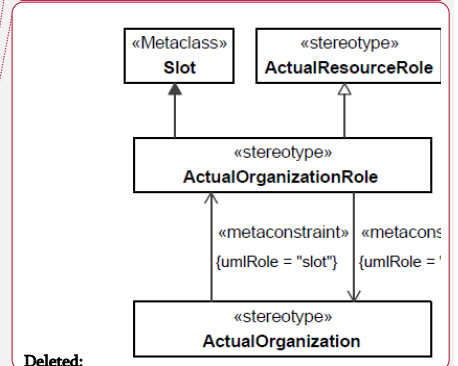


Figure 7.196 - ActualOrganizationRole

Constraints

- [1] ActualOrganizationRole.owningInstance Value for owningInstance metaproperty has to be stereotyped «ActualOrganization» or its specializations.

Commented [AM83]: [UAF11-157](#) Diagram replaced by ActualOrganizationRole.svg



## ActualResourceRole

**Package:** Structure

**isAbstract:** No

**Generalization:** [UAFElement](#)

**Extension:** Slot

**Description**

An instance of a ResourcePerformer.

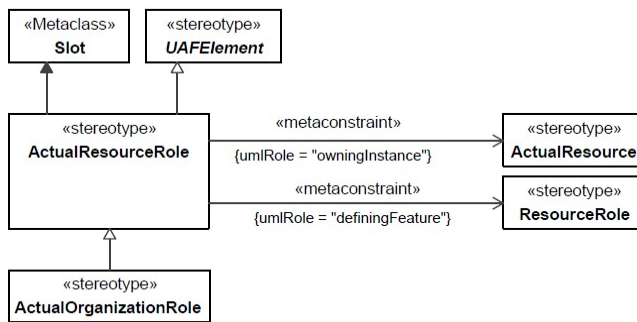


Figure 7.197 - ActualResourceRole

**Constraints**

- [1] ActualResourceRole.definingFeature Value for definingFeature metaproperty has to be stereotyped «ResourceRole» or its specializations.
- [2] ActualResourceRole.owningInstance Value for owningInstance metaproperty has to be stereotyped «ActualResource» or its specializations.

### 7.1.12.3 UAF::Actual Resources::Connectivity

Contains the elements that contribute to the Actual Resources Connectivity Viewpoint.

#### ActualResourceRelationship

**Package:** Connectivity

**isAbstract:** No

**Generalization:** [UAFElement](#), ItemFlow

**Extension:** InformationFlow

Description

An abstract element that details the ActualOrganizationalResources that are able to carry out an ActualResponsibility.

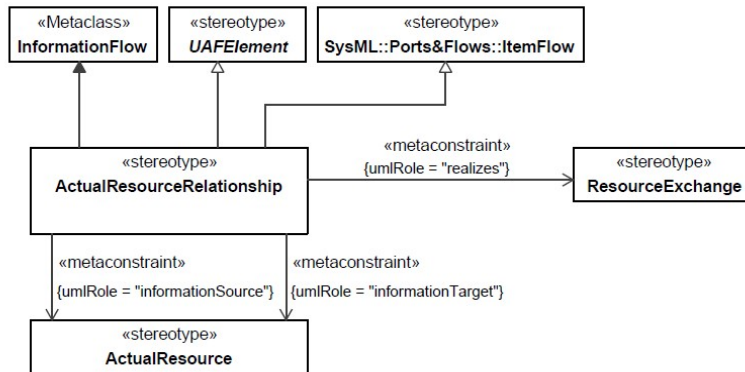


Figure 7.198 - ActualResourceRelationship

Constraints

- [1] ActualResourceRelationship.informationSource Value for informationSource metaproperty must be stereotyped «ActualResource» or its specializations.
- [2] ActualResourceRelationship.informationTarget Value for informationTarget metaproperty must be stereotyped «ActualResource» or its specializations.
- [3] ActualResourceRelationship.realizes Value for realizes metaproperty must be stereotyped «ResourceExchange» or its specializations.

**FillsPost**

**Package:** Connectivity

**isAbstract:** No

**Generalization:** [MeasurableElement](#), Allocate

**Extension:** Abstraction

Description

A dependency relationship that asserts that an ActualPerson fills an ActualPost.

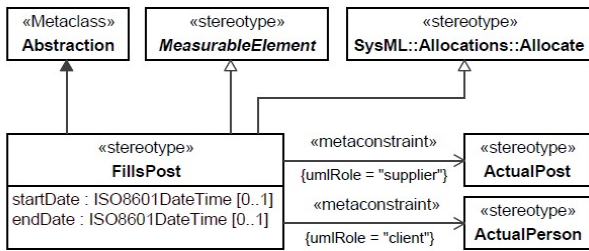


Figure 7.199 - FillsPost

Attributes

- endDate : ISO8601DateTime[0..1] End date of an ActualPerson filling an ActualPost.
- startDate : ISO8601DateTime[0..1] Start date of an ActualPerson filling an ActualPost.

Constraints

- [1] FillsPost.client Value for the client metaproperty must be stereotyped by «ActualPerson» or its specializations.
- [2] FillsPost.supplier Value for the supplier metaproperty must be stereotyped by «ActualPost» or its specializations.

7.1.12.4 UAF::Actual Resources::Constraints

Contains the elements that contribute to the Actual Resources Constraints Viewpoint.

ActualService

Package: Constraints

isAbstract: No

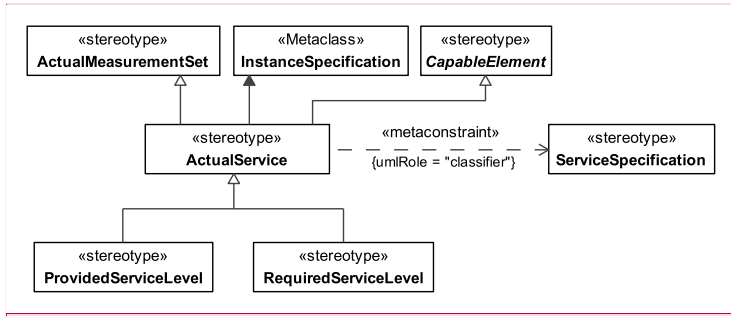
Generalization: [ActualMeasurementSet](#), [CapableElement](#)

Extension: InstanceSpecification

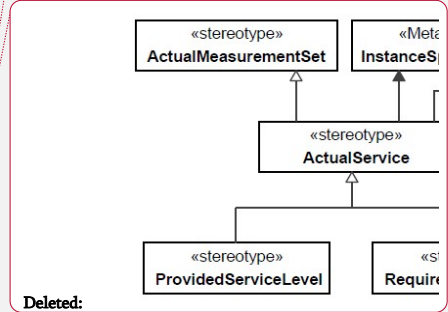
Description

An instance of a ServiceSpecification.

Commented [AM84]: [UAF11-125](#), [UAF11-122](#)  
 ActualPropertySet replaced by CapableElement in the list of Generalizations  
 Deleted: [ActualPropertySet](#)



Commented [AM85]: Figure 7.200 – ActualService replaced by ActualService.svg.



Deleted:

Figure 7.200 - ActualService

Constraints

- [1] ActualService.classifier Value for the classifier metaproperty must be stereotyped by «ServiceSpecification» or its specializations.

**ProvidedServiceLevel**

Package: Constraints

isAbstract: No

Generalization: [ActualService](#)

Extension: InstanceSpecification

Description

A sub type of ActualService that details a specific service level delivered by the provider.

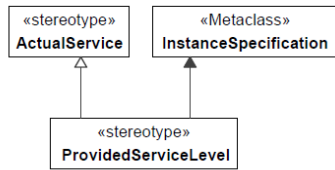


Figure 7.201 - ProvidedServiceLevel

**ProvidesCompetence**

**Package:** Constraints

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Dependency

Description

A dependency relationship that asserts that an ActualOrganizationalResource provides a specific set of Competencies.

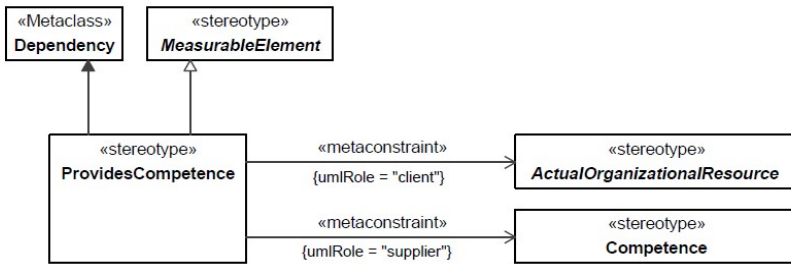


Figure 7.202 - ProvidesCompetence



#### Constraints

- [1] ProvidesCompetence.client Value for the client metaproperty must be stereotyped by a specialization of «ActualOrganizationalResource».
- [2] ProvidesCompetence.supplier Value for the supplier metaproperty must be stereotyped «Competence» or its specializations.

#### RequiredServiceLevel

**Package:** Constraints

**isAbstract:** No

**Generalization:** [ActualService](#)

**Extension:** InstanceSpecification

#### Description

A sub type of ActualService that details a specific service level required of the provider.

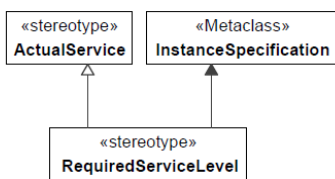


Figure 7.203 - RequiredServiceLevel

#### 7.1.12.5 UAF::Actual Resources::Traceability

Contains the elements that contribute to the Actual Resources Traceability Viewpoint.

#### OwnsProcess

**Package:** Traceability

**isAbstract:** No

**Generalization:** [MeasurableElement](#), Allocate

**Extension:** Abstraction

#### Description

A dependency relationship denoting that an ActualOrganizationResource owns an OperationalActivity.

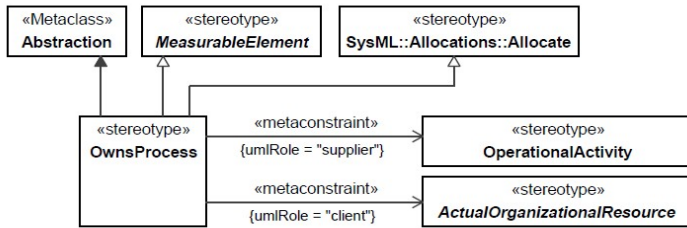


Figure 7.204 - OwnsProcess

Constraints

- [1] OwnsProcess.client Value for the client metaproperty must be stereotyped «ActualOrganizationalResource» or its specializations.
- [2] OwnsProcess.supplier Value for the supplier metaproperty must be stereotyped «OperationalActivity» or its specializations.

**7.1.13 UAF::Summary and Overview**

Stakeholders: Executives, PMs, Enterprise Architects.

Concerns: executive-level summary information in a consistent form.

Definition: provides executive-level summary information in a consistent form that allows quick reference and comparison between architectural descriptions. Includes assumptions, constraints, and limitations that may affect high-level decisions relating to an architecture-based work programme.

**ArchitecturalDescription**

**Package:** Summary and Overview

**isAbstract:** No

**Generalization:** [MeasurableElement](#)

**Extension:** Package

Description

An Architecture Description is a work product used to express the Architecture of some System Of Interest. It provides executive-level summary information about the architecture description in a consistent form to allow quick reference and comparison between architecture descriptions. It includes assumptions, constraints, and limitations that may affect high-level decisions relating to an architecture-based work program.

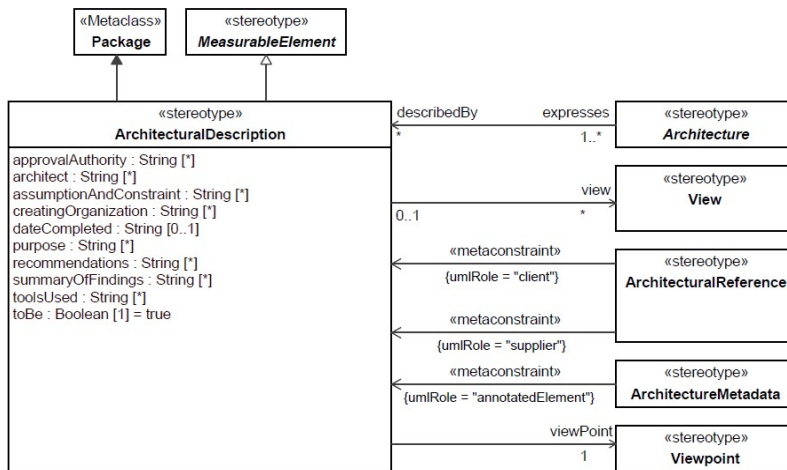


Figure 7.205 - ArchitecturalDescription

Attributes

approvalAuthority : String[*]	Someone or something that has the authority to approve the ArchitecturalDescription.
architect : String[*]	Someone responsible for the creation of ArchitecturalDescription.
assumptionAndConstraint : String[*]	Any assumptions, constraints, and limitations contained in the ArchitecturalDescription, including those affecting deployment, communications performance, information assurance environments, etc.
creatingOrganization : String[*]	The organization responsible for creating the ArchitecturalDescription.
dateCompleted : String[0..1]	Date that the ArchitecturalDescription was completed.
purpose : String[*]	Explains the need for the Architecture, what it will demonstrate, the types of analyses that will be applied to it, who is expected to perform the analyses, what decisions are expected to be made on the basis of each form of analysis, who is expected to make those decisions, and what actions are expected to result.
recommendations : String[*]	States the recommendations that have been developed based on the architecture effort. Examples include recommended system implementations, and opportunities for technology insertion.

summaryOfFindings : String[\*] Summarizes the findings that have been developed so far. This may be updated several times during the development of the ArchitecturalDescription.

toBe : Boolean[1] Indicates whether the ArchitecturalDescription represents an Architecture that exists or will exist in the future.

toolsUsed : String[\*] Identifies any tools used to develop the ArchitecturalDescription as well as file names and formats if appropriate.

Associations

architectureFramework : String[1] Indicates the type of framework used.

view : View[\*] Indicates which views are used in the ArchitecturalDescription.

viewPoint : Viewpoint[1] Indicates which Viewpoints are used in the ArchitecturalDescription. The definition of Viewpoint corresponds to the definition from ISO/IEC/IEEE 42010.

**Architecture**

**Package:** Summary and Overview

**isAbstract:** Yes

**Generalization:** [UAFElement](#)

**Extension:** Class

Description

An abstract element that represents a generic architecture. Subtypes are LogicalArchitecture and PhysicalArchitecture.

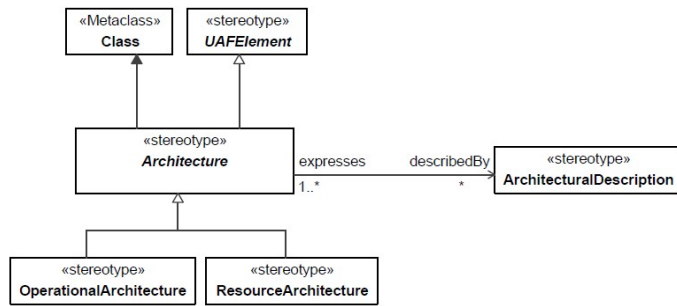


Figure 7.206 - Architecture

Associations

describedBy : ArchitecturalDescription[\*]      The description of an Architecture.

**Concern**

**Package:** Summary and Overview

**isAbstract:** No

**Generalization:** [PropertySet](#), Block

**Extension:** Class

Description

Interest in an EnterprisePhase (EnterprisePhase is synonym for System in ISO 42010) relevant to one or more of its stakeholders.

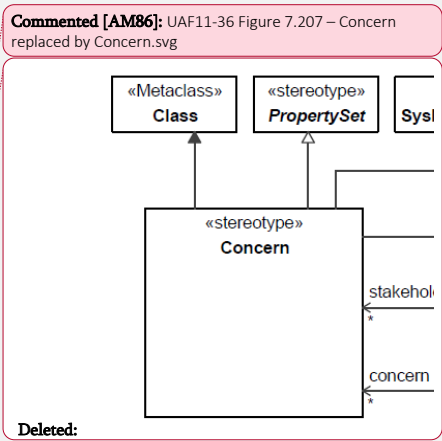
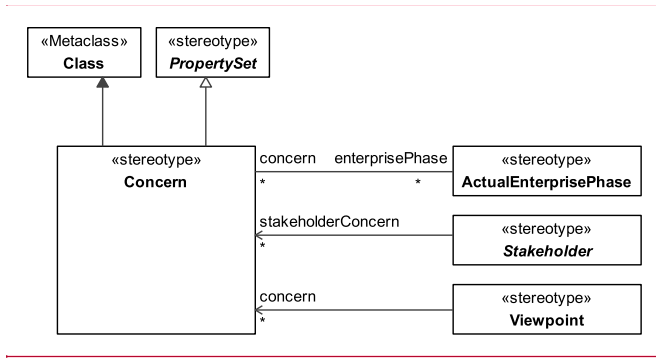


Figure 7.207 - Concern

**Associations**

systemConcern : ActualEnterprisePhase[\*]      Relates a Concern to the ActualEnterprisePhase that addresses that concern (ActualEnterprisePhase is synonym for System in ISO 42010).

**Stakeholder**

**Package:** Summary and Overview

**isAbstract:** Yes

**Generalization:** [UAFElement](#)

**Extension:** Element

**Description**

Individual, team, organization, or classes thereof, having an interest in an EnterprisePhase [ISO/IEC/IEEE 42010:2011].

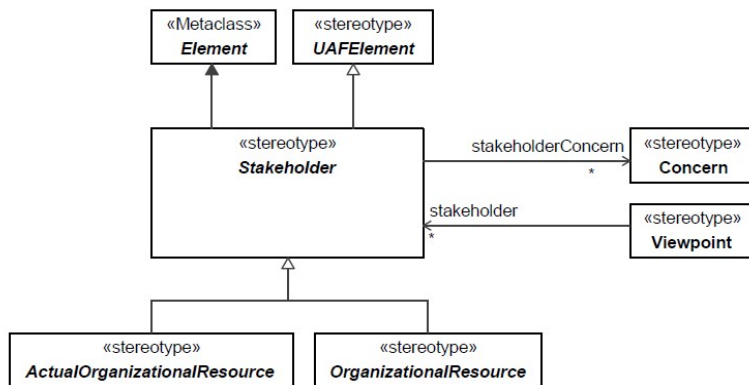


Figure 7.208 - Stakeholder

Associations

stakeholderConcern : Concern[\*] Relates a Stakeholder to a Concern.

**UAFElement**

**Package:** Summary and Overview

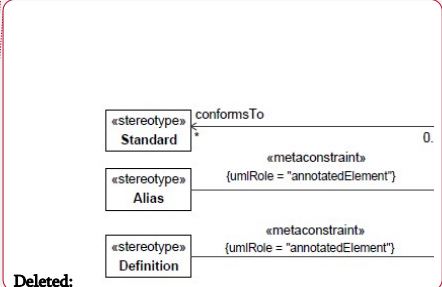
**isAbstract:** Yes

**Extension:** Element

Description

Abstract super type for all of the UAF elements. It provides a way for all of the UAF elements to have a common set of properties.

Commented [AM87]: UAF11-147. Figure 7.209 – UAFEElement replaced by UAFEElement.svg



Deleted:

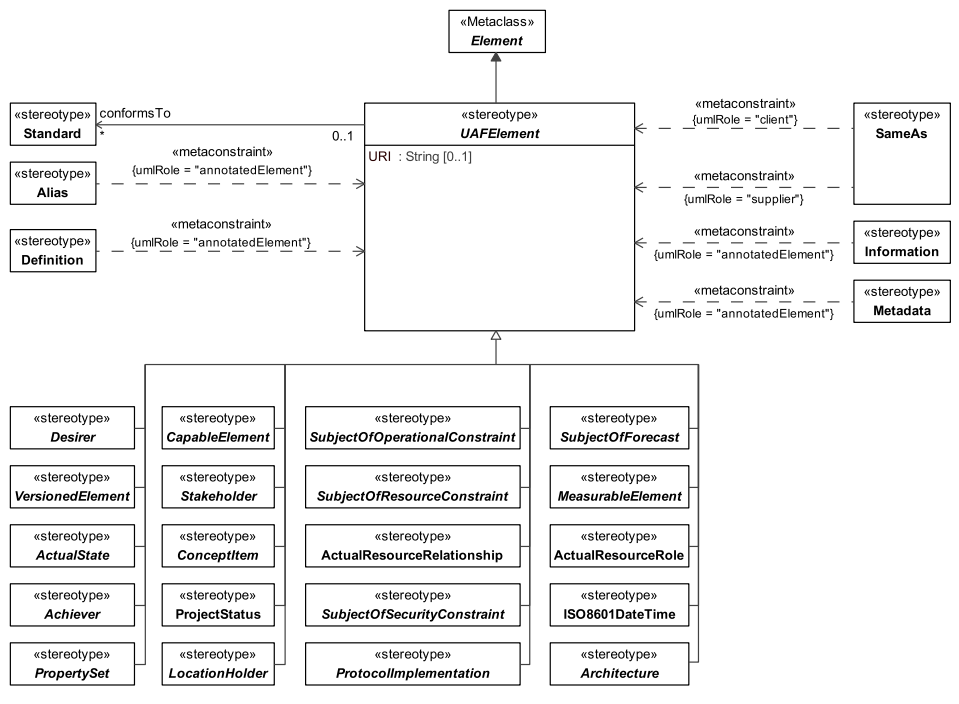


Figure 7.209 - UAFEElement

Attributes

URI : String[0..1] Captures Unique identifier for the element.

Associations

conformsTo : Standard[\*] Relates a UAFEElement to the Standard that the UAFEElement is conforming to.

View

Package: Summary and Overview



**isAbstract:** No

**Generalization:** [PropertySet](#), View

**Extension:** Class

Description

An architecture view expresses the architecture of the system-of-interest in accordance with an architecture viewpoint (or simply, viewpoint). [ISO/IEC/IEEE 42010:2011(E)].

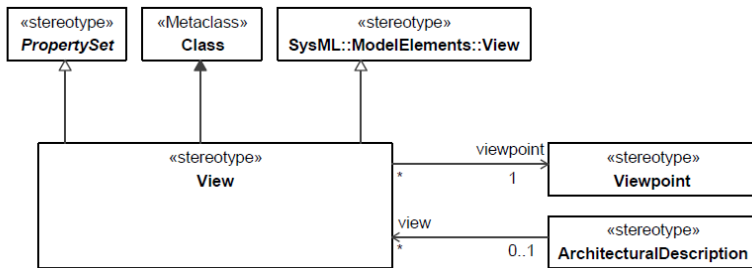


Figure 7.210 - View

Associations

viewpoint : Viewpoint[1]      Relates the View to the Viewpoint that the View conforms to.

**Viewpoint**

**Package:** Summary and Overview

**isAbstract:** No

**Generalization:** [PropertySet](#), Viewpoint

**Extension:** Class

Description

An architecture viewpoint frames (to formulate or construct in a particular style or language) one or more concerns. A concern can be framed by more than one viewpoint. [ISO/IEC/IEEE 42010:2011(E)].

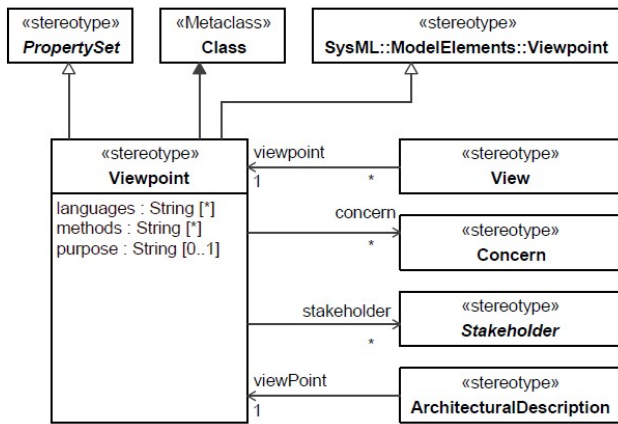


Figure 7.211 - Viewpoint

Attributes

- languages : String[\*]      The languages used to express the Viewpoint.
- methods : String[\*]      The methods employed in the development of the Viewpoint.
- purpose : String[0..1]    The purpose of the Viewpoint.

Associations

- concern : Concern[\*]      Relates the Viewpoint to the Concerns that the Viewpoint addresses.
- stakeholder : Stakeholder[\*]    Relates the Viewpoint to the Stakeholders whose Concerns are being addressed by the Viewpoint.

# Annex A: UAF Views (Profile)

(informative)

## A.1 General

This section is intended as non-normative guidance for developers and users as to what UAF elements and relationships are applicable for each of the UAF Views.

## A.2 View Specifications

MODAF: A connected and coherent set of Architectural Elements which conform to a View.

DoDAF Alias: View: DoDAF divides the problem space into manageable pieces, according to the stakeholder's Viewpoint, further defined in the framework as "Views."

### A.2.1 View Specifications::Strategic

**Stakeholders:** Capability Portfolio Managers

**Concerns:** capability management process

**Definition:** describe capability taxonomy, composition, dependencies and evolution

#### View Specifications::Strategic::Taxonomy

Stakeholders: PMs, Enterprise Architects, Executives

Concerns: capability needs

Definition: shows the taxonomy of capabilities

Recommended Implementation: SysML Block Definition Diagram

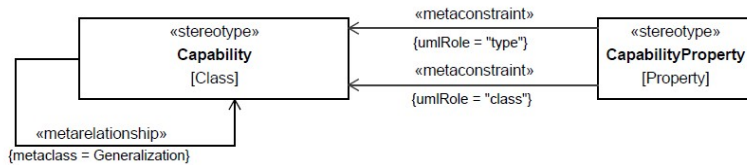


Figure A.1 - Strategic Taxonomy

Elements

- [Capability](#)
- [CapabilityProperty](#)

### View Specifications::Strategic::Structure

Stakeholders: PMs, Enterprise Architects, Executives

Concerns: capability needs

Definition: shows the relationship between EnterprisePhases and the Capabilities that are intended to be developed during the enterprise phases, and the organizations involved in the enterprise.

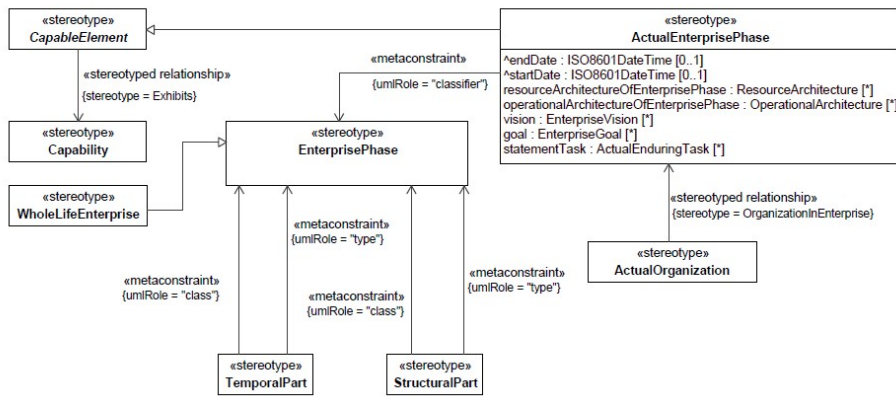


Figure A.2 - Strategic Structure

Elements

- [ActualEnterprisePhase](#)
- [ActualOrganization](#)
- [Capability](#)
- [CapableElement](#)
- [EnterprisePhase](#)
- [StructuralPart](#)
- [TemporalPart](#)
- [WholeLifeEnterprise](#)

### View Specifications::Strategic::Connectivity

Stakeholders: PMs, Executives, Enterprise Architects

Concerns: capability dependencies

Definition: describes the dependencies between planned capabilities

Recommended Implementation: SysML Block Definition Diagram, SysML Internal Block Diagram

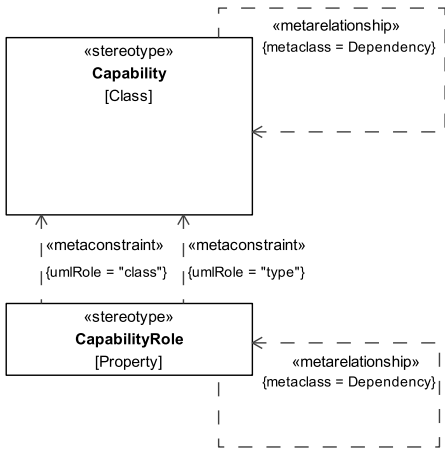


Figure A.3 - Strategic Connectivity

Elements

- [Capability](#)
  - [CapabilityRole](#)

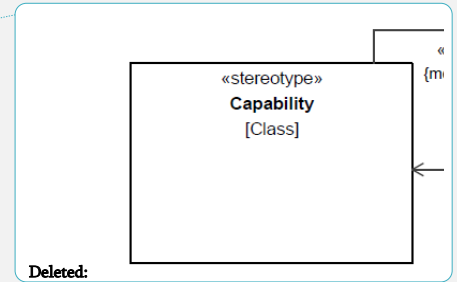
**View Specifications::Strategic::States**

Stakeholders: PMs, Enterprise Architects

Concerns: effects that the implementation(s) of capabilities are expected to deliver

Definition: captures the relationships between capability(ies) and desired effect(s) that implementation(s) of capability(ies) should achieve.

Recommended Implementation: SysML Block Definition Diagram



Deleted:

**Commented [Y88]:** UAF11-73 Replace figure with Strategic\_Connectivity.svg

**Commented [Y89]:** UAF11-73 Added CapabilityRole to elements

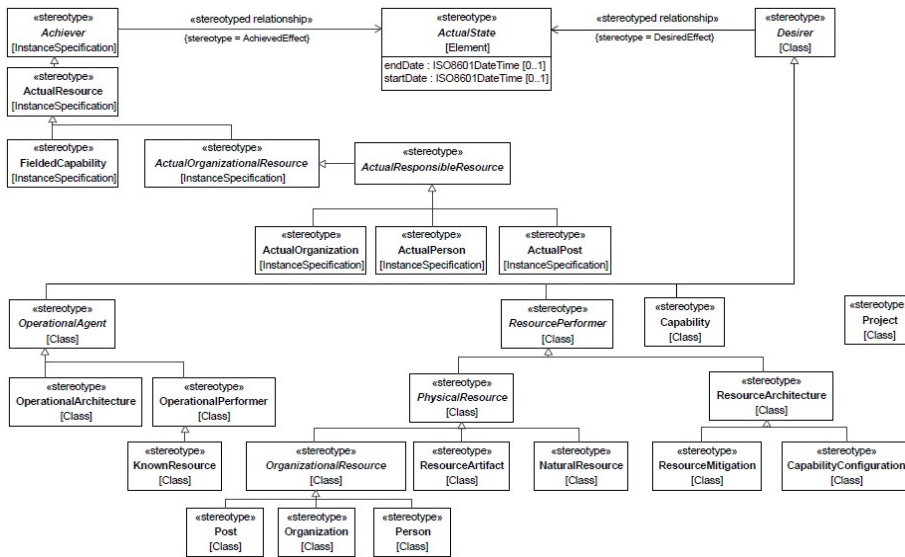


Figure A.4 - Strategic States

Elements

- [Achiever](#)
- [ActualOrganization](#)
- [ActualOrganizationalResource](#)
- [ActualPerson](#)
- [ActualPost](#)
- [ActualResource](#)
- [ActualResponsibleResource](#)
- [ActualState](#)
- [Capability](#)
- [CapabilityConfiguration](#)
- [Desirer](#)
- [FieldedCapability](#)
- [KnownResource](#)
- [NaturalResource](#)

- [OperationalAgent](#)
- [OperationalArchitecture](#)
- [OperationalPerformer](#)
- [Organization](#)
- [OrganizationalResource](#)
- [Person](#)
- [PhysicalResource](#)
- [Post](#)
- [Project](#)
- [ResourceArchitecture](#)
- [ResourceArtifact](#)
- [ResourceMitigation](#)
- [ResourcePerformer](#)

**View Specifications::Strategic::Constraints**

Stakeholders: PMs, Enterprise Architects

Concerns: capability constraints

Definition: details the measurements that set performance requirements constraining capabilities

Recommended Implementation: tabular format, SysML Block Definition Diagram

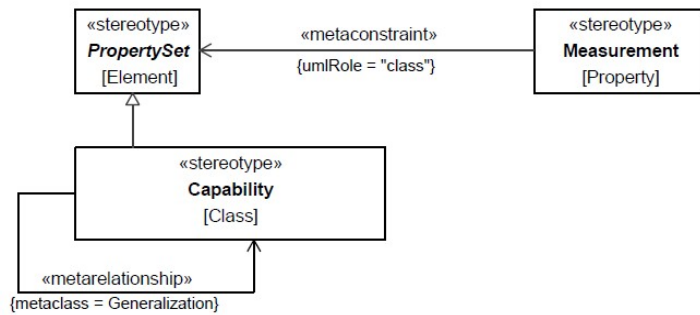


Figure A.5 - Strategic Constraints

Elements

- [Capability](#)
- [Measurement](#)
- [PropertySet](#)









- [CapabilityConfiguration](#)
- [CapableElement](#)
- [FieldedCapability](#)
- [ResourceArchitecture](#)
- [ResourcePerformer](#)

**View Specifications::Strategic::Traceability**

Stakeholders: PMs, Enterprise Architects, Business Architects

Concerns: traceability between capabilities and operational activities

Definition: describes the mapping between the capabilities required by an Enterprise and the supporting operational activities.

Recommended Implementation: matrix format, SysML Block Definition Diagram

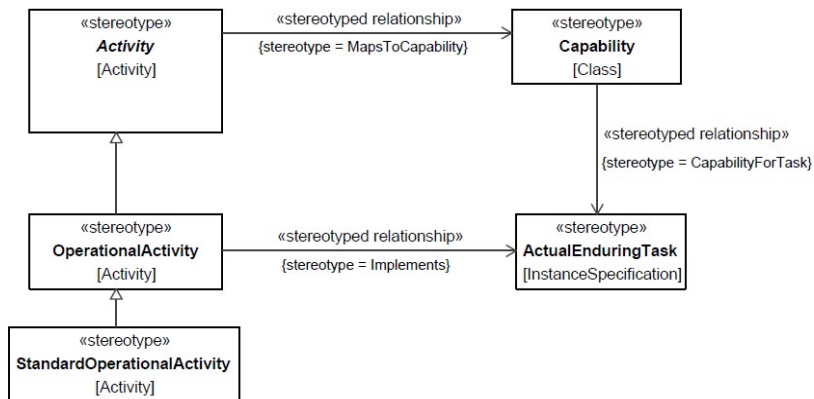


Figure A.8 - Strategic Traceability

Elements

- [Activity](#)
- [ActualEnduringTask](#)
- [Capability](#)
- [OperationalActivity](#)
- [StandardOperationalActivity](#)

## A.2.2 View Specifications::Operational

**Stakeholders:** Business Architects, Executives

**Concerns:** illustrate the Logical Architecture of the enterprise

**Definition:** describe the requirements, operational behavior, structure, and exchanges required to support (exhibit) capabilities. Defines all operational elements in an implementation/solution independent manner.

### View Specifications::Operational::Taxonomy

**Stakeholders:** Business Architects, Systems Engineers, Enterprise Architects, Owners responsible for Operational Agents

**Concerns:** OperationalAgent types

**Definition:** shows the taxonomy of types of OperationalAgents

**Recommended Implementation:** SysML Block Definition Diagram, [SysML Internal Block Diagram](#).

**Commented [GB90]:** UAF11- 69 add text, SysML Internal Block Diagram

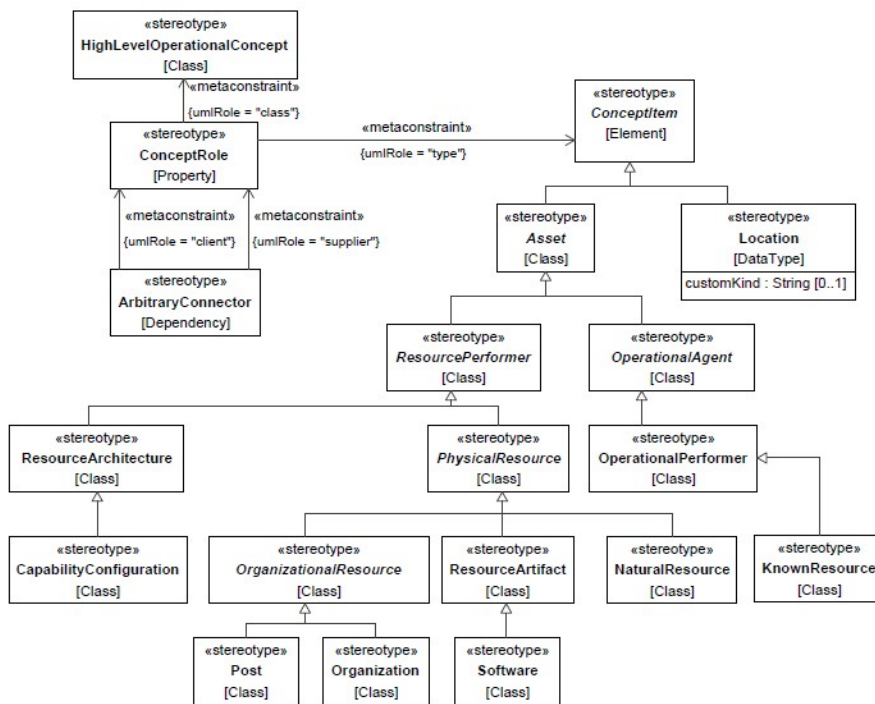


Figure A.9 - Operational Taxonomy

Elements

Unified Architecture Framework Profile (UAFP), v1.0

- [ArbitraryConnector](#)
- [Asset](#)
- [CapabilityConfiguration](#)
- [ConceptItem](#)
- [ConceptRole](#)
- [HighLevelOperationalConcept](#)
- [KnownResource](#)
- [Location](#)
- [NaturalResource](#)
- [OperationalAgent](#)
- [OperationalPerformer](#)
- [Organization](#)
- [OrganizationalResource](#)
- [PhysicalResource](#)
- [Post](#)
- [ResourceArchitecture](#)
- [ResourceArtifact](#)
- [ResourcePerformer](#)
- [Software](#)

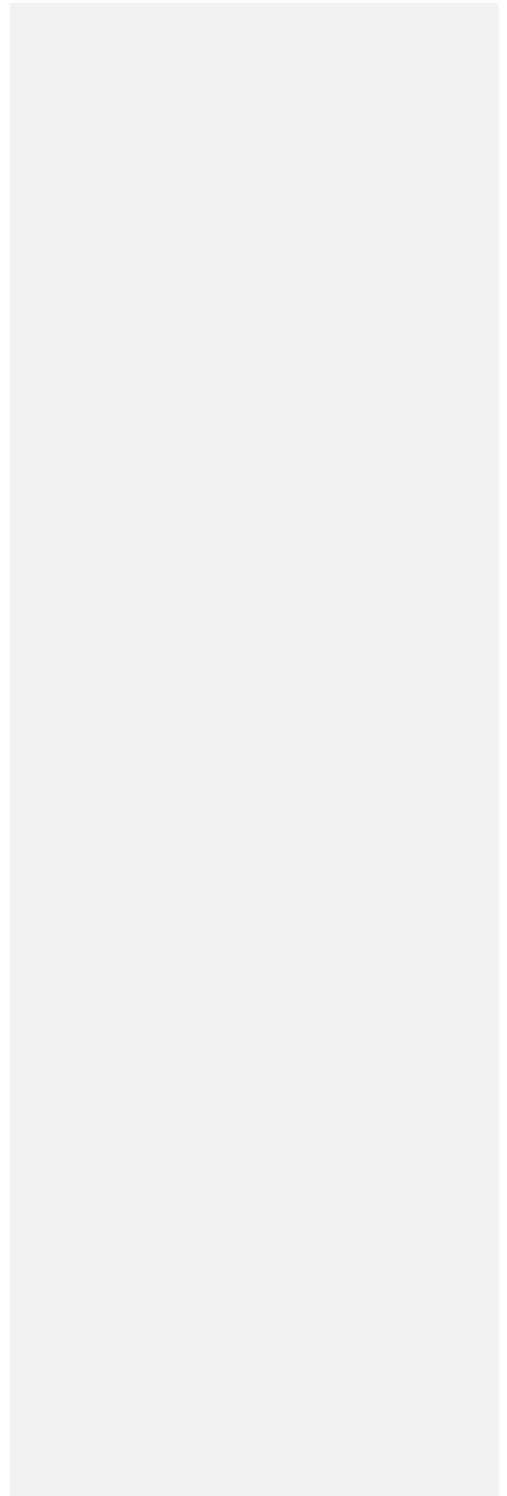
**View Specifications::Operational::Structure**

Stakeholders: Business Architects, Systems Engineers, Enterprise Architects, Owners responsible for Operational Agents

Concerns: identifies the operational exchange requirements between OperationalPerformers

Definition: defines operational architecture and exchange requirements necessary to support a specific set of Capability(ies).

Recommended Implementation: SysML Block Definition Diagram, SysML Internal Block Diagram



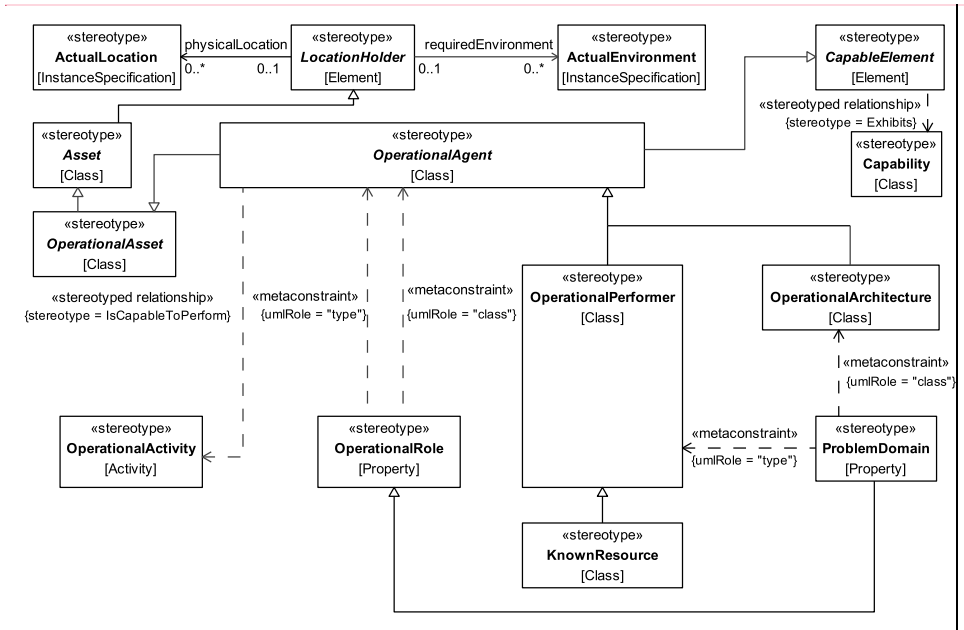


Figure A.10 - Operational Structure

Elements

- [ActualEnvironment](#)
- [ActualLocation](#)
- [Asset](#)
- [Capability](#)
- [CapableElement](#)
- [KnownResource](#)
- [LocationHolder](#)
- [OperationalActivity](#)
- [OperationalAgent](#)
- [OperationalArchitecture](#)
- [OperationalAsset](#)
- [OperationalPerformer](#)
- [OperationalRole](#)
- [ProblemDomain](#)

View Specifications::Operational::Connectivity

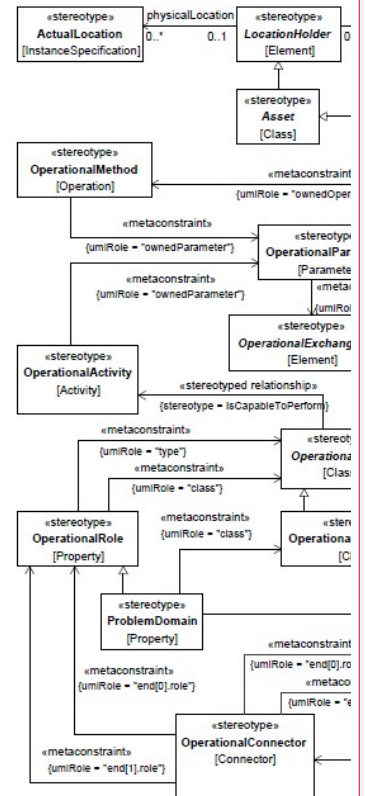
Stakeholders: Systems Engineers, Architects, Solution Providers.

Concerns: capture the interfaces between OperationalPerformers.

Definition: summarizes logical exchanges between OperationalPerformers of information, systems, personnel, energy etc. and

Unified Architecture Framework Profile (UAFP), v1.0

Commented [AM91]: UAF11-15 Figure A.10 - Operational Structure replaced by Operational\_Structure.svg



Deleted:

Commented [AM92]: UAF11-15 List of elements updated according to Operational\_Structure.svg

Deleted: • ~~ActualLocation~~

- ~~Asset~~
- ~~Capability~~
- ~~CapableElement~~
- ~~Environment~~
- ~~KnownResource~~
- ~~LocationHolder~~
- ~~OperationalActivity~~
- ~~OperationalAgent~~

the logical activities that produce and consume them. Measurements can optionally be included.  
Recommended Implementation: SysML Internal Block Diagram, tabular format.

**Commented [AM93]:** UAF11-204

**Deleted:** Stakeholders: Systems Engineers, Architects, Solution Providers  
Concerns: capture the interfaces between OperationalPerformers  
Definition: summarizes logical exchanges between OperationalPerformers of information, systems, personnel, energy, etc. and the logical activities that produce and consume them. Measurements can optionally be included.  
Recommended Implementation: tabular format



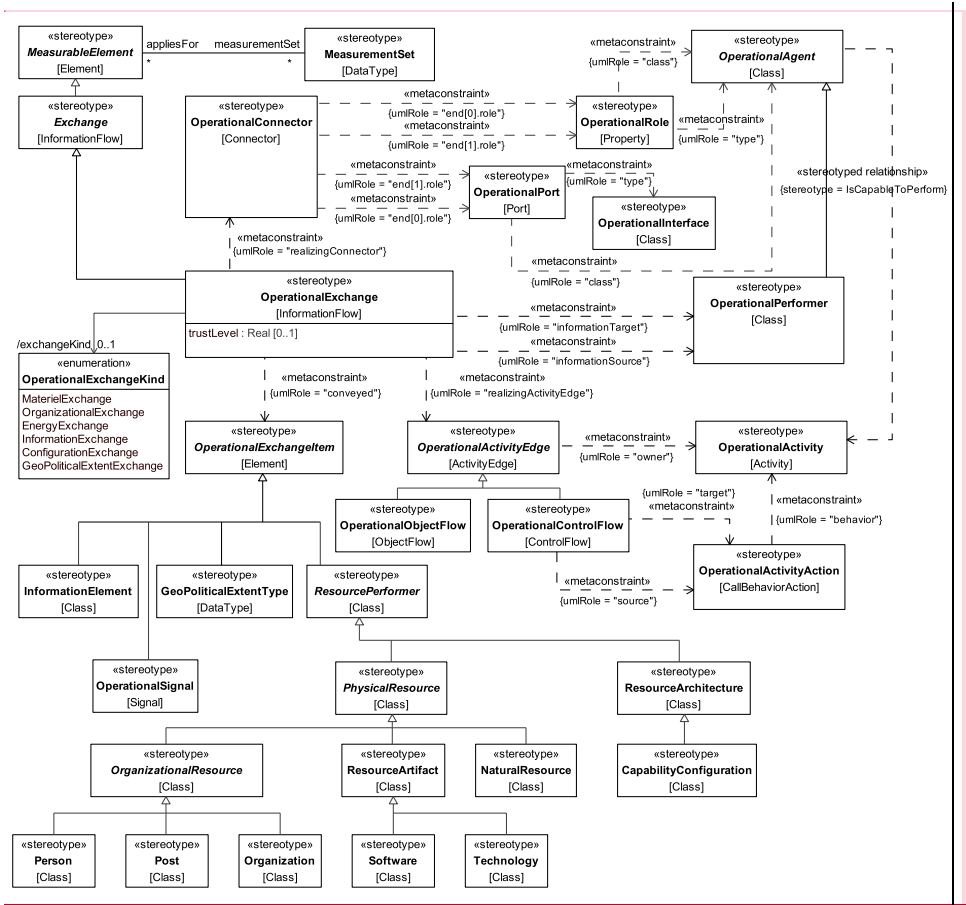
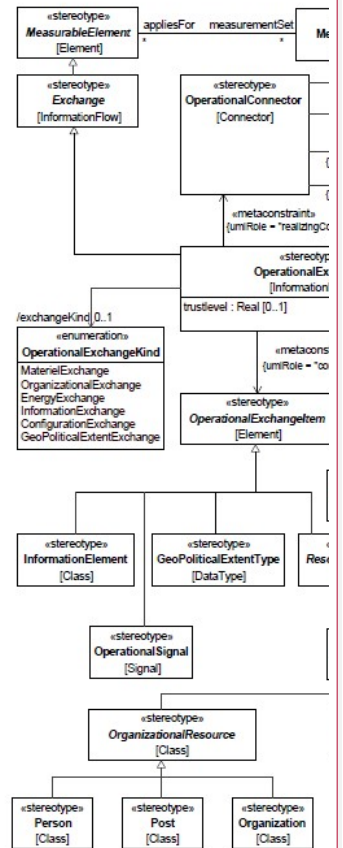


Figure A.11 - Operational Connectivity

Elements

- [CapabilityConfiguration](#)
- [Exchange](#)
- [GeoPoliticalExtentType](#)
- [InformationElement](#)
- [MeasurableElement](#)

Commented [AM94]: UAF11-204, UAF11-49. Removed original, added Operational\_Connectivity.svg



Deleted:

- [MeasurementSet](#)
- [NaturalResource](#)
- [OperationalActivity](#)
- [OperationalActivityAction](#)
- [OperationalActivityEdge](#)
- [OperationalAgent](#)
- [OperationalConnector](#)
- [OperationalControlFlow](#)
- [OperationalExchange](#)
- [OperationalExchangeItem](#)
- [OperationalExchangeKind](#)
- [OperationalInterface](#)
- [OperationalObjectFlow](#)
- [OperationalPerformer](#)
- [OperationalPort](#)
- [OperationalRole](#)
- [OperationalSignal](#)
- [Organization](#)
- [OrganizationalResource](#)
- [Person](#)
- [PhysicalResource](#)
- [Post](#)
- [ResourceArchitecture](#)
- [ResourceArtifact](#)
- [ResourcePerformer](#)
- [Software](#)
- [Technology](#)

#### **View Specifications::Operational::Processes**

Stakeholders: Business Architect, Systems Engineers, Enterprise Architects

Concerns: captures activity based behavior and flows

Definition: describes the activities that are normally conducted in the course of achieving business goals that support a capability. It describes operational activities, their Inputs/Outputs, operational activity actions and flows between them.

Recommended Implementation: SysML Activity Diagram, SysML Block Definition Diagram, BPMN Process Diagram



- [OperationalAgent](#)
- [OperationalControlFlow](#)
- [OperationalExchange](#)
- [OperationalExchangeItem](#)
- [OperationalMethod](#)
- [OperationalObjectFlow](#)
- [OperationalParameter](#)
- [OperationalPerformer](#)
- [OperationalRole](#)
- [RequiredServiceLevel](#)
- [ServiceSpecification](#)
- [StandardOperationalActivity](#)

**View Specifications::Operational::States**

Stakeholders: Systems Engineers, Software Engineers

Concerns: capture state-based behavior of an operational OperationalPerformer

Definition: it is a graphical representation of states of an operational OperationalPerformer and how that operational OperationalPerformer responds to various events and actions.

Recommended Implementation: SysML State [Machine Diagram](#).

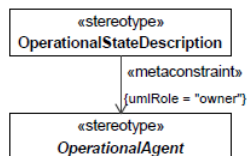


Figure A.13 - Operational States

Elements

- [OperationalAgent](#)
- [OperationalStateDescription](#)

**View Specifications::Operational::Interaction Scenarios**

Stakeholders: Systems Engineers, Business Architects

Concerns: express a time ordered examination of the operational exchanges as a result of a particular operational scenario.

Definition: provides a time-ordered examination of the operational exchanges between participating nodes.

(OperationalPerformer roles) as a result of a particular operational scenario.

Formatted: Right: 8.38 cm  
 Commented [AM95]: UAF11-93 "SysML State Diagram" changed to "SysML State Machine Diagram".

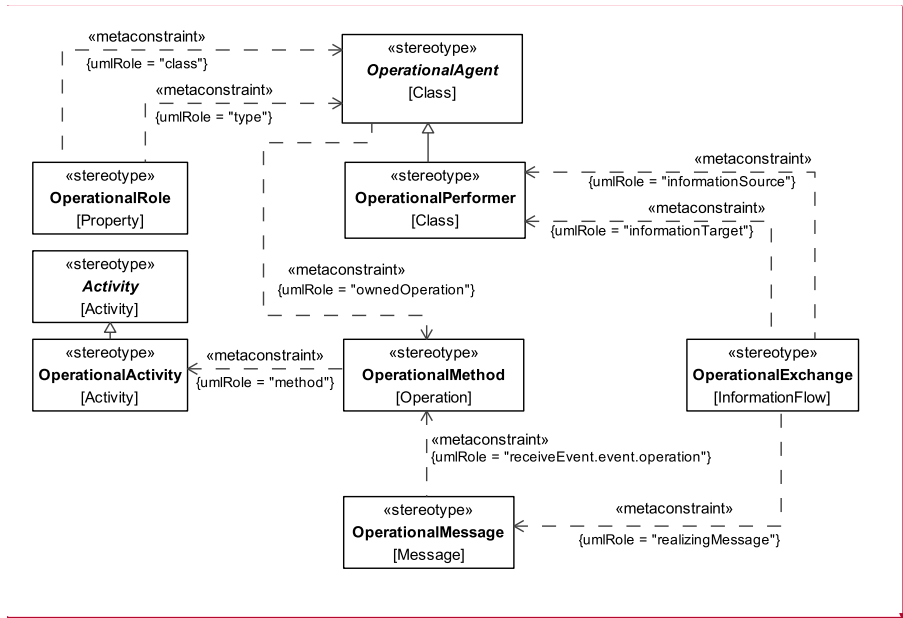


Figure A.14 - Operational Interaction Scenarios

Elements

- [Activity](#)
- [OperationalActivity](#)
- [OperationalAgent](#)
- [OperationalExchange](#)
- [OperationalMessage](#)
- [OperationalMethod](#)
- [OperationalPerformer](#)
- [OperationalRole](#)

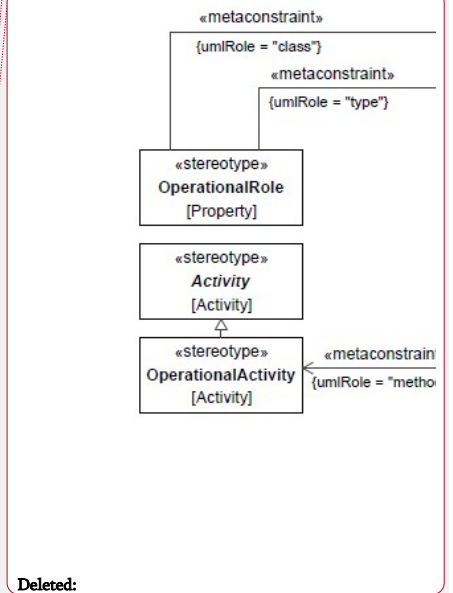
**View Specifications::Operational::Constraints**

Stakeholders: Systems Engineers, Architects, Program Sponsors

Concerns: define operational limitations, constraints and performance parameters for the enterprise

Definition: specifies traditional textual operational or business rules that are constraints on the way that business is done in the enterprise. The addition of SysML parametrics provides a computational means of defining operational constraints across

Commented [AM96]: UAF11-76 Figure A.14 - Operational Interaction Scenarios replaced by Operational\_Interaction\_Scenarios.svg



Deleted:

the enterprise or within a specific operational context.  
 Recommended Implementation: tabular format, SysML Block Definition Diagram, SysML Parametric Diagram

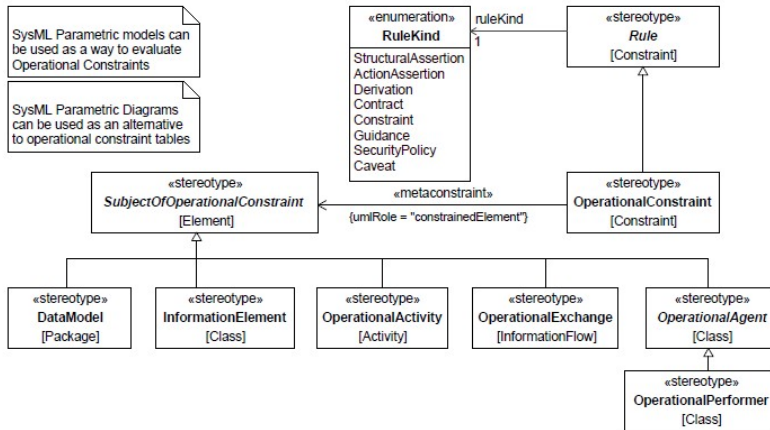


Figure A.15 - Operational Constraints

Elements

- [DataModel](#)
- [InformationElement](#)
- [OperationalActivity](#)
- [OperationalAgent](#)
- [OperationalConstraint](#)
- [OperationalExchange](#)
- [OperationalPerformer](#)
- [Rule](#)
- [RuleKind](#)
- [SubjectOfOperationalConstraint](#)

**View Specifications::Operational::Traceability**

Stakeholders: PMs, Enterprise Architects, Business Architects

Concerns: traceability between capabilities and operational activities and capabilities and operational agents

Definition: describes the mapping between the capabilities required by an Enterprise and the supporting operational activities and operational agents.

Recommended Implementation: matrix format, SysML Block Definition Diagram

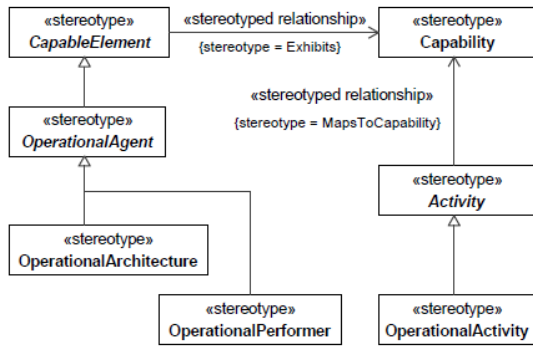


Figure A.16 - Operational Traceability

Elements

- [Activity](#)
- [Capability](#)
- [CapableElement](#)
- [OperationalActivity](#)
- [OperationalAgent](#)
- [OperationalArchitecture](#)
- [OperationalPerformer](#)

**A.2.3 View Specifications::Services**

**Stakeholders:** Enterprise Architects, Solution Providers, Systems Engineers, Software Architects, Business Architects

**Concerns:** specifications of services required to exhibit a Capability

**Definition:** shows Service Specifications and required and provided service levels of these specifications required to exhibit a Capability or to support an Operational Activity.

**View Specifications::Services::Taxonomy**

**Stakeholders:** Enterprise Architects, Solution Providers, Systems Engineers, Software Architects, Business Architects

**Concerns:** service specification types and required and provided service levels of these types

**Definition:** shows the taxonomy of types of services and the level of service that they are expected to provide or are required to meet through the display of ActualMeasurements associated with the Provided and Required Service Level. Recommended

**Implementation:** SysML Block Definition Diagram

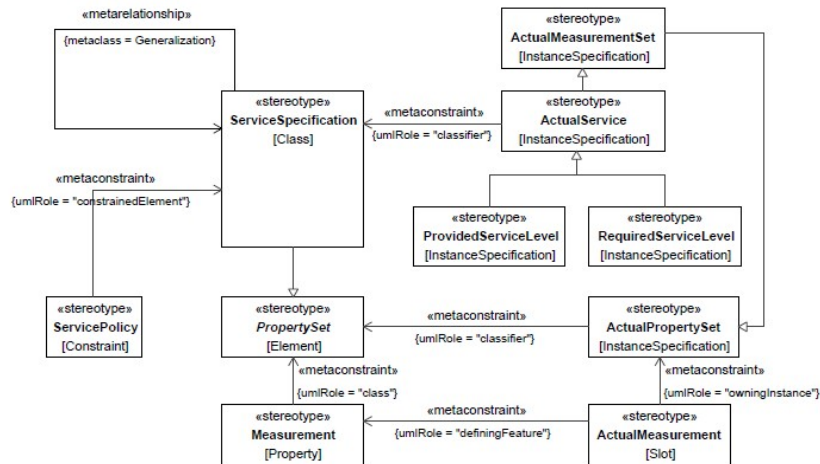


Figure A.17 - Services Taxonomy

Elements

- [ActualMeasurement](#)
- [ActualMeasurementSet](#)
- [ActualPropertySet](#)
- [ActualService](#)
- [Measurement](#)
- [PropertySet](#)
- [ProvidedServiceLevel](#)
- [RequiredServiceLevel](#)
- [ServicePolicy](#)
- [ServiceSpecification](#)

**View Specifications::Services::Structure**

Stakeholders: Solution Providers, Systems Engineers, Software Architects, Business Architects

Concerns: combination of services required to exhibit a capability

Definition: shows the composition of services and how services are combined into a higher level service required to exhibit a capability or support an operational activity.

Recommended Implementation: SysML Block Definition Diagram, SysML Internal Block Diagram



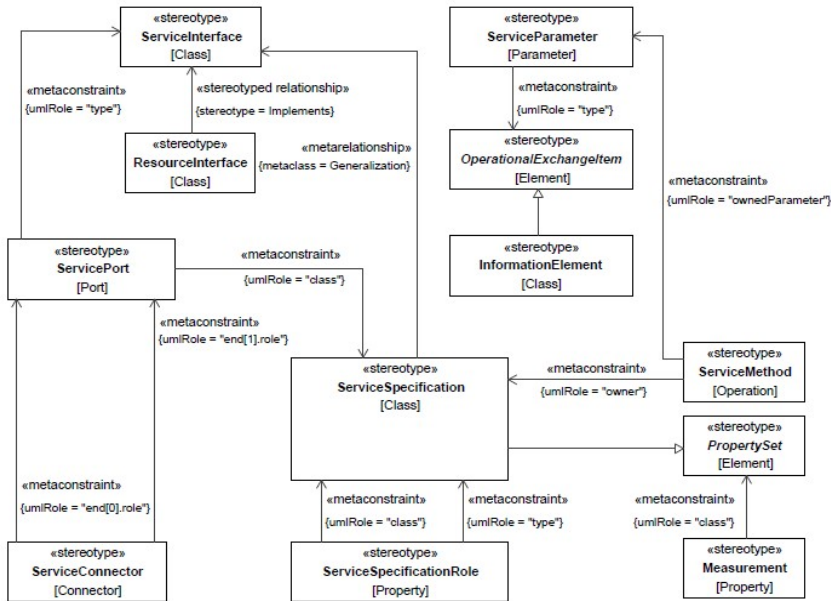


Figure A.18 - Services Structure

Elements

- [InformationElement](#)
- [Measurement](#)
- [OperationalExchangeItem](#)
- [PropertySet](#)
- [ResourceInterface](#)
- [ServiceConnector](#)
- [ServiceInterface](#)
- [ServiceMethod](#)
- [ServiceParameter](#)
- [ServicePort](#)
- [ServiceSpecification](#)
- [ServiceSpecificationRole](#)

**View Specifications::Services::Connectivity**

Stakeholders: Solution Providers, Systems Engineers, Software Architects, Business Architects

Concerns: interoperability among services

Definition: specifies service interfaces, e.g., provided and required service operations, to ensure compatibility and reusability of services.

Recommended Implementation: SysML Block Definition Diagram, SysML Internal Block Diagram, tabular format

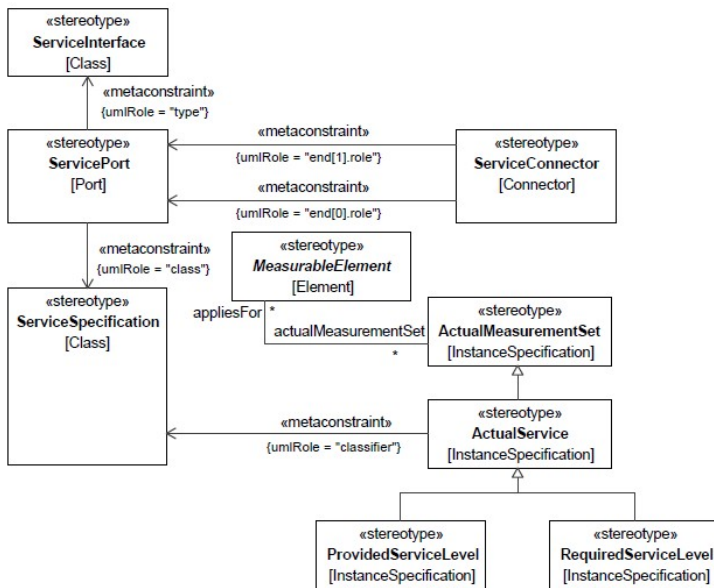


Figure A.19 - Services Connectivity

Elements

- [ActualMeasurementSet](#)
- [ActualService](#)
- [MeasurableElement](#)
- [ProvidedServiceLevel](#)
- [RequiredServiceLevel](#)
- [ServiceConnector](#)
- [ServiceInterface](#)
- [ServicePort](#)
- [ServiceSpecification](#)

**View Specifications::Services::Processes**

Stakeholders: Solution Providers, Systems Engineers, Software Architects, Business Architects  
 Concerns: the behavior of a service in terms of the operational activities it is expected to support  
 Definition: provides detailed information regarding the allocation of service functions to service specifications, and data flows between service functions.  
 Recommended Implementation: SysML Block Definition Diagram, SysML Internal Block Diagram, tabular format

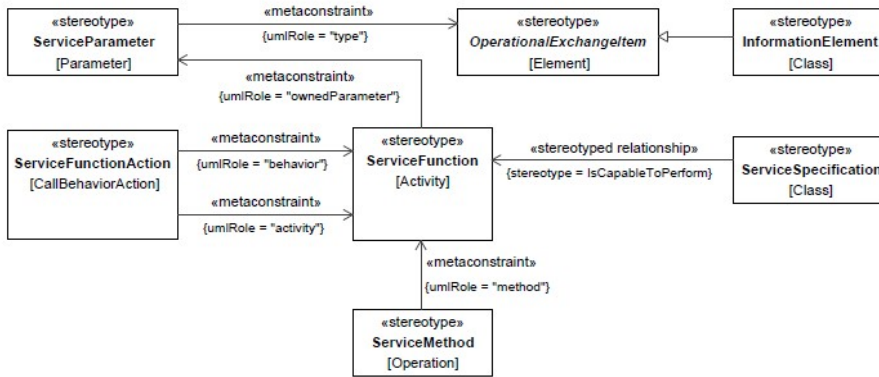


Figure A.20 - Services Processes

Elements

- [InformationElement](#)
- [OperationalExchangeItem](#)
- [ServiceFunction](#)
- [ServiceFunctionAction](#)
- [ServiceMethod](#)
- [ServiceParameter](#)
- [ServiceSpecification](#)

**View Specifications::Services::States**

Stakeholders: Solution Providers, Systems Engineers, Software Architects, Business Architects  
 Concerns: the behavior of a service specification in terms of states and events causing transitions between states  
 Definition: specifies the possible states a service specification may have, and the possible transitions between those states.  
 Recommended Implementation: SysML State Machine Diagram

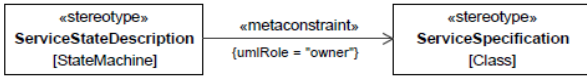


Figure A.21 - Services States

Elements

- [ServiceSpecification](#)
- [ServiceStateDescription](#)

**View Specifications::Services::Interaction Scenarios**

Stakeholders: Solution Providers, Systems Engineers, Software Architects, Business Architects

Concerns: the behavior of a service specification in terms of expected time-ordered examination of the interactions between service roles.

Definition: specifies how a service roles interact with each other, service providers and consumers, and the sequence and dependencies of those interactions.

Recommended Implementation: SysML Sequence Diagram

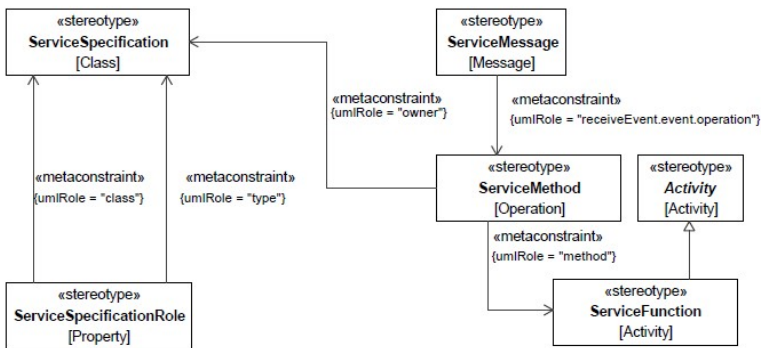


Figure A.22 - Services Interaction Scenarios

Elements

- [Activity](#)

- [ServiceFunction](#)
- [ServiceMessage](#)
- [ServiceMethod](#)
- [ServiceSpecification](#)
- [ServiceSpecificationRole](#)

**View Specifications::Services::Constraints**

Stakeholders: Solution Providers, Systems Engineers, Software Architects, Business Architects  
 Concerns: service policies that apply to implementations of service specifications  
 Definition: specifies traditional textual service policies that are constraints on the way that service specifications are implemented within resources. The addition of SysML parametrics provide a computational means of defining service policies across the enterprise or within a specific service configuration.  
 Recommended Implementation: tabular format, SysML Parametric Diagram

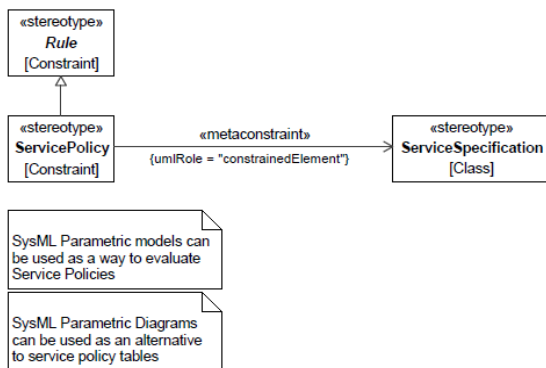


Figure A.23 - Services Constraints

Elements

- [Rule](#)
- [ServicePolicy](#)
- [ServiceSpecification](#)

**View Specifications::Services::Roadmap**

Stakeholders: Solution Providers, Systems Engineers, Software Architects, Business Architects  
 Concerns: service specification changes over time  
 Definition: provides an overview of how a service specification changes over time. It shows the combination of several

service specifications mapped against a timeline.  
 Recommended Implementation: timeline, SysML Block Definition Diagram, SysML Internal Block Diagram

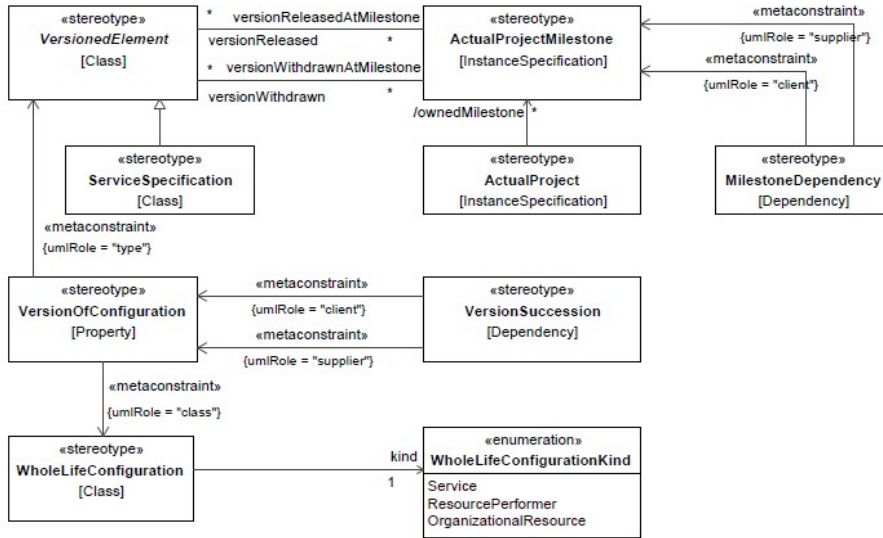


Figure A.24 - Services Roadmap

Elements

- [ActualProject](#)
- [ActualProjectMilestone](#)
- [MilestoneDependency](#)
- [ServiceSpecification](#)
- [VersionedElement](#)
- [VersionOfConfiguration](#)
- [VersionSuccession](#)
- [WholeLifeConfiguration](#)
- [WholeLifeConfigurationKind](#)

**View Specifications::Services::Traceability**

Stakeholders: Solution Providers, Systems Engineers, Software Architects, Business Architects  
 Concerns: traceability between operational activities and service specifications that support them

Definition: depicts the mapping of service specifications to operational activities and how service specifications contribute to the achievement of a capability.

Recommended Implementation: [tabular or matrix format](#).

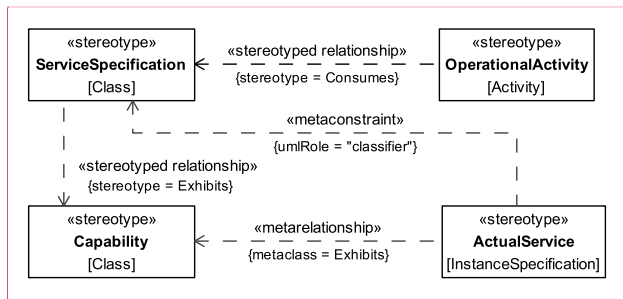


Figure A.25 - Services Traceability

Elements

- [ActualService](#)
- [Capability](#)
- [OperationalActivity](#)
- [ServiceSpecification](#)

### A.2.4 View Specifications::Personnel

**Stakeholders:** Human resources, Solution Providers, PMs

**Concerns:** human factors

**Definition:** aims to clarify the role of Human Factors (HF) when creating architectures in order to facilitate both Human Factors Integration (HFI) and systems engineering (SE).

#### View Specifications::Personnel::Taxonomy

**Stakeholders:** Human resources, Solution Providers, PMs

**Concerns:** organizational resource types

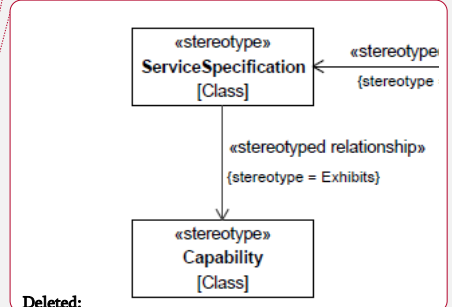
**Definition:** shows the taxonomy of types of organizational resources.

**Recommended Implementation:** SysML Block Definition Diagram

**Commented [GB97]:** UAF11-98 delete original change to "tabular or matrix format"

**Deleted:** timeline, SysML Block Definition Diagram, SysML Internal Block Diagram

**Commented [AM98]:** [UAF11-128](#) Figure A.25 - Services Traceability replaced by Services\_Traceability.svg



**Commented [AM99]:** [UAF11-128](#) ActualService added to the Elements list.

**Formatted:** OMG Bold Bullet KeepNext Paragraph, Right: 0 cm, Bulleted + Level: 1 + Aligned at: 0.63 cm + Indent at: 1.27 cm

**Formatted:** Font: Bold

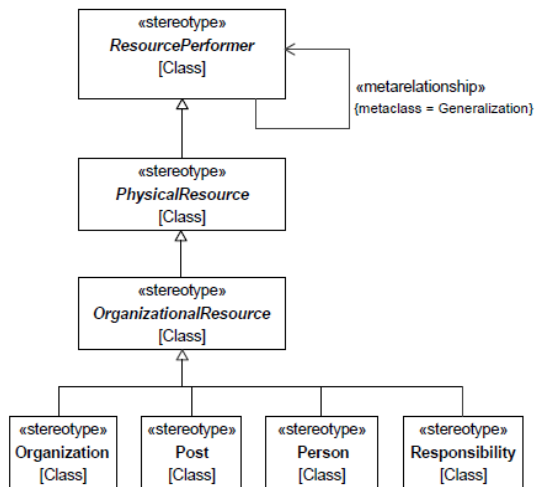


Figure A.26 - Personnel Taxonomy

Elements

- [Organization](#)
- [OrganizationalResource](#)
- [Person](#)
- [PhysicalResource](#)
- [Post](#)
- [ResourcePerformer](#)
- [Responsibility](#)

**View Specifications::Personnel::Structure**

Stakeholders: Human resources, Solution Providers, PMs

Concerns: typical organizational structure used to support a capability(ies)

Definition: shows organizational structures and possible interactions between organizational resources.

Recommended Implementation: SysML Block Definition Diagram, SysML Internal Block Diagram



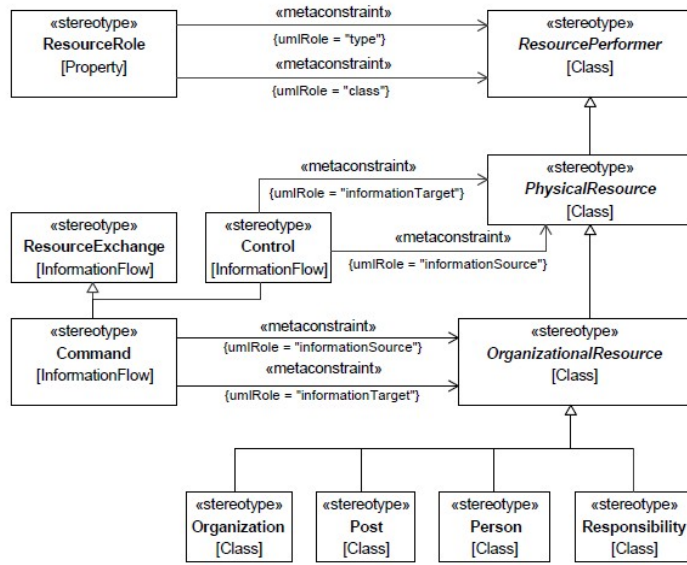


Figure A.27 - Personnel Structure

Elements

- [Command](#)
- [Control](#)
- [Organization](#)
- [OrganizationalResource](#)
- [Person](#)
- [PhysicalResource](#)
- [Post](#)
- [ResourceExchange](#)
- [ResourcePerformer](#)
- [ResourceRole](#)
- [Responsibility](#)

**View Specifications::Personnel::Connectivity**

Stakeholders: Solution providers

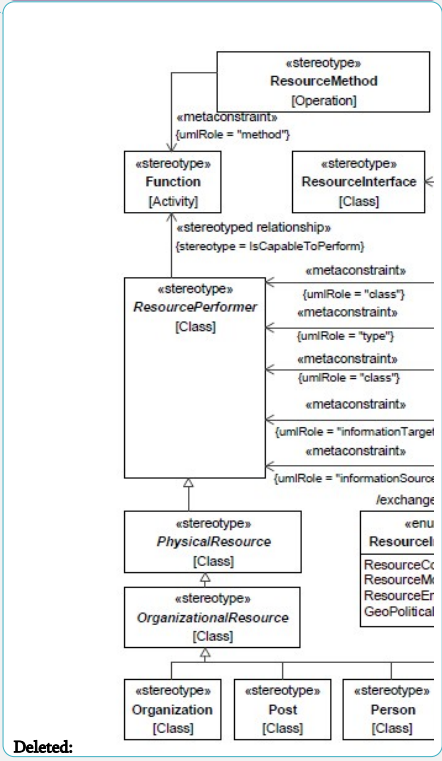
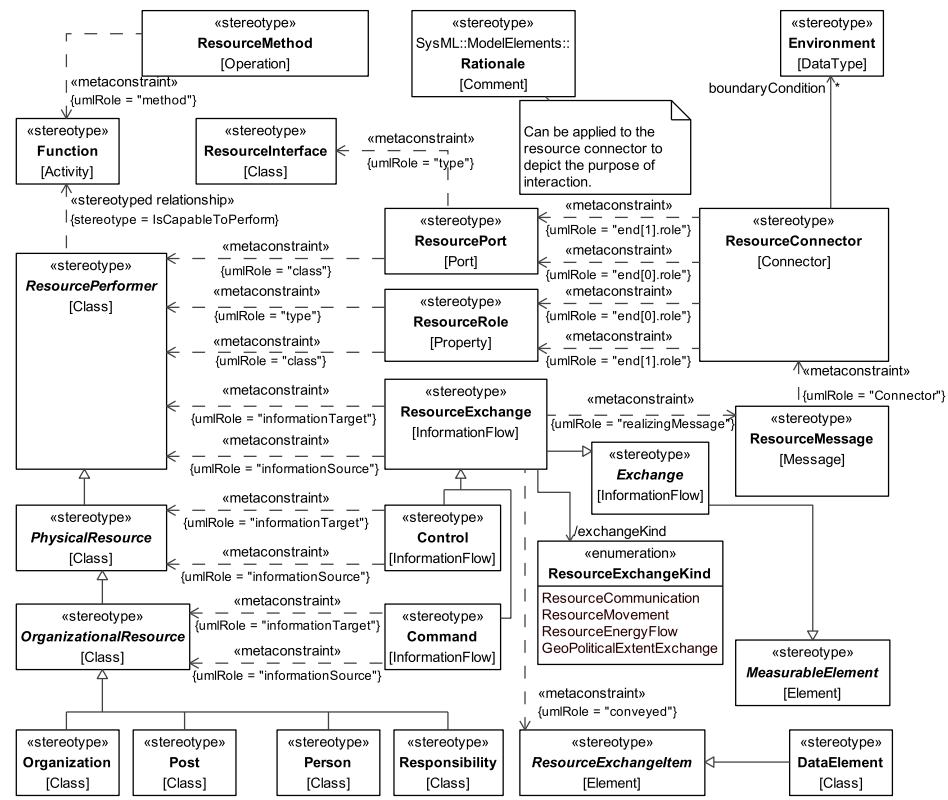
Concerns: interaction of organizational resources

Definition: captures the possible interactions between organizational resources, including command and control relationships.

Interactions typically illustrate the fundamental roles and management responsibilities.

Recommended Implementation: SysML, Internal Block Diagram, tabular format

**Commented [GB100]:** UAF11-87 add  
**Commented [GB101]:** UAF11-87 SysML, Internal Block Diagram



Deleted:

Figure A.28 - Personnel Connectivity

**Commented [Y102]:** UAF11-48 Replace image with Personnel\_Connectivity.svg

## Elements

- [DataElement](#)
- [Environment](#)
- [Exchange](#)
- [Function](#)
- [MeasurableElement](#)
- [Organization](#)
- [OrganizationalResource](#)
- [Person](#)
- [PhysicalResource](#)
- [Post](#)
- Rationale
- [ResourceConnector](#)
- [ResourceExchange](#)
- [ResourceExchangeItem](#)
- [ResourceInteractionKind](#)
- [ResourceInterface](#)
- [ResourceMessage](#)
- [ResourceMethod](#)
- [ResourcePerformer](#)
- [ResourcePort](#)
- [ResourceRole](#)
- [Responsibility](#)

### View Specifications::Personnel::Processes

Stakeholders: Systems engineers, Solution providers

Concerns: functions that have to be carried out by organizational resources

Definition: specifies organizational resource functions in relation to resource definitions.

Recommended Implementation: SysML Activity Diagram, SysML Block Definition Diagram, BPMN Process Diagram

**Commented [Y103]:** UAF11-48 replaced text  
ResourceExchangeKind with ResourceInteractionKind

**Deleted:** [ResourceInteractionKind](#)

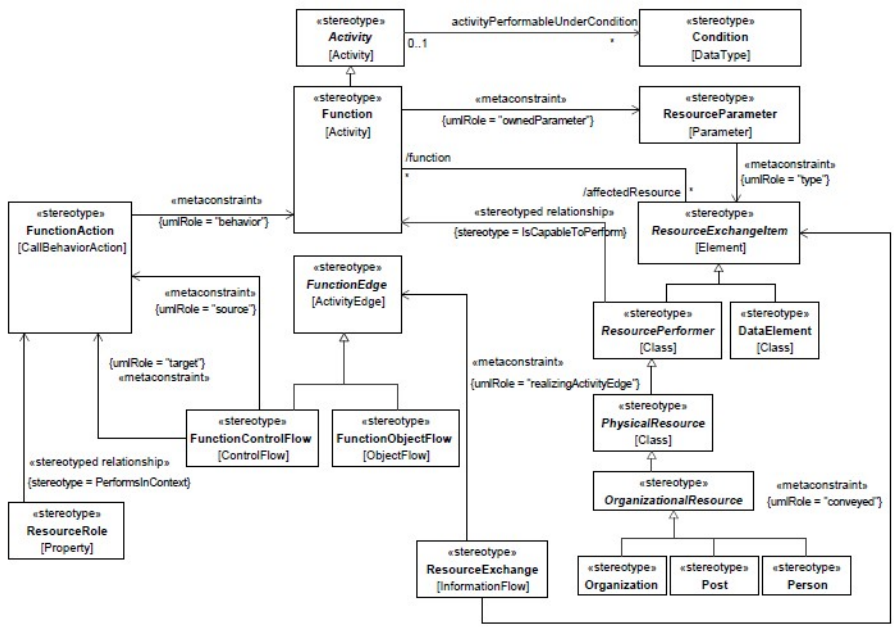


Figure A.29 - Personnel Processes

Elements

- [Activity](#)
- [Condition](#)
- [DataElement](#)
- [Function](#)
- [FunctionAction](#)
- [FunctionControlFlow](#)
- [FunctionEdge](#)
- [FunctionObjectFlow](#)
- [Organization](#)
- [OrganizationalResource](#)
- [Person](#)

- [PhysicalResource](#)
- [Post](#)
- [ResourceExchange](#)
- [ResourceExchangeItem](#)
- [ResourceParameter](#)
- [ResourcePerformer](#)
- [ResourceRole](#)

**View Specifications::Personnel::States**

Stakeholders: Systems Engineers, Software Engineers

Concerns: capture state-based behavior of an organizational resource

Definition: it is a graphical representation of states of an organizational resource and how that organizational resource responds to various events and actions.

Recommended Implementation: SysML State **Machine** Diagram

Commented [AM104]: [UAF11-93](#) "SysML State Diagram" changed to "SysML State Machine Diagram".

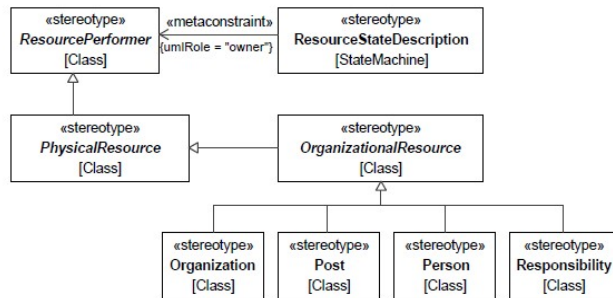


Figure A.30 - Personnel States

Elements

- [Organization](#)
- [OrganizationalResource](#)
- [Person](#)
- [PhysicalResource](#)
- [Post](#)
- [ResourcePerformer](#)
- [ResourceStateDescription](#)
- [Responsibility](#)

**View Specifications::Personnel::Interaction Scenarios**

Stakeholders: Software Engineers, Systems Engineers  
 Concerns: interactions between organizational resources (roles)  
 Definition: provides a time-ordered examination of the interactions between organizational resources.  
 Recommended Implementation: SysML Sequence Diagram, BPMN Collaboration Diagram

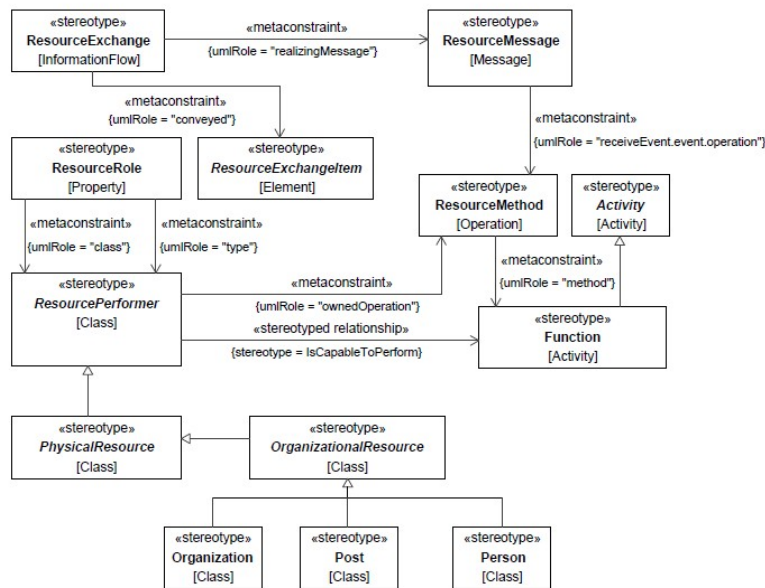


Figure A.31 - Personnel Interaction Scenarios

Elements

- [Activity](#)
- [Function](#)
- [Organization](#)
- [OrganizationalResource](#)
- [Person](#)
- [PhysicalResource](#)
- [Post](#)
- [ResourceExchange](#)

- [ResourceExchangeItem](#)
- [ResourceMessage](#)
- [ResourceMethod](#)
- [ResourcePerformer](#)
- [ResourceRole](#)

**View Specifications::Personnel::Constraints**

Stakeholders: Systems engineers, Solution providers

Concerns: allocation of competencies to actual posts

Definition: specifies requirements for actual organizational resources – by linking competencies and actual posts.

Recommended Implementation: SysML Block Definition Diagram

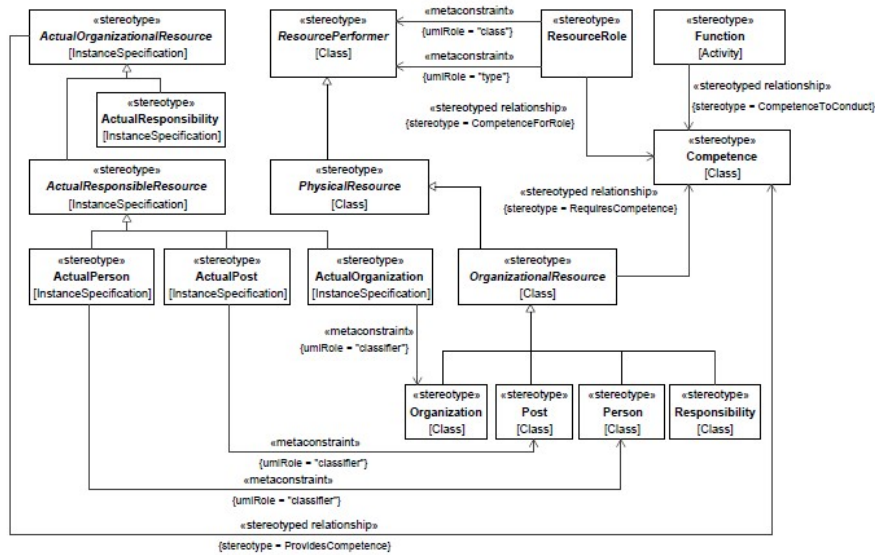


Figure A.32 - Personnel Constraints: Competence

Elements

- [ActualOrganization](#)
- [ActualOrganizationalResource](#)
- [ActualPerson](#)
- [ActualPost](#)

- [ActualResponsibility](#)
- [ActualResponsibleResource](#)
- [Competence](#)
- [Function](#)
- [Organization](#)
- [OrganizationalResource](#)
- [Person](#)
- [PhysicalResource](#)
- [Post](#)
- [ResourcePerformer](#)
- [ResourceRole](#)
- [Responsibility](#)

Stakeholders: Systems engineers, Solution providers, Human resources  
 Concerns: optimization of organizational resource behavior  
 Definition: captures the factors that affect, constrain and characterize organizational resource behavior as the basis for performance predictions at the level of actual persons and actual organizations. It creates a bridge between static architectural definitions and behavior predictions through executable models.  
 Recommended Implementation: tabular format, SysML Parametric Diagram, SysML Block Definition Diagram

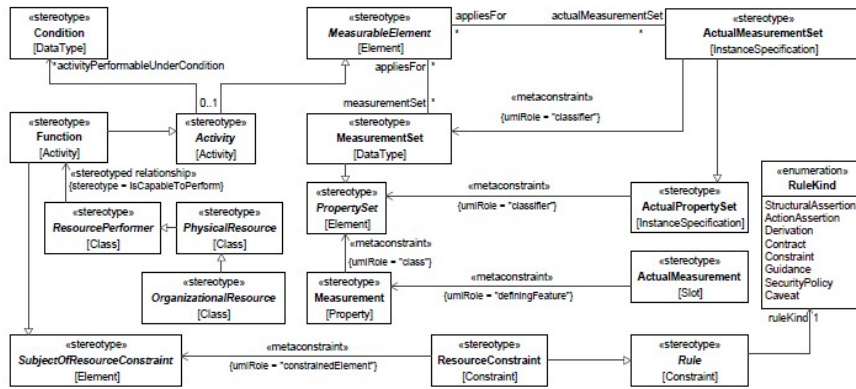


Figure A.33 - Personnel Constraints: Drivers



#### Elements

- [Activity](#)
- [ActualMeasurement](#)
- [ActualMeasurementSet](#)
- [ActualPropertySet](#)
- [Condition](#)
- [Function](#)
- [MeasurableElement](#)
- [Measurement](#)
- [MeasurementSet](#)
- [OrganizationalResource](#)
- [PhysicalResource](#)
- [PropertySet](#)
- [ResourceConstraint](#)
- [ResourcePerformer](#)
- [Rule](#)
- [RuleKind](#)
- [SubjectOfResourceConstraint](#)

Stakeholders: Human resources, solution providers

Concerns: how well an actual organizational resource matches the needs of the actual organization

Definition: provides a repository for human-related measures (i.e., quality objectives and performance criteria (HFI values)), targets and competences.

Recommended Implementation: SysML Block Definition Diagram

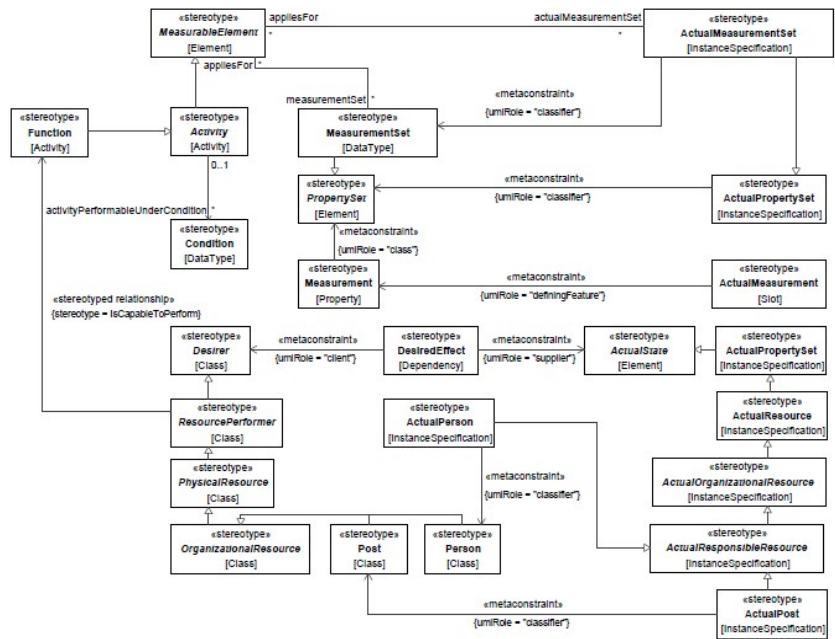


Figure A.34 - Personnel Constraints: Performance

Elements

- [Activity](#)
- [ActualMeasurement](#)
- [ActualMeasurementSet](#)
- [ActualOrganizationalResource](#)
- [ActualPerson](#)
- [ActualPost](#)
- [ActualPropertySet](#)
- [ActualResource](#)
- [ActualResponsibleResource](#)
- [ActualState](#)
- [Condition](#)

- [DesiredEffect](#)
- [Desirer](#)
- [Function](#)
- [MeasurableElement](#)
- [Measurement](#)
- [MeasurementSet](#)
- [OrganizationalResource](#)
- [Person](#)
- [PhysicalResource](#)
- [Post](#)
- [PropertySet](#)
- [ResourcePerformer](#)

**View Specifications::Personnel::Roadmap**

Stakeholders: Human Resources, Training, Logisticians, Solution Providers

Concerns: the staffing and training of resources

Definition: defines the requirements and functions to ensure that actual persons with the right competencies, and in the right numbers, are available to fulfill actual posts.

Recommended Implementation: Timeline, SysML Block Definition Diagram

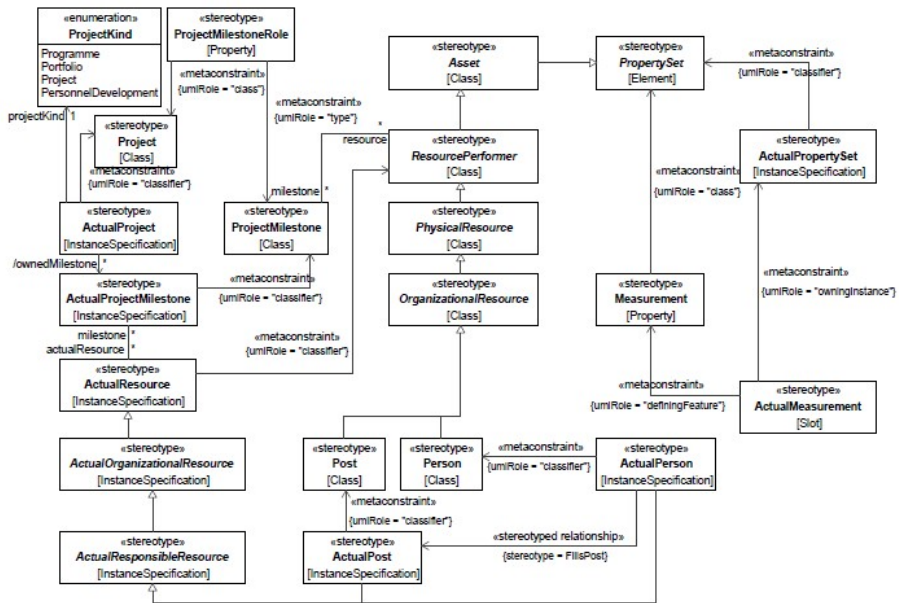


Figure A.35 - Personnel Roadmap: Availability

Elements

- [ActualMeasurement](#)
- [ActualOrganizationalResource](#)
- [ActualPerson](#)
- [ActualPost](#)
- [ActualProject](#)
- [ActualProjectMilestone](#)
- [ActualPropertySet](#)
- [ActualResource](#)
- [ActualResponsibleResource](#)
- [Asset](#)
- [Measurement](#)

- [OrganizationalResource](#)
- [Person](#)
- [PhysicalResource](#)
- [Post](#)
- [Project](#)
- [ProjectKind](#)
- [ProjectMilestone](#)
- [ProjectMilestoneRole](#)
- [PropertySet](#)
- [ResourcePerformer](#)

Stakeholders: Human resources, Solution Providers

Concerns: organizational structure changes over time

Definition: provides an overview of how a organizational structure changes over time. It shows the structure of several organizational structures mapped against a timeline.

Recommended Implementation: timeline, SysML Block Definition Diagram, SysML Internal Block Diagram

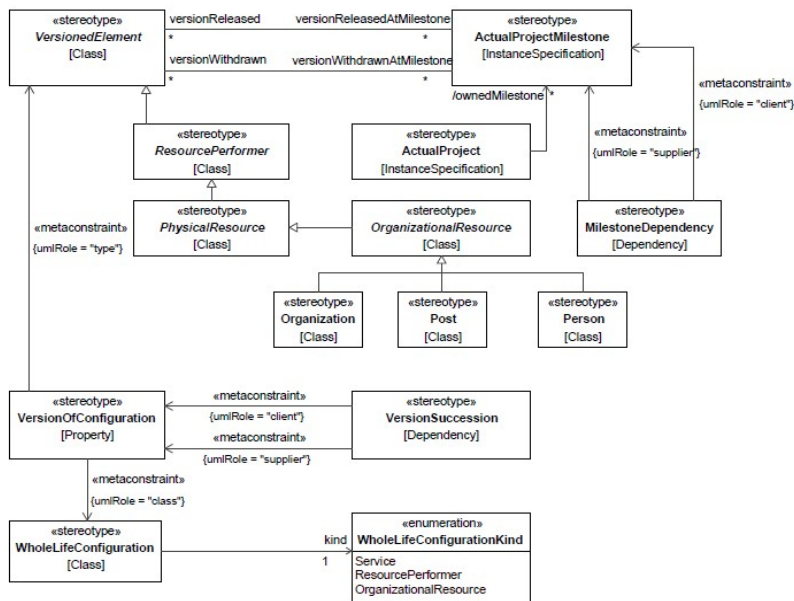


Figure A.36 - Personnel Roadmap: Evolution

Elements

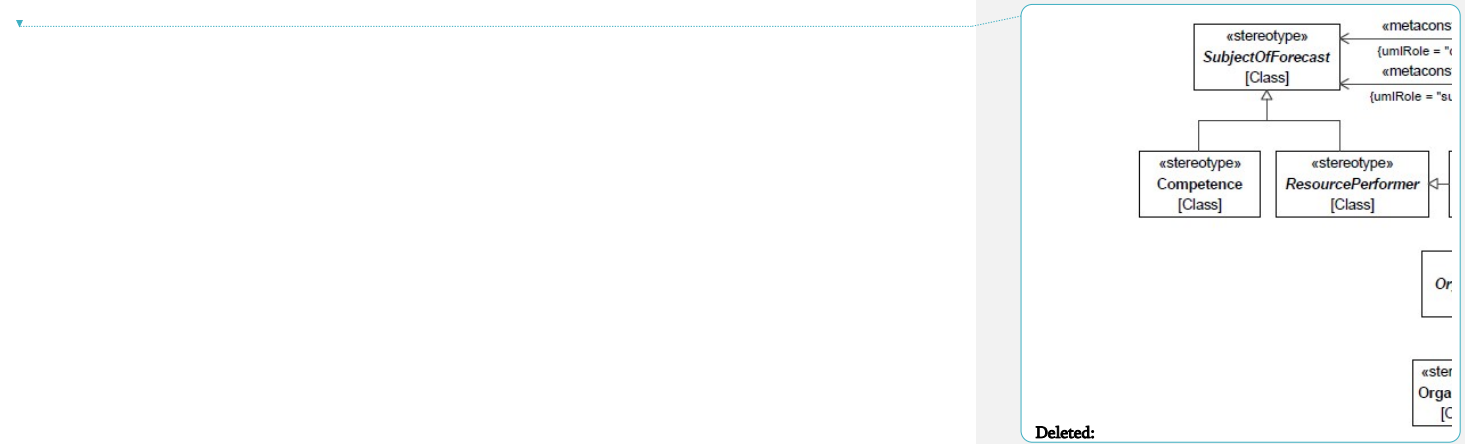
- [ActualProject](#)
- [ActualProjectMilestone](#)
- [MilestoneDependency](#)
- [Organization](#)
- [OrganizationalResource](#)
- [Person](#)
- [PhysicalResource](#)
- [Post](#)
- [ResourcePerformer](#)
- [VersionedElement](#)
- [VersionOfConfiguration](#)
- [VersionSuccession](#)
- [WholeLifeConfiguration](#)
- [WholeLifeConfigurationKind](#)

Stakeholders: Human resources, Logisticians, Solution Providers

Concerns: competencies and skills forecast

Definition: defines the underlying current and expected supporting competencies and skills of organizational resources.

Recommended Implementation: timeline, tabular format, SysML Block Definition Diagram



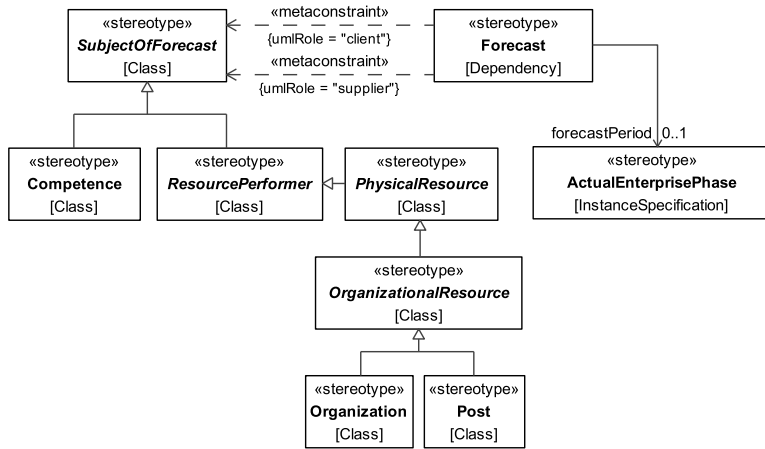


Figure A.37 - Personnel Roadmap: Forecast

Commented [Y105]: UAF11-35 replaced image with Personnel\_Roadmap\_Forecast.svg

Elements

- [ActualEnterprisePhase](#)
- [Competence](#)
- [Forecast](#)
- [Organization](#)
- [OrganizationalResource](#)
- [PhysicalResource](#)
- [Post](#)
- [ResourcePerformer](#)
- [SubjectOfForecast](#)

**View Specifications::Personnel::Traceability**

Stakeholders: Systems Engineers, Enterprise Architects, Solution Providers, Business Architects

Concerns: traceability between operational activities and functions that implements them

Definition: depicts the mapping of functions (performed by organizational resources) to operational activities and thus identifies the transformation of an operational need into a purposeful function performed by an organizational resource or solution.

Recommended Implementation: Matrix format, SysML Block Definition Diagram

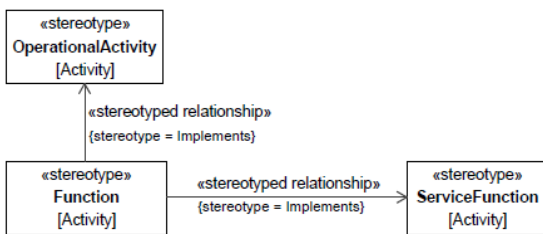


Figure A.38 - Personnel Traceability

Elements

- [Function](#)
- [OperationalActivity](#)
- [ServiceFunction](#)



## A.2.5 View Specifications::Resources

**Stakeholders:** Systems Engineers, Resource Owners, Implementers, Solution Providers, IT Architects

**Concerns:** definition of solution architectures to implement operational requirements

**Definition:** captures a solution architecture consisting of resources, e.g., organizational, software, artifacts, capability configurations, natural resources that implement the operational requirements. Further design of a resource is typically detailed in SysML or UML.

### View Specifications::Resources::Taxonomy

**Stakeholders:** Solution Providers, Systems Engineers, IT Architects, Implementers

**Concerns:** resource types

**Definition:** shows the taxonomy of types of resources.

**Recommended Implementation:** SysML Block Definition Diagram

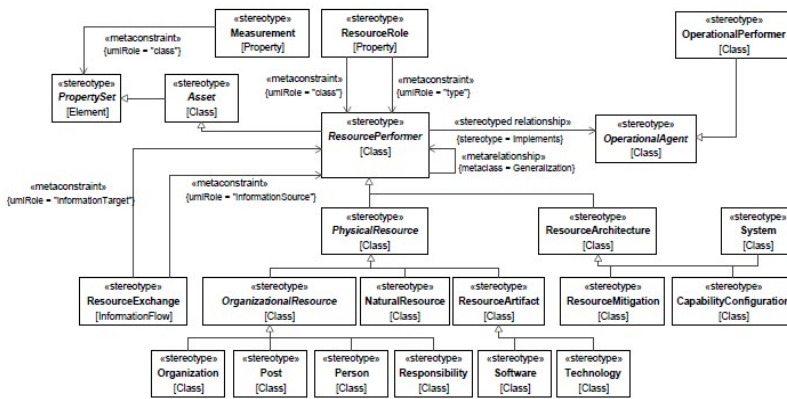


Figure A.39 - Resources Taxonomy

Elements

- [Asset](#)
- [CapabilityConfiguration](#)
- [Measurement](#)
- [NaturalResource](#)
- [OperationalAgent](#)
- [OperationalPerformer](#)
- [Organization](#)
- [OrganizationalResource](#)

- [Person](#)
- [PhysicalResource](#)
- [Post](#)
- [PropertySet](#)
- [ResourceArchitecture](#)
- [ResourceArtifact](#)
- [ResourceExchange](#)
- [ResourceMitigation](#)
- [ResourcePerformer](#)
- [ResourceRole](#)
- [Responsibility](#)
- [Software](#)
- [System](#)
- [Technology](#)

#### **View Specifications::Resources::Structure**

Stakeholders: Systems Engineers, Resource Owners, Implementers, Solution Providers

Concerns: reference the resource structure, connectors and interfaces in a specific context

Definition: defines the physical resources, e.g., capability configuration(s)/system(s) and interactions necessary to implement a specific set of OperationalPerformer(s). Can be used to represent communications networks and pathways that link communications resources and provides details regarding their configuration.

Recommended Implementation: SysML Internal Block Diagram, SysML Block Definition Diagram



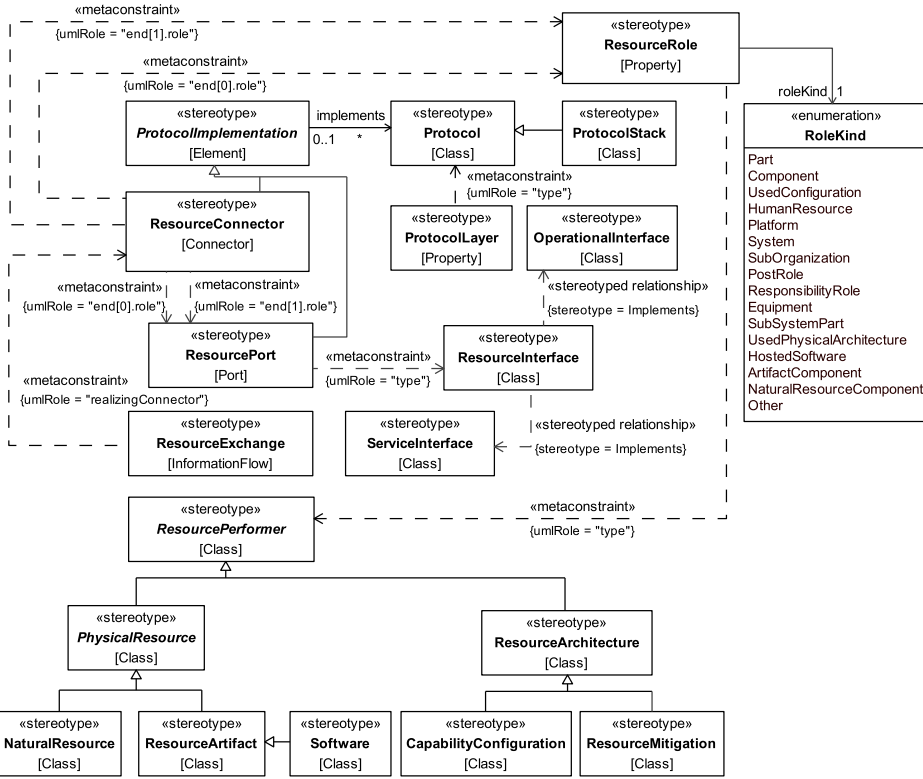
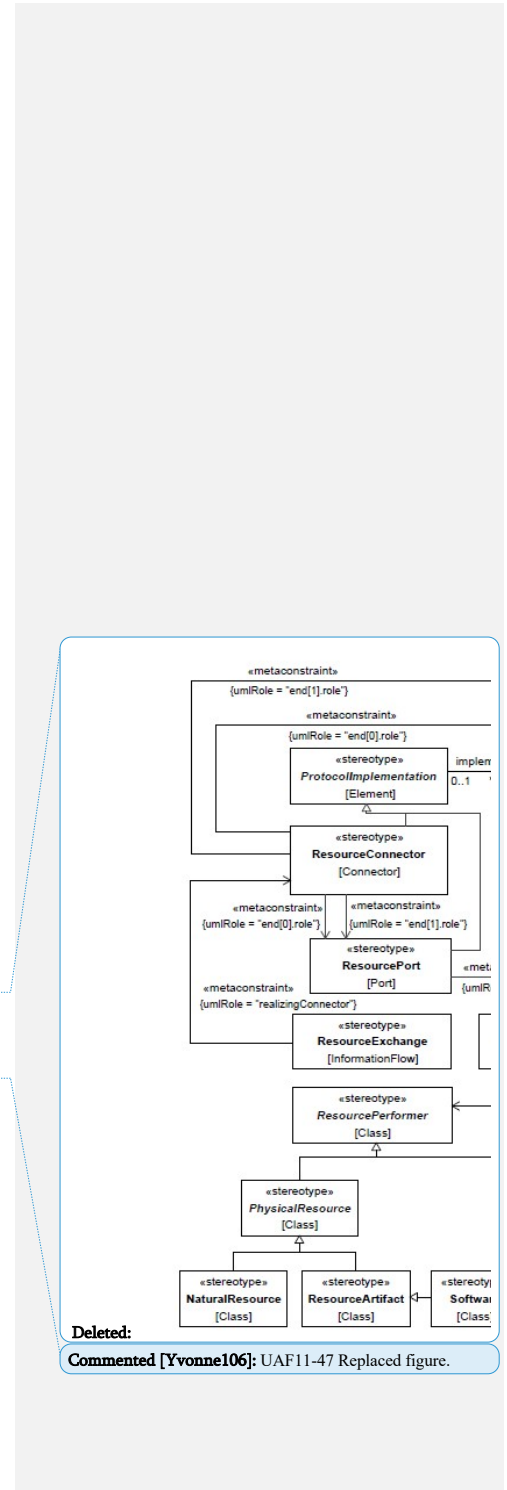


Figure A.40 - Resources Structure

Elements

- [CapabilityConfiguration](#)
- [NaturalResource](#)
- [OperationalInterface](#)
- [PhysicalResource](#)
- [Protocol](#)
- [ProtocolImplementation](#)



Deleted:

Commented [Yvonne106]: UAF11-47 Replaced figure.

- [ProtocolLayer](#)
- [ProtocolStack](#)
- [ResourceArchitecture](#)

- [ResourceArtifact](#)
- [ResourceConnector](#)
- [ResourceExchange](#)
- [ResourceInterface](#)
- [ResourceMitigation](#)
- [ResourcePerformer](#)
- [ResourcePort](#)
- [ResourceRole](#)
- [RoleKind](#)
- [ServiceInterface](#)
- [Software](#)

**View Specifications::Resources::Connectivity**

Stakeholders: Systems Engineers, IT Architects, Solution Providers, Implementers

Concerns: capture the interactions between resources

Definition: summarizes interactions between resources of information, systems, personnel, natural resources, etc. and the functions that produce and consume them. Measurements can optionally be included.

Recommended Implementation: [SysML, Internal Block Diagram](#), tabular format

**Commented [GB107]:** UAF11-87 add SysML, Internal Block Diagram



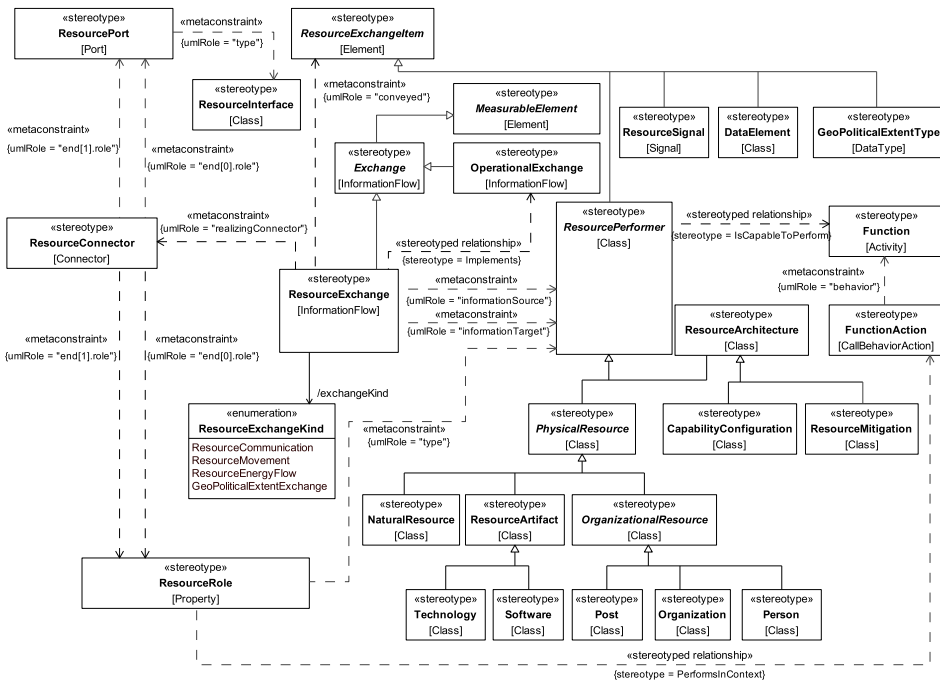


Figure A.41 - Resources Connectivity

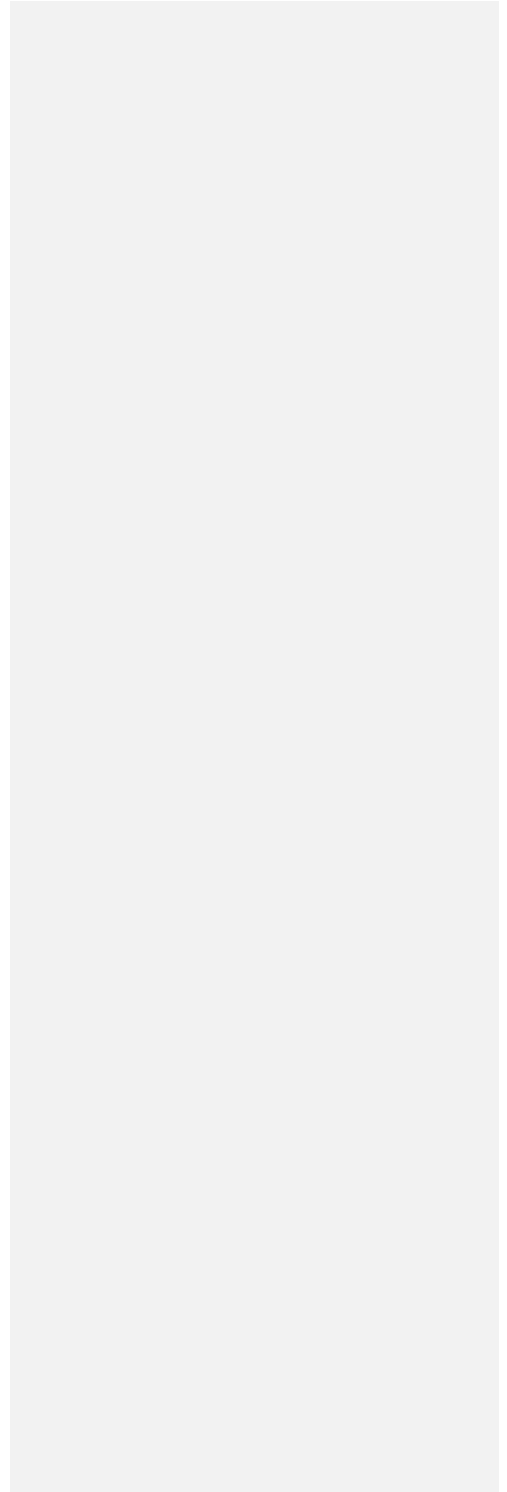
Commented [Y108]: UAF11-48 Replaced image with Resources\_Connectivity.svg

Elements

- [CapabilityConfiguration](#)
- [DataElement](#)
- [Exchange](#)
- [Function](#)
- [FunctionAction](#)
- [GeoPoliticalExtentType](#)
- [MeasurableElement](#)
- [NaturalResource](#)
- [OperationalExchange](#)
- [Organization](#)
- [OrganizationalResource](#)



- [Person](#)



- [PhysicalResource](#)
- [Post](#)
- [ResourceArchitecture](#)
- [ResourceArtifact](#)
- [ResourceConnector](#)
- [ResourceExchange](#)
- [ResourceExchangeItem](#)
- [ResourceExchangeKind](#)
- [ResourceInterface](#)
- [ResourceMitigation](#)
- [ResourcePerformer](#)
- [ResourcePort](#)
- [ResourceRole](#)
- [ResourceSignal](#)
- [Software](#)
- [Technology](#)

Deleted: [ResourceInteractionKind](#)

**View Specifications::Resources::Processes**

Stakeholders: Solution Providers, Systems Engineers, IT Architects

Concerns: captures activity based behavior and flows

Definition: describes the functions that are normally conducted in the course of implementing operational activity(ies) in support of capability(ies). It describes the functions, their Inputs/Outputs, function actions and flows between them.

Recommended Implementation: SysML Activity Diagram, SysML Block Definition Diagram

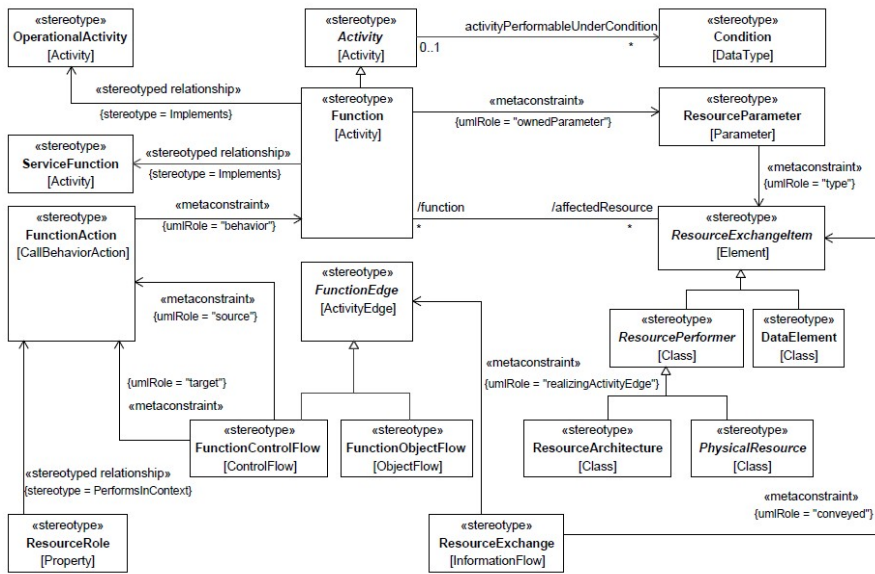


Figure A.42 - Resources Processes

Elements

- [Activity](#)
- [Condition](#)
- [DataElement](#)
- [Function](#)
- [FunctionAction](#)
- [FunctionControlFlow](#)
- [FunctionEdge](#)
- [FunctionObjectFlow](#)
- [OperationalActivity](#)
- [PhysicalResource](#)
- [ResourceArchitecture](#)
- [ResourceExchange](#)

- [ResourceExchangeItem](#)
- [ResourceParameter](#)
- [ResourcePerformer](#)
- [ResourceRole](#)
- [ServiceFunction](#)

**View Specifications::Resources::States**

Stakeholders: Systems Engineers, Software Engineers

Concerns: capture state-based behavior of a resource

Definition: it is a graphical representation of states of a resource and how that resource responds to various events and actions.

Recommended Implementation: SysML State **Machine** Diagram

Commented [AM109]: [UAF11-93](#) "SysML State Diagram" changed to "SysML State Machine Diagram".

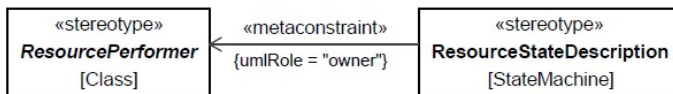


Figure A.43 - Resources States

Elements

- [ResourcePerformer](#)
- [ResourceStateDescription](#)

**View Specifications::Resources::Interaction Scenarios**

Stakeholders: Software Engineers, Systems Engineers

Concerns: interactions between resources (roles)

Definition: provides a time-ordered examination of the interactions between resources.

Recommended Implementation: SysML Sequence Diagram

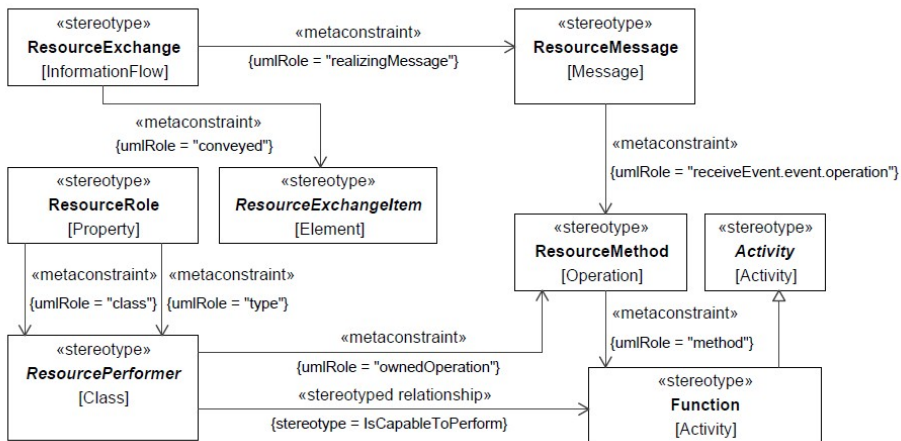


Figure A.44 - Resources Interaction Scenarios

Elements

- [Activity](#)
- [Function](#)
- [ResourceExchange](#)
- [ResourceExchangeItem](#)
- [ResourceMessage](#)
- [ResourceMethod](#)
- [ResourcePerformer](#)
- [ResourceRole](#)

**View Specifications::Resources::Constraints**

Stakeholders: Systems Engineers, IT Architects, Solution Providers, Implementers

Concerns: define limitations, constraints and performance parameters for resources, their interactions, performed functions, and data

Definition: specifies traditional textual rules/non-functional requirements that are constraints on resources, their interactions, performed functions, and data. The addition of SysML parametrics provide a computational means of defining resource constraints within a specific context.

Recommended Implementation: tabular format, SysML Block Definition Diagram, SysML Parametric Diagram, OCL

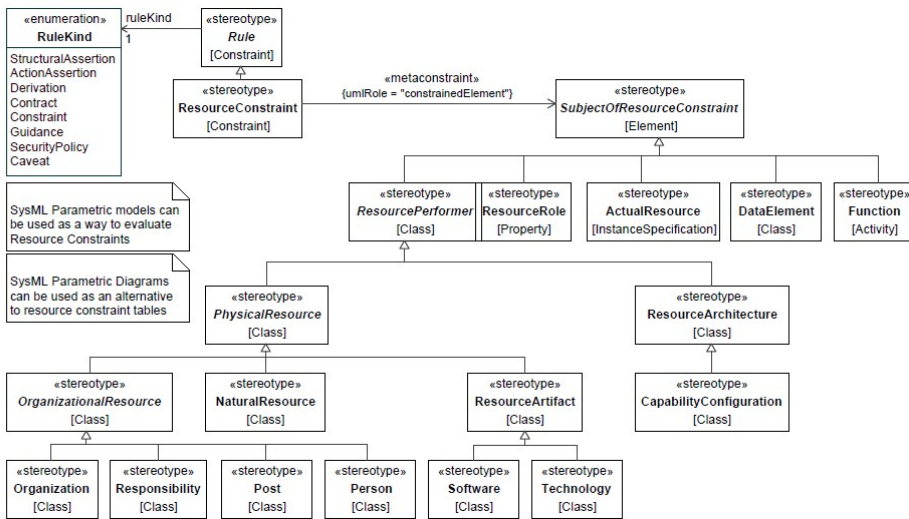


Figure A.45 - Resources Constraints

Elements

- [ActualResource](#)
- [CapabilityConfiguration](#)
- [DataElement](#)
- [Function](#)
- [NaturalResource](#)
- [Organization](#)
- [OrganizationalResource](#)
- [Person](#)
- [PhysicalResource](#)
- [Post](#)
- [ResourceArchitecture](#)
- [ResourceArtifact](#)
- [ResourceConstraint](#)

- [ResourcePerformer](#)
- [ResourceRole](#)
- [Responsibility](#)
- [Rule](#)
- [RuleKind](#)
- [Software](#)
- [SubjectOfResourceConstraint](#)
- [Technology](#)

**View Specifications::Resources::Roadmap**

Stakeholders: Systems Engineers, IT Architects, Solution Providers, Implements

Concerns: resource structure changes over time

Definition: provides an overview of how a resource structure changes over time. It shows the structure of several resources mapped against a timeline.

Recommended Implementation: timeline, SysML Block Definition Diagram, SysML Internal Block Diagram

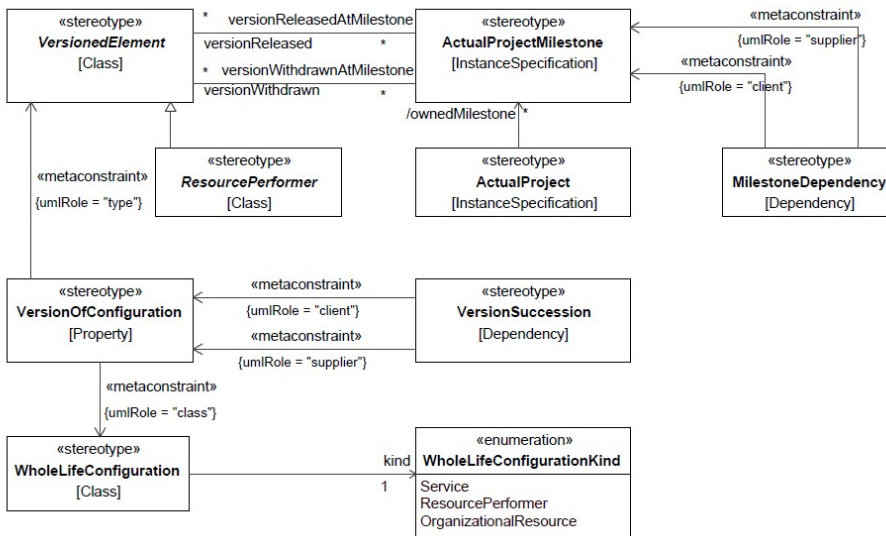


Figure A.46 - Resources Roadmap: Evolution

Elements

- [ActualProject](#)





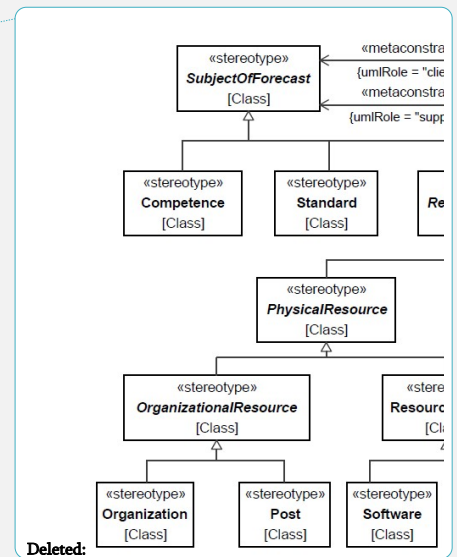
- [ActualProjectMilestone](#)
- [MilestoneDependency](#)
- [ResourcePerformer](#)
- [VersionedElement](#)
- [VersionOfConfiguration](#)
- [VersionSuccession](#)
- [WholeLifeConfiguration](#)
- [WholeLifeConfigurationKind](#)

Stakeholders: Solution Providers, Systems Engineers, IT Architects

Concerns: technology forecast

Definition: defines the underlying current and expected supporting technologies. Expected supporting technologies are those that can be reasonably forecast given the current state of technology, and expected improvements / trends.

Recommended Implementation: timeline, tabular format, SysML Block Definition Diagram



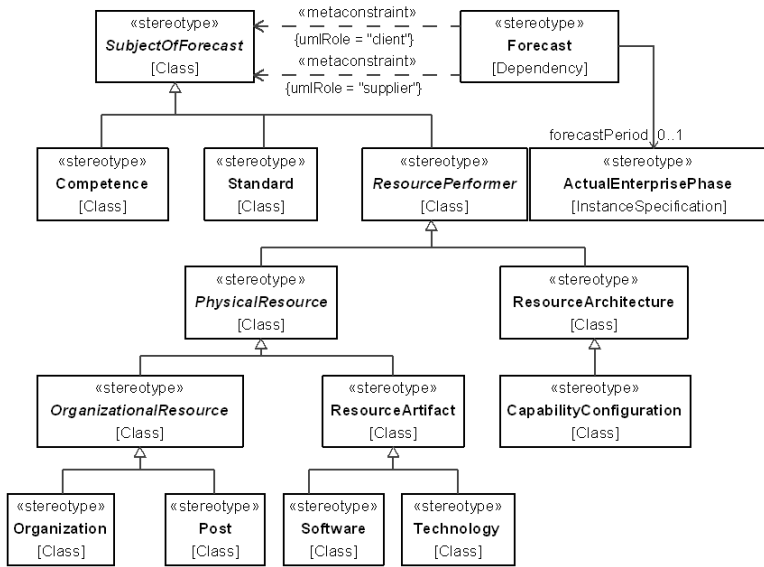


Figure A.47 - Resources Roadmap: Forecast

Deleted: ¶

Commented [Y110]: UAF11-35 replace image with Resources\_Roadmap\_Forecast.svg

Elements

- [ActualEnterprisePhase](#)
- [CapabilityConfiguration](#)

- [Competence](#)
- [Forecast](#)
- [Organization](#)
- [OrganizationalResource](#)
- [PhysicalResource](#)
- [Post](#)
- [ResourceArchitecture](#)
- [ResourceArtifact](#)
- [ResourcePerformer](#)
- [Software](#)
- [Standard](#)
- [SubjectOfForecast](#)
- [Technology](#)

**View Specifications::Resources::Traceability**

Stakeholders: Systems Engineers, Enterprise Architects, Solution Providers, Business Architects  
 Concerns: traceability between operational activities and functions that implements them  
 Definition: depicts the mapping of functions to operational activities and thus identifies the transformation of an operational need into a purposeful function performed by a resource or solution.  
 Recommended Implementation: Matrix format, SysML Block Definition Diagram

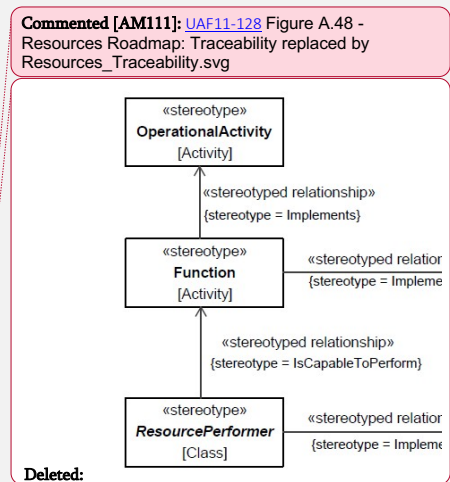
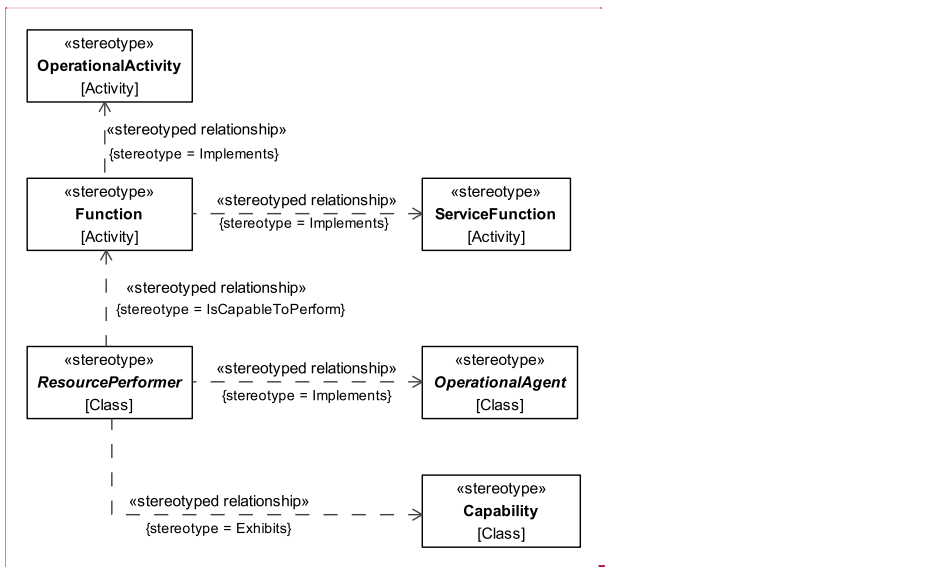


Figure A.48 - Resources Roadmap: Traceability

Elements

- Capability
- Function

**Commented [AM112]:** [UAF11-128](#) Capability added to the elements list.

**Formatted:** OMG Bold Bullet KeepNext Paragraph, Right: 0 cm, Bulleted + Level: 1 + Aligned at: 0.63 cm + Indent at: 1.27 cm, Tab stops: Not at 0.81 cm

**Formatted:** Font: Bold

- [OperationalActivity](#)
- [OperationalAgent](#)
- [ResourcePerformer](#)
- [ServiceFunction](#)

## A.2.6 View Specifications::Security

### View Specifications::Security::Taxonomy

Concerns: Security assets and security enclaves.

Definition: Defines the hierarchy of security assets and asset owners that are available to implement security, security constraints (policy, guidance, laws and regulations) and details where they are located (security enclaves)

Recommended Implementation: SysML Internal Block Diagram, SysML Block Definition Diagram

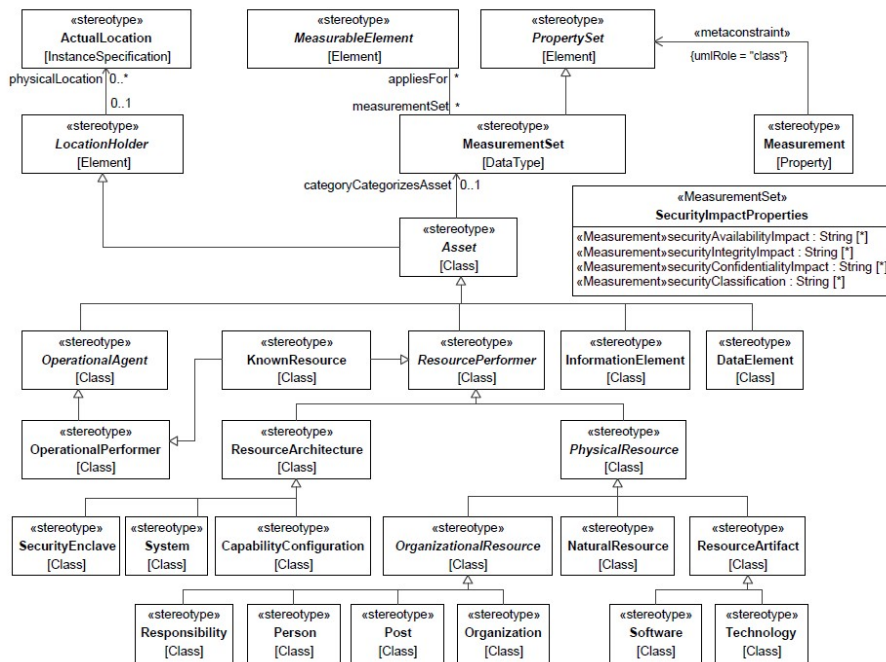


Figure A.49 - Security Taxonomy

## Elements

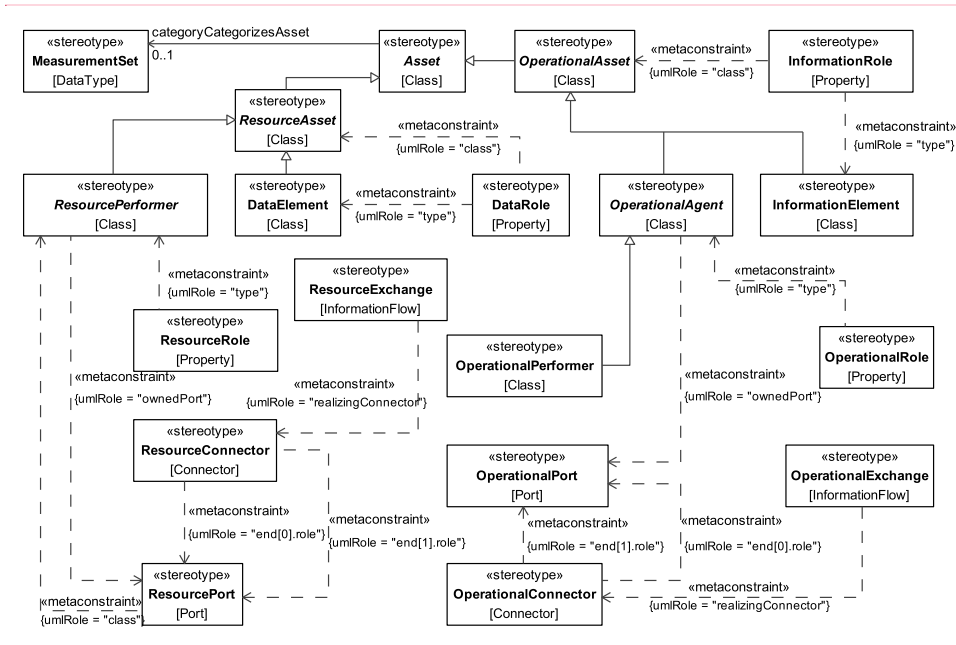
- [ActualLocation](#)
- [Asset](#)
- [CapabilityConfiguration](#)
- [DataElement](#)
- [InformationElement](#)
- [KnownResource](#)
- [LocationHolder](#)
- [MeasurableElement](#)
- [Measurement](#)
- [MeasurementSet](#)
- [NaturalResource](#)
- [OperationalAgent](#)
- [OperationalPerformer](#)
- [Organization](#)
- [OrganizationalResource](#)
- [Person](#)
- [PhysicalResource](#)
- [Post](#)
- [PropertySet](#)
- [ResourceArchitecture](#)
- [ResourceArtifact](#)
- [ResourcePerformer](#)
- [Responsibility](#)
- [SecurityEnclave](#)
- SecurityImpactProperties
- [Software](#)
- [System](#)
- [Technology](#)

### **View Specifications::Security::Structure**

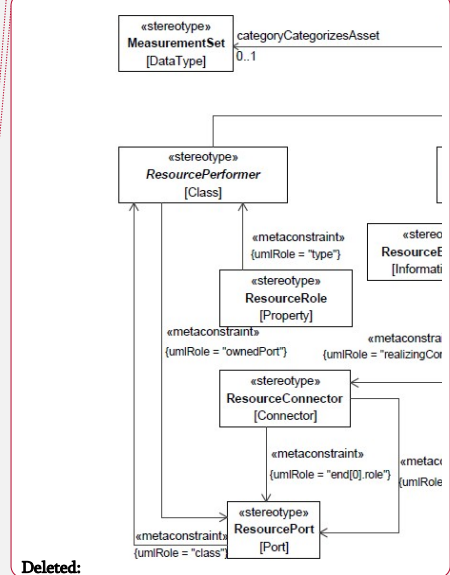
Concerns: The structure of security information and where it is used at the operational and resource level

Definition: Captures the allocation of assets (operational and resource, information and data) across the security enclaves, shows applicable security controls necessary to protect organizations, systems and information during processing, while in storage (bdd), and during transmission (flows on an ibd). This view also captures Asset Aggregation and allocates the usage of the aggregated information at a location through the use of the SecurityProperty.

Recommended Implementation: SysML Internal Block Diagram, SysML Block Definition Diagram



Commented [AM113]: UAF11-16 Figure A.50 - Security Structure replaced by Security\_Structure.svg



Deleted:

Figure A.50 - Security Structure

Elements

- [Asset](#)
- [DataElement](#)
- [DataRole](#)
- [InformationElement](#)
- [InformationRole](#)
- [MeasurementSet](#)
- [OperationalAgent](#)
- [OperationalAsset](#)
- [OperationalConnector](#)
- [OperationalExchange](#)
- [OperationalPerformer](#)
- [OperationalPort](#)
- [OperationalRole](#)
- [ResourceAsset](#)
- [ResourceConnector](#)
- [ResourceExchange](#)
- [ResourcePerformer](#)

Commented [AM114]: UAF11-16 Elements list updated by adding InformationRole, DataRole and removing SecurityProperty

- [ResourcePort](#)
- [ResourceRole](#)

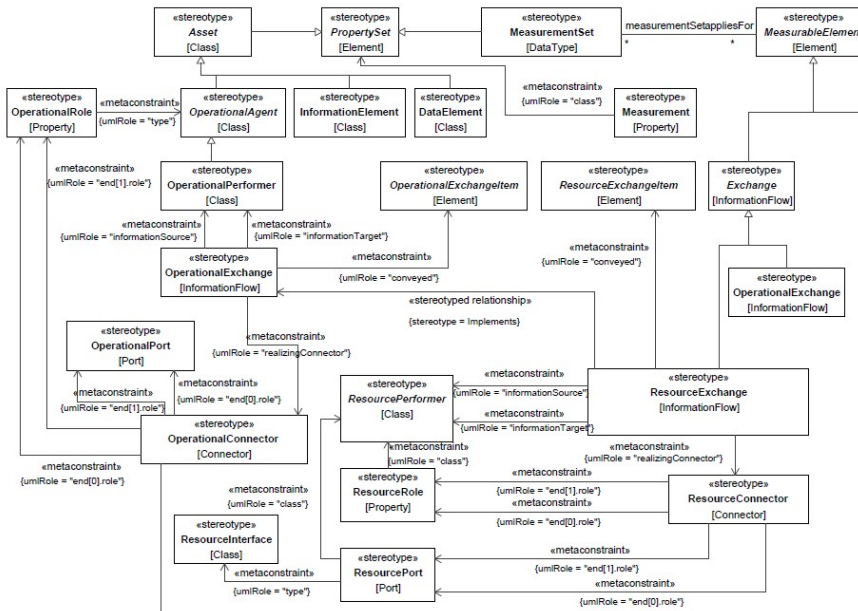
### View Specifications::Security::Connectivity

Stakeholders: Security Architects, Security Engineers

Concerns: Addresses the security constraints and information assurance attributes that exist on exchanges across resources and across performers.

Definition: Lists security exchanges across security assets; the applicable security controls; and the security enclaves that house the producers and consumers of the exchanges. Measurements can optionally be included.

Recommended Implementation: [SysML Internal Block Diagram](#), [tabular](#) format



- Deleted:**
- [Asset](#)
  - [DataElement](#)
  - [InformationElement](#)
  - [MeasurementSet](#)
  - [OperationalAgent](#)
  - [OperationalConnector](#)
  - [OperationalExchange](#)
  - [OperationalPerformer](#)
  - [OperationalPort](#)
  - [OperationalRole](#)
  - [ResourceConnector](#)
  - [ResourceExchange](#)
  - [ResourcePerformer](#)
  - [ResourcePort](#)
  - [ResourceRole](#)
  - [SecurityProperty](#)
- Commented [GB115]:** UAF11-87 Add SysML Internal Block Diagram

Figure A.51 - Security Connectivity

#### Elements

- [Asset](#)
- [DataElement](#)



- [Exchange](#)
- [InformationElement](#)
- [MeasurableElement](#)
- [Measurement](#)
- [MeasurementSet](#)
- [OperationalAgent](#)
- [OperationalConnector](#)
- [OperationalExchange](#)
- [OperationalExchangeItem](#)
- [OperationalPerformer](#)
- [OperationalPort](#)
- [OperationalRole](#)
- [PropertySet](#)
- [ResourceConnector](#)
- [ResourceExchange](#)
- [ResourceExchangeItem](#)
- [ResourceInterface](#)
- [ResourcePerformer](#)
- [ResourcePort](#)
- [ResourceRole](#)

#### **View Specifications::Security::Processes**

Stakeholders: Security Architects, Security Engineers

Concerns: The specification of the Security Control families, security controls, and measures required to address a specific security baseline.

Definition: Provides a set of Security Controls and any possible enhancements as applicable to assets. The activity diagram describes operational or resource level processes that apply (operational level) or implement (resource level) security controls/enhancements to assets located in enclaves and across enclaves. This Security Process view can be instantiated either as a variant of an activity/flow diagram or as a hierarchical work breakdown structure.

Recommended Implementation: SysML Activity Diagram, SysML Block Definition Diagram





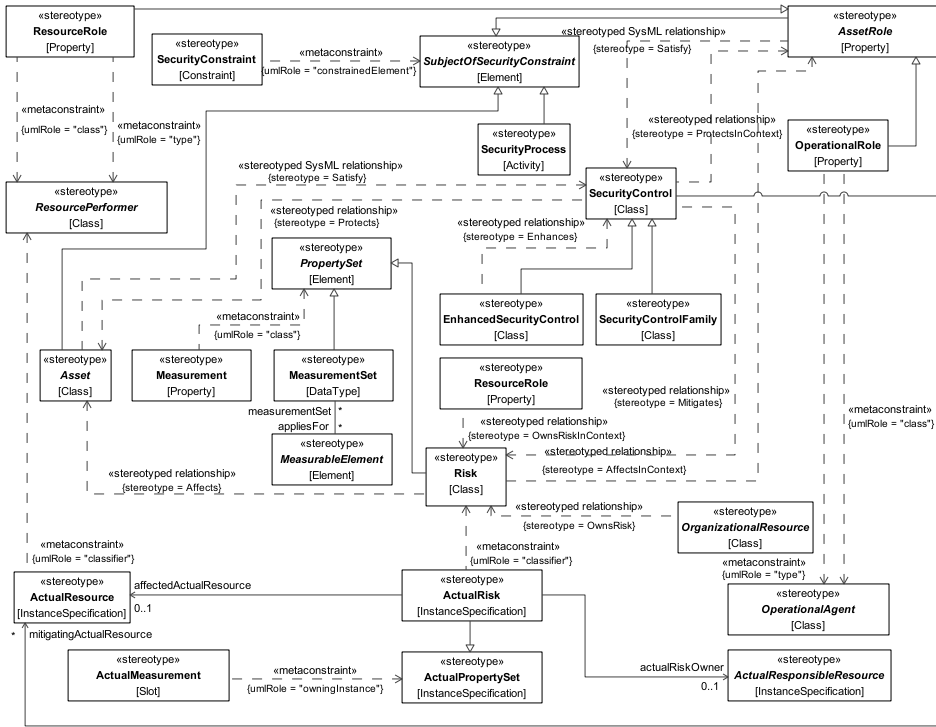


Figure A.53 - Security Constraints

Commented [Y118]: UAF11-59 replaced image with Security\_Constraints.svg

Elements

- [ActualMeasurement](#)
- [ActualPropertySet](#)
- [ActualResource](#)

- [ActualResponsibleResource](#)
  - [ActualRisk](#)
  - [Asset](#)
  - [AssetRole](#)
  - [EnhancedSecurityControl](#)
  - [MeasurableElement](#)
  - [Measurement](#)
  - [MeasurementSet](#)
  - [OperationalAgent](#)
  - [OperationalRole](#)
  - [OrganizationalResource](#)
  - [PropertySet](#)
  - [ResourcePerformer](#)
  - [ResourceRole](#)
  - [Risk](#)
  - [SecurityConstraint](#)
  - [SecurityControl](#)
  - [SecurityControlFamily](#)
  - [SecurityProcess](#)
- 
- [SubjectOfSecurityConstraint](#)

**Commented [Y119]:** UAF11-59 Added security process as a subtype of Subject of Security Constraint.

**View Specifications::Security::Traceability**

Stakeholders: Security Architects, Security Engineers, Risk Analysts

Concerns: traceability between risk and risk owner, risk mitigations, and affected asset roles

Definition: depicts the mapping of a risk to each of the following: risk owner, risk mitigations, and affected asset roles.

Recommended Implementation: Matrix format, SysML Block Definition Diagram

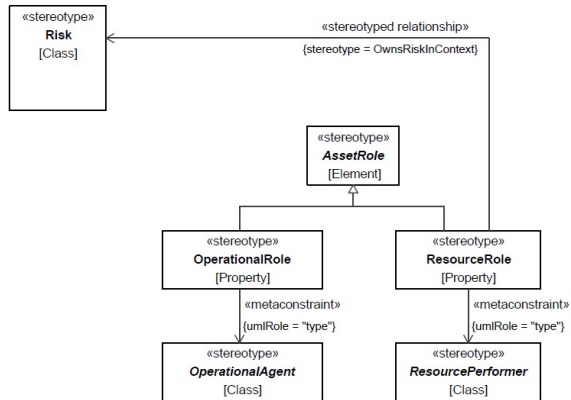


Figure A.54 - Security Traceability

Elements

- [AssetRole](#)
- [OperationalAgent](#)
- [OperationalRole](#)
- [ResourcePerformer](#)
- [ResourceRole](#)
- [Risk](#)

**A.2.7 View Specifications::Projects**

**Stakeholders:** PMs, Project Portfolio Managers, Enterprise Architects

**Concerns:** project portfolio, projects and project milestones

**Definition:** describes projects and project milestones, how those projects deliver capabilities, the organizations contributing to the projects and dependencies between projects.

**View Specifications::Projects::Taxonomy**

Stakeholders: PMs, Project Portfolio Managers, Enterprise Architects

Concerns: types of projects and project milestones

Definition: shows the taxonomy of types of projects and project milestones

Recommended Implementation: SysML Block Definition Diagram



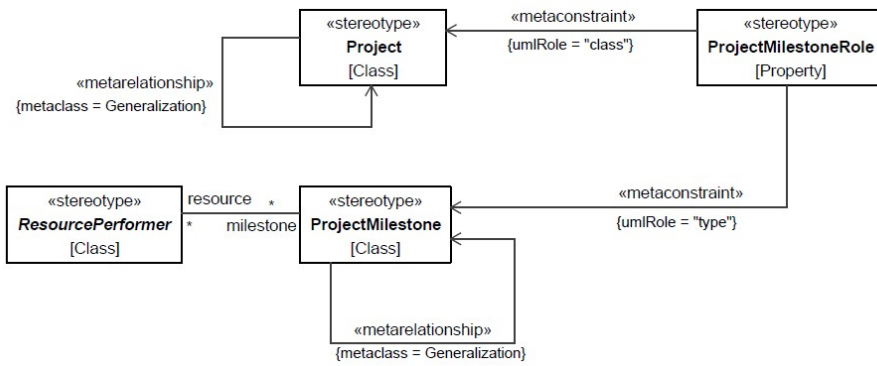


Figure A.55 - Project Taxonomy

Elements

- [Project](#)
- [ProjectMilestone](#)
- [ProjectMilestoneRole](#)
- [ResourcePerformer](#)

**View Specifications::Projects::Structure**

Stakeholders: PMs

Concerns: relationships between types of projects and project milestones

Definition: provides a template for an actual project(s) road map(s) to be implemented

Recommended Implementation: SysML Block Definition Diagram



Deleted: <object>



**View Specifications::Projects::Connectivity**

Stakeholders: PMs

Concerns: relationships between projects and project milestones

Definition: shows how projects and project milestones are related in sequence.

Recommended Implementation: SysML Block Definition Diagram

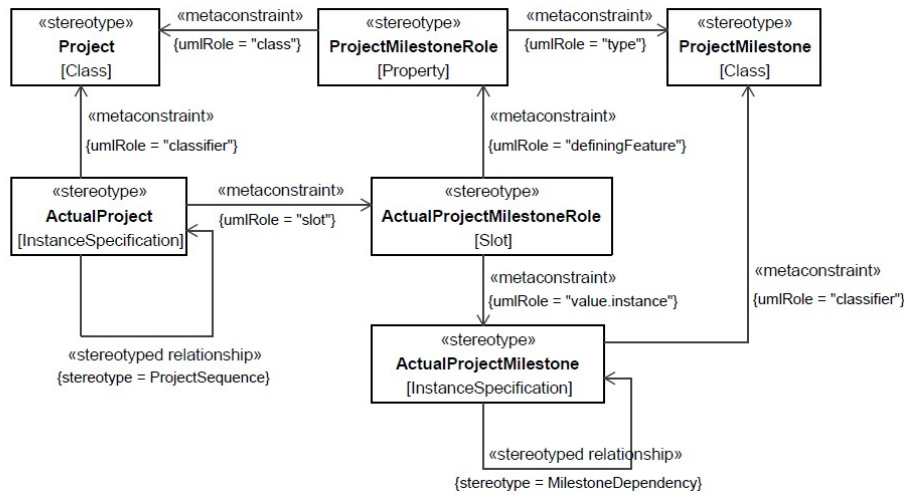


Figure A.57 - Project Connectivity

Elements

- [ActualProject](#)
- [ActualProjectMilestone](#)
- [ActualProjectMilestoneRole](#)
- [Project](#)
- [ProjectMilestone](#)
- [ProjectMilestoneRole](#)

**View Specifications::Projects::Processes**

Stakeholders: PMs

Concerns: captures project tasks (ProjectActivities) and flows between them

Definition: describes the ProjectActivities that are normally conducted in the course of projects to support capability(ies) and implement resources. It describes the ProjectActivities, their Inputs/Outputs, ProjectActivityActions and flows between them.

Recommended Implementation: SysML Activity Diagram, SysML Block Definition Diagram

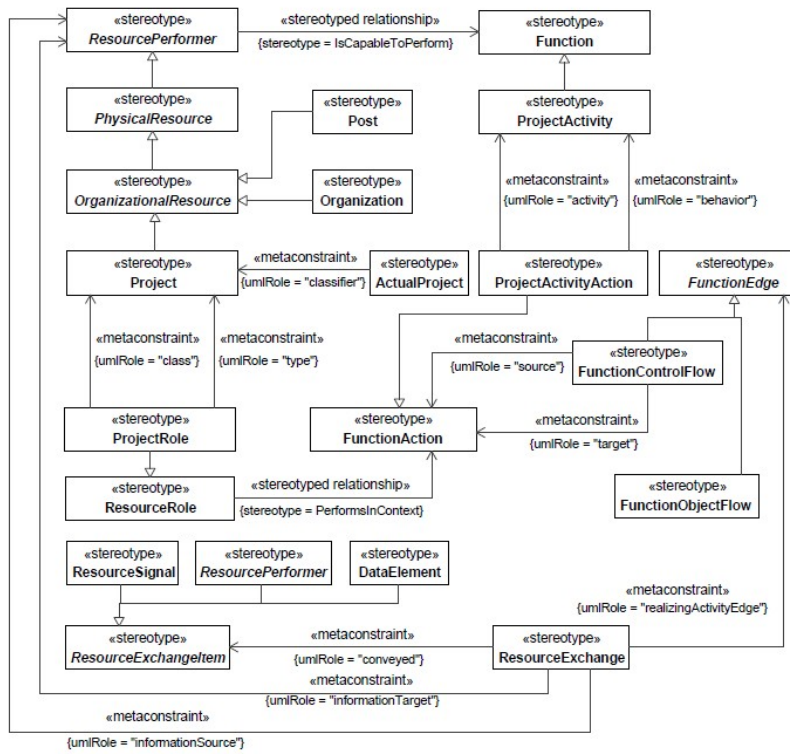


Figure A.58 - Project Processes

Elements

- [ActualProject](#)
- [DataElement](#)
- [Function](#)
- [FunctionAction](#)
- [FunctionControlFlow](#)
- [FunctionEdge](#)

- [FunctionObjectFlow](#)
- [Organization](#)
- [OrganizationalResource](#)
- [PhysicalResource](#)
- [Post](#)
- [Project](#)
- [ProjectActivity](#)
- [ProjectActivityAction](#)
- [ProjectRole](#)
- [ResourceExchange](#)
- [ResourceExchangeItem](#)
- [ResourcePerformer](#)
- [ResourceRole](#)
- [ResourceSignal](#)

**View Specifications::Projects::Roadmap**

Stakeholders: PMs, Capability Owners, Solution Providers, Enterprise Architects

Concerns: the product portfolio management; a planning of capability delivery

Definition: provides a timeline perspective on programs or projects.

Recommended Implementation: timeline, tabular format, SysML Block Definition Diagram

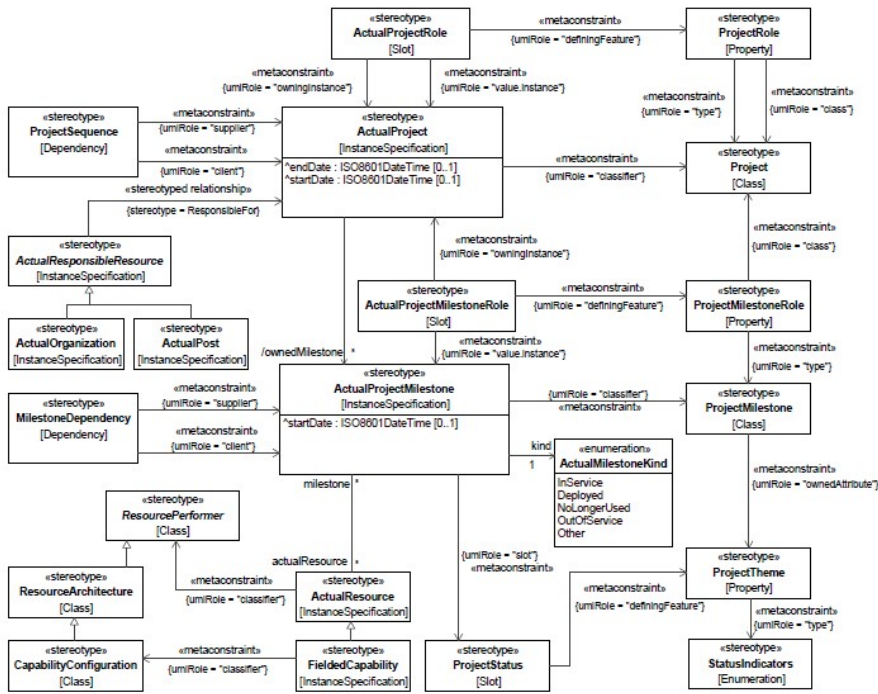


Figure A.59 - Project Roadmap

Elements

- [ActualMilestoneKind](#)
- [ActualOrganization](#)
- [ActualPost](#)
- [ActualProject](#)
- [ActualProjectMilestone](#)
- [ActualProjectMilestoneRole](#)
- [ActualProjectRole](#)
- [ActualResource](#)
- [ActualResponsibleResource](#)
- [CapabilityConfiguration](#)

- [FieldedCapability](#)
- [MilestoneDependency](#)
- [Project](#)
- [ProjectMilestone](#)
- [ProjectMilestoneRole](#)
- [ProjectRole](#)
- [ProjectSequence](#)
- [ProjectStatus](#)
- [ProjectTheme](#)
- [ResourceArchitecture](#)
- [ResourcePerformer](#)
- [StatusIndicators](#)

**View Specifications::Projects::Traceability**

Stakeholders: PMs, Project Portfolio Managers, Enterprise Architects

Concerns: traceability between capabilities and projects that deliver them

Definition: depicts the mapping of projects to capabilities and thus identifies the transformation of a capability(ies) into a purposeful implementation via projects.

Recommended Implementation: Matrix format, SysML Block Definition Diagram

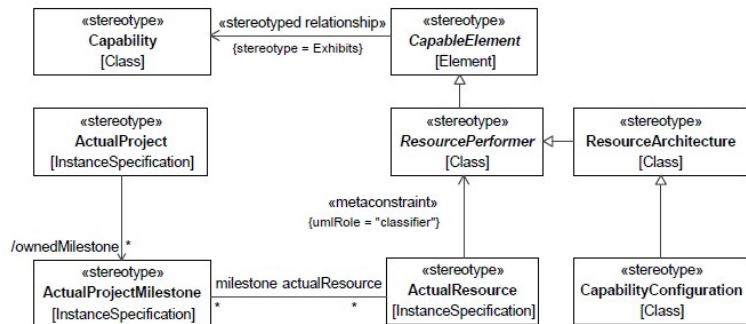


Figure A.60 - Project Traceability

Elements

- [ActualProject](#)
- [ActualProjectMilestone](#)
- [ActualResource](#)
- [Capability](#)
- [CapabilityConfiguration](#)
- [CapableElement](#)
- [ResourceArchitecture](#)
- [ResourcePerformer](#)

**A.2.8 View Specifications::Standards**

**Stakeholders:** Solution Providers, Systems Engineers, Software Engineers, Systems Architects, Business Architects

**Concerns:** technical and non-technical Standards applicable to the architecture

**Definition:** shows the technical, operational, and business Standards applicable to the architecture. Defines the underlying current and expected Standards.

**View Specifications::Standards::Taxonomy**

Stakeholders: Solution Providers, Systems Engineers, Software Engineers, Systems Architects, Business Architects

Concerns: technical and non-technical standards, guidance and policy applicable to the architecture

Definition: shows the taxonomy of types of technical, operational, and business standards, guidance and policy applicable to the architecture.

Recommended Implementation: SysML Block Definition Diagram



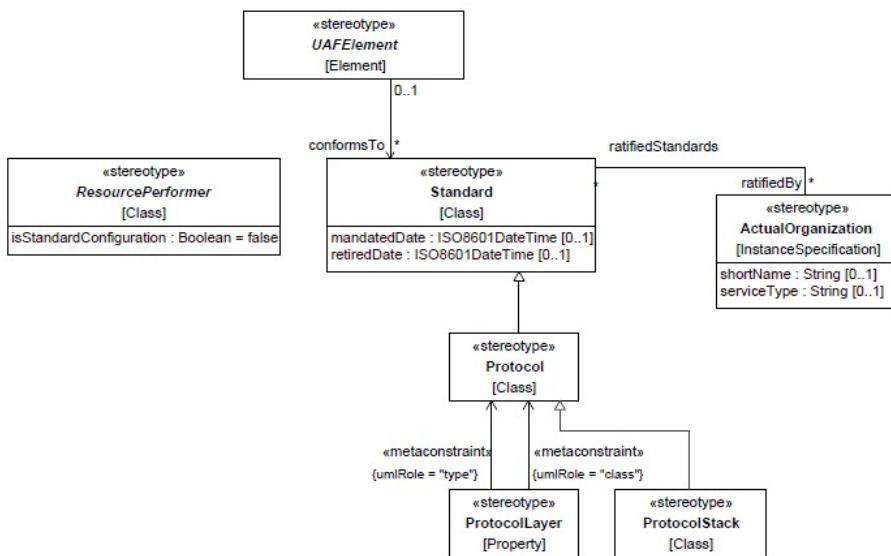


Figure A.61 - Standards Taxonomy

Elements

- [ActualOrganization](#)
- [Protocol](#)
- [ProtocolLayer](#)
- [ProtocolStack](#)
- [ResourcePerformer](#)
- [Standard](#)
- [UAFElement](#)

**View Specifications::Standards::Structure**

Stakeholders: Solution Providers, Systems Engineers, Software Engineers, Systems Architects

Concerns: the specification of the protocol stack used in the architecture

Definition: shows the composition of standards required to achieve the architecture's objectives.

Recommended Implementation: SysML Internal Block Diagram

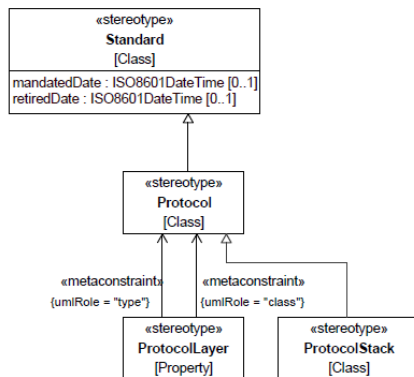


Figure A.62 - Standards Structure

Elements

- [Protocol](#)
- [ProtocolLayer](#)
- [ProtocolStack](#)
- [Standard](#)

**View Specifications::Standards::Roadmap**

Stakeholders: Solution Providers, Systems Engineers, Systems Architects, Software Engineers, Business Architects  
 Concerns: expected changes in technology-related standards and conventions, operational standards, or business standards and conventions  
 Definition: defines the underlying current and expected standards. Expected standards are those that can be reasonably forecast given the current state of technology, and expected improvements / trends.  
 Recommended Implementation: timeline, tabular format, SysML Block Definition Diagram



Deleted: <object>

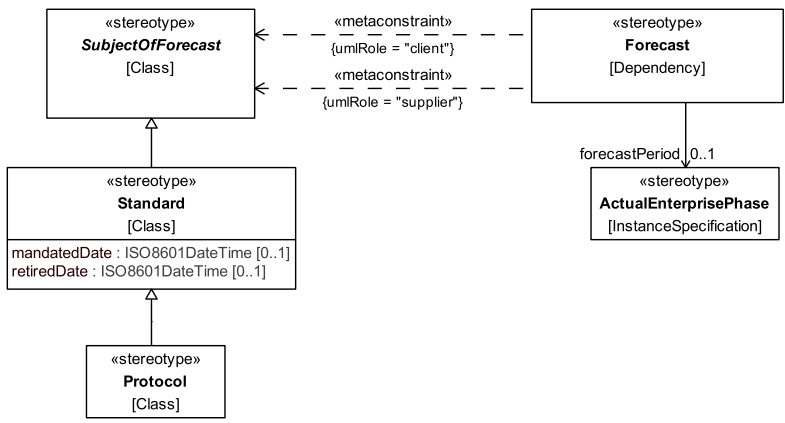


Figure A.63 - Standards Roadmap

Commented [Y121]: UAF11-35 Repacked figure with Standards\_Roadmap.svg

Elements

- [ActualEnterprisePhase](#)
- [Forecast](#)
- [Protocol](#)
- [Standard](#)
- [SubjectOfForecast](#)

**View Specifications::Standards::Traceability**

Stakeholders: Solution Providers, Systems Engineers, Software Engineers, Systems Architects, Business Architects  
Concerns: standards that need to be taken in account to ensure the interoperability of the implementation of architectural elements  
Definition: shows the applicability of standards to specific elements in the architecture.  
Recommended Implementation: tabular format, matrix format, SysML Block Definition Diagram

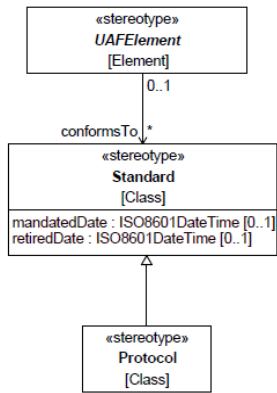


Figure A.64 - Standards Traceability

Elements

- [Protocol](#)
- [Standard](#)
- [UAFElement](#)

## A.2.9 View Specifications::Actual Resources

### View Specifications::Actual Resources::Structure

Stakeholders: Solution Providers, Systems Engineers, Business Architects

Concerns: the analysis, e.g., evaluation of different alternatives, what-if, trade-offs, V&V on the actual resource configurations as it provides a means to capture different solution architectures. The detailed analysis (trade-off, what-if, etc.) is carried out using the Resource Constraints view.

Definition: illustrates the expected or achieved actual resource configurations required to meet an operational need.

Recommended Implementation: SysML Block Definition Diagram

**Commented [GB122]:** UAF11-104/259

**Deleted:** SysML Internal Block Diagram

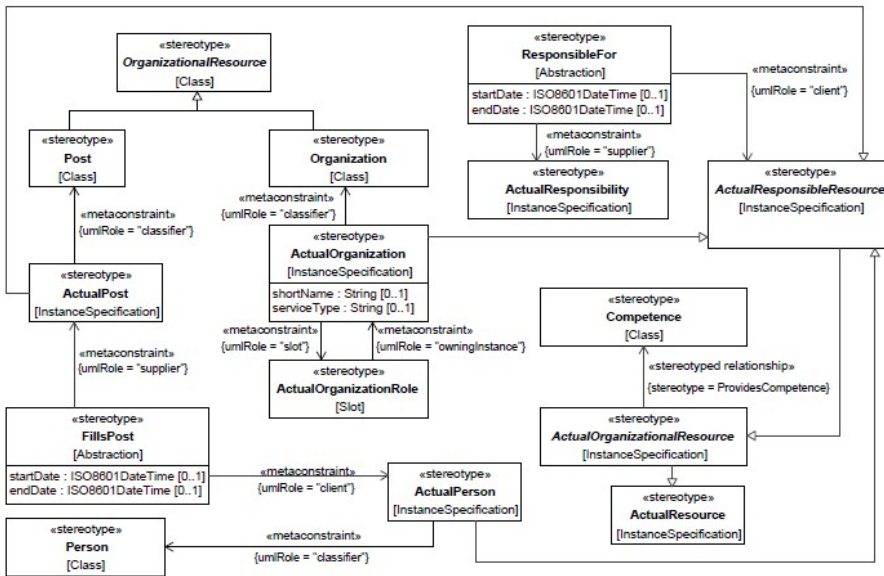


Figure A.65 - Actual Resources Structure

Elements

- [ActualOrganization](#)
- [ActualOrganizationalResource](#)
- [ActualOrganizationRole](#)
- [ActualPerson](#)
- [ActualPost](#)
- [ActualResource](#)
- [ActualResponsibility](#)
- [ActualResponsibleResource](#)
- [Competence](#)
- [FillsPost](#)
- [Organization](#)
- [OrganizationalResource](#)

- [Person](#)
- [Post](#)
- [ResponsibleFor](#)

**View Specifications::Actual Resources::Connectivity**

Stakeholders: Solution Providers, Systems Engineers, Business Architects

Concerns: the communication of actual resource

Definition: illustrates the actual resource configurations and actual relationships between them.

Recommended Implementation: tabular format, SysML Block Definition Diagram, SysML Internal Block Diagram, SysML Sequence Diagram

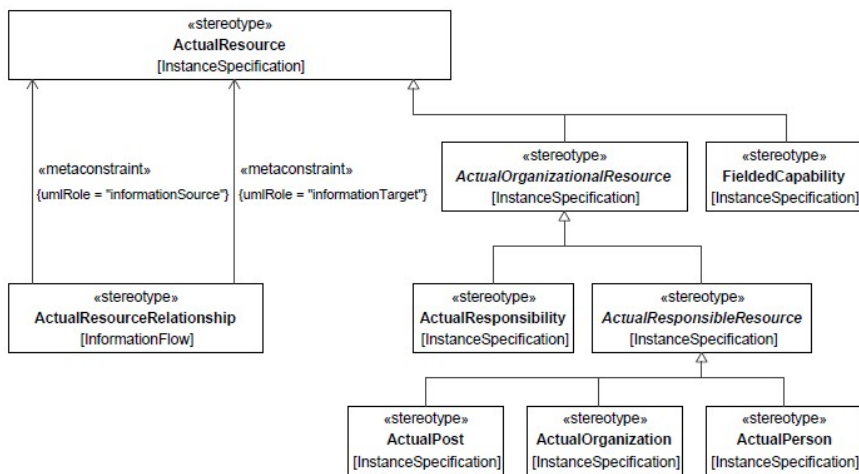


Figure A.66 - Actual Resources Connectivity

Elements

- [ActualOrganization](#)
- [ActualOrganizationalResource](#)
- [ActualPerson](#)
- [ActualPost](#)
- [ActualResource](#)
- [ActualResourceRelationship](#)
- [ActualResponsibility](#)
- [ActualResponsibleResource](#)
- [FieldedCapability](#)



Commented [AM123]: UAF11-128 Section added including Actual\_Resources\_Traceability.svg

### View Specifications::Actual Resources::Traceability

Stakeholders: Systems Engineers, Enterprise Architects, Solution Providers, Business Architects.

Concerns: traceability between operational activities and functions that implements them.

Definition: depicts the mapping of functions to operational activities and thus identifies the transformation of an operational need into a purposeful function performed by a resource or solution.

Recommended Implementation: Matrix format, SysML Block Definition Diagram.

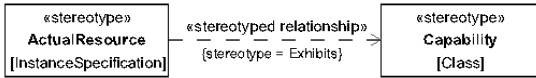


Figure 4:67 - Actual Resources Traceability Elements

- ActualResource
- Capability

### A.2.10 View Specifications::Dictionary

Stakeholders: Architects, users of the architecture, Capability Owners, Systems Engineers, Solution Providers

Concerns: Definitions for all the elements in the architecture, libraries of environments and measurements

Definition: Presents all the elements used in an architecture. Can be used specifically to capture:

- elements and relationships that are involved in defining the environments applicable to capability, operational concept, or set of systems.
- measurable properties that can be used to support analysis such as KPIs, MoEs, TPIs, etc.

Recommended Implementation: Tabular format, SysML Block Definition Diagram

### View Specifications::Dictionary::Dictionary

Stakeholders: Solution Providers, Systems Engineers, Software Architects, Business Architects

Concerns: provides a central reference for a given architecture's data and metadata. It enables the set of architecture description to stand alone, with minimal reference to outside resources.

Definition: contains definitions of terms used in the given architecture. It consists of textual definitions in the form of a glossary, their taxonomies, and their metadata (i.e., data about architecture data), including metadata for any custom-tailored views. Architects should use standard terms where possible (i.e., terms from existing, approved dictionaries, glossaries, and lexicons).

Recommended Implementation: text, table format

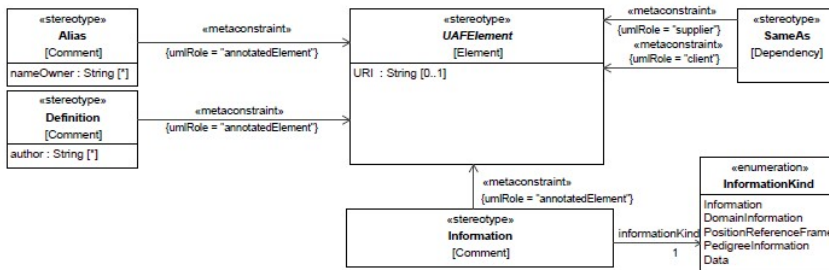


Figure A.67 - Dictionary

Elements

- [Alias](#)
- [Definition](#)
- [Information](#)
- [InformationKind](#)
- [SameAs](#)
- [UAFElement](#)

## A.2.11 View Specifications::Requirements

### View Specifications::Requirements::Requirements

Stakeholders: Requirement Engineers, Solution Providers, Systems Engineers, Software Engineers, Systems Architects, Business Architects

Concerns: provides a central reference for a set of stakeholder needs expressed as requirements, their relationship (via traceability) to more detailed requirements and the solution described by the architecture that will meet those requirements.

Definition: used to represent requirements, their properties, and relationships (trace, verify, satisfy, refine) between each other and to UAF architectural elements.

Recommended Implementation: SysML Requirement Diagram, tabular format, matrix format

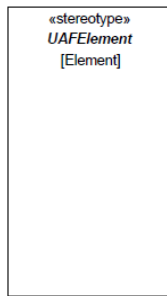


Figure A.68 - Requirements

Elements

- [UAFElement](#)

## A.2.12 View Specifications::Summary & Overview

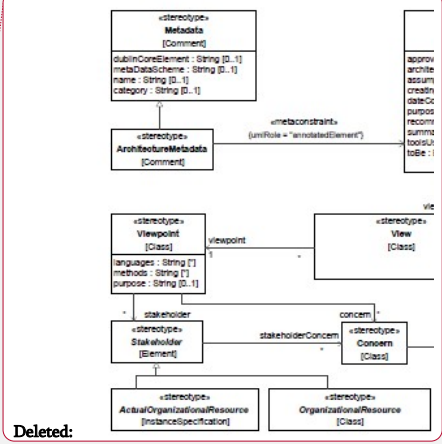
### View Specifications::Summary & Overview::Summary & Overview

Stakeholders: Decision makers, Solution Providers, Systems Engineers, Software Architects, Business Architects Concerns: quick overview of an architecture description and summary of analysis. In the initial phases of architecture development, it serves as a planning guide. Upon completion of an architecture, it provides a summary of findings, and any conducted analysis.

Definition: provides executive-level summary information in a consistent form that allows quick reference and comparison among architectures. The Summary and Overview includes assumptions, constraints, and limitations that may affect high-level decision processes involving the architecture.

Recommended Implementation: text, free form diagram, table format

Commented [AM124]: UAF11-36 Figure A.69 - Summary & Overview replaced by Summary\_&\_Overview.svg



Deleted:

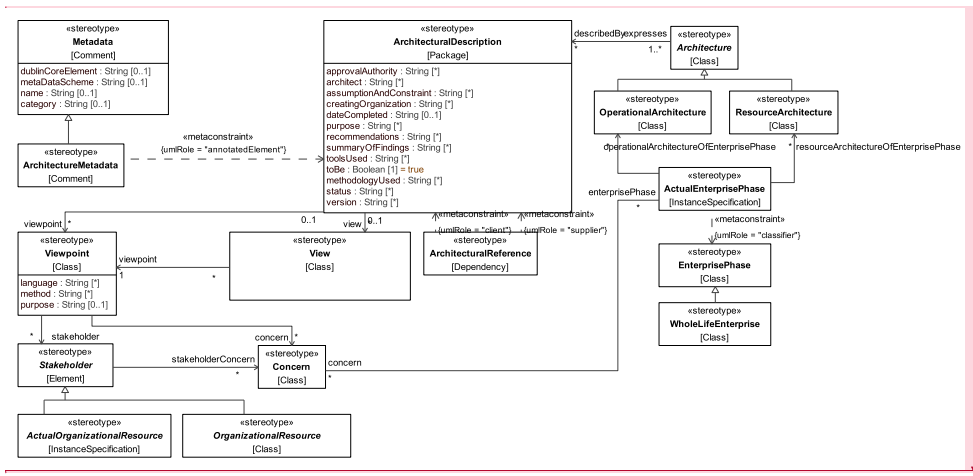


Figure A.69 - Summary & Overview

Elements

- [ActualEnterprisePhase](#)
- [ActualOrganizationalResource](#)
- [ArchitecturalDescription](#)
- [ArchitecturalReference](#)
- [Architecture](#)
- [ArchitectureMetadata](#)
- [Concern](#)
- [EnterprisePhase](#)
- [Metadata](#)
- [OperationalArchitecture](#)
- [OrganizationalResource](#)
- [ResourceArchitecture](#)
- [Stakeholder](#)
- [View](#)
- [Viewpoint](#)
- [WholeLifeEnterprise](#)

## A.2.13 View Specifications::Information

### View Specifications::Information::Information Model

Stakeholders: Data Modelers, Software Engineers, Systems Engineers

Concerns: address the information perspective on operational, service, and resource architectures.

Definition: allows analysis of an architecture's information and data definition aspect, without consideration of implementation specific issues.

Recommended Implementation: SysML Block Definition Diagram

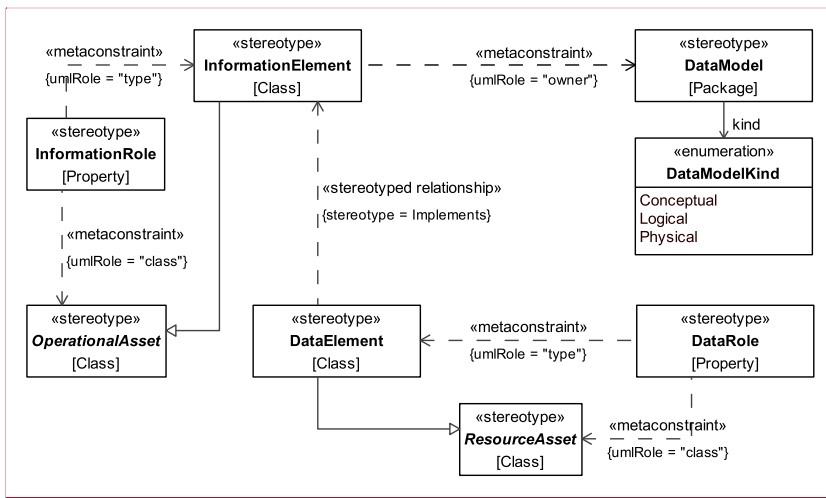


Figure A.70 - Information Model

#### Elements

- [DataElement](#)
- [DataModel](#)
- [DataModelKind](#)
- [DataRole](#)
- [InformationElement](#)
- [InformationRole](#)
- [OperationalAsset](#)
- [ResourceAsset](#)

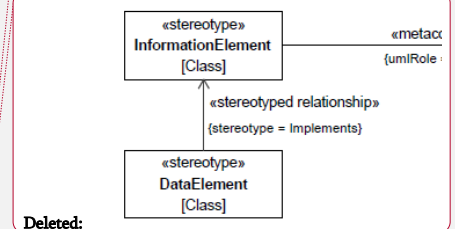
## A.2.14 View Specifications::Parameters

Stakeholders: Capability owners, Systems Engineers, Solution Providers

Concerns: identifies measurable properties that can be used to support engineering analysis and environment for the Capabilities.

Unified Architecture Framework Profile (UAFP), v1.0

Commented [AM125]: UAF11-3 Diagram replaced by Information\_Model.svg



Deleted:

Commented [AM126]: UAF11-3 Elements list updated according diagram Information\_Model.svg

Deleted: • ~~DataElement~~

- ~~DataModel~~
- ~~DataModelKind~~
- ~~InformationElement~~

Definition: Shows the measurable properties of something in the physical world and elements and relationships that are involved in defining the environments applicable to capability, operational concept or set of systems.

**View Specifications::Parameters::Parameters: Environment**

Stakeholders: Capability owners, Systems Engineers, Solution Providers

Concerns: defines the environment for the capabilities

Definition: shows the elements and relationships that are involved in defining the environments applicable to capability, operational concept or set of systems.

Recommended Implementation: SysML Block Definition Diagram

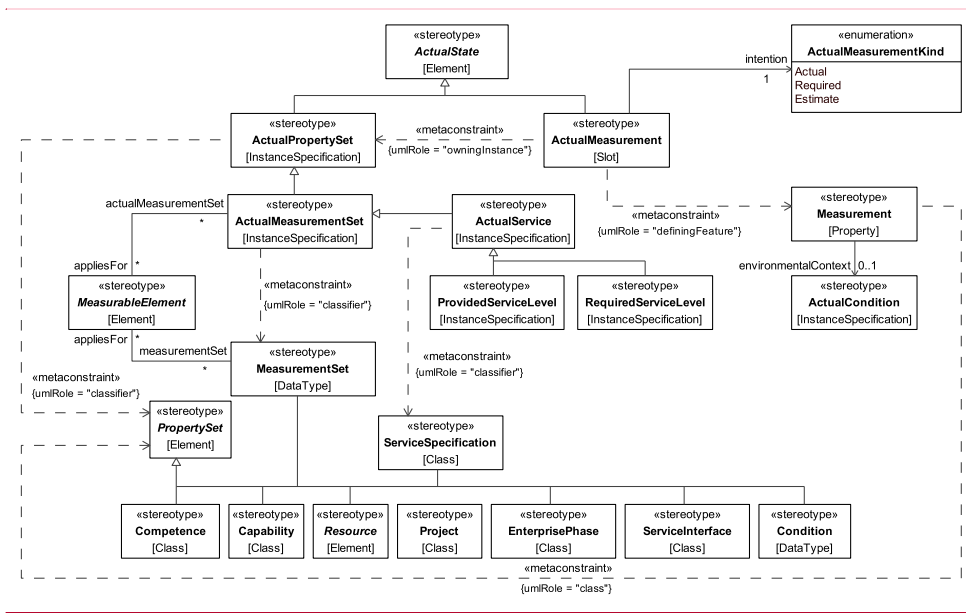


- [GeoPoliticalExtentTypeKind](#)
- [Location](#)
- [LocationHolder](#)
- [LocationKind](#)
- [LocationTypeKind](#)
- [OperationalRole](#)
- [ResourceAsset](#)
- [ResourcePerformer](#)
- [ResourceRole](#)

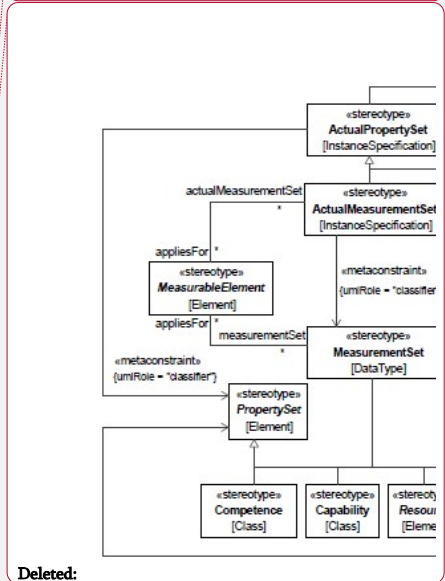
Commented [AM128]: UAF11-15 ResourceAsset added to the elements list.

**View Specifications::Parameters::Parameters: Measurements**

Stakeholders: Capability owners, Systems Engineers, Solution Providers  
 Concerns: identifies measurable properties that can be used to support analysis such as KPIs, MOs, TPIs, etc.  
 Definition: Shows the measurable properties of something in the physical world, expressed in amounts of a unit of measure that can be associated with any element in the architecture.  
 Recommended Implementation: SysML Block Definition Diagram



Commented [AM129]: UAF11-125 Figure A.72 - Parameters: Measurements Replaced by Measurements.svg



Deleted:

Figure A.72 - Parameters: Measurements



- [ActualCondition](#)

- [ActualMeasurement](#)

**Commented [AM130]:** [UAF11-125](#) Actual Condition added to the elements list.

- [ActualMeasurementKind](#)
- [ActualMeasurementSet](#)
- [ActualPropertySet](#)
- [ActualService](#)
- [ActualState](#)
- [Capability](#)
- [Competence](#)
- [Condition](#)
- [EnterprisePhase](#)
- [MeasurableElement](#)
- [Measurement](#)
- [MeasurementSet](#)
- [Project](#)
- [PropertySet](#)
- [ProvidedServiceLevel](#)
- [RequiredServiceLevel](#)
- [Resource](#)
- [ServiceInterface](#)
- [ServiceSpecification](#)

## Annex B: Class Library

### B.1 Class Library

A library of Measurements.

#### **BillingItem**

**Package:** Class Library

**isAbstract:** No

Description

Properties indicating the assurance of a piece of information.

Attributes

cost : Cost[1]	Details the cost of the BillingItem.
id : String[0..1]	Details the unique identifier of the BillingItem.
numberOfUses : Integer[0..1]	Details the numberOfUses of the BillingItem.
paymentLocation : String[0..1]	Details the location where payment should be made of the BillingItem.
paymentModality : PricingType[1]	Details if a payment is based upon Quantity, Time, or Use.
paymentPeriod : Periodicity[1]	Details the frequency of a payment period.
paymentTimeDuration : Duration[*]	Details the length of time the payments should be made i.e., 1 year.
periodDuration : Duration[0..1]	Details the time period between payments.
quantity : String[0..1]	Details the number of units to be delivered.
unit : String[0..1]	Details the units used for the BillingItem e.g., 1 gross.

#### **ClassificationAttributes**

**Package:** Class Library

**isAbstract:** No

Description

W3C XML Schema for the Intelligence Community Metadata Standard for Information Security Marking (IC-ISM), which is part of the IC standards for Information Assurance.

#### Attributes

classificationReason : String[]	One or more reason indicators or explanatory text describing the basis for an original classification decision.
classifiedBy : String[]	Details The identity, by name or personal identifier, and position title of the original classification authority for a resource.
dateOfExemptedSource : String[]	Details the specific year, month, and day of publication or release of a source document, or the most recent source document, that was itself marked with a declassification constraint. This element is always used in conjunction with typeOfExemptedSource element.
declassDate : String[]	Details a specific year, month, and day upon which the information shall be automatically declassified if not properly exempted from automatic declassification.
declassException : String[]	Details a single indicator describing an exemption to the nominal 25-year point for automatic declassification. This element is used in conjunction with the Declassification Date or Declassification Event.
DeclassManualReview : String[]	Details a true/false indicator that a manual review is required for declassification. Use this attribute to force the appearance of "//MR" in the header and footer marking titles. Use this attribute ONLY when it is necessary to override the business logic applied to classification and control markings in the document to determine whether manual review is required.
derivedFrom : String[]	Details a citation of the authoritative source or reference to multiple sources of the classification markings used in a classified resource.
DisseminationControls : String[]	Details one or more indicators identifying the expansion or limitation on the distribution of information.
FGIsorceOpen : String[]	Details one or more indicators identifying information which qualifies as foreign government information for which the source(s) of the information is not concealed.
FGIsorceProtected : String[]	Details a single indicator that information qualifies as foreign government information for which the source(s) of the information must be concealed. Within protected internal organizational spaces this element may be used to maintain a record of the one or more indicators identifying information which qualifies as foreign government information for which the source(s) of the information must be concealed. Measures must be taken prior to dissemination of the information to conceal the source(s) of the foreign government information.
nonICmarkings : String[]	Details one or more indicators of the expansion or limitation on the distribution of an information resource or portion within the domain of information originating from non-intelligence components.
ownerProducer : String[]	Details one or more indicators identifying the national government or international organization that have purview over the classification marking of an information resource or portion therein. This element is always used in conjunction with the Classification element. Taken together, the two elements specify the classification category and the type of classification (US, non-US, or

Joint). Within protected internal organizational spaces this element may include one or more indicators identifying information which qualifies as foreign government information for which the source(s) of the information must be concealed. Measures must be taken prior to dissemination of the information to conceal the source(s) of the foreign government information.

- releasableTo : String[] Details one or more indicators identifying the country or countries and/or international organization(s) to which classified information may be released based on the determination of an originator in accordance with established foreign disclosure procedures. This element is used in conjunction with the Dissemination Controls element.
- SARIdentifier : String[] Details the Authorized Special Access Required (SAR) program digraph(s) or trigraph(s) preceded by "SAR-". Either (a) a single digraph or trigraph or (b) a space-delimited list of digraphs or trigraphs. Example: "SAR-ABC SAR-DEF ..."
- SCIControls : String[] Details one or more indicators identifying sensitive compartmented information control system(s).
- typeOfExemptedSource : String[] Details a declassification marking of a source document that causes the current, derivative document to be exempted from automatic declassification. This element is always used in conjunction with the Date Of Exempted Source element.

#### Associations

- taxonomy : String[] Details a single indicator of the highest level of classification applicable to an information resource or portion within the domain of classified national security information. The Classification element is always used in conjunction with the Owner Producer element. Taken together, the two elements specify the classification category and the type of classification (US, non-US, or Joint).

### CommunicationsLinkProperties

**Package:** Class Library

**isAbstract:** No

#### Description

Properties detailing aspects of Resource Interfaces.

#### Attributes

- capacity : String[] Details how much information can be passed on the Communications Link.
- infrastructureTechnology : String[] Details the technology to be used to provide the communications infrastructure.

### **DataElementProperties**

**Package:** Class Library

**isAbstract:** No

Description

Properties detailing the aspects of a DataElement.

Attributes

accuracy : String[]	Details the accuracy of the data.
content : String[]	Specifies content of the data element (i.e., actual data to be exchanged).
formatType : String[]	Details the format of the data.
mediaType : String[]	Details the media used to transmit the data.
scope : String[]	Details in text a description of the extent or range of the data element content.
unitOfMeasurement : String[]	Details the units of measurement of the data.

### **Duration**

**Package:** Class Library

**isAbstract:** No

Description

Properties detailing aspects OperationalActivities.

Attributes

timeUnit : String[0..1]	Details the units of time e.g., second, hour, day.
value : Integer[0..1]	Details the value of the duration.

### **ExchangeProperties**

**Package:** Class Library

**isAbstract:** No

Description

Properties detailing aspects of exchange for Operational Exchange and/or Resource Interaction.

#### Attributes

accountability : String[*]	Details who or what is responsible for the exchange.
periodicity : String[*]	Details the frequency of the exchange.
size : String[*]	Details the size (in KB) of data that be exchanged.
throughput : String[*]	Details how much information can be exchanged.
timeliness : String[*]	Details the allowable time of delay this system data can tolerate and still be relevant to the receiving system.
transactionType : String[*]	Details the type of transactions used by the exchange.

#### InformationElementProperties

**Package:** Class Library

**isAbstract:** No

#### Description

Predefined additional DoDAF properties for InformationElement.

#### Attributes

accuracy : String[*]	Details the degree to which the information conforms to actual fact as required by the information producer and consumer.
content : String[*]	Specifies content of the information element (i.e., actual information to be exchanged).
language : String[*]	Details the language used to capture the information.
scope : String[*]	Details in text a description of the extent or range of the information element content.

#### OperationalActivityProperties

**Package:** Class Library

**isAbstract:** No

#### Description

Properties detailing aspects OperationalActivities.

#### Attributes

cost : String[]	Details the cost of an activity.
-----------------	----------------------------------

### Periodicity

**Package:** Class Library

**isAbstract:** No

#### Description

Enumeration of how often the information exchange occurs; may be an average or a worst case estimate and may include conditions. Its enumeration literals are:

- OnceAMonth - Indicates that an event of some sort may occur monthly.
- OnceAWeek - Indicates that an event of some sort may occur weekly.
- Anytime - Indicates that an event of some sort may occur at anytime.
- OnRequest - Indicates that an event of some sort may occur on request.

### PricingType

**Package:** Class Library

**isAbstract:** No

#### Description

Enumeration of a unit of measure of a resource. Its enumeration literals are:

- perTime - Indicates that the unit of measure of a resource is based on a unit of time.
- perUse - Indicates that the unit of measure of a resource is based upon how often the resource is used.
- perQuantity - Indicates that the unit of measure of a resource is based on a quantity.

### SecurityControlAssessmentProperties

**Package:** Class Library

**isAbstract:** No

#### Description

Properties detailing aspects of the Assessment and Authorization process.

#### Attributes

coverageOfSecurityControlAssessment : String[*]	Security controls assessment method that addresses the scope or breadth of the assessment objects included in the assessment (e.g., types of objects to be assessed and the number of objects to be assessed by type).
depthOfSecurityControlAssessment : String[*]	Security controls assessment method that addresses the rigor and level of detail associated with the application of the method.
effectivenessOfSecurityControl : String[*]	Details if security control is satisfactory or not as assessed.



### **SecurityControlProperties**

**Package:** Class Library

**isAbstract:** No

Description

Properties detailing aspects of Security Controls.

Attributes

securityControlApplicability : String[1] Details how applicable a security control is to a given security objective.

securityControlImportance : String[1] Details how important a security control is to a given security objective.

### **SecurityImpactProperties**

**Package:** Class Library

**isAbstract:** No

Description

Properties detailing aspects of Security Categories.

Attributes

securityAvailabilityImpact : String[\*] Details the potential impact on organization or individuals if the information is not available to those who need to access it.

securityClassification : String[\*] Details a classification for the exchange.

securityConfidentialityImpact : String[\*] Details the potential impact on organization or individuals due to unauthorized disclosure of information.

securityIntegrityImpact : String[\*] Details the potential impact on organization or individuals due to modification or destruction of information, and includes ensuring information non-repudiation and authenticity.

This page intentionally left blank.

Page 12: [1] Deleted	GRAHAM Bleakley	5/10/19 6:19:00 PM
Page 50: [2] Formatted	Aurelijus Morkevicius	5/1/19 2:36:00 PM
Not Expanded by / Condensed by		
Page 50: [2] Formatted	Aurelijus Morkevicius	5/1/19 2:36:00 PM
Not Expanded by / Condensed by		
Page 50: [2] Formatted	Aurelijus Morkevicius	5/1/19 2:36:00 PM
Not Expanded by / Condensed by		
Page 50: [2] Formatted	Aurelijus Morkevicius	5/1/19 2:36:00 PM
Not Expanded by / Condensed by		
Page 50: [2] Formatted	Aurelijus Morkevicius	5/1/19 2:36:00 PM
Not Expanded by / Condensed by		
Page 50: [2] Formatted	Aurelijus Morkevicius	5/1/19 2:36:00 PM
Not Expanded by / Condensed by		
Page 50: [2] Formatted	Aurelijus Morkevicius	5/1/19 2:36:00 PM
Not Expanded by / Condensed by		
Page 50: [2] Formatted	Aurelijus Morkevicius	5/1/19 2:36:00 PM
Not Expanded by / Condensed by		
Page 50: [2] Formatted	Aurelijus Morkevicius	5/1/19 2:36:00 PM
Not Expanded by / Condensed by		
Page 50: [2] Formatted	Aurelijus Morkevicius	5/1/19 2:36:00 PM
Not Expanded by / Condensed by		
Page 50: [2] Formatted	Aurelijus Morkevicius	5/1/19 2:36:00 PM
Not Expanded by / Condensed by		
Page 50: [2] Formatted	Aurelijus Morkevicius	5/1/19 2:36:00 PM
Not Expanded by / Condensed by		
Page 50: [2] Formatted	Aurelijus Morkevicius	5/1/19 2:36:00 PM
Not Expanded by / Condensed by		
Page 50: [2] Formatted	Aurelijus Morkevicius	5/1/19 2:36:00 PM
Not Expanded by / Condensed by		

Page 50: [2] Formatted	Aurelijus Morkevicius	5/1/19 2:36:00 PM
Not Expanded by / Condensed by		
Page 50: [2] Formatted	Aurelijus Morkevicius	5/1/19 2:36:00 PM
Not Expanded by / Condensed by		
Page 50: [2] Formatted	Aurelijus Morkevicius	5/1/19 2:36:00 PM
Not Expanded by / Condensed by		
Page 50: [2] Formatted	Aurelijus Morkevicius	5/1/19 2:36:00 PM
Not Expanded by / Condensed by		
Page 50: [2] Formatted	Aurelijus Morkevicius	5/1/19 2:36:00 PM
Not Expanded by / Condensed by		
Page 50: [2] Formatted	Aurelijus Morkevicius	5/1/19 2:36:00 PM
Not Expanded by / Condensed by		
Page 50: [2] Formatted	Aurelijus Morkevicius	5/1/19 2:36:00 PM
Not Expanded by / Condensed by		
Page 50: [2] Formatted	Aurelijus Morkevicius	5/1/19 2:36:00 PM
Not Expanded by / Condensed by		
Page 50: [2] Formatted	Aurelijus Morkevicius	5/1/19 2:36:00 PM
Not Expanded by / Condensed by		
Page 50: [2] Formatted	Aurelijus Morkevicius	5/1/19 2:36:00 PM
Not Expanded by / Condensed by		
Page 50: [2] Formatted	Aurelijus Morkevicius	5/1/19 2:36:00 PM
Not Expanded by / Condensed by		
Page 50: [2] Formatted	Aurelijus Morkevicius	5/1/19 2:36:00 PM
Not Expanded by / Condensed by		
Page 50: [2] Formatted	Aurelijus Morkevicius	5/1/19 2:36:00 PM
Not Expanded by / Condensed by		
Page 50: [2] Formatted	Aurelijus Morkevicius	5/1/19 2:36:00 PM
Not Expanded by / Condensed by		
Page 50: [2] Formatted	Aurelijus Morkevicius	5/1/19 2:36:00 PM
Not Expanded by / Condensed by		



Not Expanded by / Condensed by

Page 50: [3] Formatted Aurelijus Morkevicius 5/1/19 2:36:00 PM

Not Expanded by / Condensed by

Page 50: [3] Formatted Aurelijus Morkevicius 5/1/19 2:36:00 PM

Not Expanded by / Condensed by

Page 50: [3] Formatted Aurelijus Morkevicius 5/1/19 2:36:00 PM

Not Expanded by / Condensed by

Page 50: [3] Formatted Aurelijus Morkevicius 5/1/19 2:36:00 PM

Not Expanded by / Condensed by

Page 50: [3] Formatted Aurelijus Morkevicius 5/1/19 2:36:00 PM

Not Expanded by / Condensed by

Page 50: [3] Formatted Aurelijus Morkevicius 5/1/19 2:36:00 PM

Not Expanded by / Condensed by

Page 50: [3] Formatted Aurelijus Morkevicius 5/1/19 2:36:00 PM

Not Expanded by / Condensed by

Page 50: [3] Formatted Aurelijus Morkevicius 5/1/19 2:36:00 PM

Not Expanded by / Condensed by

Page 50: [3] Formatted Aurelijus Morkevicius 5/1/19 2:36:00 PM

Not Expanded by / Condensed by

Page 50: [3] Formatted Aurelijus Morkevicius 5/1/19 2:36:00 PM

Not Expanded by / Condensed by

Page 50: [3] Formatted Aurelijus Morkevicius 5/1/19 2:36:00 PM

Not Expanded by / Condensed by

Page 50: [3] Formatted Aurelijus Morkevicius 5/1/19 2:36:00 PM

Not Expanded by / Condensed by

Page 50: [3] Formatted Aurelijus Morkevicius 5/1/19 2:36:00 PM

Not Expanded by / Condensed by

Page 50: [3] Formatted Aurelijus Morkevicius 5/1/19 2:36:00 PM

Not Expanded by / Condensed by

Page 50: [3] Formatted Aurelijus Morkevicius 5/1/19 2:36:00 PM

Not Expanded by / Condensed by

Page 50: [3] Formatted Aurelijus Morkevicius 5/1/19 2:36:00 PM

Not Expanded by / Condensed by

Page 50: [3] Formatted Aurelijus Morkevicius 5/1/19 2:36:00 PM

Not Expanded by / Condensed by

Page 50: [3] Formatted	Aurelijus Morkevicius	5/1/19 2:36:00 PM
------------------------	-----------------------	-------------------

Not Expanded by / Condensed by

Page 159: [4] Deleted	Aurelijus Morkevicius	5/1/19 2:58:00 PM
-----------------------	-----------------------	-------------------

Page 225: [5] Deleted	Aurelijus Morkevicius	5/1/19 6:04:00 PM
-----------------------	-----------------------	-------------------